

# SIEMENS

## SIMATIC

### WinCC Basic V12.0 SP1

#### Systemhandbuch


|                                       |    |
|---------------------------------------|----|
| Systemübersicht STEP 7 und WinCC      | 1  |
| Liesmich                              | 2  |
| Installation                          | 3  |
| Projekte und Programme migrieren      | 4  |
| Erste Schritte                        | 5  |
| Einführung in das TIA-Portal          | 6  |
| Projekte bearbeiten                   | 7  |
| Geräte und Netze bearbeiten           | 8  |
| PLC programmieren                     | 9  |
| Prozesse visualisieren (Basic)        | 10 |
| Technologiefunktionen einsetzen       | 11 |
| Online- und Diagnosefunktionen nutzen | 12 |
| Hardware-Dokumentation                | 13 |


Ausdruck der Onlinehilfe


## Rechtliche Hinweise

### Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

|   |
|---|
|  <b>GEFAHR</b>   |
| bedeutet, dass Tod oder schwere Körperverletzung eintreten <b>wird</b> , wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden. |

|   |
|---|
|  <b>WARNUNG</b>  |
| bedeutet, dass Tod oder schwere Körperverletzung eintreten <b>kann</b> , wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden. |

|   |
|---|
|  <b>VORSICHT</b>                               |
| bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden. |

|   |
|---|
| <b>ACHTUNG</b>  |
| bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden. |


Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

### Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

### Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

|   |
|---|
|  <b>WARNUNG</b>  |
| Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden. |

### Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

### Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.



# Inhaltsverzeichnis

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Systemübersicht STEP 7 und WinCC</b> .....            | <b>25</b> |
| 1.1      | Skalierung von STEP 7 und WinCC.....                     | 25        |
| 1.2      | Optionen für STEP 7 Engineeringssystem.....              | 26        |
| 1.3      | Optionen für WinCC Engineering- und Runtime-Systeme..... | 26        |
| <b>2</b> | <b>Liesmich</b> .....                                    | <b>29</b> |
| 2.1      | Allgemeine Hinweise.....                                 | 29        |
| 2.1.1    | Allgemeine Hinweise.....                                 | 29        |
| 2.1.2    | Hinweise zur Installation.....                           | 33        |
| 2.2      | STEP 7 Basic.....  | 35        |
| 2.2.1    | Security-Hinweise.....                                   | 35        |
| 2.2.2    | Hinweise zur Benutzung.....                              | 37        |
| 2.2.3    | Geräte und Netze bearbeiten.....                         | 39        |
| 2.2.3.1  | Allgemeine Hinweise zu Geräten und Netzen.....           | 39        |
| 2.2.3.2  | Verwendung von Modulen an der S7-1200.....               | 39        |
| 2.2.3.3  | Austausch der ET 200S Positioniermodule.....             | 40        |
| 2.2.3.4  | CP 343-2 an SIMATIC S7 Embedded Controller EC31-RTX..... | 40        |
| 2.2.3.5  | Hinweise zu Online und Diagnose.....                     | 41        |
| 2.2.3.6  | Netzwerk-Komponenten.....                                | 41        |
| 2.2.4    | PLC programmieren.....                                   | 42        |
| 2.2.4.1  | Allgemeine Hinweise zur PLC-Programmierung.....          | 42        |
| 2.2.4.2  | Anweisungen.....   | 44        |
| 2.2.4.3  | Anwenderprogramm testen.....                             | 45        |
| 2.2.5    | Technologiefunktionen.....                               | 46        |
| 2.2.5.1  | Hinweise zu Technologiefunktionen.....                   | 46        |
| 2.3      | WinCC Basic.....   | 46        |
| 2.3.1    | Security-Hinweise.....                                   | 46        |
| 2.3.2    | Hinweise zur Benutzung.....                              | 49        |
| 2.3.3    | Migration.....   | 52        |
| 2.3.4    | Engineering System.....                                  | 54        |
| 2.3.4.1  | Bilder und Bildobjekte.....                              | 54        |
| 2.3.4.2  | Variablen und Verbindungen.....                          | 59        |
| 2.3.4.3  | Meldesystem und Meldeanzeigen.....                       | 61        |
| 2.3.4.4  | Systemfunktionen.....                                    | 62        |
| 2.3.4.5  | Rezepturen.....  | 62        |
| 2.3.4.6  | Benutzerverwaltung.....                                  | 62        |
| 2.3.4.7  | Kommunikation.....                                       | 63        |
| 2.3.5    | Übersetzen und Laden.....                                | 66        |
| 2.3.6    | Runtime.....   | 69        |
| 2.3.6.1  | Hinweise zum Betrieb in Runtime.....                     | 69        |
| 2.3.6.2  | Hinweise zum Betrieb von Panels in Runtime.....          | 70        |
| 2.3.7    | Bediengeräte.....  | 71        |
| 2.3.7.1  | Hinweise zu Bediengeräten.....                           | 71        |

|          |   |            |
|----------|---|------------|
| <b>3</b> | <b>Installation.....</b>                                  | <b>75</b>  |
| 3.1      | Systemvoraussetzungen für die Installation.....           | 75         |
| 3.1.1    | Hinweise zu Lizenzen.....                                 | 75         |
| 3.1.2    | Hinweise zu den Systemvoraussetzungen.....                | 76         |
| 3.1.3    | Systemvoraussetzungen STEP 7 Basic.....                   | 77         |
| 3.1.3.1  | Lizensierung von STEP 7.....                              | 77         |
| 3.1.3.2  | Umgang mit Lizenzen und License Keys.....                 | 78         |
| 3.1.3.3  | Software- und Hardwarevoraussetzungen STEP 7.....         | 80         |
| 3.1.4    | Systemvoraussetzungen WinCC Basic.....                    | 83         |
| 3.1.4.1  | Software- und Hardwarevoraussetzungen.....                | 83         |
| 3.1.4.2  | Parallelinstallationen.....                               | 87         |
| 3.1.4.3  | Lizenzen und Powerpacks.....                              | 88         |
| 3.2      | Installationsprotokoll.....                               | 93         |
| 3.3      | Installation starten.....                                 | 94         |
| 3.4      | Support Packages installieren.....                        | 96         |
| 3.5      | Installierte Software anzeigen.....                       | 97         |
| 3.6      | Installierte Produkte modifizieren oder hochrüsten.....   | 98         |
| 3.7      | Installierte Produkte reparieren.....                     | 100        |
| 3.8      | Deinstallation starten.....                               | 101        |
| 3.9      | Migrations-Tool installieren und deinstallieren.....      | 104        |
| 3.9.1    | Systemvoraussetzungen.....                                | 104        |
| 3.9.2    | Migrations-Tool installieren.....                         | 104        |
| 3.9.3    | Migrations-Tool deinstallieren.....                       | 105        |
| <b>4</b> | <b>Projekte und Programme migrieren.....</b>              | <b>107</b> |
| 4.1      | Projekte in ein TIA-Portal-Projekt migrieren.....         | 107        |
| 4.1.1    | Migration von Projekten mit dem TIA-Portal.....           | 107        |
| 4.1.2    | Vorbereitung von Projekten mit dem Migrations-Tool.....   | 109        |
| 4.1.2.1  | Migration von Projekten mit dem Migrations-Tool.....      | 109        |
| 4.1.2.2  | Migrations-Tool aufrufen.....                             | 110        |
| 4.1.2.3  | Migrations-Datei erstellen.....                           | 110        |
| 4.1.3    | Projekte migrieren.....                                   | 112        |
| 4.1.4    | Verlauf der Migration anzeigen.....                       | 113        |
| 4.1.5    | Protokoll der Migration anzeigen.....                     | 114        |
| 4.1.6    | WinCC flexible Projekte migrieren (Basic).....            | 114        |
| 4.1.6.1  | Grundlagen (WinCC flexible).....                          | 114        |
| 4.1.6.2  | Engineering-Daten migrieren (WinCC flexible).....         | 120        |
| 4.1.6.3  | Runtime-Daten migrieren (WinCC flexible).....             | 131        |
| 4.1.6.4  | Integrierte Projekte migrieren (WinCC flexible).....      | 134        |
| 4.1.6.5  | Referenz (WinCC flexible).....                            | 137        |
| 4.1.7    | Integrierte Projekte migrieren.....                       | 145        |
| 4.1.7.1  | Migration eines integrierten Projekts.....                | 145        |
| 4.1.7.2  | Integrierte Projekte nacharbeiten.....                    | 147        |
| 4.1.7.3  | Unspezifizierte CPUs in spezifizierte CPUs umwandeln..... | 149        |
| 4.1.7.4  | Integrierte HMI-Verbindung erstellen.....                 | 151        |
| 4.1.7.5  | HMI-Variablen neu verknüpfen.....                         | 153        |
| 4.1.7.6  | Unspezifizierte Verbindung löschen.....                   | 154        |

|          |   |            |
|----------|---|------------|
| 4.2      | Programmierempfehlungen.....                            | 155        |
| 4.2.1    | Die neuen Funktionen der S7-1500-CPU im Überblick.....  | 155        |
| 4.2.2    | Symbolisch adressieren.....                             | 157        |
| 4.2.3    | IEC-Zeiten und -Zähler verwenden.....                   | 159        |
| 4.2.4    | Freigabeausgang ENO flexibel einsetzen.....             | 161        |
| 4.2.5    | MOVE-Anweisungen in AWL verwenden.....                  | 162        |
| 4.2.6    | Array-Zugriff mit variablem Index realisieren.....      | 164        |
| 4.2.7    | PLC-Datentypen (UDT) verwenden.....                     | 166        |
| <b>5</b> | <b>Erste Schritte.....</b>                              | <b>169</b> |
| 5.1      | Getting Started Dokumentationen.....                    | 169        |
| <b>6</b> | <b>Einführung in das TIA-Portal.....</b>                | <b>171</b> |
| 6.1      | Bedienoberfläche und Bedienung.....                     | 171        |
| 6.1.1    | TIA-Portal starten, einstellen und beenden.....         | 171        |
| 6.1.1.1  | TIA-Portal starten und beenden.....                     | 171        |
| 6.1.1.2  | Übersicht der Programmeinstellungen.....                | 171        |
| 6.1.1.3  | Übersicht der Skript- und Texteditor-Einstellungen..... | 173        |
| 6.1.1.4  | Übersicht der Druckeinstellungen.....                   | 174        |
| 6.1.1.5  | Einstellungen ändern.....                               | 175        |
| 6.1.2    | Aufbau der Bedienoberfläche.....                        | 176        |
| 6.1.2.1  | Ansichten.....  | 176        |
| 6.1.2.2  | Portalansicht.....                                      | 176        |
| 6.1.2.3  | Projektansicht.....                                     | 178        |
| 6.1.2.4  | Bibliotheksansicht.....                                 | 181        |
| 6.1.2.5  | Projektnavigation.....                                  | 181        |
| 6.1.2.6  | Arbeitsbereich.....                                     | 184        |
| 6.1.2.7  | Inspektorfenster.....                                   | 192        |
| 6.1.2.8  | Task Cards.....   | 194        |
| 6.1.2.9  | Referenzprojekte.....                                   | 196        |
| 6.1.2.10 | Detailansicht.....                                      | 198        |
| 6.1.2.11 | Übersichtsfenster.....                                  | 199        |
| 6.1.2.12 | Bedienoberflächenlayout.....                            | 203        |
| 6.1.3    | Tastaturbedienung im TIA-Portal.....                    | 207        |
| 6.1.3.1  | Bedienung des TIA-Portals mit der Tastatur.....         | 207        |
| 6.1.3.2  | Übersicht aller Tastenkombinationen anzeigen.....       | 207        |
| 6.1.3.3  | Grundfunktionen des TIA-Portals.....                    | 207        |
| 6.1.3.4  | Nutzen projektbezogener Funktionen.....                 | 209        |
| 6.1.3.5  | Anordnung von Fenstern.....                             | 209        |
| 6.1.3.6  | Navigation durch die Programmoberfläche.....            | 210        |
| 6.1.3.7  | Anpassen von Editoren.....                              | 212        |
| 6.1.3.8  | Bearbeiten von Objekten.....                            | 213        |
| 6.1.3.9  | Textbearbeitung.....                                    | 214        |
| 6.1.3.10 | Bearbeiten von Tabellen.....                            | 216        |
| 6.1.3.11 | Nutzen von Online-Funktionen.....                       | 217        |
| 6.1.3.12 | Bildschirmtastatur verwenden.....                       | 218        |
| 6.1.4    | Betriebssystemabhängige Besonderheiten.....             | 218        |
| 6.1.4.1  | Einfluss von Benutzerrechten.....                       | 218        |
| 6.1.4.2  | Benutzerrechte erweitern.....                           | 219        |
| 6.2      | Hilfe zum Informationssystem.....                       | 220        |
| 6.2.1    | Allgemeines zum Informationssystem.....                 | 220        |

|          |  |            |
|----------|--|------------|
| 6.2.2    | Hilfe öffnen.....  | 223        |
| 6.2.3    | Hilfe nach Stichwörtern durchsuchen.....                     | 224        |
| 6.2.4    | Im Volltext suchen.....                                      | 224        |
| 6.2.5    | Favoriten verwenden.....                                     | 225        |
| 6.2.6    | Hilfethemen drucken.....                                     | 226        |
| 6.2.7    | Anzeige von Tooltips und Tooltip-Kaskaden konfigurieren..... | 227        |
| 6.2.8    | Sicherheitshinweise.....                                     | 227        |
| 6.2.9    | Individuelle Dokumentation zusammenstellen.....              | 229        |
| <b>7</b> | <b>Projekte bearbeiten.....</b>                              | <b>231</b> |
| 7.1      | Grundlagen zu Projekten.....                                 | 231        |
| 7.2      | Protokolle verwenden.....                                    | 231        |
| 7.3      | Projekte anlegen und verwalten.....                          | 232        |
| 7.3.1    | Neues Projekt erstellen.....                                 | 232        |
| 7.3.2    | Projekte öffnen.....   | 233        |
| 7.3.3    | Hinweise zur Kompatibilität.....                             | 234        |
| 7.3.4    | Projekte hochrüsten.....                                     | 235        |
| 7.3.5    | Eigenschaften des Projekts anzeigen.....                     | 237        |
| 7.3.6    | Projekte speichern.....                                      | 237        |
| 7.3.7    | Projekte schließen.....                                      | 238        |
| 7.3.8    | Projekte löschen.....  | 239        |
| 7.3.9    | Mit mehrsprachigen Projekten arbeiten.....                   | 240        |
| 7.3.9.1  | Grundlagen zu Projekttexten.....                             | 240        |
| 7.3.9.2  | Projektsprachen auswählen.....                               | 242        |
| 7.3.9.3  | Editiersprache festlegen.....                                | 243        |
| 7.3.9.4  | Alle Projekttexte tabellarisch übersetzen.....               | 243        |
| 7.3.9.5  | Einzelnen Objekten zugehörige Texte übersetzen.....          | 244        |
| 7.3.9.6  | Texte mithilfe von Referenztexten übersetzen.....            | 244        |
| 7.3.9.7  | Projekttexte exportieren und importieren.....                | 245        |
| 7.3.9.8  | Anwendungsbeispiele für mehrsprachige Projekte.....          | 247        |
| 7.3.10   | Projekte archivieren und dearchivieren.....                  | 249        |
| 7.3.10.1 | Arbeiten mit Projektarchiven.....                            | 249        |
| 7.3.10.2 | Projekte archivieren.....                                    | 249        |
| 7.3.10.3 | Projekte dearchivieren.....                                  | 251        |
| 7.4      | Referenzprojekte verwenden.....                              | 252        |
| 7.4.1    | Grundlagen zu Referenzprojekten.....                         | 252        |
| 7.4.2    | Referenzprojekt öffnen und schließen.....                    | 252        |
| 7.4.3    | Referenzprojekte vergleichen.....                            | 253        |
| 7.5      | Projektdateien bearbeiten.....                               | 254        |
| 7.5.1    | Projektdateien übersetzen und laden.....                     | 254        |
| 7.5.1.1  | Projektdateien übersetzen.....                               | 254        |
| 7.5.1.2  | Projektdateien laden.....                                    | 256        |
| 7.5.2    | Projektdateien vergleichen.....                              | 261        |
| 7.5.2.1  | Grundlagen zum Projektdateivergleich.....                    | 261        |
| 7.5.2.2  | Online/Offline-Vergleich durchführen.....                    | 262        |
| 7.5.2.3  | Offline/Offline-Vergleich durchführen.....                   | 262        |
| 7.5.2.4  | Vergleichseditor verwenden.....                              | 264        |
| 7.5.3    | Projektdateien schützen.....                                 | 274        |
| 7.5.3.1  | Schutzkonzept für Projektdateien.....                        | 274        |
| 7.5.3.2  | Zugriffsrechte für Geräte zurücknehmen.....                  | 275        |

|          |  |     |
|----------|--|-----|
| 7.5.4    | Projekthinhalte drucken.....                                       | 276 |
| 7.5.4.1  | Projektdokumentation drucken.....                                  | 276 |
| 7.5.4.2  | Modulbeschriftungsstreifen drucken.....                            | 295 |
| 7.6      | Aktionen rückgängig machen und wiederholen.....                    | 300 |
| 7.6.1    | Grundlagen zum Rückgängig machen und Wiederholen von Aktionen..... | 300 |
| 7.6.2    | Aktion rückgängig machen.....                                      | 301 |
| 7.6.3    | Aktion wiederholen.....  | 303 |
| 7.7      | In Projekten suchen und ersetzen.....                              | 304 |
| 7.7.1    | Informationen zur Suchfunktion.....                                | 304 |
| 7.7.2    | Suchen und ersetzen.....   | 304 |
| 7.8      | Mit Textlisten arbeiten.....                                       | 306 |
| 7.8.1    | Textlisten.....  | 306 |
| 7.8.2    | Anwenderdefinierte Textlisten anlegen.....                         | 308 |
| 7.8.3    | Anwenderdefinierte Textlisten bearbeiten.....                      | 309 |
| 7.8.4    | Systemdefinierte Textlisten bearbeiten.....                        | 309 |
| 7.9      | Memory Cards verwenden.....  | 310 |
| 7.9.1    | Grundlagen zu Memory Cards.....                                    | 310 |
| 7.9.2    | Benutzerdefinierten Kartenleser hinzufügen.....                    | 311 |
| 7.9.3    | Auf Memory Cards zugreifen.....                                    | 312 |
| 7.9.4    | Eigenschaften von Memory Cards anzeigen.....                       | 312 |
| 7.10     | Bibliotheken verwenden.....  | 313 |
| 7.10.1   | Grundlagen zu Bibliotheken.....                                    | 313 |
| 7.10.2   | Task Card "Bibliotheken" verwenden.....                            | 315 |
| 7.10.2.1 | Übersicht über die Task Card "Bibliotheken".....                   | 315 |
| 7.10.2.2 | Elementansicht verwenden.....                                      | 317 |
| 7.10.3   | Bibliotheksansicht verwenden.....                                  | 318 |
| 7.10.3.1 | Übersicht über die Bibliotheksansicht.....                         | 318 |
| 7.10.3.2 | Bibliotheksansicht öffnen und schließen.....                       | 320 |
| 7.10.4   | Bibliotheksverwaltung verwenden.....                               | 321 |
| 7.10.4.1 | Übersicht über die Bibliotheksverwaltung.....                      | 321 |
| 7.10.4.2 | Bibliotheksverwaltung öffnen.....                                  | 323 |
| 7.10.5   | Globalen Bibliotheken verwenden.....                               | 323 |
| 7.10.5.1 | Globale Bibliothek erstellen.....                                  | 323 |
| 7.10.5.2 | Globale Bibliothek öffnen.....                                     | 324 |
| 7.10.5.3 | Bibliotheken aus älteren Versionen hochrüsten.....                 | 325 |
| 7.10.5.4 | Eigenschaften von globalen Bibliotheken anzeigen.....              | 326 |
| 7.10.5.5 | Protokolle zu globalen Bibliotheken anzeigen.....                  | 327 |
| 7.10.5.6 | Globale Bibliothek speichern.....                                  | 328 |
| 7.10.5.7 | Globale Bibliothek schließen.....                                  | 329 |
| 7.10.5.8 | Globale Bibliothek löschen.....                                    | 330 |
| 7.10.5.9 | Globale Bibliotheken archivieren und deaktivieren.....             | 331 |
| 7.10.6   | Ordner in einer Bibliothek erstellen.....                          | 333 |
| 7.10.7   | Kopiervorlagen einsetzen.....                                      | 334 |
| 7.10.7.1 | Grundlagen zu Kopiervorlagen.....                                  | 334 |
| 7.10.7.2 | Kopiervorlagen hinzufügen.....                                     | 335 |
| 7.10.7.3 | Kopiervorlagen filtern.....  | 336 |
| 7.10.7.4 | Kopiervorlagen verwenden.....                                      | 337 |
| 7.10.8   | Typen und ihre Versionen einsetzen.....                            | 337 |
| 7.10.8.1 | Grundlagen zu Typen.....   | 337 |
| 7.10.8.2 | Zustand von Typ-Versionen.....                                     | 339 |

|          |   |            |
|----------|---|------------|
| 7.10.8.3 | Freigegebene Typ-Version anzeigen.....                                  | 340        |
| 7.10.8.4 | Eigenschaften eines Typen oder einer Version anzeigen.....              | 341        |
| 7.10.8.5 | Arbeiten mit Typen in der Projektbibliothek.....                        | 342        |
| 7.10.8.6 | Arbeiten mit Typen in globalen Bibliotheken.....                        | 354        |
| 7.10.9   | Bibliothekselemente bearbeiten.....                                     | 358        |
| 7.10.10  | Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren..... | 360        |
| 7.10.11  | Namen und Pfadstruktur harmonisieren.....                               | 362        |
| 7.10.12  | Bibliothek aufräumen.....   | 362        |
| 7.10.13  | Bibliothekselemente vergleichen.....                                    | 363        |
| 7.11     | Querverweise nutzen.....  | 364        |
| 7.11.1   | Querverweise nutzen.....  | 364        |
| 7.12     | Geräte simulieren.....  | 365        |
| 7.12.1   | Simulation von Geräten.....   | 365        |
| 7.12.2   | Simulation starten.....   | 365        |
| <b>8</b> | <b>Geräte und Netze bearbeiten.....</b>                                 | <b>367</b> |
| 8.1      | Geräte und Netze konfigurieren.....                                     | 367        |
| 8.1.1    | Hardware- und Netzwerkeditor.....                                       | 367        |
| 8.1.1.1  | Übersicht über den Hardware- und Netzwerkeditor.....                    | 367        |
| 8.1.1.2  | Netzsicht.....  | 369        |
| 8.1.1.3  | Gerätesicht.....  | 371        |
| 8.1.1.4  | Topologiesicht.....   | 374        |
| 8.1.1.5  | Drucken von Hardware- und Netzkonfigurationen.....                      | 377        |
| 8.1.1.6  | Seitenumbruchvorschau für den Druck aktivieren.....                     | 378        |
| 8.1.1.7  | Optionen für den Druck ändern.....                                      | 379        |
| 8.1.1.8  | Inspektorfenster.....   | 379        |
| 8.1.1.9  | Hardware-Katalog.....   | 381        |
| 8.1.1.10 | Informationen zu Hardware-Komponenten.....                              | 383        |
| 8.1.1.11 | Produktsupport aktivieren.....  | 384        |
| 8.1.1.12 | Tastaturbedienung: Navigation im Editor.....                            | 385        |
| 8.1.1.13 | Tastaturbedienung: Bearbeiten von Objekten.....                         | 386        |
| 8.1.2    | Geräte konfigurieren.....   | 388        |
| 8.1.2.1  | Grundlagen.....   | 388        |
| 8.1.2.2  | Einzelne Geräte konfigurieren.....                                      | 397        |
| 8.1.3    | Netze konfigurieren.....  | 413        |
| 8.1.3.1  | Geräte vernetzen.....   | 413        |
| 8.1.3.2  | Kommunikation über Verbindungen.....                                    | 433        |
| 8.1.3.3  | Topologie anzeigen und projektieren.....                                | 494        |
| 8.1.3.4  | Industrial Ethernet Security.....                                       | 511        |
| 8.1.4    | Konfigurationen erstellen.....  | 627        |
| 8.1.4.1  | Wissenswertes zum Webserver.....  | 627        |
| 8.1.4.2  | Wissenswertes zu Betriebsarten bei PROFIBUS-DP.....                     | 627        |
| 8.1.4.3  | Automatisierungssysteme konfigurieren.....                              | 628        |
| 8.1.4.4  | S7-1200-CM/CP.....  | 666        |
| 8.1.4.5  | IPv6-Protokoll.....   | 679        |
| 8.1.4.6  | SCALANCE X, W und M.....  | 680        |
| 8.1.4.7  | PROFIBUS DP konfigurieren.....  | 767        |
| 8.1.4.8  | Konfigurationen für PROFINET IO.....                                    | 791        |
| 8.1.4.9  | Buskopplung mit PN/PN Coupler.....                                      | 821        |
| 8.1.4.10 | Externe Tools einbinden.....  | 822        |
| 8.1.4.11 | Konfiguration laden.....  | 824        |

|          |  |             |
|----------|--|-------------|
| 8.1.5    | Meldungen anzeigen.....  | 827         |
| 8.1.5.1  | Übersicht über die Meldungsanzeige.....  | 827         |
| 8.1.5.2  | Archivansicht.....   | 828         |
| 8.1.5.3  | Aufbau der Meldungen in der Archivansicht.....   | 828         |
| 8.1.5.4  | Meldungen empfangen.....   | 829         |
| 8.1.5.5  | Archiv exportieren.....  | 829         |
| 8.1.5.6  | Archiv leeren.....   | 830         |
| 8.1.5.7  | Ansicht "Aktive Meldungen".....  | 830         |
| 8.1.5.8  | Aufbau der Meldungen in der Ansicht "Aktive Meldungen".....  | 830         |
| 8.1.5.9  | Status der Meldungen.....  | 830         |
| 8.1.5.10 | Meldungen quittieren.....  | 831         |
| 8.1.5.11 | Meldungen ignorieren.....  | 832         |
| 8.1.5.12 | Tastaturbedienung in der Meldungsanzeige.....  | 832         |
| 8.1.6    | Zusatzinformationen zu Konfigurationen.....  | 833         |
| 8.1.6.1  | Arbeitsweise von S7-1200 CPUs.....   | 833         |
| 8.1.6.2  | Dezentrale Peripherie.....   | 910         |
| 8.2      | Geräte und Netze diagnostizieren.....  | 964         |
| 8.2.1    | Hardware diagnostizieren.....  | 964         |
| 8.2.1.1  | Überblick über die Hardware-Diagnose.....  | 964         |
| 8.2.1.2  | Nicht änderbare und aktuelle Werte von parametrierbaren Baugruppeneigenschaften anzeigen...<br>975 |             |
| 8.2.1.3  | Aktuelle Werte von dynamischen Baugruppeneigenschaften anzeigen.....                               | 981         |
| 8.2.1.4  | Eine Baugruppe auf Störungen überprüfen.....   | 986         |
| 8.2.1.5  | Eigenschaften einer Baugruppe bzw. des PG/PC ändern.....   | 992         |
| 8.2.1.6  | Diagnose im Betriebszustand STOP.....  | 1009        |
| 8.2.1.7  | Online-Zugänge in der Online- und Diagnosesicht.....   | 1012        |
| 8.2.1.8  | PROFIBUS DP-Subnetze auf Störungen überprüfen.....   | 1015        |
| 8.2.2    | Verbindungen diagnostizieren.....  | 1018        |
| 8.2.2.1  | Überblick über die Verbindungsdiagnose.....  | 1018        |
| 8.2.2.2  | Verbindungsstatus mittels Symbolen anzeigen.....   | 1019        |
| 8.2.2.3  | Verbindungen im Detail diagnostizieren.....  | 1020        |
| <b>9</b> | <b>PLC programmieren.....</b>  | <b>1025</b> |
| 9.1      | Anwenderprogramm erstellen.....  | 1025        |
| 9.1.1    | Grundlagen zur Programmierung.....   | 1025        |
| 9.1.1.1  | Betriebssystem und Anwenderprogramm.....   | 1025        |
| 9.1.1.2  | Bausteine im Anwenderprogramm.....   | 1026        |
| 9.1.1.3  | Bausteinaufrufe.....   | 1039        |
| 9.1.1.4  | Operanden verwenden und adressieren.....   | 1056        |
| 9.1.1.5  | Datentypen.....  | 1087        |
| 9.1.1.6  | Programmflusssteuerung.....  | 1180        |
| 9.1.2    | PLC-Variablen deklarieren.....   | 1186        |
| 9.1.2.1  | Übersicht über PLC-Variablen Tabellen.....   | 1186        |
| 9.1.2.2  | Aufbau der PLC-Variablen Tabellen.....   | 1187        |
| 9.1.2.3  | Regeln für PLC-Variablen.....  | 1188        |
| 9.1.2.4  | PLC-Variablen Tabellen anlegen und verwalten.....  | 1192        |
| 9.1.2.5  | PLC-Variablen deklarieren.....   | 1194        |
| 9.1.2.6  | PLC-Variablen für Ein- und Ausgänge in Strukturen zusammenfassen.....                              | 1198        |
| 9.1.2.7  | Symbolische Konstanten deklarieren.....  | 1202        |
| 9.1.2.8  | Eigenschaften bearbeiten.....  | 1204        |
| 9.1.2.9  | PLC-Variablen beobachten.....  | 1206        |
| 9.1.2.10 | PLC-Variablen Tabellen bearbeiten.....   | 1207        |

|          |   |      |
|----------|---|------|
| 9.1.3    | Bausteine anlegen und verwalten.....                        | 1211 |
| 9.1.3.1  | Bausteine anlegen.....                                      | 1211 |
| 9.1.3.2  | Bausteineigenschaften festlegen.....                        | 1223 |
| 9.1.3.3  | Bausteine verwalten.....                                    | 1230 |
| 9.1.4    | Bausteine programmieren.....                                | 1235 |
| 9.1.4.1  | Programmiereditor.....                                      | 1235 |
| 9.1.4.2  | Codebausteine programmieren.....                            | 1257 |
| 9.1.4.3  | Datenbausteine programmieren.....                           | 1408 |
| 9.1.4.4  | PLC-Datentypen programmieren.....                           | 1436 |
| 9.1.4.5  | Externe Quelldateien verwenden.....                         | 1446 |
| 9.1.5    | PLC-Programm vergleichen.....                               | 1452 |
| 9.1.5.1  | Grundlagen zum Vergleichen von PLC-Programmen.....          | 1452 |
| 9.1.5.2  | Bausteine vergleichen.....                                  | 1456 |
| 9.1.5.3  | PLC-Variablen vergleichen.....                              | 1465 |
| 9.1.5.4  | PLC-Datentypen vergleichen.....                             | 1466 |
| 9.1.6    | Bausteine übersetzen und laden.....                         | 1468 |
| 9.1.6.1  | Bausteine übersetzen.....                                   | 1468 |
| 9.1.6.2  | Bausteine laden.....  | 1473 |
| 9.1.7    | Bausteine schützen.....                                     | 1484 |
| 9.1.7.1  | Schützen von Bausteinen.....                                | 1484 |
| 9.1.7.2  | Kopierschutz für Bausteine einrichten und entfernen.....    | 1487 |
| 9.1.7.3  | Know-how-Schutz für Bausteine einrichten.....               | 1488 |
| 9.1.7.4  | Know-how-geschützte Bausteine öffnen.....                   | 1489 |
| 9.1.7.5  | Know-how-geschützte Bausteine drucken.....                  | 1490 |
| 9.1.7.6  | Passwort ändern.....  | 1491 |
| 9.1.7.7  | Know-how-Schutz für Bausteine entfernen.....                | 1492 |
| 9.2      | Programminformationen anzeigen.....                         | 1493 |
| 9.2.1    | Übersicht der möglichen Programminformationen.....          | 1493 |
| 9.2.2    | Belegungsplan anzeigen.....                                 | 1494 |
| 9.2.2.1  | Einführung zum Belegungsplan.....                           | 1494 |
| 9.2.2.2  | Aufbau des Belegungsplans.....                              | 1495 |
| 9.2.2.3  | Symbole im Belegungsplan.....                               | 1496 |
| 9.2.2.4  | Belegungsplan anzeigen.....                                 | 1497 |
| 9.2.2.5  | Ansichtsoptionen für Belegungsplan einstellen.....          | 1498 |
| 9.2.2.6  | Filtermöglichkeiten im Belegungsplan.....                   | 1498 |
| 9.2.2.7  | Filter für Belegungsplan definieren.....                    | 1499 |
| 9.2.2.8  | Belegungsplan filtern.....                                  | 1500 |
| 9.2.2.9  | Remanente Speicherbereiche für Merker definieren.....       | 1501 |
| 9.2.2.10 | Anzeige der Remanenz für Merker aktivieren.....             | 1502 |
| 9.2.3    | Aufrufstruktur anzeigen.....                                | 1502 |
| 9.2.3.1  | Einführung zur Aufrufstruktur.....                          | 1502 |
| 9.2.3.2  | Symbole in der Aufrufstruktur.....                          | 1504 |
| 9.2.3.3  | Aufbau der Aufrufstruktur.....                              | 1505 |
| 9.2.3.4  | Aufrufstruktur anzeigen.....                                | 1506 |
| 9.2.3.5  | Ansichtsoptionen für die Aufrufstruktur einstellen.....     | 1507 |
| 9.2.3.6  | Einführung zur Konsistenzprüfung in der Aufrufstruktur..... | 1508 |
| 9.2.3.7  | Bausteinkonsistenz in der Aufrufstruktur prüfen.....        | 1508 |
| 9.2.4    | Abhängigkeitsstruktur anzeigen.....                         | 1509 |
| 9.2.4.1  | Einführung zur Abhängigkeitsstruktur.....                   | 1509 |
| 9.2.4.2  | Aufbau der Abhängigkeitsstruktur.....                       | 1510 |
| 9.2.4.3  | Symbole in der Abhängigkeitsstruktur.....                   | 1511 |
| 9.2.4.4  | Abhängigkeitsstruktur anzeigen.....                         | 1512 |



|          |  |      |
|----------|--|------|
| 9.2.4.5  | Ansichtsoptionen für die Abhängigkeitsstruktur einstellen.....     | 1512 |
| 9.2.4.6  | Einführung zur Konsistenzprüfung in der Abhängigkeitsstruktur..... | 1513 |
| 9.2.4.7  | Bausteinkonsistenz in der Abhängigkeitsstruktur prüfen.....        | 1514 |
| 9.2.5    | Speicherauslastung der CPU anzeigen.....                           | 1515 |
| 9.2.5.1  | Einführung zur Speicherauslastung.....                             | 1515 |
| 9.2.5.2  | Aufbau des Registers "Speicherauslastung".....                     | 1517 |
| 9.2.5.3  | Speicherauslastung anzeigen.....                                   | 1518 |
| 9.2.5.4  | Maximal verfügbaren Ladespeicher auswählen.....                    | 1519 |
| 9.3      | Querverweise anzeigen.....   | 1519 |
| 9.3.1    | Allgemeine Informationen zu Querverweisen.....                     | 1519 |
| 9.3.2    | Aufbau der Querverweisliste.....                                   | 1520 |
| 9.3.3    | Querverweisliste anzeigen.....                                     | 1522 |
| 9.3.4    | Querverweise im Inspektorfenster anzeigen.....                     | 1523 |
| 9.4      | Anwenderprogramm testen.....                                       | 1524 |
| 9.4.1    | Grundlagen zum Testen des Anwenderprogramms.....                   | 1524 |
| 9.4.2    | Testen mit Programmstatus.....                                     | 1525 |
| 9.4.2.1  | Einführung zum Testen mit Programmstatus.....                      | 1525 |
| 9.4.2.2  | Test mit Programmstatus ein-/ausschalten.....                      | 1526 |
| 9.4.2.3  | Bausteine während des Programmtests bearbeiten.....                | 1528 |
| 9.4.2.4  | Variablen im Programmstatus steuern.....                           | 1528 |
| 9.4.2.5  | Anzeigeformate im Programmstatus umstellen.....                    | 1529 |
| 9.4.2.6  | Beispiele für die Programmstatusanzeige.....                       | 1530 |
| 9.4.3    | Testen mit der Beobachtungstabelle.....                            | 1533 |
| 9.4.3.1  | Einführung zum Testen mit der Beobachtungstabelle.....             | 1533 |
| 9.4.3.2  | Aufbau der Beobachtungstabelle.....                                | 1534 |
| 9.4.3.3  | Basismodus und Erweiterter Modus in der Beobachtungstabelle.....   | 1535 |
| 9.4.3.4  | Symbole in der Beobachtungstabelle.....                            | 1536 |
| 9.4.3.5  | Beobachtungstabellen erstellen und bearbeiten.....                 | 1537 |
| 9.4.3.6  | Variablen in die Beobachtungstabelle eingeben.....                 | 1539 |
| 9.4.3.7  | Variablen in der Beobachtungstabelle beobachten.....               | 1546 |
| 9.4.3.8  | Variablen in der Beobachtungstabelle steuern.....                  | 1552 |
| 9.4.4    | Testen mit der Forcetabelle.....                                   | 1560 |
| 9.4.4.1  | Einführung zum Testen mit der Forcetabelle.....                    | 1560 |
| 9.4.4.2  | Sicherheitsvorkehrungen beim Forcen von Variablen.....             | 1562 |
| 9.4.4.3  | Aufbau der Forcetabelle.....                                       | 1562 |
| 9.4.4.4  | Basismodus und Erweiterter Modus in der Forcetabelle.....          | 1563 |
| 9.4.4.5  | Symbole in der Forcetabelle.....                                   | 1564 |
| 9.4.4.6  | Forcetabelle öffnen und bearbeiten.....                            | 1565 |
| 9.4.4.7  | Variablen in die Forcetabelle eingeben.....                        | 1566 |
| 9.4.4.8  | Variablen in der Forcetabelle beobachten.....                      | 1572 |
| 9.4.4.9  | Variablen in der Forcetabelle forcen.....                          | 1576 |
| 9.4.4.10 | Forcen von Variablen beenden.....                                  | 1583 |
| 9.5      | Übergreifende Projektfunktionen nutzen.....                        | 1586 |
| 9.5.1    | Importieren und Exportieren.....                                   | 1586 |
| 9.5.1.1  | Grundlagen zum Importieren und Exportieren.....                    | 1586 |
| 9.5.1.2  | Format der Exportdatei.....  | 1587 |
| 9.5.1.3  | Exportieren von PLC-Variablen.....                                 | 1587 |
| 9.5.1.4  | Importieren von PLC-Variablen.....                                 | 1588 |
| 9.6      | Programmierbeispiele.....  | 1589 |
| 9.6.1    | Programmierbeispiele KOP.....                                      | 1589 |

|          |  |      |
|----------|--|------|
| 9.6.1.1  | Beispiel für das Steuern eines Förderbands .....                               | 1589 |
| 9.6.1.2  | Beispiel für das Erfassen der Richtung eines Förderbands.....                  | 1591 |
| 9.6.1.3  | Beispiel für das Erfassen des Füllstands eines Lagerbereichs .....             | 1592 |
| 9.6.1.4  | Beispiel für das Kontrollieren der Raumtemperatur.....                         | 1595 |
| 9.6.2    | Programmierbeispiele FUP.....  | 1597 |
| 9.6.2.1  | Beispiel für das Steuern eines Förderbands .....                               | 1597 |
| 9.6.2.2  | Beispiel für das Erfassen der Richtung eines Förderbands.....                  | 1598 |
| 9.6.2.3  | Beispiel für das Erfassen des Füllstands eines Lagerbereichs .....             | 1599 |
| 9.6.2.4  | Beispiel für das Kontrollieren der Raumtemperatur.....                         | 1602 |
| 9.6.3    | Programmierbeispiele AWL.....  | 1604 |
| 9.6.3.1  | Beispiel für das Steuern eines Förderbands.....                                | 1604 |
| 9.6.3.2  | Beispiel für das Erfassen der Richtung eines Förderbands.....                  | 1606 |
| 9.6.3.3  | Beispiel für das Erfassen des Füllstands eines Lagerbereichs .....             | 1607 |
| 9.6.3.4  | Beispiel für das Berechnen einer Gleichung.....                                | 1609 |
| 9.6.3.5  | Beispiel für das Heizen eines Ofens .....                                      | 1610 |
| 9.6.3.6  | Beispiel für eine Schrittkette.....  | 1612 |
| 9.6.4    | Programmierbeispiele SCL.....  | 1614 |
| 9.6.4.1  | Beispiel für das Steuern eines Förderbands.....                                | 1614 |
| 9.6.4.2  | Beispiel für das Erfassen der Richtung eines Förderbands.....                  | 1616 |
| 9.6.4.3  | Beispiel für das Erfassen des Füllstands eines Lagerbereichs .....             | 1617 |
| 9.7      | Referenzen.....  | 1620 |
| 9.7.1    | Allgemeine Parameter der Anweisungen.....                                      | 1620 |
| 9.7.1.1  | Bedeutung von REQ, RET_VAL und BUSY bei asynchron arbeitenden Anweisungen..... | 1620 |
| 9.7.1.2  | Fehlerauswertung mit dem Ausgangsparameter RET_VAL.....                        | 1622 |
| 9.7.2    | Einfache Anweisungen.....  | 1626 |
| 9.7.2.1  | KOP.....   | 1626 |
| 9.7.2.2  | FUP.....   | 1884 |
| 9.7.2.3  | SCL.....   | 2155 |
| 9.7.3    | Erweiterte Anweisungen.....  | 2349 |
| 9.7.3.1  | Datum und Uhrzeit.....   | 2349 |
| 9.7.3.2  | String + Char.....   | 2364 |
| 9.7.3.3  | Prozessabbild.....   | 2385 |
| 9.7.3.4  | Dezentrale Peripherie.....   | 2392 |
| 9.7.3.5  | PROFenergy.....  | 2465 |
| 9.7.3.6  | Baugruppenparametrierung.....  | 2507 |
| 9.7.3.7  | Alarme.....  | 2517 |
| 9.7.3.8  | Meldungen.....   | 2547 |
| 9.7.3.9  | Diagnose.....  | 2552 |
| 9.7.3.10 | Puls.....  | 2583 |
| 9.7.3.11 | Rezepturen und Data Logging.....   | 2585 |
| 9.7.3.12 | Datenbausteinfunktionen.....   | 2611 |
| 9.7.3.13 | Adressierung.....  | 2621 |
| 9.7.4    | Technologie.....   | 2636 |
| 9.7.4.1  | S7-1200 Motion Control.....  | 2636 |
| 9.7.4.2  | Schnelle Zähler.....   | 2670 |
| 9.7.4.3  | PID Control.....   | 2672 |
| 9.7.5    | Kommunikation.....   | 2782 |
| 9.7.5.1  | Kommunikationsprozessor.....   | 2782 |
| 9.7.5.2  | S7-Kommunikation.....  | 2913 |
| 9.7.5.3  | Open User Communication.....   | 2931 |
| 9.7.5.4  | Webserver.....   | 3014 |
| 9.7.5.5  | TeleService.....   | 3016 |

|           |   |             |
|-----------|---|-------------|
| <b>10</b> | <b>Prozesse visualisieren (Basic)</b> .....                 | <b>3025</b> |
| 10.1      | Bilder erstellen.....                                       | 3025        |
| 10.1.1    | Grundlagen.....   | 3025        |
| 10.1.1.1  | Grundlagen zu Bildern .....                                 | 3025        |
| 10.1.1.2  | Geräteabhängiger Funktionsumfang von Bildern.....           | 3026        |
| 10.1.1.3  | Grundlagen.....   | 3028        |
| 10.1.1.4  | Mit Bildern arbeiten.....                                   | 3031        |
| 10.1.1.5  | Arbeiten mit Vorlagen.....                                  | 3034        |
| 10.1.2    | Arbeiten mit Objekten.....                                  | 3040        |
| 10.1.2.1  | Übersicht über die Objekte.....                             | 3040        |
| 10.1.2.2  | Bearbeitungsmöglichkeiten von Objekten.....                 | 3043        |
| 10.1.2.3  | Objekt einfügen.....  | 3044        |
| 10.1.2.4  | Objekt löschen .....  | 3046        |
| 10.1.2.5  | Objekt positionieren.....                                   | 3047        |
| 10.1.2.6  | Größe eines Objekts ändern.....                             | 3048        |
| 10.1.2.7  | Mehrere Objekte auswählen.....                              | 3050        |
| 10.1.2.8  | Objekte ausrichten.....                                     | 3052        |
| 10.1.2.9  | Objekt nach vorn oder hinten schieben.....                  | 3054        |
| 10.1.2.10 | Objekte außerhalb des Bildbereichs einblenden.....          | 3055        |
| 10.1.2.11 | Objekt drehen.....  | 3055        |
| 10.1.2.12 | Objekt spiegeln.....  | 3057        |
| 10.1.2.13 | Objekt gestalten.....                                       | 3058        |
| 10.1.2.14 | Mehrere Objekte eines Typs einfügen (stempeln).....         | 3058        |
| 10.1.2.15 | Position und Größe mehrerer Objekte verändern.....          | 3060        |
| 10.1.2.16 | Externe Grafiken.....                                       | 3060        |
| 10.1.2.17 | Externe Grafiken verwalten.....                             | 3062        |
| 10.1.2.18 | Externe Grafik in der Grafiksammlung hinterlegen.....       | 3063        |
| 10.1.2.19 | Arbeiten mit Objektgruppen.....                             | 3065        |
| 10.1.2.20 | Konfigurieren des Tastenzugriffs.....                       | 3072        |
| 10.1.2.21 | Beispiele.....  | 3075        |
| 10.1.3    | Arbeiten mit Textlisten und Grafiklisten.....               | 3078        |
| 10.1.3.1  | Arbeiten mit Textlisten.....                                | 3078        |
| 10.1.3.2  | Arbeiten mit Grafiklisten.....                              | 3086        |
| 10.1.4    | Dynamisieren von Bildern.....                               | 3096        |
| 10.1.4.1  | Grundlagen zur Dynamisierung.....                           | 3096        |
| 10.1.4.2  | Dynamisierung im Inspektorfenster.....                      | 3096        |
| 10.1.4.3  | Dynamisieren mit Animationen.....                           | 3098        |
| 10.1.4.4  | Dynamisieren mit Systemfunktionen.....                      | 3106        |
| 10.1.5    | Arbeiten mit Funktionstasten.....                           | 3108        |
| 10.1.5.1  | Arbeiten mit Funktionstasten .....                          | 3108        |
| 10.1.5.2  | Funktionstasten global belegen.....                         | 3111        |
| 10.1.5.3  | Funktionstasten lokal belegen.....                          | 3112        |
| 10.1.5.4  | Einer Funktionstaste eine Funktion zuweisen.....            | 3113        |
| 10.1.5.5  | Bedienberechtigung für eine Funktionstaste vergeben.....    | 3114        |
| 10.1.5.6  | Einer Funktionstaste eine Grafik zuweisen.....              | 3116        |
| 10.1.5.7  | LED-Variablen projektieren.....                             | 3118        |
| 10.1.5.8  | Beispiel: Funktionstasten zur Bildnavigation verwenden..... | 3120        |
| 10.1.6    | Arbeiten mit Ebenen.....                                    | 3121        |
| 10.1.6.1  | Grundlagen zum Arbeiten mit Ebenen.....                     | 3121        |
| 10.1.6.2  | Objekte auf Ebenen verschieben.....                         | 3122        |
| 10.1.6.3  | Aktive Ebene einstellen.....                                | 3123        |

|          |  |      |
|----------|--|------|
| 10.1.6.4 | Ebenen ein- und ausblenden.....                                | 3124 |
| 10.1.6.5 | Ebenen umbenennen.....   | 3125 |
| 10.1.7   | Anzeige- und Bedienobjekte.....                                | 3127 |
| 10.1.7.1 | Geräteabhängigkeit der Objekte.....                            | 3127 |
| 10.1.7.2 | Objekte.....   | 3128 |
| 10.1.8   | Bildnavigation projektieren.....                               | 3157 |
| 10.1.8.1 | Grundlagen zur Bildnavigation.....                             | 3157 |
| 10.1.8.2 | Schaltfläche mit Bildwechsel belegen.....                      | 3158 |
| 10.1.8.3 | Funktionstaste mit einem Bildwechsel belegen.....              | 3159 |
| 10.2     | Mit Variablen arbeiten.....                                    | 3160 |
| 10.2.1   | Grundlagen.....  | 3160 |
| 10.2.1.1 | Grundlagen zu Variablen.....                                   | 3160 |
| 10.2.1.2 | Übersicht über HMI-Variablen Tabellen.....                     | 3161 |
| 10.2.1.3 | Externe Variablen.....   | 3162 |
| 10.2.1.4 | Adressierung externer Variablen.....                           | 3164 |
| 10.2.1.5 | Interne Variablen.....   | 3166 |
| 10.2.2   | Arbeiten mit Variablen.....                                    | 3167 |
| 10.2.2.1 | Anlegen von Variablen.....                                     | 3167 |
| 10.2.2.2 | Bearbeiten von Variablen.....                                  | 3172 |
| 10.2.2.3 | Konfigurieren von Variablen.....                               | 3178 |
| 10.2.3   | Arbeiten mit Arrays.....                                       | 3196 |
| 10.2.3.1 | Grundlagen zu Arrays.....                                      | 3196 |
| 10.2.3.2 | Arrayvariable anlegen.....                                     | 3198 |
| 10.2.3.3 | Beispiele zu Arrays.....                                       | 3199 |
| 10.2.4   | Arbeiten mit Zyklen.....                                       | 3200 |
| 10.2.4.1 | Grundlagen zu Zyklen.....                                      | 3200 |
| 10.2.5   | Variablen darstellen.....                                      | 3201 |
| 10.2.5.1 | Ausgeben von Variablenwerten in Bildern.....                   | 3201 |
| 10.2.5.2 | Kurvenanzeige für Werte aus der Steuerung projektieren.....    | 3201 |
| 10.3     | Mit Meldungen arbeiten.....                                    | 3203 |
| 10.3.1   | Grundlagen.....  | 3203 |
| 10.3.1.1 | Das Meldesystem in WinCC.....                                  | 3203 |
| 10.3.1.2 | Meldeverfahren.....  | 3205 |
| 10.3.1.3 | Meldezustände .....  | 3207 |
| 10.3.1.4 | Meldeklassen.....  | 3208 |
| 10.3.1.5 | Quittierung.....   | 3210 |
| 10.3.1.6 | Meldegruppen.....  | 3211 |
| 10.3.1.7 | Meldenummer.....   | 3212 |
| 10.3.2   | Arbeiten mit Meldungen.....                                    | 3213 |
| 10.3.2.1 | Bestandteile und Eigenschaften von Meldungen.....              | 3213 |
| 10.3.2.2 | Projektieren von Meldungen.....                                | 3214 |
| 10.3.2.3 | Projektieren der Ausgabe von Meldungen.....                    | 3228 |
| 10.3.2.4 | Quittieren von Meldungen.....                                  | 3234 |
| 10.3.3   | Meldungen in Runtime bedienen.....                             | 3237 |
| 10.3.3.1 | Meldungen in Runtime.....                                      | 3237 |
| 10.3.3.2 | Einfache Meldeanzeige, Einfaches Meldefenster in Runtime ..... | 3239 |
| 10.3.3.3 | Meldeindikator in Runtime.....                                 | 3241 |
| 10.3.3.4 | Meldungen quittieren.....                                      | 3241 |
| 10.3.4   | Referenz.....  | 3242 |
| 10.3.4.1 | Systemfunktionen für Meldungen.....                            | 3242 |
| 10.3.4.2 | Systemmeldungen.....   | 3243 |

|          |  |      |
|----------|--|------|
| 10.3.5   | Systemdiagnose projektieren.....                               | 3268 |
| 10.3.5.1 | Grundlagen der Systemdiagnose.....                             | 3268 |
| 10.3.5.2 | Ansichten in der Systemdiagnose.....                           | 3268 |
| 10.3.5.3 | System-Diagnoseanzeige projektieren.....                       | 3271 |
| 10.4     | Mit Rezepturen arbeiten.....                                   | 3272 |
| 10.4.1   | Grundlagen.....  | 3272 |
| 10.4.1.1 | Definition und Einsatzgebiete.....                             | 3272 |
| 10.4.1.2 | Beispiele zum Einsatz von Rezepturen.....                      | 3274 |
| 10.4.1.3 | Aufbau von Rezepturen.....                                     | 3274 |
| 10.4.1.4 | Anzeige der Rezepturen.....                                    | 3276 |
| 10.4.1.5 | Datenfluss bei Rezepturen.....                                 | 3276 |
| 10.4.1.6 | Synchronisation von Rezepturdatensätzen mit der Steuerung..... | 3277 |
| 10.4.2   | Elemente und Grundeinstellungen.....                           | 3279 |
| 10.4.2.1 | Editor "Rezepturen".....                                       | 3279 |
| 10.4.3   | Anzeige und Bearbeitung von Rezepturen in Runtime.....         | 3282 |
| 10.4.3.1 | Einfache Rezepturanzeige.....                                  | 3282 |
| 10.4.3.2 | Verhalten der Rezepturanzeige in Runtime.....                  | 3283 |
| 10.4.4   | Rezepturen projektieren.....                                   | 3284 |
| 10.4.4.1 | Allgemeine Vorgehensweise beim Projektieren.....               | 3284 |
| 10.4.4.2 | Rezepturen anlegen und bearbeiten.....                         | 3285 |
| 10.4.4.3 | Anzeige der Rezeptur projektieren.....                         | 3293 |
| 10.4.5   | Rezepturen in Runtime bedienen.....                            | 3296 |
| 10.4.5.1 | Einfache Rezepturanzeige bedienen.....                         | 3296 |
| 10.4.6   | Beispiel.....  | 3302 |
| 10.4.6.1 | Beispiel für das Erstellen einer Rezeptur.....                 | 3302 |
| 10.5     | Eine Benutzerverwaltung projektieren.....                      | 3304 |
| 10.5.1   | Einsatzgebiet der Benutzerverwaltung.....                      | 3304 |
| 10.5.2   | Aufbau der Benutzerverwaltung.....                             | 3305 |
| 10.5.3   | Grundlagen.....  | 3306 |
| 10.5.3.1 | Benutzer.....  | 3306 |
| 10.5.3.2 | Arbeitsbereich Benutzer.....                                   | 3307 |
| 10.5.3.3 | Benutzergruppen.....   | 3308 |
| 10.5.3.4 | Arbeitsbereich Benutzergruppen.....                            | 3308 |
| 10.5.3.5 | Einstellungen für die Benutzerverwaltung.....                  | 3309 |
| 10.5.4   | Benutzerverwaltung aufbauen.....                               | 3311 |
| 10.5.4.1 | Grundlagen zur Benutzerverwaltung.....                         | 3311 |
| 10.5.4.2 | Benutzer für Runtime administrieren.....                       | 3312 |
| 10.5.4.3 | Benutzer in Runtime administrieren.....                        | 3319 |
| 10.5.4.4 | Zugriffsschutz projektieren.....                               | 3327 |
| 10.5.5   | Referenz.....  | 3328 |
| 10.5.5.1 | Objekte mit Zugriffsschutz.....                                | 3328 |
| 10.5.5.2 | Vordefinierte Benutzergruppen und Berechtigungen.....          | 3328 |
| 10.5.6   | Beispiele.....   | 3329 |
| 10.5.6.1 | Beispiel: Schaltfläche mit Anmeldedialog projektieren.....     | 3329 |
| 10.5.6.2 | Beispiel: Anmelden und Abmelden protokollieren.....            | 3330 |
| 10.5.6.3 | Beispiel einer Benutzerverwaltung.....                         | 3331 |
| 10.6     | Mit Systemfunktionen arbeiten.....                             | 3339 |
| 10.6.1   | Grundlagen.....  | 3339 |
| 10.6.1.1 | Systemfunktionen.....  | 3339 |
| 10.6.1.2 | Einsatz von Systemfunktionen.....                              | 3341 |
| 10.6.2   | Arbeiten mit Funktionslisten.....                              | 3341 |

|          |   |      |
|----------|---|------|
| 10.6.2.1 | Grundlagen zur Funktionsliste.....                                | 3341 |
| 10.6.2.2 | Eigenschaften einer Funktionsliste.....                           | 3342 |
| 10.6.2.3 | Funktionsliste projektieren.....                                  | 3343 |
| 10.6.2.4 | Funktionsliste bearbeiten.....                                    | 3345 |
| 10.6.2.5 | Abarbeitung der Funktionsliste in Runtime .....                   | 3346 |
| 10.6.3   | Beispiel.....   | 3346 |
| 10.6.3.1 | Wechsel der Betriebsart am Bediengerät mit aktueller Anzeige..... | 3346 |
| 10.6.4   | Referenz.....   | 3349 |
| 10.6.4.1 | Funktionsliste.....   | 3349 |
| 10.6.4.2 | Ereignisse.....   | 3415 |
| 10.7     | Aufgaben planen.....  | 3432 |
| 10.7.1   | Einsatzgebiet des Aufgabenplaners.....                            | 3432 |
| 10.7.2   | Arbeiten mit Aufgaben und Triggern.....                           | 3434 |
| 10.7.3   | Grundlagen.....   | 3435 |
| 10.7.3.1 | Arbeitsbereich des Editors Aufgabenplaner.....                    | 3435 |
| 10.7.3.2 | Funktionsliste.....   | 3436 |
| 10.7.3.3 | Trigger.....  | 3437 |
| 10.7.3.4 | Aufgabe planen mit Ereignis-Trigger.....                          | 3437 |
| 10.7.4   | Beispiele.....  | 3439 |
| 10.7.4.1 | Beispiel: Benutzer bei Benutzerwechsel aktualisieren.....         | 3439 |
| 10.8     | Mit Steuerungen kommunizieren.....                                | 3441 |
| 10.8.1   | Grundlagen zur Kommunikation.....                                 | 3441 |
| 10.8.1.1 | Kommunikation zwischen Geräten.....                               | 3441 |
| 10.8.1.2 | Geräte und Netze im Automatisierungssystem.....                   | 3442 |
| 10.8.1.3 | Datenaustausch über Variablen.....                                | 3447 |
| 10.8.1.4 | Datenaustausch über Bereichszeiger.....                           | 3448 |
| 10.8.1.5 | Kommunikationstreiber.....  | 3448 |
| 10.8.2   | Editoren für die Kommunikation.....                               | 3449 |
| 10.8.2.1 | Editor "Geräte & Netze".....                                      | 3449 |
| 10.8.2.2 | Netzsicht.....  | 3450 |
| 10.8.2.3 | Netzwerkdaten.....  | 3453 |
| 10.8.2.4 | Diagnose von Online-Verbindungen.....                             | 3455 |
| 10.8.2.5 | Gerätesicht.....  | 3456 |
| 10.8.2.6 | Topologiesicht.....   | 3459 |
| 10.8.2.7 | Inspektorfenster.....   | 3461 |
| 10.8.2.8 | Hardware-Katalog.....   | 3463 |
| 10.8.2.9 | Informationen zu Hardware-Komponenten.....                        | 3465 |
| 10.8.3   | Netze und Verbindungen.....                                       | 3467 |
| 10.8.3.1 | SIMATIC Kommunikationsnetze.....                                  | 3467 |
| 10.8.3.2 | Netze und Verbindungen projektieren.....                          | 3471 |
| 10.8.4   | Datenaustausch.....   | 3480 |
| 10.8.4.1 | Datenaustausch über Variablen.....                                | 3480 |
| 10.8.4.2 | Datenaustausch über Bereichszeiger.....                           | 3487 |
| 10.8.5   | Geräteabhängigkeit.....   | 3494 |
| 10.8.5.1 | Basic Panel.....  | 3494 |
| 10.8.6   | Mit SIMATIC S7 1500 kommunizieren.....                            | 3497 |
| 10.8.6.1 | Kommunikation mit SIMATIC S7 1500.....                            | 3497 |
| 10.8.6.2 | Kommunikation über PROFINET.....                                  | 3498 |
| 10.8.6.3 | Kommunikation über PROFIBUS.....                                  | 3511 |
| 10.8.6.4 | Datenaustausch.....   | 3520 |
| 10.8.6.5 | Leistungsmerkmale der Kommunikation.....                          | 3540 |

|           |   |      |
|-----------|---|------|
| 10.8.6.6  | Verbindungen im Editor "Verbindungen" projektieren..... | 3541 |
| 10.8.7    | Mit SIMATIC S7 1200 kommunizieren.....                  | 3546 |
| 10.8.7.1  | Kommunikation mit SIMATIC S7 1200.....                  | 3546 |
| 10.8.7.2  | Kommunikation über PROFINET.....                        | 3547 |
| 10.8.7.3  | Kommunikation über PROFIBUS.....                        | 3562 |
| 10.8.7.4  | Datenaustausch .....                                    | 3571 |
| 10.8.7.5  | Leistungsmerkmale der Kommunikation.....                | 3591 |
| 10.8.7.6  | Verbindungen im Editor "Verbindungen" anlegen.....      | 3593 |
| 10.8.8    | Mit SIMATIC S7 300/400 kommunizieren.....               | 3601 |
| 10.8.8.1  | Kommunikation mit SIMATIC S7 300/400.....               | 3601 |
| 10.8.8.2  | Kommunikation über PROFINET.....                        | 3602 |
| 10.8.8.3  | Kommunikation über PROFIBUS.....                        | 3615 |
| 10.8.8.4  | Kommunikation über MPI.....                             | 3623 |
| 10.8.8.5  | Datenaustausch.....                                     | 3631 |
| 10.8.8.6  | Leistungsmerkmale der Kommunikation.....                | 3650 |
| 10.8.8.7  | Verbindungen im Editor "Verbindungen" anlegen.....      | 3650 |
| 10.8.9    | Mit SIMATIC S7 200 kommunizieren.....                   | 3663 |
| 10.8.9.1  | Kommunikation mit SIMATIC S7 200.....                   | 3663 |
| 10.8.9.2  | Verbindung mit SIMATIC S7 200 anlegen.....              | 3663 |
| 10.8.9.3  | Parameter für die Verbindung.....                       | 3665 |
| 10.8.9.4  | Datenaustausch .....                                    | 3673 |
| 10.8.9.5  | Leistungsmerkmale der Kommunikation.....                | 3692 |
| 10.8.10   | Mit SIMATIC LOGO! kommunizieren.....                    | 3692 |
| 10.8.10.1 | Kommunikation mit SIMATIC LOGO!.....                    | 3692 |
| 10.8.10.2 | Verbindung mit SIMATIC LOGO! anlegen.....               | 3693 |
| 10.8.10.3 | Parameter für die Verbindung.....                       | 3695 |
| 10.8.10.4 | Datenaustausch.....                                     | 3699 |
| 10.8.10.5 | Leistungsmerkmale der Kommunikation.....                | 3704 |
| 10.8.11   | Kommunikation mit anderen Steuerungen.....              | 3704 |
| 10.8.11.1 | Kommunikation mit anderen Steuerungen.....              | 3704 |
| 10.8.11.2 | Besonderheiten bei der Projektierung.....               | 3705 |
| 10.8.11.3 | Kommunikationstreiber.....                              | 3706 |
| 10.8.11.4 | Datenaustausch über Bereichszeiger.....                 | 3821 |
| 10.9      | Übergreifende Funktionen nutzen.....                    | 3838 |
| 10.9.1    | Grundlagen zum Bediengeräte-Assistenten.....            | 3838 |
| 10.9.2    | Arbeiten mit Bibliotheken.....                          | 3839 |
| 10.9.2.1  | Grundlagen zu Bibliotheken.....                         | 3839 |
| 10.9.2.2  | Kopiervorlagen und Typen.....                           | 3841 |
| 10.9.2.3  | Bibliotheken in WinCC.....                              | 3842 |
| 10.9.2.4  | Bibliotheksobjekte anzeigen.....                        | 3844 |
| 10.9.2.5  | Bibliotheksobjekte verwalten.....                       | 3847 |
| 10.9.2.6  | Objekt in einer Bibliothek hinterlegen.....             | 3848 |
| 10.9.2.7  | Bibliotheksobjekt einfügen.....                         | 3849 |
| 10.9.2.8  | Globale Bibliothek anlegen.....                         | 3849 |
| 10.9.2.9  | Globale Bibliothek speichern.....                       | 3850 |
| 10.9.2.10 | Globale Bibliothek öffnen.....                          | 3851 |
| 10.9.3    | Projektdateien importieren und exportieren.....         | 3852 |
| 10.9.3.1  | Import und Export von Projektdateien.....               | 3852 |
| 10.9.3.2  | Rezepturen importieren und exportieren.....             | 3854 |
| 10.9.3.3  | Meldungen importieren und exportieren.....              | 3858 |
| 10.9.3.4  | Variablen importieren und exportieren.....              | 3865 |
| 10.9.3.5  | Textlisten importieren und exportieren.....             | 3871 |

|           |   |      |
|-----------|---|------|
| 10.9.3.6  | Projekttexte importieren und exportieren.....                 | 3875 |
| 10.9.4    | Querverweise nutzen.....                                      | 3878 |
| 10.9.4.1  | Allgemeine Informationen zu Querverweisen.....                | 3878 |
| 10.9.4.2  | Querverweisliste anzeigen.....                                | 3878 |
| 10.9.4.3  | Aufbau der Querverweisliste.....                              | 3879 |
| 10.9.4.4  | Querverweise im Inspektorfenster anzeigen.....                | 3880 |
| 10.9.5    | Sprachen verwalten.....                                       | 3882 |
| 10.9.5.1  | Sprachbegriffe in WinCC.....                                  | 3882 |
| 10.9.5.2  | Einstellungen für Sprachen im Betriebssystem.....             | 3883 |
| 10.9.5.3  | Einstellungen im Betriebssystem für asiatische Sprachen.....  | 3884 |
| 10.9.5.4  | Projektsprachen einstellen.....                               | 3885 |
| 10.9.5.5  | Ein Projekt in mehreren Sprachen erstellen.....               | 3888 |
| 10.9.5.6  | Sprachabhängigen Grafiken einsetzen.....                      | 3896 |
| 10.9.5.7  | Sprachen in Runtime.....                                      | 3900 |
| 10.9.5.8  | Beispiel zur mehrsprachigen Projektierung.....                | 3907 |
| 10.9.6    | Gerät tauschen.....   | 3910 |
| 10.9.6.1  | Grundlagen.....   | 3910 |
| 10.9.6.2  | Geräteabhängige Funktionen.....                               | 3911 |
| 10.9.6.3  | Bilder an das neue Gerät anpassen.....                        | 3914 |
| 10.9.7    | Zwischen Bediengeräten und Editoren kopieren.....             | 3920 |
| 10.9.7.1  | Grundlagen.....   | 3920 |
| 10.9.7.2  | Kopieren und Einfügen.....                                    | 3922 |
| 10.9.7.3  | Kopieren zwischen unterschiedlichen RT- und ES-Versionen..... | 3926 |
| 10.9.8    | WinCC-Versionskompatibilität nutzen.....                      | 3927 |
| 10.9.8.1  | Grundlagen zur Versionskompatibilität.....                    | 3927 |
| 10.9.8.2  | Projekt einer WinCC-Vorgängerversion bearbeiten.....          | 3929 |
| 10.9.8.3  | Projekt hochrüsten.....                                       | 3930 |
| 10.9.8.4  | Wechsel zwischen Bediengeräteversionen.....                   | 3931 |
| 10.9.8.5  | Bediengeräteversion ändern.....                               | 3932 |
| 10.9.9    | Daten einer Speicherkarte anzeigen.....                       | 3933 |
| 10.9.9.1  | Grundlagen.....   | 3933 |
| 10.9.9.2  | Arbeiten mit Backups.....                                     | 3934 |
| 10.10     | Übersetzen und Laden.....                                     | 3936 |
| 10.10.1   | Übersetzen und Laden von Projekten.....                       | 3936 |
| 10.10.1.1 | Übersicht zum Übersetzen und Laden von Projekten.....         | 3936 |
| 10.10.1.2 | Projekt übersetzen.....                                       | 3938 |
| 10.10.1.3 | Laden von Projekten.....                                      | 3939 |
| 10.10.1.4 | Runtime-Start.....  | 3942 |
| 10.10.2   | Simulieren von Projekten.....                                 | 3943 |
| 10.10.2.1 | Grundlagen zur Simulation.....                                | 3943 |
| 10.10.2.2 | Projekt simulieren.....                                       | 3944 |
| 10.10.2.3 | Arbeiten mit dem Variablen-Simulator.....                     | 3946 |
| 10.10.2.4 | Einschränkungen bei der Simulation.....                       | 3948 |
| 10.10.3   | Wartung des Bediengeräts.....                                 | 3949 |
| 10.10.3.1 | Übersicht zur Wartung des Bediengeräts (Basic Panels).....    | 3949 |
| 10.10.3.2 | ProSave.....  | 3950 |
| 10.10.3.3 | Datensicherung des Bediengeräts.....                          | 3950 |
| 10.10.3.4 | Daten des Bediengeräts sichern und wiederherstellen.....      | 3951 |
| 10.10.3.5 | Aktualisierung des Betriebssystems.....                       | 3952 |
| 10.10.3.6 | Betriebssystem des Bediengeräts aktualisieren.....            | 3953 |
| 10.10.4   | Referenz.....   | 3955 |
| 10.10.4.1 | Fehlermeldungen beim Laden von Projekten.....                 | 3955 |



|           |   |             |
|-----------|---|-------------|
| 10.10.4.2 | Anpassen des Projekts für ein anderes Bediengerät.....            | 3956        |
| 10.10.4.3 | Herstellen der Verbindung zum Bediengerät.....                    | 3958        |
| 10.11     | In Runtime bedienen.....  | 3959        |
| 10.11.1   | Grundlagen.....   | 3959        |
| 10.11.1.1 | Überblick.....  | 3959        |
| 10.11.1.2 | Variablen in Runtime.....   | 3960        |
| 10.11.1.3 | Systemfunktionen in Runtime.....                                  | 3960        |
| 10.11.2   | Projekte in Betrieb nehmen.....                                   | 3961        |
| 10.11.2.1 | Einstellungen der Runtime-Software .....                          | 3961        |
| 10.11.2.2 | Projekt laden.....  | 3962        |
| 10.11.2.3 | Runtime an der Engineering-Station starten.....                   | 3963        |
| 10.11.2.4 | Runtime auf dem Bediengerät starten.....                          | 3964        |
| 10.11.2.5 | Projekt testen.....   | 3965        |
| 10.11.2.6 | Projekt beenden.....  | 3966        |
| 10.11.2.7 | Daten des Bediengeräts sichern und wiederherstellen.....          | 3966        |
| 10.11.3   | Sprachen in Runtime.....  | 3967        |
| 10.11.3.1 | Sprachen in Runtime.....  | 3967        |
| 10.11.3.2 | Runtime-Sprache einstellen.....                                   | 3968        |
| 10.11.3.3 | Schriftart für eine Runtime-Sprache einstellen.....               | 3969        |
| 10.11.3.4 | Sprachumschaltung projektieren.....                               | 3970        |
| 10.11.3.5 | Besonderheiten asiatischer und östlicher Sprachen in Runtime..... | 3971        |
| 10.11.4   | Projekt bedienen.....   | 3972        |
| 10.11.4.1 | Grundlagen.....   | 3972        |
| 10.11.4.2 | Objekte bedienen.....   | 3980        |
| 10.11.4.3 | Sicherheit im Projekt.....  | 3992        |
| 10.11.4.4 | Meldungen bedienen.....   | 4000        |
| 10.11.4.5 | Rezepturen bedienen.....  | 4008        |
| 10.12     | Leistungsmerkmale.....  | 4019        |
| 10.12.1   | Engineering System.....   | 4019        |
| 10.12.2   | Basic Panel.....  | 4020        |
| 10.12.3   | Allgemeine technische Daten.....                                  | 4024        |
| 10.12.3.1 | Empfohlene Drucker.....   | 4024        |
| 10.12.3.2 | Speicherplatzbedarf von Rezepturen für Basic Panels.....          | 4025        |
| 10.13     | Umstieg auf WinCC V12.....  | 4026        |
| 10.13.1   | Übersicht zum Umstieg auf WinCC V12.....                          | 4026        |
| 10.13.2   | Bibliotheken.....   | 4027        |
| 10.13.3   | Bilder und Vorlagen.....  | 4029        |
| <b>11</b> | <b>Technologiefunktionen einsetzen.....</b>                       | <b>4033</b> |
| 11.1      | PID-Regelung.....   | 4033        |
| 11.1.1    | Grundlagen zum Regeln.....  | 4033        |
| 11.1.1.1  | Regelkreis und Stellglieder.....                                  | 4033        |
| 11.1.1.2  | Regelstrecken.....  | 4034        |
| 11.1.1.3  | Kennwerte der Regelstrecke.....                                   | 4036        |
| 11.1.1.4  | Impulsregler.....   | 4038        |
| 11.1.1.5  | Führungs- und Störverhalten.....                                  | 4042        |
| 11.1.1.6  | Regelverhalten bei unterschiedlichen Rückführstrukturen.....      | 4043        |
| 11.1.1.7  | Wahl der Reglerstruktur bei gegebener Regelstrecke.....           | 4050        |
| 11.1.1.8  | Einstellung der PID-Parameter.....                                | 4051        |
| 11.1.2    | Projektieren eines Software-Reglers.....                          | 4051        |
| 11.1.2.1  | Übersicht der Software-Regler.....                                | 4051        |

|           |   |      |
|-----------|---|------|
| 11.1.2.2  | Schritte für das Projektieren eines Software-Reglers.....                                     | 4053 |
| 11.1.2.3  | Technologieobjekte hinzufügen.....  | 4053 |
| 11.1.2.4  | Technologieobjekte konfigurieren.....   | 4054 |
| 11.1.2.5  | Anweisung im Anwenderprogramm aufrufen.....   | 4055 |
| 11.1.2.6  | Technologieobjekte in Gerät laden.....  | 4056 |
| 11.1.2.7  | Software-Regler in Betrieb nehmen.....  | 4058 |
| 11.1.2.8  | Optimierte PID-Parameter im Projekt speichern.....  | 4058 |
| 11.1.2.9  | Werte vergleichen.....  | 4059 |
| 11.1.2.10 | Instanz-DB eines Technologieobjekts anzeigen.....   | 4061 |
| 11.1.3    | PID_Compact einsetzen.....  | 4061 |
| 11.1.3.1  | Technologieobjekt PID_Compact.....  | 4061 |
| 11.1.3.2  | PID_Compact V2.....   | 4062 |
| 11.1.3.3  | PID_Compact V1.....   | 4078 |
| 11.1.4    | PID_3Step einsetzen.....  | 4094 |
| 11.1.4.1  | Technologieobjekt PID_3Step.....  | 4094 |
| 11.1.4.2  | PID_3Step V2.....   | 4095 |
| 11.1.4.3  | PID_3Step V1.....   | 4111 |
| 11.2      | S7-1200 Motion Control einsetzen.....   | 4127 |
| 11.2.1    | Einleitung.....   | 4127 |
| 11.2.1.1  | Motionfunktionalität der CPU S7-1200.....   | 4127 |
| 11.2.1.2  | Hardwarekomponenten für Motion Control.....   | 4128 |
| 11.2.2    | Grundlagen für das Arbeiten mit S7-1200 Motion Control.....                                   | 4131 |
| 11.2.2.1  | Für Motion Control relevante Ausgänge der CPU.....  | 4131 |
| 11.2.2.2  | Prinzip der Impulsschnittstelle.....  | 4133 |
| 11.2.2.3  | Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang...<br>4134 | 4134 |
| 11.2.2.4  | Hardware- und Software-Endschalter.....   | 4135 |
| 11.2.2.5  | Ruckbegrenzung.....   | 4136 |
| 11.2.2.6  | Referenzieren.....  | 4137 |
| 11.2.3    | Leitfaden zum Einsatz von Motion Control.....   | 4138 |
| 11.2.4    | Versionsübersicht.....  | 4139 |
| 11.2.5    | Technologieobjekt Achse.....  | 4141 |
| 11.2.5.1  | Einbindung des Technologieobjekts Achse.....  | 4141 |
| 11.2.5.2  | Werkzeuge des Technologieobjekts Achse.....   | 4143 |
| 11.2.5.3  | Technologieobjekt Achse hinzufügen.....   | 4145 |
| 11.2.5.4  | Technologieobjekt Achse konfigurieren.....  | 4146 |
| 11.2.6    | Technologieobjekt Auftragstabelle.....  | 4170 |
| 11.2.6.1  | Verwendung des Technologieobjekts Auftragstabelle.....  | 4170 |
| 11.2.6.2  | Werkzeuge des Technologieobjekts Auftragstabelle.....   | 4171 |
| 11.2.6.3  | Technologieobjekt Auftragstabelle hinzufügen.....   | 4171 |
| 11.2.6.4  | Technologieobjekt Auftragstabelle konfigurieren.....  | 4172 |
| 11.2.7    | Laden in CPU.....   | 4190 |
| 11.2.8    | Inbetriebnahme der Achse - Achssteuertafel.....   | 4191 |
| 11.2.9    | Programmieren.....  | 4194 |
| 11.2.9.1  | Übersicht über die Motion Control-Anweisungen.....  | 4194 |
| 11.2.9.2  | Anwenderprogramm erstellen.....   | 4195 |
| 11.2.9.3  | Programmierhinweise.....  | 4198 |
| 11.2.9.4  | Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart.....                         | 4200 |
| 11.2.9.5  | Verfolgung laufender Aufträge.....  | 4200 |
| 11.2.9.6  | Fehleranzeigen der Motion Control-Anweisungen.....  | 4213 |
| 11.2.10   | Achse - Diagnose.....   | 4214 |
| 11.2.10.1 | Status- und Fehlerbits.....   | 4214 |

|            |  |             |
|------------|--|-------------|
| 11.2.10.2  | Bewegungsstatus.....   | 4217        |
| 11.2.10.3  | Dynamikeinstellungen.....  | 4217        |
| 11.2.11    | Arbeiten mit Beobachtungstabellen.....                                   | 4218        |
| 11.2.12    | Anhang.....  | 4219        |
| 11.2.12.1  | Mehrere Achsen mit gleichem PTO einsetzen.....                           | 4219        |
| 11.2.12.2  | Mehrere Antriebe mit gleichem PTO einsetzen.....                         | 4223        |
| 11.2.12.3  | Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen..... | 4224        |
| 11.2.12.4  | Sonderfälle beim Einsatz von Software-Endschaltern.....                  | 4226        |
| 11.2.12.5  | Reduzierung der Geschwindigkeit bei kurzer Positionierdauer.....         | 4233        |
| 11.2.12.6  | Dynamische Anpassung der Start-/Stopp-Geschwindigkeit.....               | 4233        |
| 11.2.12.7  | Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0).....      | 4233        |
| 11.2.12.8  | Variable des Technologieobjekts Achse.....                               | 4242        |
| 11.2.12.9  | Variable des Technologieobjekts Auftragsstabelle.....                    | 4257        |
| 11.2.12.10 | Dokumentation von Funktionen aus älteren Versionen.....                  | 4258        |
| <b>12</b>  | <b>Online- und Diagnosefunktionen nutzen.....</b>                        | <b>4267</b> |
| 12.1       | Erreichbare Teilnehmer anzeigen.....                                     | 4267        |
| 12.2       | Gerätekonfiguration online ändern.....                                   | 4268        |
| 12.3       | Geräte online verbinden.....   | 4269        |
| 12.3.1     | Allgemeine Information zum Online-Modus.....                             | 4269        |
| 12.3.2     | Ansicht im Online-Modus.....   | 4271        |
| 12.3.3     | Online verbinden und Online-Verbindung trennen.....                      | 4272        |
| 12.3.4     | Online verbinden mit mehreren Geräten.....                               | 4274        |
| 12.3.5     | Online-Verbindungen von mehreren Geräten trennen.....                    | 4275        |
| 12.4       | Software und Hardware-Konfiguration eines Geräts sichern.....            | 4275        |
| 12.4.1     | Erstellen einer Sicherung für ein Gerät.....                             | 4275        |
| 12.4.2     | Sicherung der Projektierung eines Geräts.....                            | 4276        |
| 12.4.3     | Software und Hardware-Konfiguration eines Geräts wiederherstellen.....   | 4277        |
| 12.5       | PG/PC-Schnittstellen konfigurieren.....                                  | 4278        |
| 12.5.1     | Online-Zugänge.....  | 4278        |
| 12.5.2     | Grundlagen zur Parametrierung der PG/PC-Schnittstelle.....               | 4280        |
| 12.5.3     | Eigenschaften einer Schnittstelle anzeigen und ändern.....               | 4281        |
| 12.5.4     | Schnittstellen hinzufügen.....   | 4281        |
| 12.5.5     | Ethernet Schnittstelle parametrieren.....                                | 4282        |
| 12.5.5.1   | Parametrieren der Industrial Ethernet-Schnittstelle.....                 | 4282        |
| 12.5.5.2   | Betriebssystemparameter anzeigen.....                                    | 4283        |
| 12.5.5.3   | PG/PC-Schnittstelle mit einem Subnetz verbinden.....                     | 4284        |
| 12.5.5.4   | Ethernet-Schnittstelle parametrieren.....                                | 4284        |
| 12.5.5.5   | Zuweisen einer temporären IP-Adresse.....                                | 4285        |
| 12.5.5.6   | Temporäre IP-Adressen verwalten.....                                     | 4286        |
| 12.5.5.7   | TCP/IP Konfiguration zurücksetzen.....                                   | 4287        |
| 12.5.6     | MPI- und PROFIBUS-Schnittstelle parametrieren.....                       | 4287        |
| 12.5.6.1   | Parametrieren der MPI- und PROFIBUS-Schnittstelle.....                   | 4287        |
| 12.5.6.2   | MPI- oder PROFIBUS-Schnittstelle automatisch parametrieren.....          | 4288        |
| 12.5.6.3   | MPI-Schnittstelle parametrieren.....                                     | 4289        |
| 12.5.6.4   | PROFIBUS-Schnittstelle parametrieren.....                                | 4290        |
| 12.5.6.5   | Übersicht der Busparameter für PROFIBUS.....                             | 4292        |
| 12.5.6.6   | MPI- oder PROFIBUS Konfiguration zurücksetzen.....                       | 4294        |
| 12.6       | Trace- und Logikanalysatorfunktion nutzen.....                           | 4295        |
|            | Vorwort.....   | 4295        |

|          |   |      |
|----------|---|------|
| 12.6.1   | Beschreibung.....   | 4296 |
| 12.6.1.1 | Unterstützte Hardware.....  | 4296 |
| 12.6.1.2 | Messwertaufzeichnung mit der Tracefunktion.....                           | 4296 |
| 12.6.1.3 | Tracekonfiguration, Aufzeichnung und Messung.....                         | 4297 |
| 12.6.1.4 | Datenablage.....  | 4298 |
| 12.6.2   | Software-Oberfläche.....  | 4299 |
| 12.6.2.1 | Projektnavigator.....   | 4301 |
| 12.6.2.2 | Arbeitsbereich.....   | 4301 |
| 12.6.2.3 | Gerätespezifischer Bereich.....   | 4309 |
| 12.6.3   | Bedienen.....   | 4309 |
| 12.6.3.1 | Schnelleinstieg.....  | 4309 |
| 12.6.3.2 | Anwenden der Tracefunktion - Übersicht.....                               | 4314 |
| 12.6.3.3 | Trace-Editor aufrufen.....  | 4315 |
| 12.6.3.4 | Traceverwaltung.....  | 4315 |
| 12.6.3.5 | Signalabelle.....   | 4319 |
| 12.6.3.6 | Kurvendiagramm.....   | 4320 |
| 12.6.4   | Geräte.....   | 4322 |
| 12.6.4.1 | S7-1200/1500 CPUs.....  | 4322 |
| 12.7     | Fernverbindung mit TeleService aufbauen.....                              | 4335 |
| 12.7.1   | Grundlagen für das Arbeiten mit TeleService.....                          | 4335 |
| 12.7.1.1 | Einführung in TeleService.....  | 4335 |
| 12.7.1.2 | Funktionalität von TeleService.....                                       | 4336 |
| 12.7.1.3 | Telefonbuch bei TeleService.....  | 4336 |
| 12.7.2   | Arbeiten mit dem Telefonbuch.....   | 4337 |
| 12.7.2.1 | Grundlagen zum Arbeiten mit dem Telefonbuch.....                          | 4337 |
| 12.7.2.2 | Aufbau des Telefonbuchs.....  | 4338 |
| 12.7.2.3 | Symbole im Telefonbuch.....   | 4339 |
| 12.7.2.4 | Telefonbuch verwalten.....  | 4340 |
| 12.7.3   | Fernverbindung als Wählverbindung.....                                    | 4345 |
| 12.7.3.1 | Grundlagen zum Aufbauen einer Wählverbindung.....                         | 4345 |
| 12.7.3.2 | Telefonnetze und Modems.....  | 4347 |
| 12.7.3.3 | Zugriffsschutz bei Wählverbindungen.....                                  | 4350 |
| 12.7.3.4 | TS Adapter MPI.....   | 4355 |
| 12.7.3.5 | TS Adapter IE.....  | 4362 |
| 12.7.3.6 | Aufbauen einer Wählverbindung zu einer entfernten Anlage.....             | 4368 |
| 12.7.4   | Fernverbindung als VPN-Verbindung.....                                    | 4370 |
| 12.7.4.1 | Grundlagen zum Aufbauen einer VPN-Verbindung.....                         | 4370 |
| 12.7.4.2 | Grundlagen zu CA-Zertifikaten.....  | 4371 |
| 12.7.4.3 | CA-Zertifikat für VPN-Verbindung installieren.....                        | 4373 |
| 12.7.4.4 | CA-Zertifikat für VPN-Verbindung löschen.....                             | 4376 |
| 12.7.4.5 | Aufbauen einer VPN-Verbindung zu einer entfernten Anlage.....             | 4377 |
| 12.7.4.6 | TS Adapter IE Advanced.....   | 4379 |
| 12.7.5   | CPU-gesteuerte Fernverbindungen bei TeleService.....                      | 4383 |
| 12.7.5.1 | Übersicht über CPU-gesteuerte Fernverbindungen.....                       | 4383 |
| 12.7.5.2 | Verbindungsaufbau von und zu entfernten Anlagen (PG-AS-Fernkopplung)..... | 4384 |
| 12.7.5.3 | Datenaustausch zwischen entfernten Anlagen (AS-AS-Fernkopplung).....      | 4385 |
| 12.7.5.4 | Senden einer SMS von einer Anlage.....                                    | 4387 |
| 12.7.5.5 | Senden einer E-Mail von einer Anlage.....                                 | 4389 |
| 12.7.6   | Hinweise zur Fehlersuche.....   | 4391 |
| 12.7.6.1 | Allgemeine Hinweise für die Fehlersuche bei Modem-Problemen.....          | 4391 |
| 12.7.6.2 | Aufzeichnen einer Protokolldatei für das Modem.....                       | 4391 |
| 12.7.6.3 | Wählverbindung zum TS Adapter wird nicht aufgebaut.....                   | 4392 |

|           |   |             |
|-----------|---|-------------|
| 12.7.6.4  | Wählverbindung vom TS Adapter wird nicht aufgebaut.....         | 4394        |
| 12.7.6.5  | Modemverbindung wird unterbrochen.....                          | 4395        |
| 12.7.6.6  | Checkliste bei Schwierigkeiten mit dem Modem.....               | 4395        |
| 12.7.6.7  | Meldungen des Modems.....                                       | 4396        |
| 12.7.6.8  | Mögliche Fehlermeldungen bei VPN-Verbindungen.....              | 4397        |
| <b>13</b> | <b>Hardware-Dokumentation.....</b>                              | <b>4399</b> |
| 13.1      | Allgemeines zur Hardware-Dokumentation.....                     | 4399        |
| 13.2      | HMI.....  | 4399        |
| 13.2.1    | Basic Panels.....   | 4399        |
| 13.2.1.1  | Basic Panels.....   | 4399        |
| 13.2.2    | Panels.....   | 4399        |
| 13.2.2.1  | 70er Serie Panels.....  | 4399        |
| 13.2.2.2  | 170er Serie Panels.....   | 4399        |
| 13.2.2.3  | 270er Serie Panels.....   | 4400        |
| 13.2.3    | Comfort Panels.....   | 4400        |
| 13.2.3.1  | Comfort Panels.....   | 4400        |
| 13.2.4    | Multi Panels.....   | 4400        |
| 13.2.4.1  | 170er Serie.....  | 4400        |
| 13.2.4.2  | 270er Serie.....  | 4400        |
| 13.2.4.3  | 370er Serie.....  | 4400        |
| 13.2.5    | Mobile Panels.....  | 4400        |
| 13.2.5.1  | 170er Serie.....  | 4400        |
| 13.2.5.2  | 270er Serie.....  | 4401        |
| 13.2.6    | Key Panels.....   | 4401        |
| 13.2.6.1  | Key Panels.....   | 4401        |
| 13.2.6.2  | Push Button Panels.....   | 4401        |
| 13.2.7    | WinAC für Multi Panels.....                                     | 4401        |
| 13.2.7.1  | WinAC for Multi Panel.....                                      | 4401        |
| 13.3      | PLC.....  | 4401        |
| 13.3.1    | SIMATIC S7-1200.....  | 4401        |
| 13.3.1.1  | CPU.....  | 4401        |
| 13.3.1.2  | Signalboards (6ES7 2xx-xxx30-0XB0).....                         | 4403        |
| 13.3.1.3  | CB 1241 (6ES7 241-1CH30-1XB0).....                              | 4403        |
| 13.3.1.4  | BB 1297 (6ES7 297-0AX30-0XA0).....                              | 4403        |
| 13.3.1.5  | Digitaleingabemodule (6ES7 221-1Bx30-0XB0).....                 | 4403        |
| 13.3.1.6  | Digitalausgabemodule (6ES7 222-1xx30-0XB0).....                 | 4404        |
| 13.3.1.7  | Digitalein- und Digitalausgabemodule (6ES7 223-1xx30-0XB0)..... | 4404        |
| 13.3.1.8  | Analogeingabemodule (6ES7 231-xxx30-0XB0).....                  | 4404        |
| 13.3.1.9  | Analogausgabemodule (6ES7 234-4Hx30-0XB0).....                  | 4405        |
| 13.3.1.10 | Analogein- und Analogausgabemodul (6ES7 234-4HE30-0XB0).....    | 4405        |
| 13.3.1.11 | Kommunikationsmodule.....                                       | 4405        |
| 13.3.1.12 | Technologiemodule.....  | 4407        |
| 13.4      | Dezentrale Peripherie.....                                      | 4408        |
| 13.4.1    | ET 200MP.....   | 4408        |
| 13.4.1.1  | Interfacemodule.....  | 4408        |
| 13.4.2    | ET 200SP.....   | 4408        |
| 13.4.2.1  | Interfacemodule.....  | 4408        |
| 13.4.2.2  | Digitaleingabemodule.....                                       | 4411        |
| 13.4.2.3  | Digitalausgabemodule.....                                       | 4412        |
| 13.4.2.4  | Analogeingabemodule.....  | 4413        |

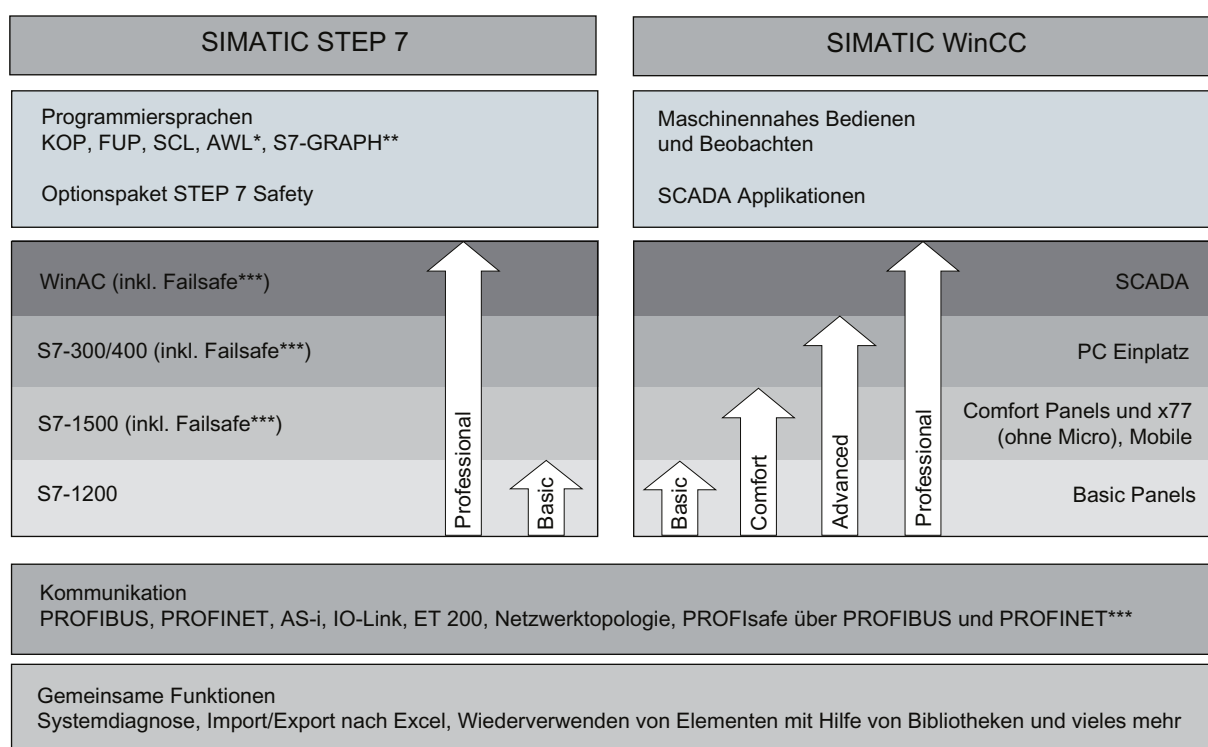
|                                    |             |
|------------------------------------|-------------|
| 13.4.2.5 Analogausgabemodule.....  | 4414        |
| 13.4.2.6 Kommunikationsmodule..... | 4415        |
| 13.4.2.7 Spezialmodule.....        | 4415        |
| 13.4.2.8 Technologiemodule.....    | 4415        |
| <b>Index.....</b>                  | <b>4417</b> |

## Systemübersicht STEP 7 und WinCC

### 1.1 Skalierung von STEP 7 und WinCC

#### Leistungsumfang der Produkte

Die folgende Grafik zeigt, welchen Leistungsumfang die einzelnen Produkte von STEP 7 und WinCC haben:



\* Nur bei Professional für S7-300/400/WinAC und S7-1500

\*\* Nur bei Professional für S7-300/400/WinAC

\*\*\* Mit installiertem Optionspaket "STEP 7 Safety"

#### STEP 7

STEP 7 (TIA-Portal) ist die Engineering-Software zur Projektierung der SIMATIC Steuerungsfamilien S7-1200, S7-1500, S7-300/400 und WinAC. STEP 7 (TIA-Portal) ist in zwei Editionen erhältlich, abhängig von den projektierbaren Steuerungsfamilien:

- STEP 7 Basic zur Projektierung der S7-1200
- STEP 7 Professional zur Projektierung von S7-1200, S7-1500, S7-300/400 und WinAC

## WinCC

WinCC (TIA-Portal) ist eine Engineering-Software zur Projektierung von SIMATIC Panels, SIMATIC Industrie PC und Standard PC mit der Visualisierungssoftware WinCC Runtime Advanced oder dem SCADA System WinCC Runtime Professional.

WinCC (TIA-Portal) ist erhältlich in vier Editionen, abhängig von den projektierbaren Bediensystemen:

- WinCC Basic zur Projektierung der Basic Panels  
WinCC Basic ist immer auch Bestandteil der Editionen STEP 7 Basic und STEP 7 Professional.
- WinCC Comfort zur Projektierung aller Panels (inklusive Comfort Panels, Mobile Panels)
- WinCC Advanced zur Projektierung aller Panels sowie PCs mit der Visualisierungssoftware WinCC Runtime Advanced  
WinCC Runtime Advanced ist eine Visualisierungssoftware für PC-basierte Einzelplatzsysteme. WinCC Runtime Advanced ist erhältlich mit Lizenzen für 128, 512, 2k, 4k sowie 8k PowerTags (Variablen mit Prozessanbindung).
- WinCC Professional zur Projektierung von Panels sowie PCs mit WinCC Runtime Advanced oder dem SCADA-System WinCC Runtime Professional. WinCC Professional ist in den Editionen WinCC Professional für 512 und 4096 Powertags sowie als "WinCC Professional max. Powertags" erhältlich.  
WinCC Runtime Professional ist ein SCADA-System zum Aufbau von Konfiguration mit Einzelplatzsystemen bis zu Mehrplatzsystemen mit Standard- oder Web-Clients. WinCC Runtime Professional ist erhältlich mit Lizenzen für 128, 512, 2k, 4k, 8k sowie 64k PowerTags (Variablen mit Prozessanbindung).

Mit WinCC (TIA-Portal) lassen sich auch SINUMERIK PC mit WinCC Runtime Advanced bzw. WinCC Runtime Professional und Bediengeräte mit SINUMERIK HMI Pro sl RT oder SINUMERIK Operate WinCC RT Basic projektieren.

## 1.2 Optionen für STEP 7 Engineering-System

### Weitere Produkte für STEP 7

Für Anwendungen mit erhöhten Sicherheitsanforderungen kann STEP 7 Professional um die Option STEP 7 Safety ergänzt werden.

Bei Einsatz der Option STEP 7 Safety können Sie fehlersichere Peripherie projektieren und Sicherheitsprogramme für F-CPU's in KOP und FUP programmieren.

## 1.3 Optionen für WinCC Engineering- und Runtime-Systeme

Die SIMATIC Panels sowie WinCC Runtime Advanced und WinCC Runtime Professional beinhalten alle wesentlichen Funktionen für das Bedienen und Beobachten von Maschinen oder Anlagen. Für einen erweiterten Aufgabenbereich kann die Funktionalität teilweise durch zusätzliche Optionen ergänzt werden.



### Optionen für Comfort Panels, Mobile Panels, Multi Panels

Für Comfort Panels, Mobile Panels und Multi Panels gibt es die folgenden Erweiterungsmöglichkeiten:

- WinCC SmartServer (Remote-Bedienung)
- WinCC Audit (Audit Trail und Elektronische Unterschrift für regulierte Applikationen)

---

#### Hinweis

Im Vergleich zu WinCC flexible 2008 sind Funktionen aus den Optionen WinCC flexible / Sm@rtService, WinCC flexible /Sm@rtAccess sowie die Option WinCC flexible /OPC-Server in die Basisfunktionalität eingeflossen.

---

### Optionen für WinCC Runtime Advanced

Für WinCC Runtime Advanced gibt es die folgenden Erweiterungsmöglichkeiten:

- WinCC SmartServer (Remote-Bedienung)
- WinCC Recipes (Rezeptursystem)
- WinCC Logging (Archivierung von Prozesswerten und Meldungen)
- WinCC Audit (Audit Trail für regulierte Applikationen)

---

#### Hinweis

Im Vergleich zu WinCC flexible 2008 sind Funktionen aus den Optionen WinCC flexible / Sm@rtService, WinCC flexible /Sm@rtAccess sowie die Option WinCC flexible /OPC-Server in die Basisfunktionalität eingeflossen.

---

### Optionen für WinCC Runtime Professional

Für WinCC Runtime Professional gibt es die folgenden Erweiterungsmöglichkeiten:

- WinCC Client (Standard-Client zum Aufbau von Mehrplatzsystemen)
- WinCC Server (Ergänzung der WinCC Runtime um Server-Funktionalität)
- WinCC Recipes (Rezeptursystem, bisher WinCC /UserArchives)
- WinCC WebNavigator (Bedienen und Beobachten über Web)
- WinCC DataMonitor (Anzeige und Auswertung von Prozesszuständen und historischen Daten)

---

#### Hinweis

Im Vergleich zu WinCC V7 sind Funktionen aus den Optionen WinCC /OPC-Server, WinCC / ConnectivityPack in die Basisfunktionalität eingeflossen. Ebenfalls in der Basisfunktionalität enthalten sind die Runtime API aus WinCC /ODK.

---

Über die Runtime-Optionen hinaus können WinCC Runtime Advanced und WinCC Runtime Professional durch kundenspezifische Controls erweitert werden. Zur Entwicklung der Controls ist die Option WinCC ControlDevelopment erforderlich.

# Liesmich

## 2.1 Allgemeine Hinweise

### 2.1.1 Allgemeine Hinweise

Die Hinweise der Liesmich sind Aussagen anderer Dokumente in der Verbindlichkeit übergeordnet.

Lesen Sie die folgenden Hinweise sorgfältig durch, da sie wichtige Informationen für die Installation und Nutzung enthalten. Lesen Sie diese Hinweise vor der Installation.

#### Funktionen für S7-1200 ab Firmwareversion V4

Im Informationssystem des TIA-Portals V12 SP1 sind Funktionen für S7-1200 ab Firmwareversion V4 dokumentiert, die in der Software noch nicht zur Verfügung stehen.

#### Online-Betrieb im Ruhezustand

Es wird empfohlen, die beiden Optionen "Ruhezustand" und "Energie sparen" im Online-Betrieb nicht zu nutzen, da es sonst zu Kommunikationsproblemen kommen kann. Passen Sie gegebenenfalls die Energieoptionen des Rechners an.

#### Erstmaliges Öffnen eines Projekts im TIA-Portal V12 SP1

Wenn Sie Ihr Projekt zum ersten Mal im TIA-Portal V12 SP1 öffnen, dann klicken Sie in der Projektansicht die verwendeten CPUs mit einem rechten Mausklick an und wählen Sie aus dem Kontextmenü "Übersetzen > Software (Bausteine komplett übersetzen)" aus.

#### Installation von neuen .Net Versionen oder .Net Service Packs

- Beenden Sie das TIA-Portal bevor Sie eine neue .Net Version oder ein neues .Net Service Pack auf Ihrem PG/PC installieren.
- Starten Sie das TIA-Portal erst nach der erfolgreichen Installation der neuen .Net Version oder des neuen .Net Service Packs.

#### Öffnen eines Projektes im TIA-Portal V11

Wenn Sie im TIA-Portal der Version 12 ein Projekt der Version V11 öffnen, können Sie dem Projekt Komponenten hinzufügen, die für die Version V11 im Rahmen eines Hardware Support Packages (HSP) nachgeliefert wurden. Das Projekt gilt dann immer noch als kompatibel mit TIA-Portal Version 11. Wenn Sie dieses Projekt nun in einem TIA-Portal Version 11 öffnen, diese TIA-Portal-Applikation aber nicht mit dem entsprechenden HSP nachgerüstet wurde, wird die entsprechende Komponente im Projekt nicht unterstützt.

### Hinweise zur Bedienung

- Wenn sich ein Projekt in der Liste der zuletzt verwendeten Projekte auf einem nicht verbundenen Netzlaufwerk befindet, kann es beim Öffnen des Menüs "Projekt" zu Verzögerungen kommen.
- Beim Einfügen einer CPU kann es zu einer längeren Wartezeit kommen, wenn gleichzeitig der Projekttext-Editor geöffnet ist. Insbesondere wenn es sich um die erste CPU in einem neu angelegten Projekt handelt, müssen Sie mit einer längeren Wartezeit rechnen. Um ein gutes Arbeitstempo zu gewährleisten, schließen Sie den Projekttext-Editor bevor Sie eine CPU einfügen.
- Unter Windows 7 kann es bei länger dauernden Funktionen (z. B. beim Laden der CPU) zur Anzeige der Meldung "Applikation reagiert nicht" kommen. In diesem Fall warten Sie bitte ab, bis die Funktion ordnungsgemäß beendet ist.
- Wenn Sie eine Microsoft-Maus mit IntelliPoint installiert haben, kann es zu Überlagerungen der Schaltflächen der Titelleiste kommen. Deinstallieren Sie in diesem Fall die IntelliPoint-Software von Microsoft.
- Die Aktivierung der Option "Virtual Desktop" bei NVIDIA-Grafikkarten kann zu Problemen führen. Deaktivieren Sie in diesem Fall den "nView virtual desktop manager" Ihres NVIDIA-Grafiktreibers.

### Verwendung des TIA-Portals über Remote Desktop

Die Verwendung des TIA-Portals über eine Remote-Desktop-Verbindung ist grundsätzlich möglich. Vermeiden Sie jedoch während einer Projektierung die Verbindung zum Desktop-Client zu trennen. In seltenen Fällen kann dies zu einer Blockierung der Software-Oberfläche führen.

Sollten Sie diese Blockierung beobachten, führen Sie bitte folgende Schritte am Desktop-Client aus.

1. Öffnen Sie den Windows Task-Manager und beenden Sie den Prozess "rdpclip.exe".
2. Geben Sie in der Eingabeaufforderung "rdpclip.exe" ein, um den Prozess neu zu starten.

Beachten Sie, dass der in diesem Moment in der Zwischenablage befindliche Inhalt dabei verloren geht. Sie können danach jedoch wie gewohnt weiter projektieren. Zur Sicherheit sollten Sie bei nächster Gelegenheit das TIA-Portal neu starten.

### Mehrfaches Öffnen des TIA-Portals

Wenn Sie mehrere Applikationen des TIA-Portals geöffnet haben und sich diese ständig wechselnd aktivieren, so können Sie um das Problem zu beheben kurzfristig in eine andere Applikation wechseln oder die Tastenkombination <ALT+Tab> drücken.

### Hinweise zu SD-Karten

Die SD-Karten sind von Siemens für den Einsatz auf S7-1200- und S7-1500-Baugruppen formatiert und eingerichtet. Dieses Format darf nicht überschrieben werden, da sonst die Karte von den Baugruppen nicht mehr akzeptiert wird. Das Formatieren mit Windows-Mitteln ist deshalb nicht zulässig.

### Verhalten bei offenem Force-Auftrag

Bitte beachten Sie, dass ein aktiver Force-Auftrag auch nach dem Laden eines neuen Projektes auf die SD-Karte bestehen bleibt. Daher sollten Sie vor dem Ziehen einer SD-Karte aus einer CPU zunächst den aktiven Force-Auftrag löschen, bevor Sie die Karte im PC mit einem neuen Projekt beschreiben. Wenn Sie eine SD-Karte mit unbekanntem Inhalt verwenden, sollten Sie die SD-Karte vor dem erneuten Laden formatieren.

### Warnungen zum Speicherplatz mit Windows XP (32 Bit)

Bei längerem Arbeiten mit Windows XP (32 Bit) kann es vorkommen, dass eine Speicherplatzwarnung erscheint, die Sie auffordert, das aktuelle Projekt zu speichern und das TIA-Portal neu zu starten. Das kann bei einem Betriebssystem mit integrierter Grafikhardware häufig vorkommen. Das Deaktivieren der Grafik-Hardwarebeschleunigung kann die Anzahl der Warnungen reduzieren. Die entsprechende Einstellung finden Sie, indem Sie auf den Desktop klicken und über die rechte Maustaste "Eigenschaften > Einstellungen > Erweitert > Problembehandlung" wählen. Schieben Sie in diesem Dialog den Schieber "Hardwarebeschleunigung" ganz nach links (Einstellung "Keine") und übernehmen Sie diese Einstellung.

### Probleme beim Herunterfahren von Windows XP

Stellen Sie bei Problemen mit dem Herunterfahren des Rechners außerdem sicher, dass das TIA Portal vollständig geschlossen ist:

1. Öffnen Sie über das Kontextmenü der Taskleiste den Task-Manager.
2. Wenn Sie im Register "Prozesse" den Prozess "Siemens.Automation.ObjectFrame.FileStorage.Server.exe" sehen, warten Sie, bis dieser Prozess beendet wird.
3. Danach können Sie den Rechner herunterfahren.

### Subnetzadressierung bei CP 1613 und CP 1623

CP 1613 und CP 1623 sind Kommunikationsbaugruppen mit Mikroprozessor. Zur sicheren Abwicklung von Kommunikationsverbindungen werden diese auf der Baugruppe bearbeitet. Für Diagnosezwecke (SNMP, DCP) wird der Protokollstack in Ihrem PC verwendet. Um beiden Protokollstacks (also CP 1613/23 Firmware und CP 1613/23 NDIS-Zugang) zu den gleichen Partnern zu ermöglichen, wird empfohlen, beide Stacks einer Baugruppe in das gleiche Subnetz zu legen.

### Editieren einer Geräte-IP-Adresse

Verwenden Sie beim Editieren einer Geräte-IP-Adresse nicht den Adressbereich von 192.168.x.241 bis 192.168.x.250. Dieser Adressbereich wird ggf. vom System automatisch einem PG zugewiesen. Dies gilt analog in Abhängigkeit der Subnetz-Maske für alle Netzklassen.

### Migration von Projekten mit dem TIA-Portal

Prüfen Sie nach der Migration von Hardware-Konfigurationen und Programmbausteinen aus früheren Automatisierungslösungen zuerst die Funktionsfähigkeit des migrierten Projektes, bevor Sie es im Produktivbetrieb verwenden.

### Arbeiten mit TeleService

Zum Liefereinsatz des TIA-Portals V12 SP1 steht der TS Adapter IE Advanced noch nicht zur Verfügung. Erst nach dem Liefereinsatz des TS Adapter IE Advanced können mit dem TIA-Portal V12 SP1 Fernverbindungen als VPN-Verbindungen projektiert werden.

### Arbeiten mit automatisch synchronisierten Netzlaufwerken

Durch automatische Synchronisation nach einer Netzwerkunterbrechung können durch Anwenderinteraktion aktuelle (lokale) Projektdaten als "Backup" auf dem Netzlaufwerk abgelegt werden und somit beim Öffnen des Projekts veraltete Projektdaten vom Netzwerklaufwerk geladen werden. Daher wird nicht empfohlen, TIA-Portal Projekte auf synchronisierten Netzlaufwerken abzulegen.

Falls dennoch auf synchronisierten Laufwerken gearbeitet wird, kann bei Netzwerkunterbrechungen lokal weitergearbeitet werden. In diesen Fällen ist unbedingt darauf zu achten, dass die TIA-Portal Applikation bei der Synchronisation der Daten geschlossen ist. Die Synchronisation selbst muss so erfolgen, dass die aktuellen (lokalen) Projektdaten die Projektdaten auf dem Netzlaufwerk ersetzen.

### Eingabe von Dezimalstellen

Bei bestimmten Windows-Spracheinstellungen kann es vorkommen, dass die Eingabe von Werten mit Komma nicht erkannt wird (Eingabe von "1,23" führt zu einem Fehler). Verwenden Sie stattdessen die internationale Schreibweise ("1.23").

### Zugriffsschutz für Memory Cards in USB-Kartenleser

Durch die Verbesserung der Sicherheitsmechanismen beim Online-Zugriff und Engineering von S7-1500-CPU's, wurde auch die Datenspeicherung auf Memory Cards verändert. Aus diesem Grund kann diese Version von STEP 7 beim Lesen von Projektdaten von Memory Cards, auf die über einen USB-Kartenleser zugegriffen wird, die Passwörter der konfigurierten Schutzstufe nicht auswerten. Das geänderte Verhalten betrifft die Memory Cards für CPU's der Baureihen S7-1200/1500. Verwenden Sie daher physikalische Schutzmechanismen, um kritische Projektdaten auf Memory Cards dieser Geräte zu schützen.

---

#### Hinweis

Diese Einschränkung steht in keinem Zusammenhang mit dem Online-Zugriff auf Geräte oder mit dem Know-how-Schutz von Programmbausteinen.

---

## Bildschirmdarstellung

Bei bestimmten Rechnerkonfigurationen mit Windows XP kann es vorkommen, dass nach längerem Arbeiten Teile der Oberfläche des TIA-Portals nicht mehr aktualisiert werden. Abhilfe kann hier ein Reduzieren der Grafik-Hardwarebeschleunigung schaffen. Die entsprechende Einstellung finden Sie, indem Sie auf den Desktop klicken und über die rechte Maustaste "Eigenschaften > Einstellungen > Erweitert > Problembehandlung" wählen. Schieben Sie in diesem Dialog den Schieber "Hardwarebeschleunigung" ganz nach links (Einstellung "Keine") und übernehmen Sie diese Einstellung.

## Informationen zum TIA-Portal im Online Support

Übersicht zu den wichtigsten technischen Informationen und Lösungen für das TIA-Portal im Industry Online Support.

Internet-Link: Auto-Hotspot

Alle Informationen zu Service und Support im Industry Online Support:

Internet-Link: Auto-Hotspot

Hier können Sie auch den Newsletter abonnieren, der Sie ständig mit den aktuellen Informationen zu Ihren Produkten versorgt.

## TIA-Portal starten

Beim Start des TIA-Portals versucht Windows die Certificate Revocation List (CRL) von "windowsupdate.com" zu aktualisieren.

Bei fehlendem Internetzugang und mehreren DNS-Servern kann es zu Zeitüberschreitungen kommen, die den Start des TIA-Portals verzögern.

## FAQs zum TIA-Portal

FAQs zum TIA-Portal finden Sie unter <http://support.automation.siemens.com>.

## 2.1.2 Hinweise zur Installation

### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

## Verwendung gleicher Versionen der TIA-Portal Produkte

Die TIA-Portal Produkte STEP 7 (inkl. PLCSIM), WinCC und Startdrive müssen auf die gleiche Version (V12 SP1) hochgerüstet werden. Sollten die Versionen unterschiedlich sein, lässt sich das TIA-Portal nicht mehr starten. Auch durch die Deinstallation eines Produktes lässt sich kein ablauffähiger Stand herstellen. Bitte stellen Sie sicher, dass die Service Packs dieser Produkte vorliegen, bevor Sie mit der Installation beginnen. Die Servicepacks können Sie im Internet

unter <http://support.automation.siemens.com> (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo2&aktprim=99&lang=de>) herunterladen.

### Zielverzeichnis der Installation

Verwenden Sie keine UNICODE-Zeichen (z. B. chinesische Schriftzeichen) im Installationspfad.

### Verwendung von Virenschernern

Während der Installation ist es notwendig, dass auf bereits installierte Dateien lesend und schreibend zugegriffen werden kann. Einige Virenscherner verhindern dies. Daher wird empfohlen, während der Installation von TIA-Portal Virenscherner zu deaktivieren und danach wieder zu aktivieren.

### Installation von STEP 7 Basic V12 und STEP 7 Professional V12 unter Windows XP mit türkischen Regions- und Sprachoptionen

Es kann zu einem Abbruch der Installation von STEP 7 Basic V12 und STEP 7 Professional V12 unter Windows XP kommen, wenn die Regions- und Sprachoptionen auf Türkisch eingestellt sind. Ändern Sie in diesem Fall die Regions- und Sprachoptionen von Türkisch auf Englisch oder auf Deutsch.

1. Öffnen Sie unter Windows die Systemsteuerung mit einem der folgenden Befehle:
  - "Start > Systemsteuerung" (Startmenü unter Windows XP)
  - "Start > Einstellungen > Systemsteuerung" (klassisches Startmenü)
2. Öffnen Sie die "Regions- und Sprachoptionen".
3. Wählen Sie das Register "Regionale Einstellungen".
4. Wählen Sie unter "Standards und Formate" aus der Klappliste die Sprache "Deutsch" oder "Englisch".
5. Klicken Sie anschließend auf die Schaltfläche "Übernehmen" und bestätigen Sie mit OK.
6. Damit die Einstellung übernommen wird, führen Sie einen Neustart Ihres PCs durch. Anschließend kann die Installation von STEP 7 Basic V12 und STEP 7 Professional V12 durchgeführt werden.
7. Nach der Installation können Sie die Regions- und Sprachoptionen wieder (wie unter Schritt 1 bis 4 beschrieben) auf Türkisch umstellen.

### Kompatibilität mit V11

In das Installationsverzeichnis unter `..\Portal V12\SampleProjects` wird ein leeres V11.0.2.5-Projekt mit dem Namen "TIA\_Portal\_Project\_V11.0.2.5.ap11" installiert, damit das TIA-Portal V12 im Kompatibilitätsmodus V11 geöffnet werden kann. Dieses Projekt muss in ein lokales Verzeichnis mit Vollzugriff kopiert werden, bevor es verwendet werden kann. Weitergehende Informationen finden Sie unter FAQ ID 66027369.



## Installation von Startdrive V12

Voraussetzung für die Installation von Startdrive V12 ist in jedem Fall die vorherige Installation von STEP 7 V12.

## Installation der SIMATIC USB Treiber unter Windows Server 2003 R2 StdE SP2

Auf dem Betriebssystem Windows Server 2003 R2 StdE SP2 kommt es zu einer Betriebssystemmeldung bzgl. der SIMATIC USB Treiber. Diese Meldung muss möglichst zeitnah zum Auftreten mit "Ja" quittiert werden. Die Meldung kann u. U. im Hintergrund stehen und somit nicht sofort sichtbar sein. Nach einer bestimmten Zeit fährt das Setup mit der nächsten Komponente fort. Die SIMATIC USB Treiber sind dann nicht installiert und können nicht verwendet werden.

## 2.2 STEP 7 Basic

### 2.2.1 Security-Hinweise

#### Aktualisierungen und Updates

Siemens bietet Automatisierungs- und Antriebsprodukte mit Industrial Security-Funktionen an, die den sicheren Betrieb der Anlage oder Maschine unterstützen. Sie sind ein wichtiger Baustein für ein ganzheitliches Industrial Security-Konzept. Die Produkte werden unter diesem Gesichtspunkt ständig weiterentwickelt. Es wird empfohlen sich regelmäßig über Aktualisierungen und Updates unserer Produkte zu informieren. Informationen und Newsletter hierzu finden Sie unter:

<http://support.automation.siemens.com> (<http://support.automation.siemens.com>)

Für den sicheren Betrieb einer Anlage oder Maschine ist es darüber hinaus notwendig, geeignete Schutzmaßnahmen (z.B. Zellschutzkonzept) zu ergreifen und die Automatisierungs- und Antriebskomponenten in ein ganzheitliches Industrial Security-Konzept der gesamten Anlage oder Maschine zu integrieren, das dem aktuellen Stand der Technik entspricht. Dabei sind auch eingesetzte Produkte von anderen Herstellern zu berücksichtigen. Weitergehende Informationen finden Sie unter:

<http://www.siemens.com/industrialsecurity> (<http://www.industry.siemens.com/topics/global/de/industrial-security/Seiten/Default.aspx>)

### Netzwerkeinstellungen

Die folgenden Tabellen zeigen die Netzwerkeinstellungen des jeweiligen Produkts, die Sie zur Analyse der Netzwerksicherheit und zur Konfiguration von externen Firewalls benötigen:

| STEP 7 Basic |             |                     |              |                  |   |
|--------------|-------------|---------------------|--------------|------------------|---|
| Name         | Port-Nummer | Transport-Protokoll | Richtung     | Funktion         | Beschreibung  |
| ALM          | 4410*       | TCP                 | In-/Outbound | Lizenz-Service   | Dieser Dienst stellt die komplette Funktionalität für Software-Lizenzen zur Verfügung und wird sowohl vom Automation License Manager als auch von allen lizenzrelevanten Softwareprodukten verwendet. |
| RFC 1006     | 102         | TCP                 | Outbound     | S7-Kommunikation | Kommunikation zum S7-Controller über Ethernet / PROFINET für Programmier- und Diagnosezwecke.   |
| DCP          | ---         | Ethernet            | Outbound     | PROFINET         | Das DCP-Protokoll (Discovery and basic Configuration Protocol) wird von PROFINET verwendet und stellt die Basisfunktionalität zum Auffinden und Konfigurieren von PROFINET-Geräten zur Verfügung.     |
| SNMP         | 161         | UDP                 | Outbound     | PROFINET         | Die SNMP Client-Funktionalität wird von STEP 7 verwendet, um Statusinformationen von PROFINET-Geräten auszulesen.   |

\* Standard-Port, der durch Anwenderkonfiguration geändert werden kann

| WinCC ES Basic (ohne Simulation) |             |                     |              |                     |   |
|----------------------------------|-------------|---------------------|--------------|---------------------|---|
| Name                             | Port-Nummer | Transport-Protokoll | Richtung     | Funktion            | Beschreibung  |
| ALM                              | 4410*       | TCP                 | In-/Outbound | Lizenz-Service      | Dieser Dienst stellt die komplette Funktionalität für Software-Lizenzen zur Verfügung und wird sowohl vom Automation License Manager als auch von allen lizenzrelevanten Softwareprodukten verwendet. |
| HMI Load                         | 1033        | TCP                 | Outbound     | HMI Load (RT Basic) | Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Basic Panels zu übertragen.   |

\* Standard-Port, der durch Anwenderkonfiguration geändert werden kann

| Simulation RT Basic |             |                     |          |                     |   |
|---------------------|-------------|---------------------|----------|---------------------|---|
| Name                | Port-Nummer | Transport-Protokoll | Richtung | Funktion            | Beschreibung  |
| HMI Load            | 1033        | TCP                 | Inbound  | HMI Load (RT Basic) | Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Basic Panels zu übertragen. |
| EtherNet/IP         | 44818       | TCP                 | Outbound | Ethernet/IP-Kanal   | Das Ethernet/IP-Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.                |
|                     | 2222        | UDP                 | Inbound  | Ethernet/IP-Kanal   | Das Ethernet/IP-Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.                |
| Modbus TCP          | 502         | TCP                 | Outbound | Modbus TCP-Kanal    | Das Modbus TCP-Protokoll wird für Verbindungen zu Schneider PLCs verwendet.                     |
| RFC 1006            | 102         | TCP                 | Outbound | S7-Kanal            | Kommunikation zum S7-Controller über Ethernet / PROFINET  |
| Mitsubishi MC       | 5002        | TCP                 | Outbound | Mitsubishi MC-Kanal | Das Mitsubishi-Protokoll wird für Verbindungen zu Mitsubishi PLCs verwendet.                    |

## 2.2.2 Hinweise zur Benutzung

### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

### Online-Betrieb

Der parallele Online-Betrieb von STEP 7 V5.5 oder älter und STEP 7 Basic V12 ist nicht freigegeben.

### Gleichzeitige Online-Verbindungen auf eine S7-1200-CPU

Es ist nicht möglich, gleichzeitig von mehreren TIA-Portal-Instanzen aus eine Online-Verbindung auf dieselbe S7-1200-CPU herzustellen.

### Baugruppen parametrieren und konfigurieren

Welche Baugruppen mit STEP 7 Basic V12 parametrieren und konfiguriert werden können, entnehmen Sie bitte <http://support.automation.siemens.com> (<http://support.automation.siemens.com/WW/view/de/28919804/133000>).

### Ziehen/Stecken der Memory Card

Führen Sie nach dem Ziehen oder Stecken einer Memory Card immer ein Umröten auf der CPU durch, um die CPU wieder in einen funktionsfähigen Zustand zu versetzen.

### Ethernet-Baugruppen ziehen und stecken

Wenn Ethernet-Baugruppen während des Betriebs gezogen und wieder gesteckt werden, dann müssen Sie den PC booten, da die Funktionalität "Erreichbare Teilnehmer" in STEP 7 oder NCM PC ansonsten nicht alle Teilnehmer anzeigt. Während der PC gebootet wird, müssen Ethernet-Baugruppen aktiviert sein.

### Hinweise zum Informationssystem

Folgende Funktion ist im Informationssystem bereits beschrieben, aber in STEP 7 Basic V12 SP1 nicht verfügbar:

- Laden von Hardware-Konfigurationen vom Zielsystem in das PG/PC.

### Projektdateien vergleichen

Die Vergleichsfunktionen (Online/Offline, Offline/Offline) berücksichtigen zur Zeit keine Hardware.

### Projektdateien mit TIA-Portal V11 und V12 laden (S7-1200)

Wenn Sie die Projektdateien einer S7-1200-CPU mit dem TIA-Portal V12 geladen haben, können Sie mit dem TIA-Portal V11 auf diese Daten nicht mehr zugreifen. Setzen Sie dazu die CPU zuerst auf die Werkseinstellungen zurück. Beachten Sie dabei die Informationen in der Online-Hilfe unter "Eine CPU auf Werkseinstellungen zurücksetzen".

### Freigabeausgang ENO aktivieren

Ab STEP 7 TIA-Portal V12.0 ist in den Programmiersprachen KOP und FUP der Freigabeausgang ENO per Voreinstellung deaktiviert und der Parameter ENO wird grau dargestellt. Bei Bedarf können Sie den Freigabeausgang aktivieren und damit gezielt steuern, bei welchen Anweisungen Sie eine Fehlerauswertung haben möchten.

Um den Freigabeausgang ENO einer Anweisung zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den Freigabeausgang ENO aktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO generieren".  
Für die Anweisung wird der ENO-Wert generiert und die Farbe des Parameters ENO wechselt von grau nach schwarz.  
Weitere Anweisungen werden mit dem Freigabeausgang eingefügt.

## Kompatibilität

Gerätekonfiguration und Programm einer S7-1200-CPU müssen grundsätzlich mit der gleichen STEP 7-Version konfiguriert werden. Im Normalfall stellt das TIA-Portal durch entsprechende Hinweise beim Laden ins Gerät sicher, dass es nicht zu Versionskonflikten kommt.

Bei S7-1200-CPU's mit der Firmwareversion V1.x ist diese automatische Überprüfung nicht möglich und daher vom Anwender sicherzustellen.

## 2.2.3 Geräte und Netze bearbeiten

### 2.2.3.1 Allgemeine Hinweise zu Geräten und Netzen

#### Inhalt

Derzeit liegen keine allgemeinen Hinweise zu Geräten und Netzen vor.

### 2.2.3.2 Verwendung von Modulen an der S7-1200

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

### Verwendung von Modulen an der S7-1200

Die folgend aufgeführten Module werden an der S7-1200 nicht unterstützt:

| Familie       | Baugruppe  | Bestellnummer  |
|---------------|------------|--|
| S7-300 FMs    | SM 338     | 6ES7 338-4BC01-0AB0  |
|               | FM 350-1   | 6ES7 350-1AH03-0AE0  |
|               | FM 350-2   | 6ES7 350-2AH00-0AE0, 6ES7 350-2AH01-0AE0   |
|               | FM 351     | 6ES7 351-1AH01-0AE0, 6ES7 351-1AH02-0AE0   |
|               | FM 352     | 6ES7 352-1AH02-0AE0  |
|               | FM 355 S   | 6ES7 355-1VH10-0AE0  |
|               | FM 355 C   | 6ES7 355-0VH10-0AE0  |
|               | FM 355-2 C | 6ES7 355-2CH00-0AE0  |
|               | FM 355-2 S | 6ES7 355-2SH00-0AE0  |
| S7-300 PtP-CP | CP 340     | 6ES7 340-1AH02-0AE0, 6ES7 340-1BH02-0AE0, 6ES7 340-1CH02-0AE0  |
|               | CP 341     | 6ES7 341-1AH01-0AE0, 6ES7 341-1AH02-0AE0, 6ES7 341-1BH01-0AE0, 6ES7 341-1BH02-0AE0, 6ES7 341-1CH01-0AE0, 6ES7 341-1CH02-0AE0 |

| Familie        | Baugruppe         | Bestellnummer  |
|----------------|-------------------|--|
| Netzkomponente | Diagnose-Repeater | 6ES7 972-0AB01-0XA0  |
| ET 200S        | 1 Count 24V       | 6ES7 138-4DA04-0AB0  |
|                | 1 Count 5V        | 6ES7 138-4DE02-0AB0  |
|                | 1 Step 5V         | 6ES7 138-4DC00-0AB0, 6ES7 138-4DC01-0AB0                       |
|                | 2 Pulse           | 6ES7 138-4DD00-0AB0, 6ES7 138-4DD01-0AB0                       |
|                | 1 SI              | 6ES7 138-4DF01-0AB0  |
|                | 1 SI Modbus       | 6ES7 138-4DF11-0AB0  |
|                | 1 SSI             | 6ES7 138-4DB02-0AB0, 6ES7 138-4DB03-0AB0                       |
|                | 1 Pos Universal   | 6ES7 138-4DL00-0AB0  |
|                | SIWAREX           | 7MH4910-0AA01, 7MH4912-0AA01, 7MH4920-0AA01                    |
| ET 200M        | SIWAREX           | 7MH4 900-2AA01, 7MH4 900-3AA01, 7MH4 950-1AA01, 7MH4 950-2AA01 |

### 2.2.3.3 Austausch der ET 200S Positioniermodule

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Austausch der ET 200S Positioniermodule

Diese Information betrifft Positioniermodule "1 Step 5V" (6ES7 138-4DC00-0AB0) aus einem Projekt, welches mit TIA-Portal V11.0 erstellt wurde. Beim Ersetzen dieser Module aus TIA-Portal V11.0 durch eine neue Version dieser Module, werden die Parametereinstellungen auf die Vorgabewerte zurückgesetzt.

Dieser Fall tritt bei einer der folgenden Vorgehensweisen auf:

- Ersetzen des Positioniermoduls 6ES7 138-4DC00-0AB0 durch dessen Nachfolgermodule 6ES7 138-4DC01-0AB0 über einen Gerätetausch.
- Aktualisieren der Modulversion über die entsprechende Schaltfläche bei den Geräteeigenschaften im Inspektorfenster.

### 2.2.3.4 CP 343-2 an SIMATIC S7 Embedded Controller EC31-RTX

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### CP 343-2 an SIMATIC S7 Embedded Controller EC31-RTX

Das Modul AS-Interface CP 343-2 (Bestell-Nr.: 6GK7 343-2AH01) kann zwar in einen Erweiterungsbaugruppenträger des SIMATIC S7 Embedded Controller EC31-RTX (Bestell-

Nr.: 6ES7 677-1DDxx-0BB0) gesteckt werden, aber der CP 343-2 darf nicht mit dem EC31-RTX betrieben werden.

### 2.2.3.5 Hinweise zu Online und Diagnose

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Hardware-Erkennung und anschließendes Online verbinden

Wenn Sie bei einer nicht spezifizierten CPU den Befehl "Online > Hardware-Erkennung" ausführen, wird die Online-Konfiguration dabei nicht aus der CPU geladen. Falls Sie die über die Hardware-Erkennung erhaltene Konfiguration nicht auf die CPU laden, hat dies zur Folge, dass in der Geräte- und Netzsicht immer ein Unterschied zwischen Offline- und Online-Konfiguration angezeigt wird. In der Online- und Diagnosesicht ist ersichtlich, dass eine unterschiedliche Projektierung vorliegt, obwohl die MLFBs bei der tatsächlich vorhandenen und der Offline-CPU identisch sind.

### 2.2.3.6 Netzwerk-Komponenten

#### Netzwerk-Komponenten

#### CP 1242-7

#### CP 1242-7 in ein anderes Projekt kopieren

Wenn Sie einen CP 1242-7 von einem Projekt in ein anderes Projekt kopieren, dann werden die folgenden Parameter in der Parametergruppe "CP-Identifikation" in der Zielstation verändert:

- Projektnummer des CP
- Stationsnummer des CP

#### Laden in Gerät

Führen Sie beim CP 1242-7 die Funktion "Laden in Gerät" über eine TeleService-Verbindung nur folgendermaßen aus:

1. Selektieren Sie in STEP 7 den CP.
2. Wählen Sie das Menü "Online" > "Laden in Gerät".
3. Wählen Sie im aufgeblendeten Dialog "Erweitertes Laden" die TeleService-Schnittstelle aus.
4. Laden Sie die Projektdaten über den Dialog "Erweitertes Laden".

## Laden von Gerät

Die Funktion "Laden von Gerät" wird vom CP 1242-7 nicht unterstützt.

## SCALANCE X

### SCALANCE XR500 als IO-Device: "Übersetzen" und "Laden in Gerät"

Bei einem SCALANCE XR500, der als PROFINET IO-Device projektiert und einem IO-Controller zugeordnet ist, werden bei den Funktionen "Übersetzen" und "Laden in Gerät" nur diejenigen Daten in den Switch geladen, die auch im Web Based Management (WBM) projektierbar sind (Layer 2, Layer 3, System, Security).

Wenn Sie die Funktionen "Übersetzen" oder "Laden in Gerät" für die PROFINET IO-Device-Daten des XR500 ausführen wollen, dann selektieren Sie zuvor den zugeordneten IO-Controller.

### Verwendung einer Link Aggregation innerhalb einer MSTP-Instanz

Wenn Sie eine Link Aggregation innerhalb einer MSTP-Instanz verwenden wollen, gehen Sie bei der Konfiguration wie folgt vor:

1. Legen Sie unter "Layer 2" > "Link Aggregation" eine Link Aggregation an.
2. Legen Sie unter "Layer 2" > "MSTP" > "MST Allgemein" eine MSTP-Instanz an.
3. Konfigurieren Sie die Link Aggregation unter "Layer 2" > "MSTP" > "MST Port".

### Automatische Aktivierung von MRP in redundanten Topologien

Wenn Sie SCALANCE X-Switches in der Topologiesicht mit redundanten Netzstrukturen verbinden, dann wird bei den betroffenen Switches automatisch MRP aktiviert.

Eine eventuell vorhandene Projektierung anderer Redundanzmechanismen, wie zum Beispiel MSTP, wird automatisch deaktiviert.

## 2.2.4 PLC programmieren

### 2.2.4.1 Allgemeine Hinweise zur PLC-Programmierung

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.



### **Inkonsistente Programme in ein Gerät laden**

Im TIA-Portal ist es nicht möglich, inkonsistente Programme ohne Konsistenzprüfung in ein Gerät zu laden. Während des Ladens werden implizit alle Bausteine des Programms geprüft und bei Inkonsistenzen erneut übersetzt. Wenn sich auf Ihrer CPU jedoch Programme befinden, die mit älteren Versionen von STEP 7 geladen wurden, können diese Programme Inkonsistenzen aufweisen.

Beachten Sie in diesem Fall folgenden Hinweis:

Wenn Sie ein inkonsistentes Programm von einem Gerät laden, können Sie das Programm anschließend nicht unverändert zurück in das Gerät laden, da während des Ladens zwangsläufig eine Konsistenzprüfung durchgeführt und bestehende Inkonsistenzen behoben werden.

### **Prozessabbild von PTO-/PWM- Ausgängen**

Benutzen Sie die Ausgänge von PTO/PWM nicht im Prozessabbild (z. B. bei Zugriffen im Anwenderprogramm, bei Onlinefunktionen oder in HMI). Die Aktualisierungsrate des Prozessabbilds ist wesentlich geringer als die Geschwindigkeit der Signaländerungen. Die Anzeige im Prozessabbild gibt daher nicht den Signalverlauf wieder.

### **Bausteine beobachten in KOP und FUP**

Falls der Beginn des Strompfads außerhalb des sichtbaren Bereichs liegt, kann es sein, dass der Eingangswert nicht ermittelt werden kann. In diesem Fall wird der Strompfad in grau dargestellt.

### **Verwendung von systemerzeugten PLC-Datentypen in Bibliotheken vermeiden**

Einige Anweisungen erstellen beim Instanzieren eigene PLC-Datentypen, die im Projektordner "PLC-Datentypen" gespeichert werden. Sie sollten diese vom System erzeugten PLC-Datentypen jedoch in keiner Bibliothek verwenden, da sie zum einen jederzeit vom System neu erstellt werden könnten und zum anderen ein ungünstiges Systemverhalten daraus entstehen kann.

### **Globale Datenbausteine in Zuweisungen verwenden**

Es ist nicht möglich, den Inhalt eines globalen Datenbausteins einem strukturell gleichen Datenbaustein z. B. über eine Move-Box zuzuweisen, wenn einer der beiden DBs eine Speicherreserve enthält.

### **Konvertieren von know-how-geschützten Bausteinen aus V10.5**

Nach der Konvertierung aus vorherigen STEP 7-Versionen (z. B. STEP 7 V10.5) muss das Programm übersetzt werden. Falls Sie know-how-geschützte Bausteine benutzen, werden Sie aufgefordert, das Passwort einzugeben.

### 2.2.4.2 Anweisungen

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Verwendung von Anweisungen (S7-1200)

Wenn bei Anweisungen, die eine Systemfunktion (SFC) oder einen Systemfunktionsbaustein (SFB) darstellen, ein parameterspezifischer Fehler auftritt, dann wird kein Fehlercode am Parameter RET\_VAL ausgegeben. Der RET\_VAL ist in diesem Fall ungültig! Um auf diese Fehler zu reagieren, haben Sie je nach eingesetzter CPU unterschiedliche Möglichkeiten.

#### Anweisung "TRCV\_C: Daten über Ethernet empfangen"

Entgegen der Aussage in der Online Hilfe wird beim Setzen des Parameters CONT auf den Wert "0" die Kommunikationsverbindung sofort getrennt und nicht erst nach dem Senden der Daten.

#### Anweisung "T\_CONFIG: Interface konfigurieren"

Nachdem Sie die Anweisung "Interface konfigurieren" ausgeführt haben, um einen IP-Parameter zu ändern, wird die CPU neu gestartet. Die CPU geht in den Betriebszustand STOP, ein Warmstart wird ausgeführt und die CPU läuft erneut an (Betriebszustand RUN). Stellen Sie sicher, dass sich der Steuerungsprozess in einem sicheren Betriebszustand befindet, nachdem die CPU nach der Ausführung der Anweisung "Interface konfigurieren" neu gestartet wurde. Ein unkontrollierter Betrieb, verursacht durch z. B. Funktionsstörungen oder Programmfehler, kann zu schweren Sach- und Personenschäden führen. Nicht remanente Daten können verloren gehen.

Parameter ERROR und STATUS

| ERROR | STATUS (DW#16#..) | ERR_LOC | Erläuterung   |
|-------|-------------------|---------|---|
| 0     | 00000000          | 0       | Nach der erfolgreichen Ausführung der Anweisung liefert der Parameter STATUS "00000000" keinen Wert zurück. |

#### Anweisung "GET\_DIAG: Diagnoseinformation lesen"

Der MODE 3 am Parameter MODE wird von der CPU S7-1200 nicht unterstützt.

## Verwendung von Anweisungen mit Parametern vom Typ VARIANT in Codebausteinen mit unterschiedlicher Zugriffsart (S7-1200)

Codebausteine (FBs/FCs) und Datenbausteine (DBs) können mit unterschiedlicher Zugriffsart erstellt werden ("Standard" und "Optimiert"). In den Codebausteinen können Sie beliebige Anweisungen aufrufen. Einige Anweisungen (z.B. "WRIT\_DBL" und "READ\_DBL") verwenden an Ein- und Ausgangsparametern Zeiger vom Typ VARIANT, um Datenbausteine zu adressieren.

Stellen Sie sicher, dass Sie diese Anweisungen nicht in Programmen verwenden, in denen sich Codebausteine mit unterschiedlicher Zugriffsart gegenseitig aufrufen, so dass folgende Fälle eintreten würden:

- Eine Struktur aus einem Standard-Datenbaustein wird unmittelbar oder mittelbar an einen optimierten Codebaustein übergeben, welcher diese Struktur unmittelbar oder mittelbar an einen der oben genannten Bausteine weitergibt.
- Der umgekehrte Fall, dass eine Struktur aus einem optimierten Codebaustein unmittelbar oder mittelbar an einen Standard-Datenbaustein übergeben wird, welcher diese Struktur unmittelbar oder mittelbar an einen der oben genannten Bausteine weitergibt.

### 2.2.4.3 Anwenderprogramm testen

#### Testen mit der Beobachtungstabelle

##### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Mehrfachzugriffe auf dieselbe CPU

Es darf von einem PG/PC aus nur mit einem geöffneten TIA-Portal online auf eine CPU zugegriffen werden. Mehrfachzugriffe auf dieselbe CPU sind nicht erlaubt und können zu Fehlern führen.

#### Laden von Datenbausteinen bei laufendem Steuerauftrag

---

##### Hinweis

Das Laden von geänderten Datenbausteinen bei laufendem Steuerauftrag kann zu unvorhergesehenen Betriebszuständen führen. Der Steuerauftrag steuert weiterhin die angegebene Adresse, während sich die Adressbelegung im Datenbaustein möglicherweise geändert hat. Beenden Sie laufende Steueraufträge vor dem Laden von Datenbausteinen.

---

#### Testen von Programmen, die aus STEP 7 V10.5 konvertiert wurden

Um ein aus STEP 7 V10.5 konvertiertes Programm zu beobachten und zu testen, müssen Sie es zunächst einmal mit STEP 7 V11.0 übersetzen und laden.

### Funktion "Peripherieausgänge freischalten"

Die Funktion "Peripherieausgänge freischalten" steht im TIA-Portal V12.0 für CPUs aus der Baureihe S7-1500 nicht zur Verfügung.  
Diese Funktion kann im TIA-Portal V12.0 ausschließlich mit einer CPU S7-300, S7-400 oder S7-1200 ausgeführt werden.

### Testen mit der Forcetabelle

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

### Forcen von Variablen bei direktem Peripheriezugriff

Wenn Sie in Ihrem Anwenderprogramm einen direkten Peripheriezugriff für eine S7-300 CPU verwenden, so ist ein Forcen dieser Peripherieadresse nicht erlaubt.

#### Beispiel

Erfolgt im Anwenderprogramm ein Peripheriezugriff auf die Adresse "EB0:P", so dürfen folgende Peripherieadressbereiche nicht geforct werden: E0.0:P, EB0:P, EW0:P und ED0:P.

## 2.2.5 Technologiefunktionen

### 2.2.5.1 Hinweise zu Technologiefunktionen

Es liegen keine Hinweise zu den Technologiefunktionen vor.

## 2.3 WinCC Basic

### 2.3.1 Security-Hinweise

#### Security-Hinweise

Siemens bietet Automatisierungs- und Antriebsprodukte mit Industrial Security-Funktionen an, die den sicheren Betrieb der Anlage oder Maschine unterstützen. Sie sind ein wichtiger Baustein für ein ganzheitliches Industrial Security-Konzept. Die Produkte werden unter diesem Gesichtspunkt ständig weiterentwickelt. Es wird empfohlen sich regelmäßig über

Aktualisierungen und Updates unserer Produkte zu informieren. Informationen und Newsletter hierzu finden Sie unter:

<http://support.automation.siemens.com> (<http://support.automation.siemens.com>)

Für den sicheren Betrieb einer Anlage oder Maschine ist es darüber hinaus notwendig, geeignete Schutzmaßnahmen (z.B. Zellschutzkonzept) zu ergreifen und die Automatisierungs- und Antriebskomponenten in ein ganzheitliches Industrial Security-Konzept der gesamten Anlage oder Maschine zu integrieren, das dem aktuellen Stand der Technik entspricht. Dabei sind auch eingesetzte Produkte von anderen Herstellern zu berücksichtigen. Weitergehende Informationen finden Sie unter:

<http://www.siemens.com/industrialsecurity> (<http://support.automation.siemens.com>)

## Kennwörter

In WinCC sind verschiedene Kennwörter voreingestellt. Aus Sicherheitsgründen sollten Sie diese Kennwörter ändern.

- Für den Benutzer "Administrator" ist das Kennwort "administrator" voreingestellt.

## Kommunikation über Ethernet

Bei Ethernet-basierter Kommunikation ist der Endanwender für die Sicherheit seines Datennetzes selbst verantwortlich, da z. B. bei gezielten Angriffen, die zur Überlast des Gerätes führen, die Funktionsfähigkeit nicht garantiert ist.

## Automatisches Beenden der Runtime

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie im Control Panel ein Kennwort.

## Netzwerkeinstellungen

Die folgenden Tabellen zeigen die Netzwerkeinstellungen des jeweiligen Produkts, die Sie zur Analyse der Netzwerksicherheit und zur Konfiguration von externen Firewalls benötigen:

| WinCC Basic (ohne Simulation) |             |                     |                   |                     |   |
|-------------------------------|-------------|---------------------|-------------------|---------------------|---|
| Name                          | Port-Nummer | Transport-Protokoll | Richtung          | Funktion            | Beschreibung  |
| ALM                           | 4410*       | TCP                 | Inbound, Outbound | Lizenz-Service      | Dieser Dienst stellt die komplette Funktionalität für Software-Lizenzen zur Verfügung und wird sowohl vom Automation License Manager als auch von allen lizenzrelevanten Softwareprodukten verwendet. |
| HMI Load                      | 1033        | TCP                 | Outbound          | HMI Load (RT Basic) | Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Basic Panels zu übertragen.   |

\* Standard-Port, der durch Anwenderkonfiguration geändert werden kann

| WinCC Simulation für Basic Panels |             |                     |          |                     |   |
|-----------------------------------|-------------|---------------------|----------|---------------------|---|
| Name                              | Port-Nummer | Transport-Protokoll | Richtung | Funktion            | Beschreibung  |
| HMI Load                          | 1033        | TCP                 | Inbound  | HMI Load (RT Basic) | Dieser Dienst wird verwendet, um Images und Konfigurationsdaten auf Basic Panels zu übertragen. |
| EtherNet/IP                       | 44818       | TCP                 | Outbound | Ethernet/IP-Kanal   | Das Ethernet/IP-Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.                |
|                                   | 2222        | UDP                 | Inbound  | Ethernet/IP-Kanal   | Das Ethernet/IP-Protokoll wird für Verbindungen zu Allen Bradley PLCs verwendet.                |
| Modbus TCP                        | 502         | TCP                 | Outbound | Modbus TCP-Kanal    | Das Modbus TCP-Protokoll wird für Verbindungen zu Schneider PLCs verwendet.                     |
| RFC 1006                          | 102         | TCP                 | Outbound | S7-Kanal            | Kommunikation zum S7-Controller über Ethernet / PROFINET  |
| Mitsubishi MC                     | 5002        | TCP                 | Outbound | Mitsubishi MC-Kanal | Das Mitsubishi-Protokoll wird für Verbindungen zu Mitsubishi PLCs verwendet.                    |

### Siehe auch

<http://www.siemens.com/industrialsecurity> (<http://www.industry.siemens.com/topics/global/de/industrial-security/Seiten/Default.aspx>)

## 2.3.2 Hinweise zur Benutzung

### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

### Kopieren von HMI-Geräten mit HMI-Verbindungen

Wenn Sie ein HMI-Gerät mit HMI-Verbindungen zu einer Steuerung kopieren, wird die HMI-Verbindung im neuen HMI-Gerät nicht automatisch mit einer vorhandenen namensgleichen Steuerung verbunden. Dies gilt sowohl für das Kopieren innerhalb eines Projekts als auch für das projektübergreifende Kopieren.

Um auch im neuen HMI-Gerät über eine HMI-Variable auf die Steuerungsvariable zugreifen zu können, sollten Sie unmittelbar nach dem Kopieren die Projektierung der HMI-Verbindung wieder vervollständigen. Gehen Sie dazu wie folgt vor:

1. Öffnen Sie den Editor "Geräte & Netze".
2. Verbinden Sie das neue HMI-Gerät mit dem gewünschten Netz.
3. Öffnen Sie die Verbindungstabelle.
4. Wählen Sie die HMI-Verbindung des neuen HMI-Gerätes aus.
5. Wählen Sie unter Partner die gewünschte Steuerung aus.

Wenn Sie zwischen Kopieren des HMI-Geräts und Vervollständigen der Verbindung das neue HMI-Gerät übersetzen oder weitere PLC-Variablen verbinden, wird in manchen Fällen eine zusätzliche HMI-Verbindung zur gleichen Steuerung erzeugt. Dies ist insbesondere der Fall, wenn HMI-Variablen mit DB-Array-Elementen verbunden sind.

### Gerätewechsel

Nach einem Bediengerätewechsel sollten Sie das Aussehen der projektierten Bilder überprüfen. Aufgrund geänderter Displaygrößen kann sich Position und Aussehen von Bildobjekten, z. B. Rezepturanzeige und Meldeanzeige, verändern.

### Gerätewechsel - Kommunikation

Bei einem Bediengerätewechsel können Fehlermeldungen vom Typ "... wird in der neuen Konfiguration nicht unterstützt. Es wird entfernt." auftreten. Diese Meldungen beziehen sich auf projektierte Verbindungen des Geräts und werden z. B. durch eine unterschiedliche Anzahl von Schnittstellen der Bediengeräte ausgelöst. Nach dem Gerätewechsel werden diese Verbindungen rot markiert. Wenn Sie diese Verbindungen weiterhin benutzen möchten, müssen Sie die Projektierung der Verbindung anpassen. Gehen Sie dazu wie folgt vor:

1. Öffnen Sie den Editor "Geräte und Netze".
2. Klicken Sie in der Funktionsleiste der Netzsicht auf "Vernetzen".
3. Vernetzen Sie die Schnittstelle des Bediengeräts mit der Schnittstelle der CPU.
4. Klicken Sie im tabellarischen Bereich der Netzsicht auf die Tabelle "Verbindungen".

5. Selektieren Sie die rot markierte Verbindung.
6. Stellen Sie im Inspektorfenster "Eigenschaften > Allgemein > Schnittstelle" die neue Schnittstelle ein.

### Angabe des Änderungszeitpunkts im Übersichtsfenster

Die im Übersichtsfenster angezeigten Änderungszeitpunkte beziehen sich nur auf Änderungen am Objekt selbst. Änderungen an unterlagerten Objekten, z. B. den Bildobjekten in einem Bild, führen nicht dazu, dass im Übersichtsfenster der Zeitpunkt der letzten Änderung des Bildes verändert wird.

### Bediengeräte-Assistent

Wenn Sie mit dem Bediengeräte-Assistenten ein Gerät mit einem Farb-Display anlegen, kann es vorkommen, dass die Grafiken der Navigationsschaltflächen in schwarz-weiß angezeigt werden. Dieser Fehler tritt aber nur auf, wenn das neue Gerät mit dem gleichen Namen angelegt wird, den ein zwischenzeitlich gelöscht Gerät mit monochromen Display besessen hat.

Sie können den Fehler vermeiden, indem Sie immer wenn Sie ein Gerät aus dem Projekt löschen auch die zugehörigen Grafiken in der Grafiksammlung löschen.

### Objekte mit Objektreferenzen in der Projektbibliothek

In WinCC flexible konnten zwei Kopiermethoden verwendet werden.

- Beim "Einfachen Kopieren" wird z. B. ein WinCC flexible Bild inklusive eines EA-Felds kopiert. Von einer am EA-Feld projektierten Variable wird nur der Objektname kopiert, da es sich hierbei um eine Referenz handelt.
- Beim "Kopieren" werden ein Bild, ein enthaltenes EA-Feld und die am EA-Feld projektierte Variable mit ihren Eigenschaften kopiert.

Auch beim Ablegen eines Objektes in einer Bibliothek konnten diese beiden Methoden angewendet werden. Durch die Migration werden Projektbibliotheken und darin enthaltene Objekte migriert und sind in WinCC nutzbar.

In WinCC steht jedoch nur noch eine Kopiermethode zur Verfügung. Diese verhält sich bezüglich Variablen wie das "Einfache Kopieren" in WinCC flexible. Bezüglich Grafiken, Grafiklisten und Textlisten verhält sie sich wie das "Kopieren" in WinCC flexible.

Wenn Sie in einer Bibliothek in WinCC flexible Objekte mit Referenzen auf Variablen abgelegt haben, müssen Sie beim Verwenden dieser Objekte in WinCC die referenzierten Objekte neu projektieren.

### Ostasiatische Projektsprachen projektieren auf einem PC ohne asiatisches Betriebssystem

Wenn Sie auf Ihrem PC kein asiatisches Betriebssystem installiert haben, aber in der Projektierung eine ostasiatische Projektsprache wählen, wird im Editor "Runtime-Einstellungen > Sprache & Schriftart" die Standardschrift als ungültig markiert.

Um das Problem zu beheben, aktivieren Sie in der Systemsteuerung "Regions- und Sprachoption > Sprachen" die Option "Dateien für ostasiatische Sprachen installieren".



## Lizenztransfer auf ein Panel unter 64-Bit-Betriebssystemen

Wenn auf einem 64-Bit-Betriebssystem im Automation License Manager der Menübefehl "Bearbeiten > Zielsysteme verbinden > Bediengerät verbinden" nicht zur Verfügung steht, öffnen Sie bitte die Kommandozeile und führen Sie, mit Administrator-Rechten, folgenden Befehl aus:

```
"%WINDIR%\system32\RegSvr32.exe" "%CommonProgramFiles%\siemens\AlmPanelPlugin\ALMPanelParam.dll"
```

## Installationsreihenfolge für Startdrive

Wenn Sie Startdrive auf einem PC installieren, dann beachten Sie bei der Installation folgende Reihenfolge:

- Installieren Sie STEP7 V12.0.
- Installieren Sie Startdrive.

## SIMOTION

SIMOTION wird mit WinCC V12 SP1 nicht unterstützt.

## Sicherheitseinstellungen bei der Installation

Wenn Sie WinCC V12 installieren, dann werden in Ihrem Betriebssystem Sicherheitseinstellungen verändert.

Welche Sicherheitseinstellungen das sind, wird während der Installation aufgelistet.

Die Änderungen der Sicherheitseinstellungen müssen Sie bestätigen.

Wenn Sie nach der Installation an Ihrem Betriebssystem Änderungen vornehmen können dadurch die Änderungen der Sicherheitseinstellungen durch die Installation des TIA Portals verändert werden.

Die Änderungen der Sicherheitseinstellungen durch die Installation des TIA Portals können Sie wiederherstellen:

"Start > Alle Programme > Siemens Automation > Security Controller > Einstellungen wiederherstellen"

## SQL Instanz von WinCC V12

Wenn Sie ein WinCC V11 Produkt installiert haben und ein WinCC V12 Produkt installieren wollen, dann müssen Sie vor der Installation alle WinCC V11 Produkte und die Instanz WinCC vom SQL Server 2005 nacheinander deinstallieren.

Mit der Installation von WinCC V12 wird eine neue WINCC SQL 2008 Instanz installiert.

### 2.3.3 Migration

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Änderung der Namen von Meldeklassen

Im Unterschied zu WinCC flexible sind die Namen der vordefinierten Meldeklassen nicht abhängig von der aktuell verwendeten Oberflächensprache. Bei der Migration werden die Namen der Meldeklassen wie folgt vergeben:

| WinCC flexible | WinCC    |
|----------------|----------|
| Fehler         | Errors   |
| System         | System   |
| Warnungen      | Warnings |

Nach der Migration können die Anzeigenamen der Meldeklassen beliebig verändert werden.

#### Projektsprachen in WinCC

In WinCC V12 werden nicht alle Projektsprachen unterstützt, die in WinCC flexible einstellbar waren, z.B. Arabisch. Wenn Sie als Ergebnis der Migration ein leeres Projekt erhalten, überprüfen Sie bitte die verwendete Bearbeitungssprache. Die nicht unterstützten Projektsprachen dürfen nicht als Bearbeitungssprache im Quellprojekt eingestellt sein. Gehen Sie wie folgt vor:

1. Öffnen Sie das Projekt mit WinCC flexible.
2. Ändern Sie die Bearbeitungssprache in zum Beispiel Englisch.
3. Speichern Sie das Projekt.
4. Starten Sie die Migration erneut.

#### Objekte mit Objektreferenzen in der Projektbibliothek

In WinCC flexible konnten zwei Kopiermethoden verwendet werden.

- Beim "Einfachen Kopieren" wird z. B. ein WinCC flexible Bild inklusive eines EA-Felds kopiert. Von einer am EA-Feld projektierten Variable wird nur der Objektname kopiert, da es sich hierbei um eine Referenz handelt.
- Beim "Kopieren" werden ein Bild, ein enthaltenes EA-Feld und die am EA-Feld projektierte Variable mit ihren Eigenschaften kopiert.

Auch beim Ablegen eines Objektes in einer Bibliothek konnten diese beiden Methoden angewendet werden. Durch die Migration werden Projektbibliotheken und darin enthaltene Objekte migriert und sind in WinCC nutzbar.

In WinCC steht jedoch nur noch eine Kopiermethode zur Verfügung. Diese verhält sich wie das "Einfache Kopieren" in WinCC flexible.

Wenn Sie in einer Bibliothek in WinCC flexible Objekte mit Referenzen auf andere Objekte abgelegt haben, müssen Sie beim Verwenden dieser Objekte in WinCC die referenzierten Objekte neu projektieren.

### Migration eines integrierten Projekts mit ProTool-Objekten

Wenn Sie ein in STEP 7 integriertes WinCC flexible Projekt migrieren und die Fehlermeldung "Optionspaket(e) PROTOOL in STEP 7 fehlen" auftritt, haben Sie WinCC flexible 2008 SP3 installiert. Zudem enthält das Projekt noch Objekte, die mit ProTool projiziert worden sind. Öffnen Sie das Projekt nicht mit WinCC flexible 2008 SP3! Um das Projekt zu migrieren gehen Sie wie folgt vor:

1. Kopieren Sie das Projekt auf einen Rechner auf dem WinCC flexible 2008 SP2 und STEP 7 installiert ist.
2. Öffnen Sie das Projekt im SIMATIC Manager.
3. Entfernen Sie alle ProTool-Objekte aus dem Projekt.
4. Führen Sie im Menü "Datei" den Befehl "Speichern unter" aus.
5. Aktivieren Sie im Dialog "Projekt speichern untern" die Option "Mit Reorganisation".
6. Klicken Sie auf OK.
7. Kopieren Sie das Projekt wieder auf den ursprünglichen Rechner.
8. Starten Sie die Migration erneut.

### Migration eines WinCC V7 Projekts mit Runtime-Sprache "Chinesisch (Taiwan)"

Wenn Sie WinCC mit Unterstützung für die Oberflächensprache "Chinesisch" installiert haben, werden bei der Migration von WinCC V7 Projekten Texte und Report Layouts der Runtime-Sprache "Chinesisch (Taiwan)" nicht migriert. Migrieren Sie derartige Projekte entweder mit dem Migrations-Tool oder auf einem PC auf dem WinCC ohne Oberflächensprache "Chinesisch" installiert ist.

### Fortschrittsanzeige

Wenn die Fortschrittsanzeige 100% anzeigt, werden von der Software noch verschiedene Restarbeiten, z. B. das Schließen von Referenzen, durchgeführt. In dieser Zeit wird die Software nicht auf Eingaben im User Interface reagieren.

### Prinzip

Nach dem Öffnen eines V11-Projektes mit einer V12-Version ist das anschließende Öffnen dieses Projektes mit einer älteren Version nicht mehr möglich.

### ActiveX-Controls von Drittanbietern

Die Migration unterstützt auch die ActiveX-Controls von Drittanbietern. Die Controls müssen allerdings im Betriebssystem registriert sein. Wenn ein ActiveX-Control nicht registriert ist, wird die Migration abgebrochen.

Wenn Sie ein Projekt mit dem Migrationstool speichern und die Migration selbst auf einem anderen PC durchführen, dann müssen die Controls auch auf diesem PC registriert sein.

### Sprachumschaltung in RT

Hochrüsten des Projekts von V11 nach V12 oder Migration von WinCC V7.0 SP3 nach V12:

Wenn Sie in V11 oder V7.0 SP3 eine Sprachumschaltung mit LCID nach Chinesisch (VR China) per Skript programmiert haben, dann funktioniert eine Sprachumschaltung nach Chinesisch (VR China) in Runtime nach Hochrüsten oder Migration nicht mehr.

Ändern Sie im Skript die LCID von "1028" auf "2052".

### Integrierte Projekte mit Meldeanzeigen migrieren

In einem integrierten Projekt ist eine Meldeanzeige mit allen Meldeklassen aktiviert. Bei der Migration des Projekts können die Meldeklassen deaktiviert werden.

Prüfen Sie nach der Migration des Projekts die Einstellungen in der Meldeanzeige.

Aktivieren Sie gegebenenfalls im Inspektorfenster der Meldeanzeige unter "Eigenschaften > Allgemein" die erforderlichen Meldeklassen.

### Migration von Projekten aus WinCC V7

Für die Migration umfangreicherer Projekte aus WinCC V7.0 SP3 empfehlen wir die Verwendung eines 64-Bit Betriebssystems.

## 2.3.4 Engineering System

### 2.3.4.1 Bilder und Bildobjekte

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Textformat von Ausgabefeldern in Meldetexten

Es ist nicht möglich Variablen und Textlisteneinträge unterstrichen darzustellen.

#### Anzeigeobjekte zwischen zwei Projekten oder zwei Geräten kopieren

In Projekt\_1 projektieren Sie z.B. ein Meldefenster im Globalen Bild. Sie kopieren das Meldefenster und fügen es in das Globale Bild in Projekt\_2 ein.

Die aktivierten Meldeklassen sind im Meldefenster nach dem Einfügen teilweise nicht aktiviert.

Das Verhalten trifft für folgende Anzeigeobjekte auf:

- Meldefenster
- Meldeindikator
- Meldeanzeige

### Darstellung der Querverweise im Inspektorfenster

Die von einem Bildobjekt verwendeten Objekte werden im Inspektorfenster im Register "Info > Querverweis" angezeigt.

Ein Bild ist geöffnet und ein Objekt ausgewählt. Sie verwenden eine HMI-Variable an dem Objekt als Prozessvariable.

In den Querverweisen wird das Objekt und die verschaltete HMI-Variable angezeigt. Zusätzlich werden alle Verwendungsstellen des Objekts und der HMI-Variablen aufgelistet.

Wenn die HMI-Variable mit einer PLC-Variable oder einer DB-Variablen verschaltet ist, dann werden auch die Verwendungsstellen der verschalteten PLC-Variablen bzw. DB-Variablen angezeigt.

### Bezeichnung der Ereignisse bei Meldungen im Inspektorfenster im Register "Info"

In einigen Meldungen des Inspektorfensters im Register "Info" werden Bezeichnungen für die Ereignisse verwendet, die von den Bezeichnungen im Register "Eigenschaften" abweichen.

| Name im Inspektorfenster im Register "Eigenschaften" | Name im Inspektorfenster im Register "Info" |
|--|---|
| Abgebaut   | ClearScreen                                 |
| Aufgebaut  | GenerateScreen                              |
| Aktivieren   | Activate                                    |
| Änderung   | Change                                      |
| Beim Öffnen eines Dialogs                            | ONMODALBEGIN                                |
| Beim Schließen eines Dialogs                         | ONMODALEND                                  |
| Benutzerwechsel                                      | PASSWORD                                    |
| Bildwechsel  | SCREEN                                      |
| Deaktivieren   | Deactivate                                  |
| Drücken  | Press                                       |
| Gegangen   | Going                                       |
| Gekommen   | Coming                                      |
| Grenzwert "Oberer Fehler" überschritten              | AboveUpperLimit                             |
| Grenzwert "Unterer Fehler" unterschritten            | BelowLowerLimit                             |
| Klicken  | Click                                       |
| Loop-In-Alarm  | LoopInAlarm                                 |
| Loslassen  | Release                                     |
| Meldepufferüberlauf                                  | OVERFLOW                                    |
| Quittieren   | Acknowledgement                             |
| Runtime-Stopp  | Shutdown                                    |
| Taste drücken  | KeyDown                                     |

| Name im Inspektorfenster im Register "Eigenschaften" | Name im Inspektorfenster im Register "Info" |
|--|---|
| Taste loslassen                                      | KeyUp                                       |
| Umschalten EIN                                       | SwitchOn                                    |
| Umschalten AUS                                       | SwitchOff                                   |
| Wertänderung   | Change value                                |

### Dynamisierung der Eigenschaften von Objekten in einer Gruppe

In einer Gruppe ist die Dynamisierung von Eigenschaften für alle Objekte der Gruppe, die diese Eigenschaft besitzen, nicht möglich. In WinCC V12 können die Eigenschaften der zu einer Gruppe gehörenden Objekte nur für jedes Objekt selbst dynamisiert werden.

### Unleserliche Zeichen bei Runtime Professional

Bei Runtime Professional können auf dem Zielsystem nur Zeichen angezeigt werden, die zu dem Sprachraum gehören, der mit der Betriebssystemeinstellung "Sprache für Programme, die Unicode nicht unterstützen" festgelegt wird. Im Projekt können jedoch auch Texte mit Zeichen aus anderen Sprachräumen projiziert werden.

Bei den Objekten Textfeld, Symbolisches EA-Feld, Zeigerinstrument und Schieberegler können im Engineering System unleserliche Zeichen auftreten, wenn die Einstellungen im Betriebssystem bezüglich der "Sprache für Programme, die Unicode nicht unterstützen" nicht mit der gewählten Editiersprache übereinstimmt und die Objekte in einem anderen Design als "WinCC Classic" dargestellt werden. Im Inspektorfenster und dem Editor "Projekttexte" werden die Zeichen korrekt dargestellt.

Überprüfen Sie daher zunächst, ob in der Systemsteuerung unter "Regions- und Sprachoptionen > Erweitert" die Einstellung bei "Sprache für Programme, die Unicode nicht unterstützen" der Editiersprache entspricht. Ansonsten können Sie die korrekten Texte im Inspektorfenster oder dem Editor "Projekttexte" kontrollieren oder ändern.

### Bildbausteine

Bildbausteine können nicht gedreht oder gespiegelt werden.

### Persistenz bei Anzeigeobjekten in WinCC Runtime Professional

Die Objekte f(t)-Kurvenanzeige, f(x)-Kurvenanzeige, Meldeanzeige, Rezepturanzeige, Tabellenanzeige und Wertetabelle verfügen über Einstellungen zur Persistenz von Online-Projektierungen. Wenn Sie für "Online-Projektierung" "Persistenz" und für "Verhalten bei Bildwechsel" "Änderungen beibehalten" projiziert haben, können Sie in Runtime Änderungen an den Konfigurationsdialogen durchführen, die auch nach einem Bildwechsel und dem Beenden von Runtime erhalten bleiben.

Bei den genannten Einstellungen führen Online-Projektierungen jedoch dazu, dass Änderungen an der Projektierung der Objekte im Engineering System nur noch dann in Runtime übernommen werden, wenn Sie das Gerät mit "Übersetzen > Software (komplett übersetzen)" neu übersetzen.

### Basic Panels, OP73, OP77A und TP177A: Anzeige von Texten in Runtime

Die im Editor "Runtime-Einstellungen > Sprachen & Schriftart" gewählte Standardschrift, wirkt sich auf die Anzeige von Texten in Runtime aus.

Wenn Sie die Schriftgröße bzw. Schriftstil unvorteilhaft wählen, kann es dazu führen, dass Texte abgeschnitten dargestellt werden

Dabei sind z. B. folgende Texte betroffen:

- Tooltips
- langer Meldetext
- Texte in Dialogen

### Tab-Reihenfolge in Bildern mit Bildbausteinen

Wenn Sie in WinCC V12 oder WinCC V12 SP1 in Bildern mit Bildbausteinen eine Tab-Reihenfolge projiziert hatten, sollten Sie in WinCC V12 SP2 die Tab-Reihenfolge dieser Bilder überprüfen. Die Tab-Reihenfolge kann sowohl im Bild als auch im Bildbaustein geändert sein.

### Variablen-Präfix eines Bildfensters in WinCC Runtime Professional

Das für ein Bildfenster projektierbare Variablen-Präfix wird von den Objekten der Palette "Controls" nicht unterstützt.

### E/A-Feld mit Anzeigeformat "Dezimal" und Darstellungsformat ohne Vorzeichen "s"

Sie haben eine Prozessvariable mit einem E/A-Feld verbunden. Das Anzeigeformat des E/A-Feldes ist "Dezimal".

Als Darstellungsformat können Sie ein Format mit und ohne Vorzeichen wählen,

Wenn Sie als "Darstellungsformat" ein Format ohne "s" wählen z. B. "999", dann hat das folgende Auswirkungen:

1. In Runtime können Sie keine negativen Werte über das E/A-Feld setzen.
2. Wenn die Variable einen negativen Wert annimmt, dann erstellt das E/A-Feld ein Zweierkomplement und ein verfälschter positiver Wert wird ausgegeben.

### Kurvenanzeige auf Basic Panel

Auf Basic Panels werden die Schaltflächen der Kurvenanzeige nicht angezeigt. Sie können die Kurvenanzeige über Funktionstasten bedienen, indem Sie dafür entsprechende Systemfunktionen an die Funktionstasten des Bediengeräts projizieren.

### Darstellung von Controls in Protokollen

Ein Projekt mit WinCC V11 SP2 mit Update 4 oder früher wird auf WinCC V12 hochgerüstet.

Dabei kann es vorkommen, dass Archivdaten in Controls bei Protokollen nicht angezeigt werden.

Betroffen sind folgende Controls:

- f(t)-Kurvenanzeige
- f(x)-Kurvenanzeige
- Tabellenanzeige

Abhilfe bei f(t)-Kurvenanzeige und f(x)-Kurvenanzeige

1. Aktivieren Sie im Control unter "Eigenschaften > Eigenschaften > Allgemein > Anzeige > Online".
2. Übersetzen Sie Ihr Projekt neu.
3. Laden Sie das Projekt auf Ihr Bediengerät.

Abhilfe bei Tabellenanzeige:

1. Aktivieren Sie im Control unter "Eigenschaften > Eigenschaften > Allgemein > Beim Öffnen des Bildes > Aktualisierung starten".
2. Übersetzen Sie Ihr Projekt neu.
3. Laden Sie das Projekt auf Ihr Bediengerät.

### Gruppierung von Bildobjekten

Wenn Sie Bildobjekte in WinCC gruppieren dann kann es bei tieferen Verschachtelungen zu Performance-Problemen von WinCC kommen.

### Status/Steuern

Das Bildobjekt "Status/Steuern" ist für folgende Steuerungen freigegeben:

- SIMATIC S7-300
- SIMATIC S7-400

### ActiveX- und .NET-Controls

In Runtime werden ActiveX- und .NET-Controls immer im Vordergrund positioniert.

Die Projektierung von ActiveX- und .NET-Controls in Ebenen wird nicht unterstützt.

### Rahmenlinie von Rechtecken

Sie haben in einem WinCC V7 Projekt ein Rechteck mit den Einstellungen "Linienstärke=1" und "Rahmen innen zeichnen= ja" projiziert.

Sie migrieren das WinCC V7 Projekt nach WinCC V12. Damit das Rechteck korrekt dargestellt wird, gehen Sie wie folgt vor.

1. Öffnen Sie das Inspektorfenster des Rechtecks.
2. Öffnen Sie die Eigenschaftsliste.
3. Deaktivieren Sie "Rahmenlinie nach innen verbreitern".



## Grafiken in Bildbausteinen

Sie fügen eine Grafikanzeige in einen Bildbaustein ein und definieren die Eigenschaft "Grafik" als Schnittstelle des Bildbausteins. Über die Schnittstelle der Bildbaustein-Instanz kann die Eigenschaft "Grafik" nun dynamisiert werden.

Verwenden Sie folgende Schreibweise um die Eigenschaft mit einem Bild über ein Script anzusprechen:

```
"..\.\.\Bildname".
```

## Dynamisierung von Instanzen eines Bildbaustein-Typs in einer Gruppe

Sie verwenden die Instanz eines Bildbaustein-Typs in einer Objektgruppe. Die Eigenschaften der Instanz werden auch als Eigenschaften der Gruppe angezeigt. Jede Dynamisierung mit Variablen, Skripten oder Animationen der Gruppe wird in Runtime nicht angezeigt.

## System-Diagnoseindikator für RT Advanced

Sie finden das Bibliotheksobjekt "Diagnosticsindicator" in der Bibliothek "Buttons and Switches > DiagnosticsButtons (Comfort Panels)". Das Objekt ist auch für Geräte mit RT Advanced verwendbar.

## Vorschau im Bildfenster

Sie verwenden eigene Designs mit Schatten bei Bildobjekten. Die Bildobjekte lassen Sie in einem Bildfenster anzeigen.

Der Schatten der Bildobjekte wird in der Vorschau des Bildfensters nicht angezeigt. Das Verhalten tritt nur im Engineering System auf. In Runtime wird es korrekt dargestellt.

### 2.3.4.2 Variablen und Verbindungen

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Namen von Variablen

Die Namen von HMI-Variablen dürfen nicht mit dem Zeichen @ beginnen.

#### Anzeige von gelöschten Arrayelementen an den Verwendungsstellen von HMI-Variablen

An den Verwendungsstellen von HMI-Variablen, z. B. Prozesswert von EA-Feldern, wird in der Regel der Variablenname angezeigt. Wenn das Element einer Arrayvariablen verwendet wird, wird der Variablenname um den in eckigen Klammern stehenden Index des Arrayelements erweitert.

Wenn eine verwendete Variable nicht mehr im Projekt vorhanden ist, wird an der Verwendungsstelle weiterhin der Variablenname angezeigt. Um auf die fehlende Variable hinzuweisen, ist das Feld rot hinterlegt. Falls ein verwendetes Arrayelement oder die

Arrayvariable selbst nicht mehr vorhanden ist, wird nur der Index des Arrayelements in eckigen Klammern angezeigt. Der Variablenname wird nicht angezeigt. Das Feld ist rot hinterlegt. In diesem Falle kann der Name der zugehörigen Arrayvariablen nicht mehr über die Verwendungsstellen identifiziert werden.

Wenn nicht bekannt ist, welche Arrayvariable an der Verwendungsstelle angebunden war, ist es evtl. erforderlich, das Arrayelement neu zu verbinden.

Wenn eine Variable oder Arrayvariable entsprechend der Referenz angelegt wurde, dann wird die markierte Referenz automatisch geschlossen.

Wenn eine HMI-Variable mit dem Arrayelement einer PLC-Variable verbunden ist und die PLC-Variable im Projekt nicht mehr vorhanden ist, tritt das gleiche Verhalten im Editor "HMI-Variablen" auf.

### Arrayvariablen als Listeneintrag von Multiplexvariablen

Arrayvariablen des Datentyps Char können wie Variablen des Datentyps String verwendet werden.

Die Verwendung einer Arrayvariablen des Datentyps Char als Listeneintrag einer Multiplexvariablen im Editor "HMI-Variablen" wird aber nicht unterstützt.

### Multiplexen von Variablen auf einem Basic Panel

Wenn Sie auf einem Basic Panel eine Variable mit einer externen Variablen multiplexen, wird in Runtime beim ersten Lesezyklus die Adresse aus der PLC gelesen. Erst beim zweiten Lesezyklus steht der Wert der gelesenen Adresse zur Verfügung.

### Runtime Advanced und Panels: Import von Arrayelementen und Strukturelementen

Arrayvariablen und Strukturvariablen werden immer im Ganzen importiert, einschließlich ihrer Elemente. Die Elemente der Arrayvariablen und Strukturvariablen werden beim Import nicht weiter befüllt.

Wenn in der Importdatei der Name einer Variable mit dem Namen eines Arrayelements bzw. Strukturelements übereinstimmt, wird eine neue Variable angelegt.

Beispiel:

Die Importdatei enthält eine Arrayvariable namens "Otto" mit 10 Arrayelementen. Die Arrayelemente heißen dann z. B. Otto[1], Otto[2].

Wenn die Importdatei eine Variable mit dem Namen "Otto[1]" enthält, wird nicht das erste Element der Arrayvariablen befüllt sondern eine neue Variable im Engineering System angelegt.

### Lokale ID von HMI-Verbindungen

Der Wert für die "Lokale ID" in den Eigenschaften einer HMI-Verbindung kann nicht verändert werden. Die lokale ID wird z.B für die Kommunikation über AR\_SEND benötigt. Um die "Lokale ID" für die Kommunikation verwenden zu können, gehen Sie wie folgt vor:

1. Öffnen Sie im Editor "Geräte & Netze" die Netzsicht.
2. Klicken Sie auf "Verbindungen".
3. Wählen Sie eine S7-Verbindung.
4. Wählen Sie im Kontextmenü der PLC den Befehl "Neue Verbindung hinzufügen".
5. Klicken Sie auf die Schnittstelle.
6. Legen Sie die "Lokale ID (hex)" fest.
7. Klicken Sie auf "Hinzufügen" und anschließend auf "Schließen".
8. Geben Sie für die neu angelegte Verbindung im Inspektorfenster unter "Eigenschaften > Allgemein" im Bereich Partner die IP-Adresse des HMI-Gerätes ein.
9. Projektieren Sie die für Kommunikation im HMI-Gerät die notwendigen Rohdatenvariablen.

### Variablen mit Datentyp DTL

Variablen die elementweise den Datentyp "DTL" verwenden, können nur lesend verwendet werden.

### 2.3.4.3 Meldesystem und Meldeanzeigen

#### Inhalt

Hinweise, die nicht mehr in der Onlinehilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Darstellung von Sonderzeichen in Meldetexten

Im Engineering System wird für die Projektierung von Meldetexten ein fester Zeichensatz verwendet. Dieser Zeichensatz bietet die Möglichkeit, verschiedenste Sonderzeichen in den Meldetexten zu verwenden.

Für die Darstellung in Runtime werden sprachspezifische Schriftarten verwendet, z. B. MS PGothic, SimSun. Die in Runtime verwendeten Schriftarten unterstützen nicht alle Sonderzeichen. Daher werden einige Sonderzeichen in Runtime nicht angezeigt.

#### Verwendung von Multiplexvariablen in Ausgabefeldern bei Meldetexten

Im Engineering System ist es möglich, in Ausgabefeldern von Meldetexten auch Multiplexvariablen zu verwenden. In Runtime führt dies zu einer fehlerhaften Darstellung der Meldung, da von den Basic Panels die Verwendung von Multiplexvariablen nicht unterstützt wird.

#### Parameter in Anwendermeldungen

Anders als in der Onlinehilfe beschrieben, ist es nicht möglich, für Anwendermeldungen Meldeparameter zu projektieren.

Im Inspektorfenster ist der Menüpunkt "Eigenschaften > Eigenschaften > Meldeparameter" nicht vorhanden.

#### 2.3.4.4 Systemfunktionen

##### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Systemfunktion "SimuliereVariable" auf Basic Panels

Wenn Sie auf einem Basic Panel die Systemfunktion "SimuliereVariable" in Verbindung mit einer kleinen Zykluszeit verwenden, kann das Bediengerät überlastet werden.

#### 2.3.4.5 Rezepturen

##### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Arrays in Rezepturelementen

Wenn Sie an Rezepturelementen einer Rezeptur sowohl ein Array als auch Elemente dieses Arrays projiziert haben, wird das Laden mit Datensätzen mit folgender Fehlermeldung abgebrochen: "290055: Import der Datensätze mit Fehler abgebrochen"

Verwenden Sie für Rezepturelemente der gleichen Rezeptur entweder nur die Arrays oder nur die Arrayelemente.

#### 2.3.4.6 Benutzerverwaltung

##### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### SIMATIC Logon für WinCC Runtime Advanced und Panels

Wenn Sie SIMATIC Logon verwenden um die Zugriffe auf ein Panel oder ein Gerät mit WinCC Runtime Advanced zu administrieren, müssen Sie beachten, dass in den Namen von Windows-Benutzergruppen und Windows-Benutzern die Zeichen '/' und '\' nicht verwendet werden dürfen.

### 2.3.4.7 Kommunikation

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Verbindungsunterbrechungen bei Mitsubishi Steuerungen

Nach mehrfachen Verbindungsunterbrechungen kann es zu der Situation kommen, dass in der Mitsubishi-Steuerung alle Verbindungsressourcen belegt sind und die Verbindung nicht mehr aufgebaut werden kann. Es wird empfohlen, im SPS-Programm der Steuerung diese Verbindungsressourcen zu kontrollieren und sie auch wieder freizugeben

#### Bereichszeiger "Datum/Uhrzeit" oder "Datum/Uhrzeit Steuerung"

Wenn Sie in der Kommunikation mit einer S7-1200 die Bereichszeiger "Datum/Uhrzeit" oder "Datum/Uhrzeit Steuerung" einsetzen, müssen sie in der Projektierung der PLC den Datentyp "DTL" verwenden.

#### Genauigkeit des Datentyps "DTL"

Der Datentyp "DTL" unterstützt Zeitangaben bis im Bereich von Nanosekunden. Da Panels Zeitangaben lediglich bis in den Bereich von Millisekunden unterstützen, ergeben sich bei der Verwendung an den Bereichszeigern folgende Einschränkungen:

- Bereichszeiger "Datum/Uhrzeit"  
Bei der Übertragung der Zeitangaben vom einem Panel zur Steuerung gilt als kleinste Zeiteinheit 1 Millisekunde. Der Wertebereich von Mikrosekunden bis Nanosekunden des Datentyps "DTL" wird mit Nullen befüllt.
- Bereichszeiger "Datum/Uhrzeit Steuerung"  
Bei der Übertragung der Zeitangaben von einer Steuerung zu einem Panel wird der Bereich von Mikrosekunden bis Nanosekunden ignoriert. Auf dem Panel wird die Zeitangabe bis einschließlich der Millisekunden weiter verarbeitet.

#### Beschränkte Anzahl von möglichen HMI-Verbindungen

Beim Übersetzen eines Geräts tritt eine Fehlermeldung auf, dass die Konfiguration der HMI-Verbindung im Editor "Geräte & Netze" ungültig ist. Die Ursache kann darin liegen, dass die maximale Anzahl der möglichen Verbindungen des HMI-Geräts bzw. der PLC überschritten ist.

Prüfen Sie die maximale Anzahl der möglichen Verbindungen. Informieren Sie sich bitte im Gerätehandbuch der von Ihnen verwendeten Geräte.

#### Kommunikation über Routing mit S7 300/400

Die Kommunikation von Verbindungspartner in unterschiedlichen Subnetzen ist über Routing mit folgenden Verbindungen möglich: PROFINET, PROFIBUS, MPI.

### Verwendung von PROFINET IO bei Panel-Bediengeräten

Bei der Verwendung von PROFINET IO zur Verbindung der Direkttasten und LEDs von Bediengeräten mit der Steuerung können Sie bei der Projektierung in HW Konfig einen Offset für den Adressbereich der Eingänge und der Ausgänge festlegen.

Bei der Verwendung einer PROFINET IO-fähigen CPU der 400er Baureihe in Verbindung mit einem der unten aufgeführten Bediengeräte gilt folgende Einschränkung:

Der Offset für den Anfang des Adressbereichs der Eingänge darf nicht größer sein als der Offset für den Anfang des Adressbereichs der Ausgänge.

Die Einschränkung gilt für folgende Bediengeräte:

- OP 177B
- OP 277
- Mobile Panel 177

Für die Projektierung der Adressparameter öffnen Sie die Steuerung mit der CPU der 400er Baureihe in HW Konfig. Selektieren Sie im Stationsfenster von HW Konfig das über PROFINET IO verbundene Bediengerät. Im unteren Bereich des Stationsfensters wird in der Detailansicht eine Tabelle mit den Eigenschaften des Bediengeräts angezeigt. Selektieren Sie in der Tabelle die Zeile mit den Adressen des Bediengeräts und öffnen Sie die Objekteigenschaften über das Kontextmenü.

Aktivieren Sie im Dialog "Objekteigenschaften" die Registerkarte "Adressen". Unter "Eingänge > Anfang" projektieren Sie den Offset für die Eingänge. Unter "Ausgänge > Anfang" projektieren Sie den Offset für die Ausgänge.

### Wertebereichsüberschreitung bei Mitsubishi MC und Mitsubishi FX

Die Kommunikationstreiber Mitsubishi MC und Mitsubishi FX überprüfen bei einigen Datentypen nicht, ob der Wert einer Rezepturvariablen den Wertebereich der Steuerungsvariablen überschreitet. Betroffen sind die Datentypen:

- 4-bit block
- 12-bit block
- 20-bit block
- 24-bit block
- 28-bit block

### Bereichszeiger Koordinierung in einer OPC-Verbindung

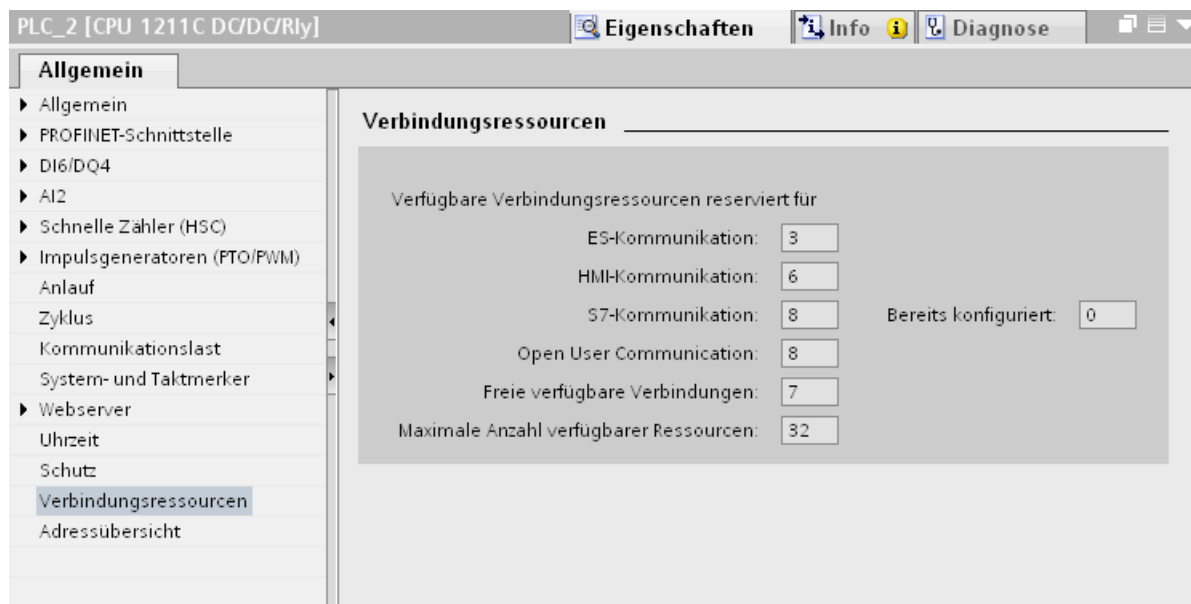
In einer OPC-Verbindung kann grundsätzlich achtmal der Bereichszeiger Koordinierung verwendet werden. Wenn Sie eine OPC-Verbindung projektieren und über "Hinzufügen" automatisch eine weitere OPC-Verbindung anlegen, wird in der neu angelegten Verbindung der Bereichszeiger Koordinierung nur einmal angezeigt. In diesem Fall sollten Sie den Kommunikationstreiber der Verbindung ändern. Wenn Sie anschließend als Kommunikationstreiber wieder OPC einstellen, steht der Bereichszeiger Koordinierung wieder achtmal zur Verfügung.

## Verbindungsressourcen: SIMATIC S7 1200

Die Steuerung SIMATIC S7 1200 verfügt über 6 Verbindungsressourcen für die HMI-Kommunikation.

Wie viele HMI-Verbindungen Sie tatsächlich projektieren können, hängt davon ab, welche Bediengeräte Sie mit der SIMATIC S7 1200 verbinden.

Ein HMI-Panel belegt 1 Verbindungsressource pro Verbindung.



## IP Einstellungen und Gerätenamen einer Steuerung im Control Panel des Bediengeräts ändern

Auf dem Bediengerät ist das Control Panel im Menü "Service and Commissioning > IP-Adaptation" geöffnet. Wenn Sie die IP Einstellungen oder den Gerätenamen einer Steuerung ändern wollen, beachten Sie Folgendes:

Im Engineering System haben Sie zuvor im Inspektorfenster der Steuerung unter "Eigenschaften > Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen" folgende Optionen aktiviert:

- "IP-Adresse auf anderem Weg beziehen" und
- "PROFINET-Gerätenamen auf anderem Weg beziehen".

## HMI-Verbindungen in WinCC V12

HMI-Verbindungen zu Steuerungen SIMATIC S7-1200 mit der Firmware-Version kleiner als V2.0 ist in WinCC V12 nicht möglich.

## Verbindungen über PROFIBUS DP

Wenn eine Verbindung über PROFIBUS DP zwischen einer Steuerung und einem Bediengerät unterbrochen und dann wieder aufgebaut wird, dann werden sporadisch alle weiteren PROFIBUS DP-Verbindungen im Kommunikationsnetz abgebrochen und wieder aufgebaut.

Schalten Sie den getrennten Teilnehmer spannungslos, bevor Sie den Teilnehmer wieder verbinden.

### "Setzen der IP-Suite (Adresse) der PLC im Control Panel" bei SIMATIC S7-1200 V1

Die Funktion "Setzen der IP Suite (Adresse) der PLC im Control Panel" ist für folgende Steuerungen nicht freigegeben:

- SIMATIC S7-1200 V1

### Wechsel einer Verbindung

Bei einem Wechsel der Verbindung von einem Bediengerät zu einer Steuerung SIMATIC S7-300/400, zu einer Steuerung SIMATIC S7-1500 oder SIMATIC S7-1200, kann es zum Abbruch der Verbindung kommen.

Beachten Sie folgende Einstellungen in den Steuerungen SIMATIC S7 1500 oder SIMATIC S7 1200:

- Absolute Addressierung der Variablen
- Option "Disable PUT-GET Kommunikation" muss ausgewählt sein
- Schutzstufe "Kompletter Schutz" darf nicht projiziert sein

## 2.3.5 Übersetzen und Laden

### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

### Übersetzen und Laden

Wenn beim Übersetzen interne Fehler oder Warnungen auftreten, übersetzen Sie das Projekt vollständig über den Befehl "Übersetzen > Software (komplett übersetzen)" im Kontextmenü des Bediengeräts.

Bevor Sie mit Ihrem Projekt in den Produktivbetrieb gehen, übersetzen Sie das Projekt vollständig über den Befehl "Übersetzen > Software (komplett übersetzen)" im Kontextmenü des Bediengeräts.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch alle geänderten Bausteine über den Befehl "Übersetzen > Software" im Kontextmenü.

### Speichern des WinCC-Projekts

Wenn Sie in WinCC ein Projekt mit "Speichern unter..." speichern, so hat dies keine Auswirkung auf die Benennung der für die Geräte generierten Runtime-Projekte. Wenn Sie



im Dialog "Erweitertes Laden" den Zielpfad der Geräte nicht anpassen, werden die Runtime-Projekte auf den Zielgeräten überschrieben.

### **Einstellungen beim Aktualisieren des Betriebssystems**

Wenn Sie von WinCC aus den Befehl "Online > Bediengeräte Wartung > Betriebssystem aktualisieren" wählen, können sie Einstellung wie den Typ der PG/PC-Schnittstelle oder die Baudrate nicht verändern. Es werden immer die beim letzten Laden verwendeten Einstellungen verwendet.

Um Einstellungen zu ändern öffnen Sie über "Online > Erweitertes Laden in Gerät" den Dialog "Erweitertes Laden" und verändern Sie die Einstellungen. Wenn Sie die Schaltfläche "Laden" drücken, werden die geänderten Einstellungen gespeichert.

Alternativ können Sie eine Aktualisierung des Betriebssystems mit geänderten Einstellungen mit ProSave durchführen. Sie starten ProSave über das Windows-Startmenü "Siemens Automation > Options and Tools > HMI Tools > SIMATIC ProSave".

### **Fehlerhafte Installation von ProSave**

Wenn Sie beim Laden der Daten in ein Zielgerät oder bei der Bediengerätewartung eine Fehlermeldung bezüglich der Installation von ProSave erhalten, können Sie den Fehler nicht mit der Reparaturfunktion des Setup beheben. Deinstallieren Sie ProSave über die Systemsteuerung. Starten Sie anschließend das Setup und installieren Sie die Komponente "ProSave" erneut.

### **Prüfen der Adressparameter**

Beim Übersetzen eines Bediengerätes in der Projektnavigation über den Befehl "Übersetzen > Software" im Kontextmenü werden die Adressparameter des Bediengeräts, wie z. B. die IP-Adresse, nicht überprüft. Um sicherzugehen, dass auch die Adressparameter überprüft werden, müssen Sie das Bediengerät im Editor "Geräte & Netze" über die Schaltfläche "Übersetzen" in der Symbolleiste übersetzen.

### **Fehlermeldung beim Laden von Daten auf die PLC**

Ein Panel und eine PLC sind verbunden und kommunizieren miteinander.

Wenn beim Laden von Daten auf die PLC vom Panel aus auf eine Variable zugegriffen wird, wird am Panel eine Fehlermeldung ausgegeben.

### **Verzögerte Reaktion im Dialog "Erweitertes Laden"**

Wenn im Dialog "Erweitertes Laden" die Einstellungen für "Typ der PG/PC Schnittstelle" und "PG/PC-Schnittstelle" nicht mit den Einstellungen auf dem Bediengerät übereinstimmen, dann kann dies dazu führen, dass die Applikation bis zu eine Minute nicht reagiert.

### Erweitertes Laden bei einer S7-1200 und einem Comfort Panel

Eine PLC S7-1200 und ein Comfort Panel befinden sich im gleichen physikalischen Netz wie der PG/PC. Sie öffnen den Dialog "Erweitertes Laden" für das Comfort Panel.

Wenn Sie die Option "Alle erreichhbaren Teilnehmer anzeigen" aktivieren, kann es vorkommen, dass die Applikation nicht mehr reagiert.

### OP77A, OP73, TP177A: Laden eines Projekts

Beim Laden eines Projekts auf ein Bediengerät, kann es dazu kommen, dass trotz aktiviertem "Remote Transfers" im Panel die Runtime nicht automatisch beendet wird.

In diesem Fall stoppen Sie Runtime und stellen den Transfermodus am Bediengerät manuell ein.

### Laden einer SIMATIC HMI Applikation in einer PC-Station

Unter folgenden Umständen kann es zu einer Fehlermeldung beim ersten Laden einer SIMATIC PC-Station kommen:

- im Projekt ist eine SIMATIC HMI Applikation in einer PC-Station projiziert
  - WinCC Runtime Advanced
  - WinCC Runtime Professional
  - WinCC Standby
  - oder WinCC Client
- die Eigenschaft "S7RTM ist installiert" ist aktiviert.

Bevor Sie das erste Mal eine SIMATIC PC-Station laden, markieren Sie in der Projektnavigation das projizierte Gerät HMI\_RT (WinCC...). Öffnen Sie den Dialog "Erweitertes Laden" und wählen Sie die entsprechenden Schnittstellen- und Parametereinstellungen. Drücken Sie "Laden".

Danach laden Sie wie gewohnt die PC-Station.

### Projekttransfer über USB

Wenn Sie mehr als ein Bediengerät an ihren Projektierungs-PC über USB angeschlossen haben, dann ist ein Projekttransfer nur auf das zuletzt angeschlossene Bediengerät möglich.

### Delta-Ladefähigkeit

Wenn Sie Änderungen an einem UDT vornehmen, geht die Delta-Ladefähigkeit des Runtime Professional-Projekts verloren.

Laden Sie das Projekt komplett neu.

### Öffnen von Projektdateien

Wenn Sie die Datei "HmiIRTm.exe" ausführen, dann öffnet sich ein Dialog mit der Abfrage ob Sie eine Projektdatei (.fwc) öffnen wollen.

Folgende Möglichkeiten stehen Ihnen zur Auswahl:

- "Ja": Öffnet einen Dialog in dem Sie eine Projektdatei (.fwc) auswählen.
- "Nein": Dialog wird geschlossen.

## 2.3.6 Runtime

### 2.3.6.1 Hinweise zum Betrieb in Runtime

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Sonderzeichen in der Benutzeranzeige

Bei der Eingabe eines Namens oder des Kennworts in der Benutzeranzeige können Sonderzeichen wie / " § \$ % & ' ? nicht verwendet werden.

#### Sprachenverhalten - Layout der Bildschirmtastatur

Das Layout der Bildschirmtastatur wird nicht umgeschaltet, wenn die Runtime-Sprache gewechselt wird.

#### Variablenwerte überschreiben die maximale Länge

Sie geben eine Zeichenkette in eine String Variable über ein E/A-Feld ein. Wenn die Zeichenfolge die projektierte Länge der Variablen überschreitet, wird die Zeichenkette auf die projektierte Länge gekürzt.

#### Leere Meldetexte

Die Runtime läuft mit einem Projekt. Das Projekt ist auf einem Netzlaufwerk gespeichert.

Bei Unterbrechungen zum Netzlaufwerk kommt es gegebenenfalls vor, dass die Runtime versucht Meldetexte vom Netzlaufwerk zu laden.

Wenn die Verbindung getrennt ist, bleibt das Meldefenster bzw. die Meldeanzeige leer.

Um das Verhalten zu vermeiden, kopieren Sie das Projekt auf ein lokales Laufwerk bevor Sie das Projekt in der Runtime starten.

#### Komplettes Laden im Service Mode

Wenn von der Engineering Station ein "Komplett Laden" auf die OS im Service Mode notwendig ist, wird automatisch Runtime beendet und danach erneut gestartet.

Das Projekt ist dann nicht mehr im Service Mode.

Wenn in diesem Zustand die Stromzufuhr unterbrochen wird, wird an der OS WinCC Runtime nicht mehr automatisch gestartet.

Abhilfe:

1. Schalten Sie das Projekt manuell in den Service Mode, nachdem Sie das "Komplett Laden" durchgeführt haben.
2. Beenden Sie das Projekt manuell
3. Aktivieren den Service Mode.
4. Starten Runtime erneut über das Surrogate-Icon in der Taskbar.

### 2.3.6.2 Hinweise zum Betrieb von Panels in Runtime

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

#### Verwendung des Mausekorns in Runtime

Auf allen Panels wird die Verwendung des Mausekorns in Runtime nicht unterstützt.

#### Basic Panels: Verbindungen zu S7-1200 und S7-1500 bei Backup/Restore

Wenn Sie die Funktion "Backup/Restore" verwenden, sind zeitgleich maximal zwei Verbindungen von Basic Panels zu folgenden Steuerungen möglich:

- SIMATIC S7-1200
- SIMATIC S7-1500

#### Basic Panels: Backup auf der Speicherkarte der PLC

Sie erstellen die Backup-Datei "A.psb" auf der Speicherkarte der PLC. Beim Erstellen des Backups tritt ein Fehler auf z. B. ein Verbindungsabbruch.

Dadurch entsteht eine fehlerhafte Datei auf der Speicherkarte der PLC. Eine solche Datei hat als Präfix "~\$".

Löschen Sie die Datei mit dem Präfix "~\$", wenn Sie ein Backup unter dem gleichen Namen "A.psb" erneut speichern wollen.

#### Basic Panels: Panel Data Storage und S7-1500F

Die Funktion "Panel Data Storage" PDS kann auf Basic Panels im Zusammenspiel mit S7-1500F nicht verwendet werden, wenn das Kennwort für die Schutzstufe "Full access incl. fail-safe" verwendet wird.

## Basic Panels auf WinCC V12 hochrüsten

Bevor Sie Basic Panels von der Version V11 auf die Version V12 hochrüsten, übertragen Sie das Image des V11 SP2 Updates 5 oder höher auf die Geräte.

Wählen Sie im Dialog "SIMATIC ProSave [OS-Update]" die Einstellung "Rücksetzen auf Werkseinstellungen". Damit starten Sie immer eine funktionsfähige Aktualisierung des Images.  
Betroffene Geräte:

- KP300 Basic mono PN,
- KP400 Basic color PN
- und KTP400 Basic color PN .

## 2.3.7 Bediengeräte

### 2.3.7.1 Hinweise zu Bediengeräten

#### Inhalt

Hinweise, die nicht mehr in der Online-Hilfe berücksichtigt werden konnten und wichtige Hinweise zu Produkt-Eigenschaften.

Geht PC in den Zustand Hibernate oder Standby, während der Transfer läuft, ist der Panel-Zustand nach der Übertragungsunterbrechung nicht definiert.

#### Mehrtastenbedienung

Durch eine Mehrtastenbedienung können unbeabsichtigte Aktionen ausgelöst werden:

- Wenn Sie ein Tasten-Gerät verwenden, dürfen Sie nicht mehr als zwei Funktionstasten gleichzeitig betätigen.
- Wenn Sie ein Touch-Gerät, einen Standard-PC oder einen Panel-PC verwenden, dürfen Sie immer nur eine Funktionstaste oder Schaltfläche gleichzeitig bedienen.

#### TS Adapter mit Ethernet Schnittstelle

Wenn ein Bediengerät über Ethernet und einen TS Adapter verbunden ist, kann es nicht auf Werkseinstellungen zurückgesetzt werden.

#### Simulation der Basic Panels

Für die Ausgabe einer externen Variablen verwenden Sie ein Ausgabefeld in einem Meldetext. Bei der Simulation wird dann der Inhalt dieses Ausgabefelds immer mit "0" angezeigt.

#### Simulation mit realer Steuerungsanbindung

Der von der Simulation verwendete Zugangspunkt ist unabhängig von den Einstellungen im Engineering System und kann nur über das Tool " PG/PC Interface einstellen" in der

Systemsteuerung verändert werden. Wenn die Verbindung zu einer Steuerung direkt nach dem Start der Simulation mit Meldung 140001 wieder abgebaut wird, sollten Sie den von der Simulation verwendeten Zugangspunkt mit "PG/PC Interface einstellen" überprüfen.

1. Doppelklicken Sie in der Systemsteuerung auf "PG/PC Interface einstellen". Ein Dialog wird geöffnet.
2. Wählen Sie im Bereich "Zugangspunkt der Applikation" "S7ONLINE" standardmäßig für HMI.
3. Wählen Sie im Bereich "Benutzte Schnittstellenparametrierung" die Schnittstelle.
4. Beenden Sie den Dialog "PG/PC Interface einstellen" mit OK

### Laden von Projekten ohne Rezepturdatensätze

In einem Projekt verwenden Sie Rezepturen. Sie übertragen das Projekt auf ein Basic Panel jedoch ohne die Rezepturdatensätze.

Wenn Sie die Struktur der Rezeptur im Engineering System geändert haben und auf dem Gerät bereits Rezepturdatensätze vorhanden waren, dann kann es zu Inkonsistenzen kommen.

Überprüfen Sie in diesem Fall die Konsistenz der Datensätze. Auf dem Gerät wird nicht bei allen strukturellen Änderungen ein Hinweis ausgegeben.

### Gleitpunktzahlen auf MP 277, MP 377, TP 177B 4" und CP4

Auf den Bediengeräten MP 277, MP 377, TP 177B 4" und CP4 werden Gleitpunktzahlen nur im Bereich von  $10^{-293}$  ...  $10^{+307}$  korrekt angezeigt. Befindet sich der Wert der Variable außerhalb dieses Bereichs wird er als 0 angezeigt.

### USB Gerätetreiber unter Windows XP

Wenn die Verbindung zwischen einem Projektierungs-PC mit Windows XP und einem Comfort Panel über USB erfolgt, werden nach einem Neustart des Bediengeräts möglicherweise die S7-USB-Treiber neu installiert. Dabei werden möglicherweise die Geräte-Einstellungen nicht wieder hergestellt.

### Mobile Panels V2

Wenn Sie in einem Projekt Mobile Panels V2 verwenden, ist es nicht möglich, das Projekt mit WinCC V11 SP1 zu öffnen. Dies betrifft Projekte mit folgenden Geräten:

- Mobile Panel 277F IWLAN (RFID Tag)
- Mobile Panel 277F IWLAN V2
- Mobile Panel 277 IWLAN V2

### Variable für "Zonen-ID/Anschlusspunkt-ID" eines Mobile Panel 277 IWLAN V2

Die für "Zonen-ID/Anschlusspunkt-ID" verwendete Variable muss bei Mobile Panel 277 IWLAN V2 Geräten den Datentyp INT besitzen. Achten Sie bei der Migration eines Projektes darauf den Datentyp gegebenenfalls anzupassen.

### Bediengeräte mit Betriebssystem Windows CE 5.0 und höher

Aufgrund einer geänderten Sicherheitseinstellung bei der Client-Server-Kommunikation darf der Zeitunterschied zwischen Bediengerät (Client) und einem PC (Server) nicht größer als 1 Tag sein. Wenn Sie z. B. Rezepturdaten vom Bediengerät auf ein Netzlaufwerk sichern, stellen Sie sicher, dass die Uhrzeit auf PC (Server) und Bediengerät (Client) korrekt eingestellt ist.

### Bediengeräte mit hoher Kommunikationslast

Wenn bei einem Panel viele Verbindungen zu Steuerungen oder anderen Bediengeräten projektiert sind, sollte die S7-Diagnose deaktiviert sein. Andernfalls besteht die Gefahr der Überlastung des Panel.

### Gerätetausch im Engineering System

Im Engineering System tauschen Sie ein Gerät mit projektierten LED-Tasten gegen ein Gerät ohne LED-Tasten. Nach dem Transfer der Projektdaten auf das Gerät startet die Runtime nicht.

Löschen Sie daher die LED-Tasten-Projektierung, bevor Sie den Gerätetausch durchführen.

### Einschränkungen beim Bediengerät MP 377 15" Touch daylight readable

In WinCC V12 werden für das Bediengerät MP 377 15" Touch daylight readable folgende Funktionen nicht unterstützt:

- Option: Sm@rtServer
- Systemfunktion: SetzeUndLeseHelligkeit
- Direkttasten

### Basic Panels auf WinCC V12 hochrüsten

Bevor Sie Basic Panels von der Version V11 auf die Version V12 hochrüsten, übertragen Sie das Image des V11 SP2 Updates 5 oder höher auf die Geräte.

Wählen Sie im Dialog "SIMATIC ProSave [OS-Update]" die Einstellung "Rücksetzen auf Werkseinstellungen".

Damit starten Sie immer eine funktionsfähige Aktualisierung des Images.

Betroffene Geräte:

- KP300 Basic mono PN
- KP400 Basic color PN
- KTP400 Basic color PN

### **Verbindungswechsel im Control Panel bei Basic Panels**

Wenn Sie die Funktion "Override projected connection information" verwenden, gilt folgende Einschränkung:

Der Verbindungswechsel im Control Panel eines Basic Panels von einer Steuerung ohne Schutzstufe zu einer Steuerung mit der Schutzstufe "Kompletter Schutz" ist nicht möglich.



# Installation

## 3.1 Systemvoraussetzungen für die Installation

### 3.1.1 Hinweise zu Lizenzen

#### Bereitstellung der Lizenzen

Die Lizenzen für die Produkte des TIA-Portals werden in der Regel auf einem Installationsdatenträger ausgeliefert und über den Automation Licence Manager während des Installationsprozesses des TIA-Portals automatisch übertragen.

Bevor Sie das TIA-Portal deinstallieren, müssen Sie noch benötigte Lizenzen transferieren und sichern. Verwenden Sie dazu den Automation License Manager.

#### Bereitstellung des Automation License Managers

Der Automation License Manager wird auf dem Installationsdatenträger mitgeliefert und während des Installationsprozesses automatisch übertragen.

Wenn Sie das TIA-Portal deinstallieren, bleibt der Automation License Manager weiterhin auf Ihrem System installiert.

#### Arbeiten mit dem Automation License Manager

Der Automation License Manager ist ein Produkt der Siemens AG und wird zur Handhabung von License Keys (technische Repräsentanten von Lizenzen) eingesetzt.

Software-Produkte, für deren Betrieb License Keys notwendig sind, wie z. B. das TIA-Portal, melden den Bedarf an License Keys automatisch beim Automation License Manager an. Wenn der Automation License Manager für diese Software einen gültigen License Key findet, so kann die Software entsprechend der mit diesem License Key verbundenen Lizenz-Nutzungsbedingungen verwendet werden.

---

#### Hinweis

Weiterführende Informationen, wie Sie mit dem Automation License Manager Ihre Lizenzen verwalten können, finden Sie in der mit dem Automation License Manager mitgelieferten Dokumentation.

---

### **3.1 Systemvoraussetzungen für die Installation**

#### **Siehe auch**

- Hinweise zu den Systemvoraussetzungen (Seite 76)
- Installation starten (Seite 94)
- Installierte Software anzeigen (Seite 97)
- Installierte Produkte modifizieren oder hochrüsten (Seite 98)
- Installierte Produkte reparieren (Seite 100)
- Deinstallation starten (Seite 101)
- Installationsprotokoll (Seite 93)

### **3.1.2 Hinweise zu den Systemvoraussetzungen**

#### **Systemvoraussetzungen für einzelne Produkte**

Je nachdem, welche Produkte Sie installieren möchten, gelten unterschiedliche Systemvoraussetzungen. Beachten Sie daher die individuellen Systemvoraussetzungen Ihrer Produkte.

Möchten Sie mehrere Produkte installieren, berücksichtigen Sie bitte die Systemvoraussetzungen des Produkts mit den höchsten Anforderungen.

#### **Anzeigen von PDF-Dateien**

Um mitgelieferte PDF-Dateien zu lesen, benötigen Sie einen PDF Reader, der das PDF-Format 1.7 unterstützt, z. B. Adobe (R) Reader Version 9.

#### **Anzeigen der Welcome Tour**

Um die Welcome Tour zum TIA-Portal zu starten, benötigen Sie den Adobe (R) Flashplayer ab Version 9.

#### **Siehe auch**

- Hinweise zu Lizenzen (Seite 75)
- Installation starten (Seite 94)
- Installierte Software anzeigen (Seite 97)
- Installierte Produkte modifizieren oder hochrüsten (Seite 98)
- Installierte Produkte reparieren (Seite 100)
- Deinstallation starten (Seite 101)

### 3.1.3 Systemvoraussetzungen STEP 7 Basic

#### 3.1.3.1 Lizenzierung von STEP 7

##### Einführung

Für die Lizenzierung der folgenden Editionen von STEP 7 benötigen Sie einen License Key:

- STEP 7 Basic
- STEP 7 Professional

Sie können den entsprechenden License Key für STEP 7 bei der Installation gleich mitinstallieren oder nach der Installation mit dem Automation License Manager übertragen.

##### Lizenzen für STEP 7

Folgende Lizenzen mit den entsprechenden License Keys stehen Ihnen zur Verfügung:

- STEP 7 Basic
- STEP 7 Professional
- STEP 7 Professional Combo

##### Gültigkeit der License Keys für ältere Versionen von STEP 7

Mit einem gültigen License Key für die V12.x von STEP 7 Professional und STEP 7 Professional Combo können auch ältere Versionen von STEP 7 ohne Einschränkungen betrieben werden. Genauere Informationen dazu entnehmen Sie der nachfolgenden Tabelle:

| Edition                   | Lizenz                    | Gültig für  |
|---------------------------|---------------------------|---|
| STEP 7 Basic V12.x        | STEP 7 Basic              | <ul style="list-style-type: none"> <li>• STEP 7 Basic V12.x</li> </ul>  |
| STEP 7 Professional V12.x | STEP 7 Professional       | <ul style="list-style-type: none"> <li>• STEP 7 Basic V10.5</li> <li>• STEP 7 Basic V11.0</li> <li>• STEP 7 Professional V11.0</li> </ul> |
| STEP 7 Professional V12.x | STEP 7 Professional Combo | <ul style="list-style-type: none"> <li>• STEP 7 Basic V10.5</li> <li>• STEP 7 Basic V11.0</li> <li>• STEP 7 Professional V11.0</li> </ul> |

##### Starten ohne gültigen License Key

Wenn Sie eine Edition von STEP 7 ohne gültigen License Key starten, weist Sie das System mit einer Meldung auf den nicht lizenzierten Modus hin. Sie haben die Möglichkeit, einmalig eine Trial License zu aktivieren. Diese hat jedoch nur eine begrenzte Gültigkeit und läuft nach 21 Tagen ab.

Wenn die Trial License abgelaufen ist, können folgende Fälle auftreten:

3.1 Systemvoraussetzungen für die Installation

- Auf dem betreffenden PC war STEP 7 noch nie lizenziert:
  - In STEP 7 lassen sich keine lizenzpflichtigen Aktionen mehr durchführen.
- Auf dem betreffenden PC war STEP 7 bereits lizenziert:
  - Der nicht lizenzierte Modus wird alle 10 Minuten und bei jeder lizenzpflichtigen Aktion durch ein quittierpflichtiges Fenster angezeigt.

**Lizenzbedarf bei Simulation**

Wenn Sie in STEP 7 die Simulation über den Menübefehl "Online > Simulation" starten, benötigen Sie keine weiteren Lizenzen.

Wenn folgende Bedingungen erfüllt sind, benötigen Sie auch bei der Simulation die entsprechenden Lizenzen für von Ihnen installierte Edition von STEP 7:

- Die Engineering Station ist mit einer PLC verbunden.
- Die Verbindung zur PLC ist projektiert und aktiv.

**Siehe auch**

Umgang mit Lizenzen und License Keys (Seite 78)

**3.1.3.2 Umgang mit Lizenzen und License Keys**

**Einleitung**

Zur Nutzung von STEP 7 Basic und STEP 7 Professional benötigen Sie jeweils einen gültigen License Key.

**Installation von License Keys**

Bei der Installation von STEP 7 Basic wird der benötigte License Key vom Setup automatisch mit installiert. Bei der Installation von STEP 7 Professional werden Sie am Ende des Setups aufgefordert, die Lizenz vom mitgelieferten Datenträger auf ihren PC zu übertragen.

Falls Sie zusätzliche License Keys installieren möchten, müssen Sie dazu den Automation License Manager nutzen.

Wenn Sie eine Lizenz installieren, wird der zugehörige License Key vom Ablageort der License Keys entfernt.

**ACHTUNG**

**Zerstörung von License Keys durch Kopieren**

Ein License Key ist nicht kopierbar. Das verwendete Kopierschutzverfahren verhindert das Kopieren der License Keys. Beim Versuch, einen License Key zu kopieren, wird dieser zerstört!

## Deinstallation von License Keys

Die Deinstallation von License Keys erfolgt immer mit dem Automation License Manager. Sie deinstallieren einen License Key in folgenden Fällen:

- Wenn Sie Daten sichern.
- Wenn Sie die Lizenz nicht mehr benötigen.

Sie können eine gültige Lizenz auch an einem anderen PC oder Bediengerät verwenden.

## Datensicherung

Zur Datensicherung am Bediengerät oder als Backup beim Gerätetausch deinstallieren Sie die License Keys am Bediengerät. Rufen Sie dazu den Automation License Manager auf und sichern Sie den zu deinstallierenden License Key auf einen anderen Ablageort.

|   |
|---|
| <b>ACHTUNG</b>  |
| <b>Zerstörung von License Keys auf PCs</b>  |
| Deinstallieren Sie in folgenden Fällen zuerst alle License Keys:  |
| <ul style="list-style-type: none"><li>• Bevor Sie die Festplatte formatieren.</li><li>• Bevor Sie die Festplatte komprimieren.</li><li>• Bevor Sie die Festplatte restaurieren.</li><li>• Bevor Sie ein Optimierungsprogramm starten, das feste Blöcke verschiebt.</li><li>• Bevor Sie ein neues Betriebssystem installieren.</li></ul> |
| Lesen Sie die Beschreibung des Automation License Manager ("Start > Siemens Automation > Dokumentation"). Beachten Sie alle Warnhinweise und Hinweise.  |

Bei PC-basierten Bediengeräten sowie beim Einsatz des Automation License Manager auf den nicht PC-basierten Bediengeräten ist der Ablageort der License Keys multilizenzfähig. Multilizenzfähig bedeutet, Sie können am gleichen Ablageort mehrere Lizenzen des gleichen Typs ablegen. Nutzen Sie beim Backup einen einzigen Ablageort für alle auf einem Bediengerät vorhandenen License Keys.

|  |
|--|
| <b>ACHTUNG</b>   |
| Behalten Sie auf jeden Fall den Original-Ablageort der License Keys. |

### Ungültige Lizenz nach Zeitonenwechsel

Die installierte Lizenz funktioniert im folgenden Fall nicht mehr.

- Wenn Sie auf einem PC die Zeitzone folgendermaßen umstellen:  
Von einer auf die volle Stunde bezogenen Zeit auf eine Zeit, die sich nicht auf eine volle Stunde bezieht.

**Beispiel:** Sie stellen die Zeitzone von GMT +3:00 auf GMT +3:30 Uhr um.

Um dieses Verhalten zu umgehen, deinstallieren Sie den License Key mit dem Automation License Manager in der Zeitzoneneinstellung, die bei der Installation des License Keys eingestellt war.

Dieses Verhalten betrifft nicht die Trial License.

### Defekte Lizenz

Eine Lizenz ist in folgenden Fällen defekt:

- Wenn der License Key auf dem Ablageort nicht mehr zugänglich ist.
- Wenn der License Key beim Übertragen auf das Ziellaufwerk verschwindet.

Sie können die defekte Lizenz mit Hilfe des Automation License Manager reparieren lassen. Benutzen Sie dazu die Funktion "Wiederherstellen" bzw. den "Wiederherstellen-Assistenten" des Automation License Manager. Für die Wiederherstellung ist es erforderlich, Kontakt mit dem Customer Support aufzunehmen.

Nähere Hinweise hierzu finden Sie im Internet unter: <http://support.automation.siemens.com> (<http://support.automation.siemens.com>)

### Siehe auch

Lizensierung von STEP 7 (Seite 77)

### 3.1.3.3 Software- und Hardwarevoraussetzungen STEP 7

#### Voraussetzungen für die Installation

Die folgende Tabelle zeigt die minimalen Software- und Hardwarevoraussetzungen, die für die Installation des Softwarepakets "SIMATIC STEP 7 Basic" erfüllt sein müssen:

| Hardware/Software          | Voraussetzung                         |
|----------------------------|---------------------------------------|
| Prozessor                  | 2,0 GHZ CORE 2 DUO                    |
| RAM                        | 1 GB (Windows XP)<br>2 GB (Windows 7) |
| Freier Festplattenspeicher | 2 GB auf Systemlaufwerk "C:"          |

| Hardware/Software   | Voraussetzung  |
|---------------------|--|
| Betriebssysteme *   | <p><b>Windows XP (32 Bit)</b></p> <ul style="list-style-type: none"> <li>• Windows XP Home SP3</li> <li>• Windows XP Professional SP3</li> </ul> <p><b>Windows 7 (32 Bit)</b></p> <ul style="list-style-type: none"> <li>• Windows 7 Home Premium</li> <li>• Windows 7 Home Premium SP1</li> <li>• Windows 7 Professional</li> <li>• Windows 7 Professional SP1</li> <li>• Windows 7 Enterprise</li> <li>• Windows 7 Enterprise SP1</li> <li>• Windows 7 Ultimate</li> <li>• Windows 7 Ultimate SP1</li> </ul> <p><b>Windows 7 (64 Bit)</b></p> <ul style="list-style-type: none"> <li>• Windows 7 Home Premium</li> <li>• Windows 7 Home Premium SP1</li> <li>• Windows 7 Professional</li> <li>• Windows 7 Professional SP1</li> <li>• Windows 7 Enterprise</li> <li>• Windows 7 Enterprise SP1</li> <li>• Windows 7 Ultimate</li> <li>• Windows 7 Ultimate SP1</li> </ul> |
| Grafikkarte         | 32 MB RAM<br>24 Bit Farbtiefe  |
| Bildschirmauflösung | 1024x768   |
| Netzwerk            | Ab Ethernet 10 Mbit/s  |
| Optisches Laufwerk  | DVD-ROM  |

\* Weitere Informationen zu den Betriebssystemen finden Sie in der Hilfe zu Microsoft Windows oder auf der Homepage von Microsoft.

## Empfohlene Hardware

Die folgende Tabelle zeigt die empfohlene Hardware für den Betrieb von STEP 7.

| Hardware           | Voraussetzung  |
|--------------------|--|
| Rechner            | SIMATIC FIELD PG M2 PREMIUM (oder vergleichbarer PC) |
| Prozessor          | 2,2 GHZ CORE 2 DUO (T7500)                           |
| RAM                | 1X2GB DDR2 RAM                                       |
| Festplatte         | 250GB S-ATA HDD                                      |
| Bildschirm         | 15" SXGA+ DISPLAY (1400 X 1050)                      |
| Optisches Laufwerk | DL MULTISTANDARD DVD RW                              |

### Unterstützte Virtualisierungsplattformen

Sie können das Softwarepaket "SIMATIC STEP 7 Basic" in einer virtuellen Maschine installieren. Verwenden Sie dazu eine der folgenden Virtualisierungsplattformen:

- VMware vSphere Hypervisor (ESXi) 5
- VMware Workstation 8
- VMware Player 4
- Microsoft Windows Server 2008 R2 SP1 Hyper-V

Folgende Betriebssysteme können diesen Virtualisierungsplattformen als Host-Betriebssystem dienen:

- Windows 7 Professional/Ultimate/Enterprise (32-Bit)
- Windows 7 Professional/Ultimate/Enterprise (32-Bit)
- Windows Server 2008 R2 (64 Bit)

Innerhalb der gewählten Virtualisierungsplattform können Sie die folgenden Gast-Betriebssysteme verwenden, um "SIMATIC STEP 7 Basic" zu installieren:

- Windows 7 Professional/Ultimate/Enterprise (32-Bit)
- Windows 7 Professional/Ultimate/Enterprise (64-Bit)

---

#### Hinweis

- Für die Gast-Betriebssysteme gelten dieselben Hardware-Anforderungen wie für die jeweiligen TIA-Produkte selbst.
  - Der Betreiber der Anlage muss dafür Sorge, dass den Gast-Betriebssystemen ausreichend Systemressourcen zur Verfügung gestellt werden.
  - Für den Einsatz von HyperV-Server und ESXi wird die von den Herstellern zertifizierte Hardware empfohlen.
  - Bei der Verwendung von Microsoft Hyper-V ist das Anzeigen der erreichbaren Teilnehmer nicht möglich.
-



## Unterstützte Sicherheitsprogramme

Folgende Sicherheitsprogramme sind kompatibel zu "SIMATIC STEP 7 Basic":

- Virenschanner:
  - Symantec Endpoint Protection 12.1
  - Trend Micro Office Scan Corporate Edition 10.6
  - McAfee VirusScan Enterprise 8.8
- Verschlüsselungssoftware:
  - Microsoft Bitlocker
  - Utimaco SafeGuard Easy 4.2
- Host-basiertes Intrusion Detection System:
  - McAfee Application Control 6.0

## 3.1.4 Systemvoraussetzungen WinCC Basic

### 3.1.4.1 Software- und Hardwarevoraussetzungen

#### Einleitung

Für die Installation müssen bestimmte Anforderungen an Betriebssystem und Software-Konfiguration erfüllt sein.

---

#### Hinweis

WinCC ist grundsätzlich für den Betrieb in einer Domäne oder Arbeitsgruppe freigegeben.

Beachten sie jedoch, dass Domänen-Gruppenrichtlinien und Einschränkungen der Domäne die Installation behindern können. Entfernen Sie in diesem Fall den Rechner vor der Installation von Microsoft Message Queuing, Microsoft SQL Server 2005 und WinCC aus der Domäne. Melden Sie sich an dem betreffenden Rechner lokal mit Administratorrechten an. Führen Sie die Installation durch. Nach der erfolgreichen Installation können Sie den WinCC-Rechner wieder in die Domäne aufnehmen. Wenn die Domänen-Gruppenrichtlinien und Einschränkungen der Domäne die Installation nicht behindern, dann muss der Rechner während der Installation nicht aus der Domäne entfernt werden.

Beachten sie, dass Domänen-Gruppenrichtlinien und Einschränkungen der Domäne auch den Betrieb behindern können. Wenn Sie diese Einschränkungen nicht aufheben können, dann betreiben Sie die WinCC-Rechner in einer Arbeitsgruppe.

Wenden Sie sich gegebenenfalls an den Domänen-Administrator.

---

### Voraussetzungen für die Installation

Die folgende Tabelle zeigt die minimalen Software- und Hardwarevoraussetzungen, die für die Installation des Softwarepakets "SIMATIC WinCC Basic" erfüllt sein müssen:

| Hardware/Software          | Voraussetzung   |
|----------------------------|---|
| Prozessortyp               | 2,0 GHz Core 2 Duo Prozessor  |
| RAM                        | 1 GB (Windows XP)<br>2 GB (Windows 7; Windows Server)   |
| Freier Festplattenspeicher | 2 GB auf Systemlaufwerk "C:"  |
| Betriebssysteme *          | <p><b>Windows XP</b></p> <ul style="list-style-type: none"> <li>• Windows XP Home SP3</li> <li>• Windows XP Professional SP3</li> </ul> <p><b>Windows 7 (32 bit)</b></p> <ul style="list-style-type: none"> <li>• Windows 7 Home Premium</li> <li>• Windows 7 Home Premium SP1</li> <li>• Windows 7 Professional</li> <li>• Windows 7 Professional SP1</li> <li>• Windows 7 Enterprise</li> <li>• Windows 7 Enterprise SP1</li> <li>• Windows 7 Ultimate</li> <li>• Windows 7 Ultimate SP1</li> </ul> <p><b>Windows 7 (64 bit)</b></p> <ul style="list-style-type: none"> <li>• Windows 7 Home Premium</li> <li>• Windows 7 Home Premium SP1</li> <li>• Windows 7 Professional</li> <li>• Windows 7 Professional SP1</li> <li>• Windows 7 Enterprise</li> <li>• Windows 7 Enterprise SP1</li> <li>• Windows 7 Ultimate SP1</li> </ul> <p><b>Windows Server (32 bit)</b></p> <ul style="list-style-type: none"> <li>• Windows Server 2003 R2 Standard Edition SP2</li> <li>• Windows Server 2008 R2 Standard Edition SP2</li> </ul> <p><b>Windows Server (64 bit)</b></p> <ul style="list-style-type: none"> <li>• Windows Server 2008 R2 Standard Edition</li> <li>• Windows Server 2008 R2 Standard Edition SP1</li> </ul> |
| Grafikkarte                | 32 MB RAM<br>24 Bit Farbtiefe   |
| Bildschirmauflösung        | 1024x768  |
| Netzwerk                   | Ab Ethernet 10 Mbit/s   |
| Optisches Laufwerk         | DVD-ROM   |
| Software                   | Microsoft .Net Framework 3.5 SP1<br>Microsoft Windows Message Queuing   |

\* Weitere Informationen zu den Betriebssystemen finden Sie in der Hilfe zu Microsoft Windows oder auf der Homepage von Microsoft.

Durch gleichzeitiges Öffnen mehrerer Instanzen von WinCC auf einem Projektierungs-PC, können auch höhere Hardwareanforderungen notwendig werden.

---

#### Hinweis

##### "Aero Glass Style" von Microsoft Windows 7

Für "Aero Glass Style" ist eine leistungsstarke Grafikkarte erforderlich. Voraussetzungen sind DirectX9-Fähigkeit und ein 128MB separater Grafikspeicher.

Die Leistung der Architektur des Grafiksystems kann deutliche Einflüsse auf die Leistung von WinCC haben.

---

## Empfohlene Hardware

Die folgende Tabelle zeigt die empfohlene Hardware für den Betrieb von SIMATIC WinCC.

| Hardware           | Voraussetzung                   |
|--------------------|---------------------------------|
| Rechner            | SIMATIC FIELD PG M2 PREMIUM     |
| Prozessor          | 2,2 GHZ CORE 2 DUO (T7500)      |
| RAM                | 1X2GB DDR2 RAM                  |
| Festplatte         | 250GB S-ATA HDD                 |
| Bildschirm         | 15" SXGA+ DISPLAY (1400 X 1050) |
| Optisches Laufwerk | DL MULTISTANDARD DVD RW         |

## Unterstützte Virtualisierungsplattformen

Sie können das Softwarepaket "SIMATIC WinCC Basic" in einer virtuellen Maschine installieren. Verwenden Sie dazu eine der folgenden Virtualisierungsplattformen:

- VMware vSphere Hypervisor (ESXi) 5
- VMware Workstation 8
- VMware Player 4
- Microsoft Windows Server 2008 R2 SP1 Hyper-V

Folgende Betriebssysteme können diesen Virtualisierungsplattformen als Host-Betriebssystem dienen:

- Windows 7 Professional/Ultimate/Enterprise (32-Bit)
- Windows 7 Professional/Ultimate/Enterprise (32-Bit)
- Windows Server 2008 R2 (64 Bit)

Innerhalb der gewählten Virtualisierungsplattform können Sie die folgenden Gast-Betriebssysteme verwenden, um "SIMATIC STEP 7 Basic" zu installieren:

- Windows 7 Professional/Ultimate/Enterprise (32-Bit)
- Windows 7 Professional/Ultimate/Enterprise (64-Bit)

---

**Hinweis**

- Für die Gast-Betriebssysteme gelten dieselben Hardware-Anforderungen wie für die jeweiligen TIA-Produkte selbst.
  - Der Betreiber der Anlage muss dafür Sorgen, dass den Gast-Betriebssystemen ausreichend Systemressourcen zur Verfügung gestellt werden.
  - Für den Einsatz von HyperV-Server und ESXi wird die von den Herstellern zertifizierte Hardware empfohlen.
- 

### Unterstützte Sicherheitsprogramme

Folgende Sicherheitsprogramme sind kompatibel zu "SIMATIC WinCC Basic":

- Virens Scanner:
  - Symantec Endpoint Protection 12.1
  - Trend Micro Office Scan Corporate Edition 10.6
  - McAfee VirusScan Enterprise 8.8
- Verschlüsselungssoftware:
  - Microsoft Bitlocker
  - Utimaco SafeGuard Easy 4.2
- Host-basiertes Intrusion Detection System:
  - McAfee Application Control 6.0

### Microsoft .Net Framework installieren

Das .Net Framework 3.5 SP1 ist auf dem Installationsdatenträger bereits enthalten. Beim Start der Installation prüft die Installationsroutine, ob das .Net Framework bereits installiert ist. Wenn das .Net Framework nicht installiert ist, dann werden Sie über einen Dialog aufgefordert, die Installation durchzuführen. Wenn Sie die Aufforderung zur Installation bestätigen, wird zuerst das .Net Framework installiert. Nach der Installation des .Net Framework müssen Sie den Rechner neu starten. Wenn Sie die Installation des .Net Framework nicht durchführen, wird die Installation von WinCC Runtime Professional abgebrochen.

### Microsoft Windows Message Queuing unter Windows XP installieren

Die Windows Komponente Message Queuing installieren Sie über die Systemsteuerung von Windows.

Klicken Sie auf Start > Systemsteuerung. Doppelklicken Sie auf "Software", der Dialog "Software" wird geöffnet. Klicken Sie im Dialog "Software" auf "Windows-Komponenten hinzufügen/ entfernen". Der Assistent für Windows-Komponenten wird geöffnet. Aktivieren Sie im Assistent für Windows-Komponenten die Komponente "Message Queuing". Klicken Sie auf "Weiter", die Komponente "Message Queuing" wird installiert.

### Microsoft Windows Message Queuing unter Windows 7 installieren

Die Windows Komponente Message Queuing installieren Sie über die Systemsteuerung von Windows.

Klicken Sie auf Start > Systemsteuerung. Klicken Sie auf "Programme", der Dialog "Programme" wird geöffnet. Klicken Sie im Bereich "Programme und Funktionen" auf "Windows-Funktionen aktivieren oder deaktivieren". Der Dialog "Windows-Funktionen" wird geöffnet. Aktivieren Sie im Dialog "Windows-Funktionen" die Funktion "Microsoft-Message Queue-Server". Klicken Sie auf "OK", die Funktion "Microsoft-Message Queue-Server" wird aktiviert.

### Online-Hilfe bei Windows 7 / Windows Server 2008

Windows 7 und Windows Server 2008 unterstützen standardmäßig nicht mehr alle Onlinehilfe-Formate. Bei WinCC werden diese Onlinehilfe-Formate in folgenden Fällen genutzt:

- Aufruf der WinCC Direkthilfe
- Aufruf des WinCC Information System aus den WinCC-Editoren bzw. über die Links der Direkthilfe

Um die WinCC-Direkthilfe weiterhin aufzurufen, wird darum bei der Installation von WinCC folgende Komponente installiert:

- Microsoft Help Engine

Das WinCC Information System können Sie auch unter Windows 7 und Windows Server 2008 aus dem Windows Startmenü oder aus dem Installations-Ordner aufrufen.

Für den Aufruf des WinCC Information System aus den WinCC-Editoren bzw. über Direkthilfe-Links sind Anpassungen im Betriebssystem nötig. Hinweise dazu finden Sie im Microsoft Support-Artikel "917607" im Abschnitt "Weitere Informationen für Benutzer mit fortgeschrittenen Kenntnissen": <http://support.microsoft.com/kb/917607> (<http://support.microsoft.com/kb/917607>)

### Siehe auch

Lizenzierung des WinCC Engineering Systems (Seite 88)

### 3.1.4.2 Parallelinstallationen

#### Parallelinstallationen im TIA-Portal V12

Wenn Sie eine nicht-zulässige Parallelinstallation von STEP 7 und WinCC durchführen, lässt sich das TIA-Portal nicht starten. Folgende Parallelinstallationen im TIA-Portal sind zulässig:

- STEP 7 V12 und WinCC V12

Beim Installieren werden Sie durch einen Dialog auf eine nicht-einheitliche Parallelinstallation hingewiesen. Folgende Parallelinstallationen sind zulässig:

### 3.1 Systemvoraussetzungen für die Installation

- WinCC V12 und RT Advanced V12
- WinCC V12 und RT Professional V12

Engineering System und Runtime müssen immer die gleiche Version nach einer Installation haben.

#### Parallelinstallation von WinCC V12 und anderen SIMATIC HMI Produkten

Die parallele Installation von WinCC V12 mit Versionen von WinCC flexible vor WinCC flexible 2008 ist nicht zulässig.

Die parallele Installation von WinCC V12 mit Versionen von WinCC vor WinCC V7.0 SP2 ist nicht zulässig. Die parallele Installation von WinCC V12 mit WinCC V7.0 SP2 oder WinCC V7.0 SP3 ist nur zulässig für:

- WinCC V12 Basic
- WinCC V12 Runtime Advanced

#### Parallele Nutzung

Sollte die Software im Namen oder der Licence-Key nach der Installation den Zusatz "Combo" tragen, ist Ihnen gemäß Ziffer 1.6 der Allgemeinen Bedingungen (siehe auch Setup-Text) die Nutzung folgender Produkte/Versionen erlaubt:

- Mit der Lizenz "WinCC V12 Comfort Combo": WinCC flexible 2008 Standard
- Mit der Lizenz "WinCC V12 Advanced Combo": WinCC flexible 2008 Advanced

### 3.1.4.3 Lizenzen und Powerpacks

#### Lizenzierung des WinCC Engineering Systems

Sie benötigen für Folgendes einen License Key:

- WinCC Engineering System, z. B. WinCC Professional
- Optionen für WinCC Engineering System

Sie können den License Key für WinCC bei der Installation gleich mitinstallieren. Lizenzen für die WinCC Optionen übertragen Sie nach der Installation mit dem Automation License Manager.

#### Starten ohne gültigen License Key

Wenn Sie WinCC ohne gültigen License Key starten, weist Sie das System mit einer Meldung auf den nicht lizenzierten Modus hin. Sie haben die Möglichkeit, einmalig eine Trial License zu aktivieren. Bei den Engineering Editionen WinCC Basic, Comfort, Advanced und Professional läuft die Trial License nach 21 Tagen ab.

Wenn die Trial License abgelaufen ist, können folgende Fälle auftreten:

- Auf dem betreffenden PC war WinCC noch nie lizenziert.
  - In WinCC lassen sich keine lizenzpflichtigen Aktionen mehr durchführen.
- Auf dem betreffenden PC war WinCC bereits lizenziert.
  - Der nicht lizenzierte Modus wird alle 10 Minuten und bei jeder lizenzpflichtigen Aktion durch ein quittierpflichtiges Fenster angezeigt.

### Lizenzbedarf bei Simulation

Wenn Sie in WinCC die Simulation über den Menübefehl "Online > Simulation > Mit Variablen-Simulator" starten, benötigen Sie keine Lizenzen für WinCC Runtime oder für lizenzpflichtige Optionen.

Wenn folgende Bedingungen erfüllt sind, benötigen Sie auch bei der Simulation die entsprechenden Lizenzen für WinCC Runtime und für lizenzpflichtige Optionen:

- Die Engineering Station ist mit einer PLC verbunden.
- Die Verbindung zur PLC ist projektiert und aktiv.

Sie starten den Simulator über den Menübefehl "Online > Simulation > starten"

### Siehe auch

Software- und Hardwarevoraussetzungen (Seite 83)

Lizenzierung von Bediengeräten (Seite 89)

Umgang mit License Keys (Seite 90)

### Lizenzierung von Bediengeräten

Nicht PC-basierte Bediengeräte werden immer mit dem maximalen Mengengerüst ausgestattet. Ein License Key für den Runtimebetrieb ist nicht erforderlich.

Zu einer Option für nicht PC-basierte Bediengeräte ist jeweils eine Lizenz erforderlich. Der License Key der jeweiligen Lizenz schaltet jeweils eine Option zur Nutzung frei.

### License Key

Um nicht PC-basierte Bediengeräte mit License Keys lizenzieren zu können, ist das Add-On "SIMATIC HMI License Manager Panel Plugin" notwendig.

Das WinCC Setup installiert standardmäßig dieses Add-On. Das License Manager Panel Plugin öffnen Sie im Automation License Manager über den Menübefehl "Bearbeiten > Zielsystem verbinden > Bediengerät verbinden".

Wenn WinCC nicht installiert ist, ist eine Installation von ProSave 7.2 oder höher notwendig.

---

### Hinweis

Weitere Informationen zur Handhabung der Lizenzen finden Sie in der Hilfe des Automation License Manager.

---

---

**Hinweis**

Vergewissern Sie sich, bevor Sie mit Lizenzierung beginnen, dass das Bediengerät mindestens mit einem aktuellen Stand des Betriebssystems ausgestattet ist. Gegebenenfalls aktualisieren Sie das Betriebssystem mit ProSave.

---

**Datensicherung**

**ACHTUNG**

**Zerstörung von License Keys auf nicht PC basierten Bediengeräten**

Bei folgenden Bediengeräten werden installierte License Keys und Autorisierungen durch Backup-/Restore-Vorgänge zerstört.

- 270er Reihe
- 370er Reihe

Bevor Sie mit Wiederherstellen beginnen, führen Sie Folgendes aus:

- Überprüfen Sie mit dem Automation License Manager und ProSave, ob auf dem Bediengerät License Keys installiert sind.
- Deinstallieren Sie die auf dem Bediengerät vorhandenen License Keys.  
Nach dem Wiederherstellen installieren Sie die License Keys wieder auf das Bediengerät.

**Nicht lizenzierter Modus**

Die Runtime-Optionen sind ohne gültige Lizenz uneingeschränkt nutzbar. Der nicht lizenzierte Modus wird alle 10 Minuten durch ein quittierpflichtiges Fenster angezeigt.

**Siehe auch**

Lizenzierung des WinCC Engineering Systems (Seite 88)

**Umgang mit License Keys**

**Einleitung**

Sie installieren einen License Key in folgenden Fällen:

- Zur Nutzung vom WinCC Engineering System
- Zur Nutzung von Optionen vom WinCC Engineering System
- Zur Nutzung vom WinCC Runtime
- Zur Nutzung von Optionen von WinCC Runtime auf PC basierten Bediengeräten
- Zur Nutzung von Optionen auf nicht PC basierten Bediengeräten

Sie deinstallieren einen License Key in folgenden Fällen:



- Wenn Sie Daten sichern
- Wenn Sie die Lizenz nicht mehr benötigen

Sie können diese Lizenz an einem anderen PC oder Bediengerät verwenden.

Wenn Sie eine Lizenz installieren, wird der zugehörige License Key vom Ablageort der License Keys entfernt.

---

**Hinweis**

Ein License Key ist nicht kopierbar. Das verwendete Kopierschutzverfahren verhindert das Kopieren der License Keys.

---

## Datensicherung

Zur Datensicherung am Bediengerät oder als Backup beim Gerätetausch deinstallieren Sie die License Keys am Bediengerät.

Sie sichern License Keys von einem Bediengerät mit dem Automation License Manager auf den Ablageort der License Keys.

**ACHTUNG****Zerstörung von License Keys auf nicht PC basierten Bediengeräten**

Bei folgenden Bediengeräten werden installierte License Keys durch Backup-/Restore-Vorgänge zerstört.

- 270er Reihe
- 370er Reihe

Bevor Sie mit Wiederherstellen beginnen, führen Sie Folgendes aus:

- Überprüfen Sie mit dem Automation License Manager und ProSave, ob auf dem Bediengerät License Keys vorhanden sind.
- Deinstallieren Sie die auf dem Bediengerät vorhandenen License Keys  
Nach dem Wiederherstellen übertragen Sie die License Keys wieder auf das Bediengerät.

**ACHTUNG****Zerstörung von License Keys auf PCs**

Deinstallieren Sie in folgenden Fällen zuerst alle License Keys:

- Bevor Sie die Festplatte formatieren
- Bevor Sie die Festplatte komprimieren
- Bevor Sie die Festplatte restaurieren
- Ein Optimierungsprogramm starten, das feste Blöcke verschiebt
- Ein neues Betriebssystem installieren

Lesen Sie die Beschreibung des Automation License Manager ("Start > Siemens Automation > Dokumentation"). Beachten Sie alle Warnhinweise und Hinweise.

3.1 Systemvoraussetzungen für die Installation

Bei PC-basierten Bediengeräten sowie beim Einsatz des Automation License Manager auf den nicht PC-basierten Bediengeräten ist der Ablageort der License Keys multilizenzfähig. Multilizenzfähig bedeutet, Sie können am gleichen Ablageort mehrere Lizenzen des gleichen Typs ablegen. Nutzen Sie beim Backup einen einzigen Ablageort für alle auf einem Bediengerät vorhandenen License Keys.

|  |
|--|
| <b>ACHTUNG</b>   |
| Behalten Sie auf jeden Fall den Original Ablageort der License Keys. |

**Ungültige Lizenz nach Zeitonenwechsel**

Die installierte Lizenz funktioniert im folgenden Fall nicht mehr.

- Wenn Sie auf einem WinCC PC die Zeitzone folgendermaßen umstellen:
  - Von einer auf die volle Stunde bezogenen Zeit auf eine Zeit, die sich nicht auf eine volle Stunde bezieht.  
Beispiel: Sie stellen die Zeitzone von GMT +3:00 auf GMT +3:30 Uhr um.

Um dieses Verhalten zu umgehen, deinstallieren Sie den License Key mit einer Zeitzoneneinstellung, die bei der Installation des License Keys eingestellt war.

Beispiel:

Sie haben den License Key bei einer Zeitzoneneinstellung mit Bezug zur vollen Stunde installiert. Dann deinstallieren Sie den License Key ebenfalls mit einer Zeitzoneneinstellung mit Bezug zur vollen Stunde.

Das Verhalten betrifft nicht die Trial License.

**Defekte Lizenz**

Eine Lizenz ist in folgenden Fällen defekt:

- Wenn der License Key auf dem Ablageort nicht mehr zugänglich ist.
- Wenn der License Key beim Übertragen auf das Ziellaufwerk verschwindet.

Sie können die defekte Lizenz mit Hilfe des Automation License Manager reparieren lassen. Benutzen Sie dazu die Funktion "Wiederherstellen" bzw. den "Wiederherstellen-Assistenten" des Automation License Manager. Für die Wiederherstellung ist es erforderlich, Kontakt mit dem Customer Support aufzunehmen. Nähere Hinweise hierzu finden Sie unter: <http://support.automation.siemens.com>

**Hinweis**

Ein störungsfreier Betrieb der Runtime Software ist auch bei fehlender bzw. defekter Lizenz sichergestellt. In kurzen Zeitabständen weist Sie das System mit einer Meldung auf den nicht lizenzierten Modus hin.

**ACHTUNG**

Wenn Sie das WinCC Engineering System ohne gültigen License Key starten, weist Sie das System mit einer Meldung auf den nicht lizenzierten Modus hin. Sie haben die Möglichkeit, einmalig eine Trial-Lizenz zu aktivieren. Die Trial-Lizenz läuft nach 21 Tagen ab.

Wenn die Trial-Lizenz abgelaufen ist, können folgende Fälle auftreten:

- Auf dem betreffenden PC war WinCC noch nie lizenziert.  
WinCC lässt sich nicht mehr starten.
- Auf dem betreffenden PC war WinCC bereits lizenziert.  
WinCC lässt sich starten. Der nicht lizenzierte Modus wird alle 10 Minuten durch ein quittierpflichtiges Fenster angezeigt.

**Siehe auch**

Lizenzierung des WinCC Engineering Systems (Seite 88)

## 3.2 Installationsprotokoll

### Funktion des Installationsprotokolls

Während folgender Installationsprozesse werden die Fortschritte in einer Datei protokolliert:

- Produkte installieren
- Bereits installierte Produkte modifizieren oder hochrüsten
- Eine bereits vorhandene Installation reparieren
- Produkte deinstallieren

Wenn während des Installationsprozesses Fehler auftreten oder Warnungen ausgegeben werden, können diese mithilfe der Protokolldatei ausgewertet werden. Dies können Sie entweder selbst tun oder Sie wenden sich an den Produkt-Support.

### Ablageort der Installationsprotokolle

Die Protokolldatei ist die neueste Datei mit der Dateinamenerweiterung ".log", deren Name mit "SIA" beginnt.

Der Ablageort der Protokolldatei ist in der Umgebungsvariablen "%autinstlog%" gespeichert. Diese Umgebungsvariable können Sie in die Adressleiste des Windows Explorers eingeben, um den Ordner mit den Protokolldateien zu öffnen. Alternativ können Sie mit "CD %autinstlog%" über die Kommandozeile zum entsprechenden Verzeichnis navigieren.

Der Ablageort ist abhängig vom Betriebssystem, z. B. "C:\Documents and Settings\All Users\Application Data\Siemens\Automation\Logfiles\Setup" in englischem Windows XP.

### Setup\_Report (CAB-Datei)

Damit Sie alle notwendigen Dateien an den Produkt-Support liefern können, wird zusätzlich eine Archiv-Datei im CAB-Format abgelegt, die das Installationsprotokoll und alle anderen benötigten Dateien enthält. Dieses Archiv finden Sie unter "%autinstlog%\Reports\Setup\_report.cab". Übermitteln Sie diese CAB-Datei dem Produkt-Support, wenn Sie Unterstützung bei der Installation benötigen. Damit kann der Produkt-Support schnell feststellen, ob die Installation einwandfrei durchgelaufen ist. CAB-Dateien, die bei früheren Installationsprozessen erzeugt wurden, werden mit einer Datumskennung im Verzeichnis "Reports" gespeichert.

### Siehe auch

Hinweise zu Lizenzen (Seite 75)

Installation starten (Seite 94)

Support Packages installieren (Seite 96)

Installierte Software anzeigen (Seite 97)

Installierte Produkte modifizieren oder hochrüsten (Seite 98)

Installierte Produkte reparieren (Seite 100)

Deinstallation starten (Seite 101)

## 3.3 Installation starten

### Einführung

Die Installation der Softwarepakete wird durch ein Setup-Programm automatisch durchgeführt. Das Setup-Programm startet nach dem Einlegen des Installationsdatenträgers in das entsprechende Laufwerk.

### Voraussetzung

- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle laufenden Programme sind beendet.

### Vorgehen

Um die Softwarepakete zu installieren, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein. Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe". Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.

3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten.
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Hinweise lesen" bzw. "Installationshinweise".  
Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".  
Der Dialog zur Auswahl der Produktsprachen wird geöffnet.
6. Wählen Sie die Sprachen für die Produktoberfläche und klicken Sie auf die Schaltfläche "Weiter".

---

**Hinweis**

Die Produktsprache "Englisch" wird als Basis immer installiert.

---

Der Dialog zur Auswahl der Produktkonfiguration wird geöffnet.

7. Wählen Sie die Produkte, die Sie installieren möchten:
  - Wenn Sie das Programm mit einer minimalen Programmzusammenstellung installieren möchten, klicken Sie auf die Schaltfläche "Minimal".
  - Wenn Sie das Programm mit produkttypischen Programmzusammenstellungen installieren möchten, klicken Sie auf die Schaltfläche "Typisch".
  - Wenn Sie selbst auswählen möchten, welche Produkte installiert werden sollen, klicken Sie auf die Schaltfläche "Benutzerdefiniert". Aktivieren Sie anschließend die Optionskästchen für die Produkte, die Sie installieren möchten.
8. Wenn eine Verknüpfung auf dem Desktop angelegt werden soll, aktivieren Sie das Optionskästchen "Desktop-Verknüpfung anlegen".
9. Klicken Sie auf die Schaltfläche "Durchsuchen", wenn Sie das Zielverzeichnis für die Installation ändern möchten. Beachten Sie dabei, dass die Länge des Installationspfades 89 Zeichen nicht überschreiten darf.
10. Klicken Sie auf die Schaltfläche "Weiter".  
Der Dialog zu den Lizenzbedingungen wird geöffnet.
11. Um die Installation fortzusetzen, lesen und akzeptieren Sie alle Lizenzvereinbarungen und klicken Sie auf "Weiter".  
Für den Fall, dass für die Installation des TIA-Portals Sicherheits- und Rechteeinstellungen geändert werden müssen, wird der Dialog zu den Sicherheitseinstellungen geöffnet.
12. Um die Installation fortzusetzen, akzeptieren Sie die Änderungen der Sicherheits- und Rechteeinstellungen und klicken Sie auf die Schaltfläche "Weiter".  
Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
13. Überprüfen Sie die gewählten Installationseinstellungen. Wenn Sie Änderungen vornehmen möchten, klicken Sie auf die Schaltfläche "Zurück" bis Sie die zu ändernde Stelle im Dialog erreicht haben. Wenn Sie die gewünschten Änderungen vorgenommen haben, kehren Sie mit "Weiter" wieder zur Übersicht zurück.

### 3.4 Support Packages installieren

14. Klicken Sie auf die Schaltfläche "Installieren".  
Die Installation wird gestartet.

---

**Hinweis**

Wenn während der Installation kein License Key gefunden wird, erhalten Sie die Möglichkeit diesen auf Ihren PC zu übertragen. Wenn Sie den Lizenztransfer überspringen, können Sie dies später mit dem Automation License Manager nachholen.

Nach der Installation erhalten Sie eine Meldung darüber, ob die Installation erfolgreich durchgeführt wurde.

---

15. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
16. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

### Ergebnis

Auf Ihrem Computer wurden das TIA-Portal zusammen mit den erworbenen Produkten und Lizenzen und der Automation License Manager installiert.

### Siehe auch

- Installationsprotokoll (Seite 93)
- Hinweise zu den Systemvoraussetzungen (Seite 76)
- Hinweise zu Lizenzen (Seite 75)
- Installierte Software anzeigen (Seite 97)
- Installierte Produkte modifizieren oder hochrüsten (Seite 98)
- Installierte Produkte reparieren (Seite 100)
- Deinstallation starten (Seite 101)

## 3.4 Support Packages installieren

Sie können im TIA-Portal nachträglich Support Packages installieren, z. B. Hardware Support Packages (HSPs).

---

**Hinweis**

Support Packages für STEP7 V5.4 oder V5.5 können nicht verwendet werden.

---

## Vorgehen

Um ein Support Package zu installieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Extras" auf "Support Packages".  
Der Dialog "Detailinformation" wird geöffnet. In einer Tabelle werden alle Support Packages aus dem Verzeichnis aufgelistet, das Sie als Speicherort für Support Packages in den Einstellungen festgelegt haben.
2. Falls Sie ein Support Package installieren möchten, das in der Liste nicht aufgeführt wird, haben Sie folgende Möglichkeiten:
  - Wenn sich das Support Package bereits auf Ihrem Computer befindet, können Sie es über "Aus dem Dateisystem hinzufügen" in die Liste aufnehmen.
  - Wenn Sie ein Support Package von der "Service & Support"-Seite im Internet hinzufügen möchten, laden Sie es zunächst mit "Aus dem Internet laden" herunter. Anschließend können Sie es aus dem Dateisystem hinzufügen.
3. Selektieren Sie das Support Package, das Sie installieren möchten.
4. Klicken Sie auf "Installieren".
5. Beenden Sie das TIA-Portal und starten Sie es erneut.

## Siehe auch

Installationsprotokoll (Seite 93)

## 3.5 Installierte Software anzeigen

Sie können sich jederzeit darüber informieren, welche Software installiert ist. Zusätzlich können Sie sich weitere Informationen zur installierten Software anzeigen lassen.

## Vorgehen

Um eine Übersicht über die installierte Software aufzurufen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Menü "Hilfe" auf "Installierte Software".  
Der Dialog "Installierte Software" wird geöffnet. Innerhalb des Dialogs sehen Sie die installierten Software-Produkte. Klappen Sie die Einträge auf, um die jeweils installierte Version zu sehen.
2. Wenn Sie sich weitere Informationen zu der installierten Automatisierungssoftware anzeigen lassen möchten, klicken Sie auf den Link innerhalb des Dialogs "Weitere Informationen zu installierter Software".  
Der Dialog "Detailinformation" wird geöffnet.
3. Wählen Sie in der Bereichsnavigation das Thema, zu dem Sie sich weitere Informationen ansehen möchten.

### Siehe auch

- Hinweise zu den Systemvoraussetzungen (Seite 76)
- Hinweise zu Lizenzen (Seite 75)
- Installation starten (Seite 94)
- Installierte Produkte modifizieren oder hochrüsten (Seite 98)
- Installierte Produkte reparieren (Seite 100)
- Deinstallation starten (Seite 101)
- Installationsprotokoll (Seite 93)

## 3.6 Installierte Produkte modifizieren oder hochrüsten

Sie haben die Möglichkeit, installierte Produkte über das Setup-Programm zu ändern oder auf eine neue Version hochzurüsten (Update).

### Voraussetzung

- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle laufenden Programme sind beendet.

### Vorgehen

Um installierte Produkte zu ändern oder hochzurüsten, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.  
Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".  
Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Hinweise lesen" bzw. "Installationshinweise".  
Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".  
Der Dialog zur Auswahl der Installationsvariante wird geöffnet.
6. Aktivieren Sie das Optionsfeld "Modifizieren/Hochrüsten" und klicken Sie auf die Schaltfläche "Weiter".  
Der Dialog zur Auswahl der Produktsprachen wird geöffnet.



7. Aktivieren Sie die Optionskästchen der Produktsprachen, die Sie installieren möchten. Bereits installierte Produktsprachen können Sie durch Deaktivieren der entsprechenden Optionskästchen deinstallieren.

---

**Hinweis**

Beachten Sie, dass Sie die Produktsprache "Englisch" nicht deinstallieren können.

---

8. Klicken Sie auf die Schaltfläche "Weiter".  
Der Dialog zur Auswahl der Produktkonfiguration wird geöffnet.
9. Aktivieren Sie die Optionskästchen der Bestandteile, die Sie installieren möchten. Bereits installierte Bestandteile können Sie durch Deaktivieren der entsprechenden Optionskästchen deinstallieren.
10. Klicken Sie auf die Schaltfläche "Weiter".

---

**Hinweis**

Beachten Sie, dass Sie das Zielverzeichnis nicht ändern können, da die vorhandene Installation verändert wird.

---

Für den Fall, dass für die Installation des TIA-Portals Sicherheits- und Rechteeinstellungen geändert werden müssen, wird der Dialog zu den Sicherheitseinstellungen geöffnet.

11. Um die Installation fortzusetzen, akzeptieren Sie die Änderungen der Sicherheits- und Rechteeinstellungen und klicken Sie auf die Schaltfläche "Weiter".  
Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
12. Klicken Sie auf die Schaltfläche "Modifizieren".  
Die Installation der zusätzlichen Bestandteile wird gestartet.

---

**Hinweis**

Nach der Installation erhalten Sie eine Meldung darüber, ob die bestehende Installation erfolgreich geändert wurde.

---

13. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
14. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

## Ergebnis

Auf Ihrem Computer wurde die bestehende Installation geändert.

## Siehe auch

- Hinweise zu den Systemvoraussetzungen (Seite 76)
- Hinweise zu Lizenzen (Seite 75)
- Installation starten (Seite 94)
- Installierte Software anzeigen (Seite 97)
- Installierte Produkte reparieren (Seite 100)
- Deinstallation starten (Seite 101)
- Installationsprotokoll (Seite 93)

## 3.7 Installierte Produkte reparieren

Sie haben die Möglichkeit, installierte Produkte mithilfe des Setup-Programms durch eine vollständige Reinstallation zu reparieren.

### Voraussetzung

- Hardware und Software des PG/PC entsprechen den Systemanforderungen.
- Sie haben Administrationsrechte auf Ihrem Computer.
- Alle laufenden Programme sind beendet.

### Vorgehen

Um installierte Produkte zu reparieren, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.  
Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".  
Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten
4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Hinweise lesen" bzw. "Installationshinweise".  
Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter".  
Der Dialog zur Auswahl der Installationsvariante wird geöffnet.
6. Aktivieren Sie das Optionsfeld "Reparieren" und klicken Sie auf die Schaltfläche "Weiter".  
Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.

7. Klicken Sie auf die Schaltfläche "Reparieren".  
Das Reparieren der bestehenden Installation wird gestartet.

---

**Hinweis**

Nach der Installation erhalten Sie eine Meldung darüber, ob die installierten Produkte erfolgreich repariert wurden.

---

8. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
9. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

**Ergebnis**

Die installierten Produkte wurden erneut installiert.

**Siehe auch**

- Hinweise zu den Systemvoraussetzungen (Seite 76)
- Hinweise zu Lizenzen (Seite 75)
- Installation starten (Seite 94)
- Installierte Software anzeigen (Seite 97)
- Installierte Produkte modifizieren oder hochrüsten (Seite 98)
- Deinstallation starten (Seite 101)
- Installationsprotokoll (Seite 93)

## 3.8 Deinstallation starten

**Einführung**

Die Deinstallation der Softwarepakete wird durch ein Setup-Programm automatisch durchgeführt. Nach seiner Ausführung führt Sie das Setup-Programm Schritt für Schritt durch den gesamten Vorgang der Deinstallation.

Sie haben zwei Möglichkeiten für die Deinstallation:

- Deinstallation von ausgewählten Komponenten über die Systemsteuerung
- Deinstallation eines Produkts mithilfe des Installationsdatenträgers

---

### Hinweis

Der Automation License Manager wird mit der Deinstallation der Softwarepakete nicht automatisch entfernt, da dieser für die Verwaltung der License Keys mehrerer Produkte der Siemens AG eingesetzt wird.

---

## Deinstallation von ausgewählten Komponenten über die Systemsteuerung

Um ausgewählte Softwarepakete zu deinstallieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Systemsteuerung über "Start > Einstellungen > Systemsteuerung".
2. Doppelklicken Sie in der Systemsteuerung auf den Eintrag "Software".  
Der Dialog "Software" wird geöffnet.
3. Wählen Sie im Dialog "Software" die zu deinstallierende Softwarepakete und klicken Sie auf die Schaltfläche "Entfernen".  
Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
4. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten, und klicken Sie auf die Schaltfläche "Weiter".  
Der Dialog zur Auswahl der Produkte, die Sie deinstallieren möchten, wird geöffnet.
5. Aktivieren Sie die Optionskästchen für die Produkte, die Sie deinstallieren möchten, und klicken Sie auf die Schaltfläche "Weiter".  
Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
6. Überprüfen Sie die Liste mit den Produkten, die deinstalliert werden. Wenn Sie Änderungen vornehmen möchten, klicken Sie auf die Schaltfläche "Zurück".
7. Klicken Sie auf die Schaltfläche "Deinstallieren".  
Die Deinstallation wird gestartet.
8. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
9. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

## Deinstallation eines Produkts mithilfe des Installationsdatenträgers

Um alle Softwarepakete zu deinstallieren, gehen Sie folgendermaßen vor:

1. Legen Sie den Installationsdatenträger in das entsprechende Laufwerk ein.  
Das Setup-Programm startet automatisch, falls der Autostart auf dem PG/PC nicht deaktiviert ist.
2. Wenn das Setup-Programm nicht automatisch startet, starten Sie es manuell durch Doppelklick auf die Datei "Start.exe".  
Der Dialog zur Auswahl der Setup-Sprache wird geöffnet.
3. Wählen Sie die Sprache, in der Sie die Dialoge des Setup-Programms angezeigt bekommen möchten.

4. Um die Produkt- und Installationshinweise zu lesen, klicken Sie auf die Schaltfläche "Produktinformation lesen" bzw. "Installationshinweise lesen". Die entsprechende Hilfedatei mit den Hinweisen wird geöffnet.
5. Wenn Sie die Hinweise gelesen haben, schließen Sie die Hilfedatei und klicken Sie auf die Schaltfläche "Weiter". Der Dialog zur Auswahl der Installationsvariante wird geöffnet.
6. Aktivieren Sie das Optionsfeld "Deinstallieren" und klicken Sie auf die Schaltfläche "Weiter". Im nächsten Dialog wird eine Übersicht der Installationseinstellungen angezeigt.
7. Klicken Sie auf die Schaltfläche "Deinstallieren". Die Deinstallation wird gestartet.
8. Möglicherweise muss der Computer neu gestartet werden. Aktivieren Sie dann das Optionsfeld "Ja, Computer jetzt neu starten". Klicken Sie anschließend auf die Schaltfläche "Neu starten".
9. Wenn der Computer nicht neu gestartet werden muss, klicken Sie auf die Schaltfläche "Beenden".

**Siehe auch**

Installationsprotokoll (Seite 93)

Hinweise zu den Systemvoraussetzungen (Seite 76)

Hinweise zu Lizenzen (Seite 75)

Installation starten (Seite 94)

Installierte Software anzeigen (Seite 97)

Installierte Produkte modifizieren oder hochrüsten (Seite 98)

Installierte Produkte reparieren (Seite 100)

## 3.9 Migrations-Tool installieren und deinstallieren

### 3.9.1 Systemvoraussetzungen

#### Systemvoraussetzungen für das Migrations-Tool

Zur Verwendung des Migrations-Tools gelten folgende Systemvoraussetzungen:

- Es sind alle Produkte installiert, mit denen das Quellprojekt erstellt wurde. Folgende Produkte werden unterstützt:
  - WinCC flexible 2008 SP2 und SP3
  - WinCC V7.0 SP3
  - STEP 7 V5.5
  - Integrierte Projekte aus STEP 7 V5.5 und den oben genannten WinCC-Produkten
  - SINUMERIK
  - STARTER und Startdrives
  - SIMOTION SCOUT V4.4  
Für die Migration von SIMOTION SCOUT V4.4-Projekten wird das SCOUT Migration Tool PlugIn V4.4 benötigt.
- Es sind alle Optionspakete installiert, welche für das Bearbeiten des STEP 7 Projekts notwendig sind. Dies umfasst beispielsweise alle HSPs für alle im Quellprojekt verwendeten Geräte.

### 3.9.2 Migrations-Tool installieren

#### Vertrieb des Migrations-Tools

Das Migrations-Tool befindet sich auf der Installations-DVD des TIA-Portals im Verzeichnis "Support". Alternativ steht es im Service- und Supportbereich der Siemens-Webseite zum Download bereit. Für manche Produkte sind zusätzliche Plugins für das Migrations-Tool erforderlich, z. B. für SIMATIC Failsafe oder SIMOTION. Die Plugins können Sie ebenfalls von der Service- und Support-Seite herunterladen oder von der Installations-DVD der jeweiligen Produkte installieren.

In der Regel wird das Migrations-Tool ohne das TIA-Portal installiert. Da das TIA-Portal eine eigene Migrations-Funktion bereits eingebunden hat, ist eine separate Installation des Migrations-Tools nicht notwendig.

## Vorgehen

Um das Migrations-Tool zu installieren, gehen Sie folgendermaßen vor:

1. Laden Sie die Installations-Datei aus dem Service- und Supportbereich der Siemens-Webseite herunter oder verwenden Sie für die Installation die Installations-Datei aus dem Verzeichnis "Support" der Installations-DVD des TIA-Portals.
2. Starten Sie die Installations-Datei.  
Das Setup-Programm für das Migrations-Tool wird geöffnet.
3. Wählen Sie im ersten Schritt, in welcher Sprache das Setup angezeigt werden soll und klicken Sie auf die Schaltfläche "Weiter".  
Die Seite zur Auswahl der Software-Sprache wird angezeigt.
4. Da das Migrations-Tool ausschließlich in englischer Sprache geliefert wird, können Sie keine weitere Sprache zur Installation wählen. Klicken Sie daher auf "Weiter", um zum nächsten Schritt zu gelangen.  
Die Seite zur Auswahl der Produkte wird angezeigt.
5. Das Migrations-Tool besteht lediglich aus einer Software-Komponente. Daher ist das Migrations-Tool bereits ausgewählt.  
Um ein Symbol zum Start des Migrations-Tools auf dem Desktop anzulegen, aktivieren Sie das Optionskästchen "Symbol auf dem Desktop erstellen". Klicken Sie anschließend auf die Schaltfläche "Weiter".  
Die Seite zur Bestätigung der Lizenzbestimmungen wird geladen.
6. Klicken Sie auf einen Eintrag in der Liste der Lizenzbestimmungen, um die gewählte Lizenzbestimmung zu lesen. Sind Sie mit allen Lizenzbestimmungen einverstanden, aktivieren Sie das Optionskästchen "Ich akzeptiere die Bedingungen der angezeigten Lizenzbedingungen". Klicken Sie anschließend auf die Schaltfläche "Weiter".  
Eine Übersicht zur Installation wird angezeigt.
7. Klicken Sie auf die Schaltfläche "Installieren".  
Die Installation wird mit den angezeigten Einstellungen durchgeführt.

### 3.9.3 Migrations-Tool deinstallieren

Das Migrations-Tool kann über die Systemsteuerung deinstalliert werden.

## Vorgehen

Um das Migrations-Tool zu deinstallieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Systemsteuerung.
2. Doppelklicken Sie in der Systemsteuerung auf den Eintrag "Software".  
Der Dialog "Software" wird geöffnet.
3. Wählen Sie im Dialog "Software" den Eintrag für das Migrations-Tool aus und klicken Sie auf die Schaltfläche "Entfernen".  
Es erscheint eine Sicherheitsabfrage.
4. Bestätigen Sie die Sicherheitsabfrage, indem Sie auf die Schaltfläche "Deinstallieren" klicken.  
Das Migrations-Tool wird deinstalliert.





## Projekte und Programme migrieren

### 4.1 Projekte in ein TIA-Portal-Projekt migrieren

#### 4.1.1 Migration von Projekten mit dem TIA-Portal

##### Migration von bereits bestehenden Projekten

Sie können Projekte aus früheren Automatisierungslösungen in das TIA-Portal migrieren. Bei jeder Migration wird ein neues Projekt für die migrierten Daten angelegt, mit dem Sie anschließend weiterarbeiten können. Eventuell bereits geöffnete TIA-Portal-Projekte werden zuvor geschlossen.

Die Migration wird anschließend in der Tabelle zum Projektverlauf angezeigt. Von dort aus haben Sie Zugriff auf das Migrationsprotokoll, das für die Migration automatisch angelegt wird.

##### Unterstützte Produkte für die Migration

Im Kapitel "Systemübersicht STEP 7 und WinCC" erfahren Sie, welche Produkte für das TIA-Portal erhältlich sind. Alle dort gelisteten Produkte werden grundsätzlich vom TIA-Portal bei der Migration unterstützt.

Welche Voraussetzungen zusätzlich erfüllt sein müssen, ist von den verwendeten Ausgangsprodukten und von den aktuell installierten Produkten abhängig. Nähere Informationen zu den Migrationsmöglichkeiten für Ihre Produkte erhalten Sie beispielsweise auf den Service & Support-Internetseiten und in der Dokumentation zu Ihren Softwareprodukten.

Siehe auch: Skalierung von STEP 7 und WinCC (Seite 25)

## Vorgehen bei der Migration

Der Migrationsprozess gliedert sich grundsätzlich in die folgenden Schritte:

1. Vorbereiten des Ausgangsprojekts  
Wenn auf dem PG/PC mit dem TIA-Portal die Software zum Bearbeiten des Ausgangsprojekts nicht oder nicht in vollem Umfang installiert ist, oder wenn es sich bei dem Ausgangsprojekt um ein integriertes Projekt handelt, dann müssen Sie das Ausgangsprojekt zunächst in eine Migrations-Datei wandeln. Installieren Sie dafür das Migrations-Tool auf einem PG/PC, auf dem die notwendige Software zur Bearbeitung des Ausgangsprojekts installiert ist. Wandeln Sie anschließend das Ausgangsprojekt mit Hilfe des Migrations-Tools um und kopieren Sie die Datei auf den PG/PC mit der Installation des TIA-Portals. Sie können diesen Schritt auslassen, wenn sich das Ausgangsprojekt und die dazugehörige Ausgangssoftware auf demselben PG/PC wie das TIA-Portal befindet und wenn es sich bei dem Ausgangsprojekt nicht um ein integriertes Projekt handelt.
2. Migration durchführen  
Führen Sie die Migration innerhalb des TIA-Portals durch. Geben Sie bei der Migration entweder die Migrations-Datei, die Sie mit dem Migrations-Tool erstellt haben, als Quelle an, oder geben Sie das Ausgangsprojekt an, wenn alle notwendige Software installiert ist.
3. Kontrollieren des Migrationsprotokolls  
Zu jeder Migration wird ein Migrationsprotokoll angelegt. Darin finden Sie Hinweise auf veränderte Projektbestandteile. Das Protokoll können Sie in der Projektnavigation unter "Gemeinsame Daten > Protokolle" oder über den Projektverlauf aufrufen. Nach dem Abschluss der Migration wird das Migrationsprotokoll im TIA-Portal angezeigt. Kontrollieren Sie das Protokoll nach Abschluss der Migration.  
Wenn die Migration fehlgeschlagen ist, dann wird im Projektverzeichnis unter "Logs" eine XML-Datei als Protokoll angelegt. Diese können Sie mit einem beliebigen XML-Editor öffnen und die Ursachen für die fehlgeschlagene Migration ersehen.
4. Nacharbeiten des migrierten Projekts  
Da die Konfigurationen des Ausgangsprojekts nicht immer vollständig mit dem TIA-Portal kompatibel sein können, werden im migrierten Projekt gegebenenfalls nicht alle Konfigurationen identisch übernommen. Arbeiten Sie daher im Migrationsprotokoll verzeichnete Hinweise systematisch nach. Wenn Sie die Hardware-Konfiguration nicht in die Migration eingeschlossen haben, müssen Sie zusätzlich die unspezifizierten Geräte in die passende Hardware umwandeln.

## Einschließen der Hardware-Konfiguration in die Migration

Standardmäßig werden bei der Migration nur die Software-Bestandteile des Projekts berücksichtigt. Für die im Ausgangsprojekt enthaltenen Geräte wird jeweils ein unspezifiziertes Gerät im migrierten Projekt erzeugt. Die Hardware- und Netzkonfigurationen sowie Verbindungen werden nicht migriert. Wandeln Sie daher im Anschluss an die Migration die unspezifizierten Geräte wieder in ein passendes Gerät um und stellen Sie manuell etwaige Netzkonfigurationen und Verbindungen wieder her.

Wenn Sie sich sicher sind, dass die verwendete Hardware im Ausgangsprojekt ein passendes Äquivalent im TIA-Portal besitzt, dann können Sie die Hardware-Konfiguration bei der Migration einschließen. In diesem Fall wird sowohl die Hardware-Konfiguration als auch die Software migriert.

## Siehe auch

- Protokoll der Migration anzeigen (Seite 114)
- Skalierung von STEP 7 und WinCC (Seite 25)

## 4.1.2 Vorbereitung von Projekten mit dem Migrations-Tool

### 4.1.2.1 Migration von Projekten mit dem Migrations-Tool

#### Vorbereitung der Migration

In vielen Fällen wird sich ein Projekt, das Sie migrieren möchten, nicht auf demselben PG/PC befinden, auf dem die aktuelle Version des TIA-Portals installiert ist. Das Ausgangsprojekt muss daher für die Migration zunächst in ein kompatibles Format umgewandelt werden. Dasselbe gilt für integrierte Projekte.

Nach dem Erstellen der Migrations-Datei kopieren Sie die Migrations-Datei auf den PG/PC mit der aktuellen Installation des TIA-Portals. Im TIA-Portal geben Sie die Migrations-Datei als Quelle für die Migration an und können ein Projekt im aktuellen Dateiformat des TIA-Portals erstellen.

#### Vorgehen bei der Migration mit dem Migrations-Tool

Um eine Migration mit dem Migrations-Tool vorzubereiten, sind die folgenden Schritte notwendig:

1. Installieren Sie auf dem PG/PC mit dem Quellprojekt das Migrations-Tool. Laden Sie dazu die Installationsdatei aus dem Service- & Support-Bereich der Siemens-Internetseite herunter oder installieren Sie das Migrations-Tool von der Setup-DVD des TIA-Portals.
2. Starten Sie das Migrations-Tool und wandeln Sie mit Hilfe des Migrations-Tools das Quellprojekt in das Migrations-Dateiformat mit der Dateiendung ".am12" um. Achten Sie bei diesem Schritt besonders darauf, dass die gesamte, für die Bearbeitung des Quellprojekts notwendige Software auf dem PG/PC installiert ist. Dazu gehören auch alle notwendigen Servicepacks, Hardware-Support-Packages und die gesamte Erweiterungssoftware, die zur Bearbeitung des Ausgangsprojekts notwendig ist. Sind einzelne Produkte nicht installiert, kann die Migration nicht oder nur unvollständig durchgeführt werden.
3. Kopieren Sie die Migrationsdatei auf das Zielsystem mit einer aktuellen Installation des TIA-Portals. Beachten Sie, dass auf dem Zielsystem sämtliche Software installiert sein muss, die zur Projektierung aller in der Migrationsdatei enthaltenen Geräte notwendig ist.
4. Führen Sie die Migration innerhalb des TIA-Portals durch und geben Sie als Quelle die Migrationsdatei mit der Endung ".am12" an.
5. Kontrollieren Sie nach der Migration das Migrationsprotokoll und arbeiten Sie darin verzeichnete Hinweise im neu erstellten Projekt systematisch nach. Beachten Sie nach dem ersten Übersetzen der Konfiguration insbesondere auch die Hinweise im Inspektorfenster.

### Einschließen der Hardware-Konfiguration in die Migration

Standardmäßig werden bei der Migration nur die Software-Bestandteile des Projekts berücksichtigt. Für die im Ausgangsprojekt enthaltenen Geräte wird jeweils ein unspezifiziertes Gerät im migrierten Projekt erzeugt. Die Hardware- und Netzkonfigurationen sowie Verbindungen werden nicht migriert. Wandeln Sie daher im Anschluss an die Migration die unspezifizierten Geräte wieder in ein passendes Gerät um und stellen Sie manuell etwaige Netzkonfigurationen und Verbindungen wieder her.

Wenn Sie sich sicher sind, dass die verwendete Hardware im Ausgangsprojekt ein passendes Äquivalent im TIA-Portal besitzt, dann können Sie die Hardware-Konfiguration bei der Migration einschließen. In diesem Fall wird sowohl die Hardware-Konfiguration als auch die Software migriert.

#### Siehe auch

Migration von Projekten mit dem TIA-Portal (Seite 107)

Projekte migrieren (Seite 112)

Migrations-Tool aufrufen (Seite 110)

Migrations-Datei erstellen (Seite 110)

#### 4.1.2.2 Migrations-Tool aufrufen

##### Migrations-Tool starten

Bei der Installation wird im Startmenü unter "Siemens Automation > Migration Tool" standardmäßig eine Verknüpfung "Migration to TIA Portal V12" angelegt. Rufen Sie diese Verknüpfung auf.

Alternativ können Sie das Migrations-Tool direkt über den Windows Explorer aufrufen. Standardmäßig wird das Migrations-Tool bei der Installation im Ordner "C:\Program Files \Siemens\Automation\Portal V12\bin" gespeichert. Um das Migrations-Tool zu starten, rufen Sie die Datei "Siemens.Automation.MigrationApplication.exe" in diesem Verzeichnis auf.

#### Siehe auch

Migrations-Datei erstellen (Seite 110)

#### 4.1.2.3 Migrations-Datei erstellen

Im Folgenden erfahren Sie, wie Sie das Ausgangsprojekt mit Hilfe des Migrations-Tools in eine für das TIA-Portal lesbare Migrations-Datei umwandeln. Diese Datei wird nach dem Umwandeln auf das Zielsystem übertragen und dort migriert.

Sie können festlegen, ob die Migrationsdatei das gesamte Projekt inklusive der kompletten Hardware-Konfiguration und der zugehörigen Software enthalten soll, oder ob Sie nur die Software migrieren möchten.

## Voraussetzung

- Es ist für alle im Ausgangsprojekt verwendeten Konfigurationen die passende, ursprüngliche Software mit einer gültigen Lizenz installiert.
- Das Ausgangsprojekt ist nicht mit einem Zugriffsschutz versehen.
- Das Ausgangsprojekt muss sich in einem konsistenten Zustand befinden, da eine einwandfreie Migration sonst nicht sichergestellt ist.

## Vorgehen

Um die Migrations-Datei zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Feld "Storage Location (Path)" den Pfad der Quelldatei für die Migration.
2. Geben Sie an, welche Projektanteile migriert werden sollen:
  - Aktivieren Sie das Optionskästchen "Include HW and Network data during the migration", um nicht nur die Software, sondern auch die vollständigen Hardware-Anteile und die Netzkonfiguration des Projekts zu migrieren.
  - Aktivieren Sie das Optionskästchen "Copy SCADA runtime data", wenn Sie zusätzlich zu den Daten des Engineering-Systems auch die Runtime-Daten, wie z. B. Meldearchive, Variablenarchive oder Anwenderarchive, migrieren möchten.
3. Wählen Sie im Feld "Intermediate file" den Pfad und den Dateinamen für die Migrations-Datei.
4. Klicken Sie auf die Schaltfläche "Migrate".

## Ergebnis:

Eine Migrationsdatei wird erstellt. Kopieren Sie im Anschluss diese Datei auf das Zielsystem und migrieren Sie diese Datei im TIA-Portal.

## Siehe auch

Projekte migrieren (Seite 112)

Migrations-Tool aufrufen (Seite 110)

Migration von Projekten mit dem Migrations-Tool (Seite 109)

### 4.1.3 Projekte migrieren

#### Voraussetzung

- Es liegt eine bereits in das Format ".am12" gewandelte Datei vor oder es ist für alle im Ausgangsprojekt verwendeten Konfigurationen die passende, ursprüngliche Software mit einer gültigen Lizenz installiert.
- Das Ausgangsprojekt ist nicht mit einem Zugriffsschutz versehen.
- Das Ausgangsprojekt muss sich in einem konsistenten Zustand befinden, da eine einwandfreie Migration sonst nicht sichergestellt ist.

Beachten Sie zusätzliche Hinweise zu den Voraussetzungen in der Hilfe zu den jeweiligen, installierten Produkten.

---

#### Hinweis

##### Ruhezustand des Systems während der Migration

Das System sollte während einer laufenden Migration nicht in Standby oder in den Ruhezustand versetzt werden. Andernfalls wird die Migration abgebrochen.

---

#### Vorgehen

Um ein Projekt zu migrieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Projekt migrieren".  
Der Dialog "Projekt migrieren" wird geöffnet.
2. Geben Sie im Feld "Quellpfad" den Pfad und den Dateinamen für das zu migrierende Projekt an. Wählen Sie entweder ein Projekt im Migrations-Format ".am12" oder im Format des Ausgangsprojekts.
3. Aktivieren Sie das Optionskästchen "Hardware-Konfiguration einschließen", um die Hardware-Konfiguration in die Migration einzuschließen.  
Wenn Sie eine Migrationsdatei gewählt haben, die mit dem Migrations-Tool erstellt wurde, ist das Optionskästchen nicht aktivierbar. In diesem Fall müssen Sie bereits bei der Umwandlung mit dem Migrations-Tool angeben, ob Sie die Hardware-Konfiguration bei der Migration einschließen möchten.
4. Aktivieren Sie das Optionskästchen "Daten von WinCC Runtime Professional kopieren", wenn Sie zusätzlich zu den Daten des Engineering-Systems auch die Runtime-Daten, wie z. B. Meldearchive, Variablenarchive oder Anwenderarchive, migrieren möchten.  
Wenn Sie eine Migrationsdatei gewählt haben, die mit dem Migrations-Tool erstellt wurde, ist das Optionskästchen nicht aktivierbar. In diesem Fall müssen Sie bereits bei der Umwandlung mit dem Migrations-Tool angeben, ob Sie die SCADA Runtime-Daten bei der Migration einschließen möchten.
5. Wählen Sie im Feld "Projektname" einen Namen für das neue Projekt.
6. Wählen Sie im Feld "Zielpfad" einen Pfad, unter dem das neue Projekt angelegt werden soll.
7. Geben Sie im Feld "Autor" Ihren Namen oder den Namen eines anderen Projekt-Verantwortlichen an.

8. Tragen Sie ggf. einen Kommentar in das Feld "Kommentar" ein.
9. Klicken Sie auf "Migrieren".

## Ergebnis

Das Ausgangsprojekt wird umgewandelt und nach dem Fertigstellen wird eine Meldung angezeigt. Anschließend wird das neu angelegte Projekt in der Projektansicht geöffnet und das Migrationsprotokoll wird im TIA-Portal geöffnet.

Sollte die Migration fehlgeschlagen sein, wird dennoch ein Projektverzeichnis angelegt und darin ein Migrations-Protokoll in Form einer XML-Datei erzeugt. In der Abschlussmeldung nach der Migration finden Sie einen Link zu dieser XML-Datei. Klicken Sie auf den Link, um die XML-Datei zu öffnen. Alternativ finden Sie die XML-Datei im Projektverzeichnis unter "\Logs".

## Siehe auch

- Integrierte Projekte nacharbeiten (Seite 147)
- Protokoll der Migration anzeigen (Seite 114)
- Protokolle verwenden (Seite 231)
- Migration von Projekten mit dem Migrations-Tool (Seite 109)
- Migrations-Datei erstellen (Seite 110)

### 4.1.4 Verlauf der Migration anzeigen

Ist ein Projekt durch eine Migration entstanden, dann wird die Migration in der Tabelle zum Projektverlauf aufgeführt. In der Tabelle kann das Migrationsprotokoll aufgerufen werden. Außerdem wird der Zeitpunkt der Migration angezeigt.

## Vorgehen

Um eine Übersichtstabelle mit der Migration anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das geöffnete Projekt.
2. Wählen Sie im Kontextmenü des Projekts den Befehl "Eigenschaften".  
Der Dialog mit den Eigenschaften des Projekts wird geöffnet.
3. Wählen Sie in der Bereichsnavigation die Gruppe "Projektverlauf".  
Die Übersichtstabelle wird geöffnet.

## Siehe auch

- Eigenschaften des Projekts anzeigen (Seite 237)

### 4.1.5 Protokoll der Migration anzeigen

Zu jeder erfolgreichen Migration wird ein Protokoll angelegt. Das Protokoll enthält folgende Informationen:

- Migrierte Objekte
- Änderungen an Objekten, die während der Migration vorgenommen wurden
- Fehler, die bei der Migration aufgetreten sind
- Gegebenenfalls einen Link zu weiterer Hilfe bei bestimmten Ereignissen.  
Weitere Hilfe erhalten Sie in diesem Fall durch einen Klick auf das Fragezeichen.

#### Vorgehen

Um die Protokolldatei zur Migration anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Gemeinsame Daten > Protokolle" in der Projektnavigation.
2. Doppelklicken Sie das gewünschte Protokoll in der Liste.  
Die Inhalte des Protokolls werden im Arbeitsbereich angezeigt.

#### Siehe auch

Migration von Projekten mit dem TIA-Portal (Seite 107)

Protokolle verwenden (Seite 231)

### 4.1.6 WinCC flexible Projekte migrieren (Basic)

#### 4.1.6.1 Grundlagen (WinCC flexible)

#### Migration (WinCC flexible)

##### Einleitung

In WinCC können Sie Projekte aus WinCC flexible weiter verwenden. Folgende Version von WinCC flexible wird unterstützt:

- WinCC flexible 2008 SP2

In den folgenden Kapiteln wird beschrieben, welche Bediengeräte unterstützt werden und welche Randbedingungen für eine erfolgreiche Migration gelten.

Projekte aus ProTool und aus früheren WinCC flexible Versionen können Sie nicht direkt nach WinCC migrieren. Wenn Sie solche Projekte in WinCC weiter verwenden wollen, müssen Sie diese erst in eine unterstützte Version von WinCC flexible migrieren.



## Siehe auch

- Objektunterstützung bei der Migration (WinCC flexible) (Seite 121)
- Projekte aus WinCC flexible migrieren (WinCC flexible) (Seite 117)
- Übersetzen und Laden eines migrierten Projekts (WinCC flexible) (Seite 119)
- Migration von Runtime-Daten (WinCC flexible) (Seite 131)
- Migration integrierter Projekte (WinCC flexible) (Seite 134)
- Unterstützte Bediengeräte (WinCC flexible) (Seite 120)
- Migration der Datentypen (WinCC flexible) (Seite 137)

## Grundlagen zur Migration (WinCC flexible)

### Einleitung

Bei der Migration werden die Projektdaten aus einem WinCC flexible-Projekt in das neue Datenformat von WinCC umgesetzt. Dabei wird nicht geprüft, ob die Daten in dem zu migrierenden Projekt konsistent sind. Wenn in einem Quellprojekt beim Übersetzen Fehler oder Warnungen ausgegeben werden, dann werden diese bei der Migration nicht behoben. Daher sollte das zu migrierende Projekt vor der Migration fehlerfrei übersetzbar sein. Beachten Sie bei der Migration den Umfang eines Projekts. Für die Migration gelten die Leistungsmerkmale von WinCC. Nähere Hinweise hierzu finden Sie in der Online-Hilfe im Kapitel "Prozesse visualisieren > Leistungsmerkmale > Engineering System".

### Eindeutigkeit von Objektnamen

In WinCC flexible wird die Eindeutigkeit der Objekte durch den enthaltenden Ordner bestimmt. Bei Bildelementen in Gruppen wird die Eindeutigkeit durch den Gruppennamen bestimmt.

In WinCC muss ein Objektnamen innerhalb eines Bediengeräts eindeutig sein. Bei Bildelementen muss der Name innerhalb eines Bilds eindeutig sein.

Bei der Migration wird die Eindeutigkeit der Namen geprüft. Wenn ein Name nach der neuen Regelung nicht eindeutig ist, wird das betreffende Objekt umbenannt. Ein umbenanntes Objekt erhält das Suffix "#Mign", wobei "n" eine fortlaufende Nummer ist.

### Beispiel:

In WinCC flexible dürfen Variablen, die in verschiedenen Ordnern abgelegt sind, den gleichen Namen haben. In WinCC muss der Variablenname auf dem projektierten Bediengerät eindeutig sein. Bei der Migration werden daher gleichnamige Variablen aus verschiedenen Ordnern umbenannt.

Die Variablen werden nach folgendem Schema umbenannt:

| Vor der Migration | Nach der Migration |
|-------------------|--------------------|
| Folder_1/Tag_1    | Folder_1/Tag_1     |
| Folder_1/Tag_2    | Folder_1/Tag_2     |

| Vor der Migration | Nach der Migration  |
|-------------------|---------------------|
| Folder_2/Tag_1    | Folder_2/Tag_1#Mig1 |
| Folder_2/Tag_2    | Folder_2/Tag_2#Mig1 |
| Folder_3/Tag_1    | Folder_3/Tag_1#Mig2 |
| Folder_3/Tag_2    | Folder_3/Tag_2#Mig2 |

### Betroffene Objekte

Folgende Objekte werden bei Bedarf umbenannt:

- Bilder
- Bildobjekte
- Rezepturen
- Variablen

### Abbruch der Migration

In folgenden Fällen wird die Migration abgebrochen:

- Wenn das zu migrierende Projekt im Engineering-System oder in Runtime geöffnet ist.
- Wenn nicht genügend freier Speicherplatz auf der Festplatte zur Verfügung steht, um eine Kopie für die Migration des Projekts anzulegen.
- Wenn die Migration wegen Problemen mit dem installierten SQL-Server die Projekt-Datenbank nicht ansprechen kann.
- Wenn die Migration wegen fehlender Benutzerberechtigung die Projekt-Datenbank nicht ansprechen kann.
- Wenn Sie in einem integrierten Projekt die "\*.hmi" Datei für die Migration auswählen. In einem integrierten Projekt müssen Sie für die Migration die Datei "\*.s7" auswählen.
- Wenn das Projekt mit einer von der Migration nicht unterstützten Version erstellt wurde.

### Projekt im Migrationsformat speichern

Die Migration eines WinCC flexible-Projekts müssen Sie nicht vollständig auf dem PC durchführen, auf dem das Projekt vorhanden ist. Sie können die Migration vorbereiten, indem Sie das Projekt im Migrationsformat speichern. Für die Speicherung eines WinCC flexible-Projekts im Migrationsformat steht Ihnen das Migrationstool zur Verfügung. Das Migrationstool exportiert die Engineering-Daten aus dem WinCC flexible-Projekt und speichert die Daten im Migrationsformat "\*.AM11".

Für die eigentliche Migration kopieren Sie die Daten im Migrationsformat auf einen PC, auf dem das TIA-Portal installiert ist.

Nähere Hinweise zum Migrationstool finden Sie im Kapitel "Auto-Hotspot".

## **Projekte aus WinCC flexible migrieren (WinCC flexible)**

### **Einleitung**

Bei der Migration eines Projekts werden Daten aus einem WinCC flexible-Projekt in ein neues Projekt für WinCC geladen. Deshalb wird für die Projektmigration automatisch ein neues Projekt angelegt. Sie können nicht in ein bestehendes Projekt migrieren.

Die Migration lässt sich sowohl in der Portalansicht als auch in der Projektansicht aufrufen.

Migrieren Sie ein Projekt nur in einem neu gestarteten TIA-Portal.

Hinweise zur Migration eines integrierten Projekts finden Sie im Kapitel Migration integrierter Projekte (WinCC flexible) (Seite 134).

Wenn Sie das Projekt nur im Migrationsformat speichern wollen, steht Ihnen das Migrationstool zur Verfügung. Nähere Hinweise hierzu finden Sie unter Grundlagen zur Migration (WinCC flexible).

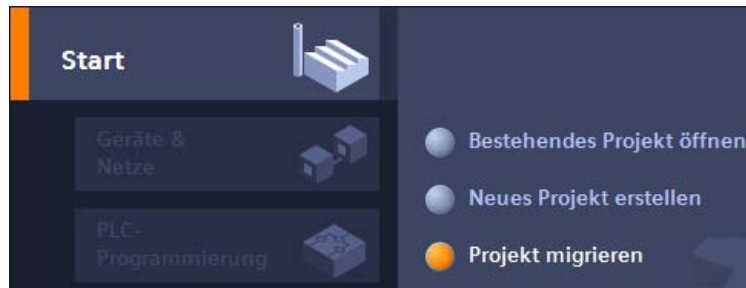
### **Voraussetzung**

- Ein Projekt aus WinCC flexible ist vorhanden.
- Das Projekt ist nicht in WinCC flexible geöffnet.

### Vorgehen

Ein Projekt migrieren Sie in der Portalansicht wie folgt:

1. Wählen Sie die Aktion "Start > Projekt migrieren".



2. Navigieren Sie im Feld "Quellpfad" zu dem Projekt, das Sie migrieren wollen.

The image shows a dialog box titled "Projekt migrieren". It contains the following fields:

- Wählen Sie das zu migrierende Projekt aus.**
  - Projektname: MigrationProject02
  - Quellpfad: Z:\Shared\_Public\_Host\MigrationProjects\MigrationProject02.hmi
- Ziel**
  - Projektname: MigrationProject02
  - Zielpfad: Z:\Shared\_Public\_Host\MigrationProjects\_V11
  - Autor: vmadmin
  - Kommentar: (empty text area)

A "Migrieren" button is located at the bottom right of the dialog.

3. Selektieren Sie die WinCC flexible-Projektdatei "\*.hmi".

4. Ändern Sie bei Bedarf die Angaben für das neu zu erstellende Projekt. Ändern Sie z. B. den Projektnamen oder den Projektpfad. In dem neuen Projekt werden die zu migrierenden Daten angelegt.
5. Klicken Sie auf "Migrieren".  
Ein neues Projekt wird erstellt und die Migration der Daten wird gestartet:
  - Die Projektansicht wird geöffnet.
  - Der Fortschritt der Migration wird in einem Migrationsfenster angezeigt.
  - Im Inspektorfenster unter "Info > Allgemein" werden Informationen, Warnungen und Fehler zum Migrationsvorgang ausgegeben.
  - Sämtliche Informationen zur Migration werden in einer Protokolldatei gespeichert.
  - Nach Abschluss der Migration wird das Projekt gespeichert und eine Meldung ausgegeben. In der Meldung finden Sie einen Link, über den Sie die Protokolldatei öffnen können.

Nach Abschluss der Migration finden Sie in der Projektnavigation für jedes migrierte Bediengerät ein neu angelegtes Gerät. Diese Geräte beinhalten die migrierten Daten, z. B. Bilder, Meldungen und Variablen.

### Migrationsprotokoll zu einem späteren Zeitpunkt öffnen

Das Migrationsprotokoll wird zusammen mit dem migrierten Projekt gespeichert. Sie können das Protokoll zu einem späteren Zeitpunkt ansehen. Die Protokolldatei öffnen Sie folgendermaßen:

1. Öffnen Sie in der Projektnavigation den Ordner "Gemeinsame Daten > Protokolle". Er enthält die Protokolle aller bisher durchgeführten Migrationen.
2. Doppelklicken Sie auf das gewünschte Migrationsprotokoll.  
Das Protokoll wird geöffnet.

### Siehe auch

Migration integrierter Projekte (WinCC flexible) (Seite 134)

### Übersetzen und Laden eines migrierten Projekts (WinCC flexible)

#### Übersetzen eines migrierten Projekts

Wenn Sie ein WinCC flexible-Projekt erfolgreich migriert haben, müssen Sie es vor dem Laden auf das Bediengerät neu übersetzen. Das Projekt wird nur dann erfolgreich übersetzt, wenn es vor der Migration fehlerfrei übersetzbar war.

Wenn beim Übersetzen des migrierten Projekts Fehler auftreten, müssen Sie diese beseitigen.

Nachdem das Projekt fehlerfrei übersetzt wurde, laden Sie es auf das Bediengerät.

### Einstellungen zum Laden auf das Bediengerät

Bei der Migration werden die Einstellungen für das Laden auf das Bediengerät nicht übernommen. Wenn Sie das Projekt migriert haben, müssen Sie die Einstellungen für das Laden neu projektieren.

Selektieren Sie das Bediengerät in der Projektnavigation und wählen Sie den Kontextmenübefehl "Laden in Gerät > Software (komplett laden)". Der Dialog "Erweitertes Laden" wird geöffnet. Projektieren Sie die erforderlichen Einstellungen für die Schnittstelle. Klicken Sie auf die Schaltfläche "Laden". Das Projekt wird neu übersetzt und der Dialog "Vorschau laden" wird geöffnet.

Expandieren Sie den Eintrag "Überschreiben" und kontrollieren Sie die Einstellungen für die folgenden Optionen:

- Wollen Sie die bestehenden Benutzerverwaltungsdaten dieses Geräts überschreiben
- Wollen Sie die bestehenden Rezepturdaten auf dem Bediengerät überschreiben

Projektieren Sie die Optionen so, wie Sie sie im Projekt künftig verwenden wollen. Laden Sie anschließend das Projekt auf das Bediengerät.

### 4.1.6.2 Engineering-Daten migrieren (WinCC flexible)

#### Unterstützte Bediengeräte (WinCC flexible)

##### Einleitung

Bei der Migration von Projekten aus WinCC flexible müssen Sie beachten, dass WinCC folgende Bediengerätetypen unterstützt:

- KTP400 Basic mono PN
- KTP400 Basic mono PN Portrait
- KTP600 Basic DP
- KTP600 Basic DP Portrait
- KTP600 Basic PN
- KTP600 Basic PN Portrait
- KTP600 Basic mono PN
- KTP600 Basic mono PN Portrait
- KTP1000 Basic DP
- KTP1000 Basic PN
- TP1500 Basic PN

In WinCC werden nur die Funktionalitäten unterstützt, die diese Bediengerätetypen anbieten.

Wenn Ihr WinCC flexible-Projekt ein Bediengerät enthält, das von WinCC nicht unterstützt wird, wird der Migrationsvorgang abgebrochen. Um das Projekt zu migrieren, müssen Sie in WinCC flexible das Bediengerät in einen von WinCC unterstützten Bediengerätetyp ändern.

In einem WinCC flexible-Projekt können Funktionalitäten enthalten sein, die von einem Basic Panel nicht unterstützt werden, z. B. wegen einer durchgeführten Gerätetypumschaltung. Diese nicht unterstützten Funktionalitäten werden nicht migriert.

## Anpassungen vor der Migration

Wenn in dem zu migrierenden Projekt das Bediengerät in ein Bediengerät mit anderer Bildgröße geändert wurde, müssen Sie das Projekt vor der Migration neu kompilieren. Durch den Kompilierungsvorgang werden die Bilder und Bildelemente in ihrer Größe angepasst.

## Siehe auch

Objektunterstützung bei der Migration (WinCC flexible) (Seite 121)

Migration (WinCC flexible) (Seite 114)

Migration von Meldeklassen und Meldegruppen (WinCC flexible) (Seite 125)

Migration von sprachabhängigen Inhalten (WinCC flexible) (Seite 128)

Migration von Bibliotheken (WinCC flexible) (Seite 130)

Migration von Variablen (WinCC flexible) (Seite 124)

Wertänderungen an Objekteigenschaften durch die Migration (WinCC flexible) (Seite 123)

## Objektunterstützung bei der Migration (WinCC flexible)

### Einleitung

Bei der Migration von Projekten aus WinCC flexible werden alle Projektierungsdaten migriert, die ein von WinCC unterstütztes Bediengerät enthält. Grundsätzlich werden alle Objekttypen und Funktionen vollständig migriert, die in der neuen Projektumgebung zur Verfügung stehen und abgebildet werden können.

Einige globale Objekttypen werden nicht migriert, z. B. Wörterbücher und globale Bibliotheken.

### Unterstützte Objekttypen

Folgende Objekttypen werden von der Migration unterstützt:

- Animationen
- Aufgabenplaner
- Benutzerverwaltung
- Bereichszeiger
- Bilder
- Bildvorlage
- Datentypen
- Funktionslisten

- Grafiklisten
- Anzeige- und Bedienobjekte  
Von der Migration werden alle Anzeige- und Bedienobjekte unterstützt, die auf den unterstützten Bediengeräten verfügbar sind.
- Meldungen
- Meldeklassen
- Meldegruppen
- Projektbibliothek
- Projektsprachen
- Rezepturen
- Runtime-Sprachen
- Runtime-Schriften
- Systemfunktionen
- Texte
- Textlisten
- Variablen
- Verbindungen

### Nicht unterstützte Objekttypen

Folgende Funktionalitäten und Objekttypen werden von der Migration nicht unterstützt:

- Globale Bibliotheken
- Wörterbücher
- Projektversionen
- Änderungslogbuch

### Migration der Bildvorlage

In WinCC steht Ihnen ein erweitertes Konzept zur Arbeit mit Bildvorlagen zur Verfügung. In WinCC stehen Ihnen ein globales Bild und mehrere Vorlagen für jedes Gerät zur Verfügung. Bei der Migration einer Vorlage aus WinCC flexible werden die enthaltenen Objekte und die in der Vorlage projektierten Eigenschaften in verschiedene Vorlagen von WinCC migriert.

Folgende Objekte werden in das "Globale Bild" von WinCC migriert:

- Meldefenster
- Meldeindikator
- Funktionstasten von Bediengeräten mit Funktionstasten

Alle weiteren Objekte und Eigenschaften werden in eine Vorlage von WinCC migriert.

Die Verbindung der Objekte und Eigenschaften zur jeweiligen Vorlage wird von der Migration automatisch angepasst.



## Migration von Systemfunktionen

In WinCC haben sich die Namen einiger Systemfunktionen geändert.

Systemfunktionen, deren Namen sich geändert haben, werden umbenannt.

Folgende Systemfunktionen sind betroffen:

| Funktionsname in WinCC flexible | Funktionsname in WinCC |
|---------------------------------|------------------------|
| IncreaseValue                   | IncreaseTag            |
| DecreaseValue                   | DecreaseTag            |
| SetValue                        | SetTag                 |

## Siehe auch

Unterstützte Bediengeräte (WinCC flexible) (Seite 120)

Wertänderungen an Objekteigenschaften durch die Migration (WinCC flexible) (Seite 123)

## Wertänderungen an Objekteigenschaften durch die Migration (WinCC flexible)

### Einleitung

Aufgrund der Vereinheitlichung von Objekteigenschaften aus WinCC V7 und WinCC flexible sind im Verlauf der Migration an einigen Objekteigenschaften Änderungen erforderlich. Die Migration berechnet die Änderungen so, dass die Darstellung der Objekte nach der Migration gleich ist, wie vor der Migration. Durch die Änderungen während der Migration ergeben sich für einige Objekteigenschaften andere Maßeinheiten und Werte in der Projektierung.

### Migration der Schrifteinstellungen eines Objekts

In WinCC V7 und in WinCC flexible wird für die Größe der verwendeten Schriften an einem Objekt die Maßeinheit "Punkt" verwendet. In WinCC wird für die Größe der verwendeten Schriften an einem Objekt die Maßeinheit "Pixel" verwendet. Bei der Migration wird die Schriftgröße entsprechend umgerechnet, so dass die Darstellung der Schrift bei der Zoomstufe 100% gleich groß ist. Aufgrund der unterschiedlichen Maßeinheiten, ändern sich nach der Migration die Zahlenwerte für die Schriftgrößen.

#### Beispiel:

| Schriftart vor der Migration | Schriftart nach der Migration |
|------------------------------|-------------------------------|
| Arial 10 Punkt               | Arial 13 Pixel                |
| Arial 16 Punkt               | Arial 21 Pixel                |
| Tahoma 10 Punkt              | Tahoma 13 Pixel               |
| Tahoma 16 Punkt              | Tahoma 21 Pixel               |

## Migration von Objekträndern

In WinCC flexible gestatten einige Objekte bei der Projektierung der Darstellung die Eingabe von Werten  $<0$  und  $>127$  für die Einrichtung der Objektränder. In WinCC ist der Wertebereich für Objektränder begrenzt auf Werte zwischen 0 und 127. Die Migration ändert Werte  $<0$  auf den Wert "0" und Werte  $>127$  auf den Wert "127".

## Siehe auch

Unterstützte Bediengeräte (WinCC flexible) (Seite 120)

Objektunterstützung bei der Migration (WinCC flexible) (Seite 121)

## Migration von Variablen (WinCC flexible)

### Einleitung

Bei der Migration von Variablen müssen Sie einige Besonderheiten beachten. Folgende Aspekte sind zu unterscheiden:

- Migration der Datentypen von Variablen
- Migration interner Variablen
- Migration externer Variablen
- Namen von Variablen

### Migration der Datentypen

WinCC verfügt teilweise über andere Datentypen und verwendet andere Datentypnamen als WinCC flexible. Bei der Migration einer betreffenden Variable wird der Datentyp aus WinCC flexible auf den entsprechenden Datentyp in WinCC abgebildet. Nähere Hinweise hierzu finden Sie im Kapitel Migration der Datentypen (WinCC flexible) (Seite 137).

### Migration von Variablen

Variablen werden immer vollständig migriert. Lediglich die Datentypnamen und Variablennamen können sich durch die Migration ändern.

### Migration der Namen von Variablen

In WinCC flexible dürfen Variablen, die in verschiedenen Ordnern abgelegt werden, den gleichen Namen haben. In WinCC muss der Variablenname auf dem projektierten Bediengerät eindeutig sein. Bei der Migration werden daher gleichnamige Variablen aus verschiedenen Ordnern umbenannt. Nähere Hinweise hierzu finden Sie im Kapitel Grundlagen zur Migration (WinCC flexible) (Seite 115).

**Siehe auch**

- Grundlagen zur Migration (WinCC flexible) (Seite 115)
- Migration der Datentypen (WinCC flexible) (Seite 137)
- Unterstützte Bediengeräte (WinCC flexible) (Seite 120)

**Migration von Meldeklassen und Meldegruppen (WinCC flexible)**

**Änderung der Namen von Meldeklassen**

Im Unterschied zu WinCC flexible sind die Namen der vordefinierten Meldeklassen nicht abhängig von der aktuell verwendeten Oberflächensprache. Bei der Migration werden die Namen der Meldeklassen wie folgt vergeben:

| WinCC flexible | WinCC  |
|----------------|--------|
| Fehler         | Alarms |
| System         | System |
| Warnungen      | Events |

Nach der Migration können Sie die Namen der Meldeklassen beliebig verändern.

**Migration von Meldegruppen**

Bei der Migration werden nur die tatsächlich benutzten Meldegruppen migriert.

Meldegruppen mit einer ID von 1-31 werden 1:1 migriert.

In WinCC wird systemseitig pro Meldeklasse eine entsprechende Meldegruppe angelegt. Diese vom System angelegten Meldegruppen erhalten IDs ab Nummer 32 und werden fortlaufend hochgezählt. Die in jedem WinCC-Projekt vorhandenen 4 vordefinierten Meldeklassen belegen durch ihre Meldegruppen automatisch die IDs 32-35. Für jede benutzerdefinierte Meldeklasse wird eine weitere Meldegruppe angelegt und dafür eine weitere ID vergeben. Daher können sich bei Meldegruppen mit IDs > 31 die IDs nach der Migration ändern. Dadurch ändert sich auch die Zuordnung der Meldegruppennamen zu den IDs.

Beispiel:

In dem Beispiel sehen Sie die Vergabe der IDs in WinCC durch die Migration.

| Meldegruppen      | ID in WinCC flexible | ID in WinCC |  |
|-------------------|----------------------|-------------|--|
| Meldegruppe 1-16  | 1-16                 | 1-16        | Vorbelegt für die Meldegruppen von Systemmeldungen                       |
| Meldegruppe 17-31 | 17-31                | 17-31       | Benutzerdefinierte Meldegruppen  |
|                   |                      | 32-35       | In WinCC vorbelegt für die Meldegruppen der vordefinierten Meldeklassen. |

| Meldegruppen   | ID in WinCC flexible | ID in WinCC |  |
|----------------|----------------------|-------------|--|
| Meldegruppe 32 | 32                   | 36          | Geänderte Zuordnung von ID zu Meldegruppe in WinCC |
| Meldegruppe 33 | 33                   | 37          | Geänderte Zuordnung von ID zu Meldegruppe in WinCC |

Beachten Sie außerdem:

Bei der Migration von Meldegruppen mit vermeintlich gleichem Gruppennamen passt die Migration die Namen an. Dieser Fall tritt ein, wenn ein Gruppenname z. B. ein Leerzeichen am Namensende enthält. Die Migration löscht alle vorhandenen Leerzeichen am Namensende. Wenn 2 Gruppen durch den Löschvorgang den gleichen Gruppennamen erhalten, erweitert die Migration den Gruppennamen nachfolgender Meldegruppen mit dem Suffix "#Mign", wobei "n" eine fortlaufende Nummer ist.

Beispiel:

In WinCC flexible existieren folgende Meldegruppen:

"AlarmGroup\_18"

"AlarmGroup\_18 " - Gruppenname enthält 1 Leerzeichen

"AlarmGroup\_18 " - Gruppenname enthält 2 Leerzeichen

Die Meldegruppe "AlarmGroup\_18" ist die Meldegruppe mit der höchsten Nummer.

Ergebnis nach der Migration:

"AlarmGroup\_18"

"AlarmGroup\_18#Mig1"

"AlarmGroup\_18#Mig1.1"

### Änderung der Namen von Meldeklassen

Im Unterschied zu WinCC flexible sind die Namen der vordefinierten Meldeklassen nicht abhängig von der aktuell verwendeten Oberflächensprache. Bei der Migration werden die Namen der Meldeklassen wie folgt vergeben:

| WinCC flexible | WinCC    |
|----------------|----------|
| Fehler         | Errors   |
| System         | System   |
| Warnungen      | Warnings |

Nach der Migration können die Namen der Meldeklassen beliebig verändert werden.

### Anzeige von ALARM\_S Meldungen und SIMATIC SFM Meldungen

In WinCC flexible können Sie in integrierten Projekten die Anzeigeklassen für ALARM\_S-Meldungen aktivieren. Die Anzeige von SIMATIC SFM-Meldungen aktivieren Sie in WinCC flexible über eine separate Einstellung. In WinCC entfällt die separate Einstellung für die Aktivierung der Anzeige von SIMATIC SFM-Meldungen. Die Anzeige von SIMATIC SFM-Meldungen steuern Sie, wie auch die Anzeige von ALARM\_S-Meldungen, in WinCC ausschließlich über die Aktivierung der entsprechenden Anzeigeklasse.

Durch das geänderte Konzept, kann sich die Anzeige der Meldungen nach der Migration ändern.

Wenn im WinCC flexible-Projekt alle Anzeigeklassen für ALARM\_S-Meldungen aktiviert und die Anzeige von SIMATIC SFM-Meldungen deaktiviert sind, werden nach der Migration ALARM\_S-Meldungen und SIMATIC SFM-Meldungen angezeigt.

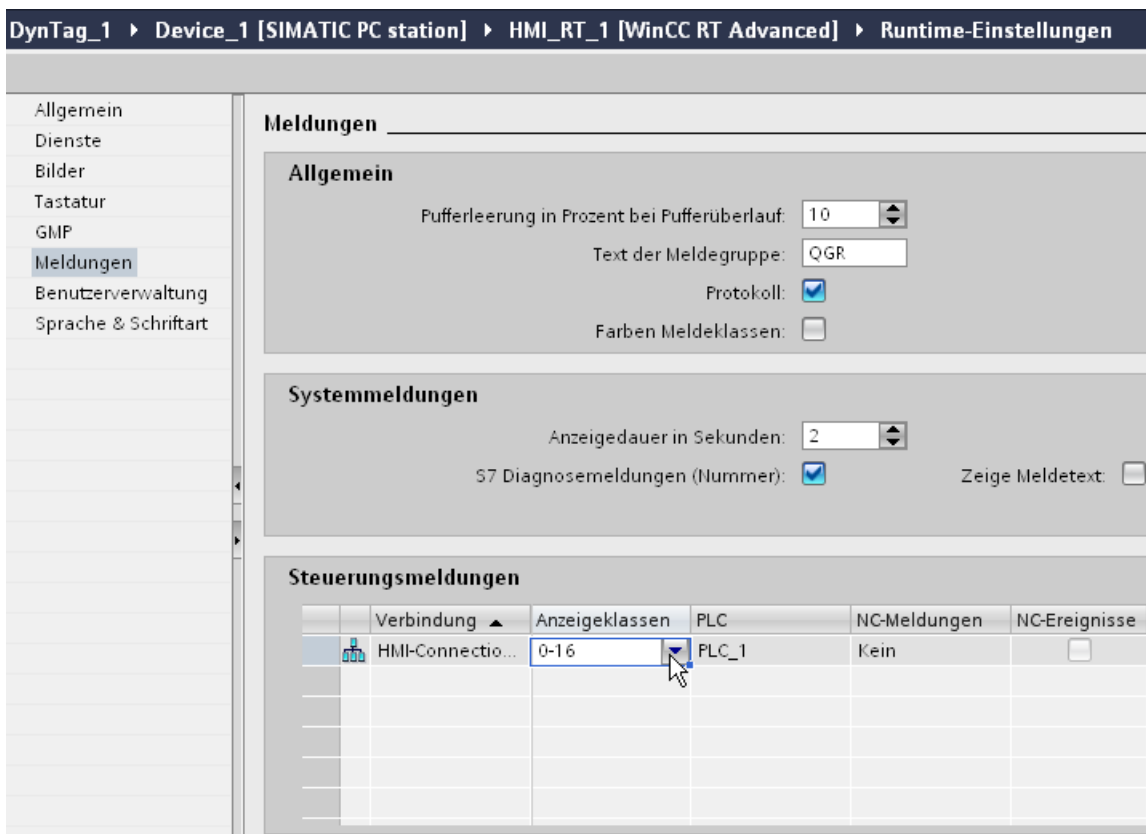
Damit nach der Migration nur ALARM\_S-Meldungen angezeigt werden, müssen Sie nach der Migration in STEP 7 die SIMATIC SFM-Meldungen einer nicht verwendeten Anzeigeklasse zuordnen. In WinCC müssen Sie dann diese Anzeigeklasse deaktivieren.

Wenn im WinCC flexible-Projekt alle Anzeigeklassen für ALARM\_S-Meldungen deaktiviert und die Anzeige von SIMATIC SFM-Meldungen aktiviert sind, werden nach der Migration keine ALARM\_S-Meldungen und keine SIMATIC SFM-Meldungen angezeigt.

Damit nach der Migration nur SIMATIC SFM-Meldungen angezeigt werden, müssen Sie nach der Migration in STEP 7 die SIMATIC SFM-Meldungen einer nicht verwendeten Anzeigeklasse zuordnen. In WinCC müssen Sie dann diese Anzeigeklasse aktivieren.

Die Anzeigeklasse ist abhängig von den Einstellungen in STEP 7. Die Standardeinstellung für SIMATIC SFM-Meldungen in Step 7 ist die Anzeigeklasse "0". Um die Anzeige in WinCC zu aktivieren, muss die Anzeigeklasse "0" aktiviert werden.

Die Anzeigeklassen aktivieren Sie in WinCC in den Runtime-Einstellungen des jeweiligen Bediengeräts in der Kategorie "Meldungen".



## Siehe auch

Unterstützte Bediengeräte (WinCC flexible) (Seite 120)

## Migration von sprachabhängigen Inhalten (WinCC flexible)

### Einleitung

In WinCC stehen Ihnen für die Projektierung in unterschiedlichen Sprachen die gleichen Möglichkeiten zur Verfügung wie in WinCC flexible. Bei der Migration eines Projekts werden alle Sprachen migriert, die WinCC unterstützt.

### Migration sprachabhängiger Inhalte

Bei der Migration werden folgende sprachabhängigen Inhalte migriert:

- Projektsprachen
- Projekttexte
- Schriftarten für die Darstellung in Runtime
- Sprachabhängige Grafiken

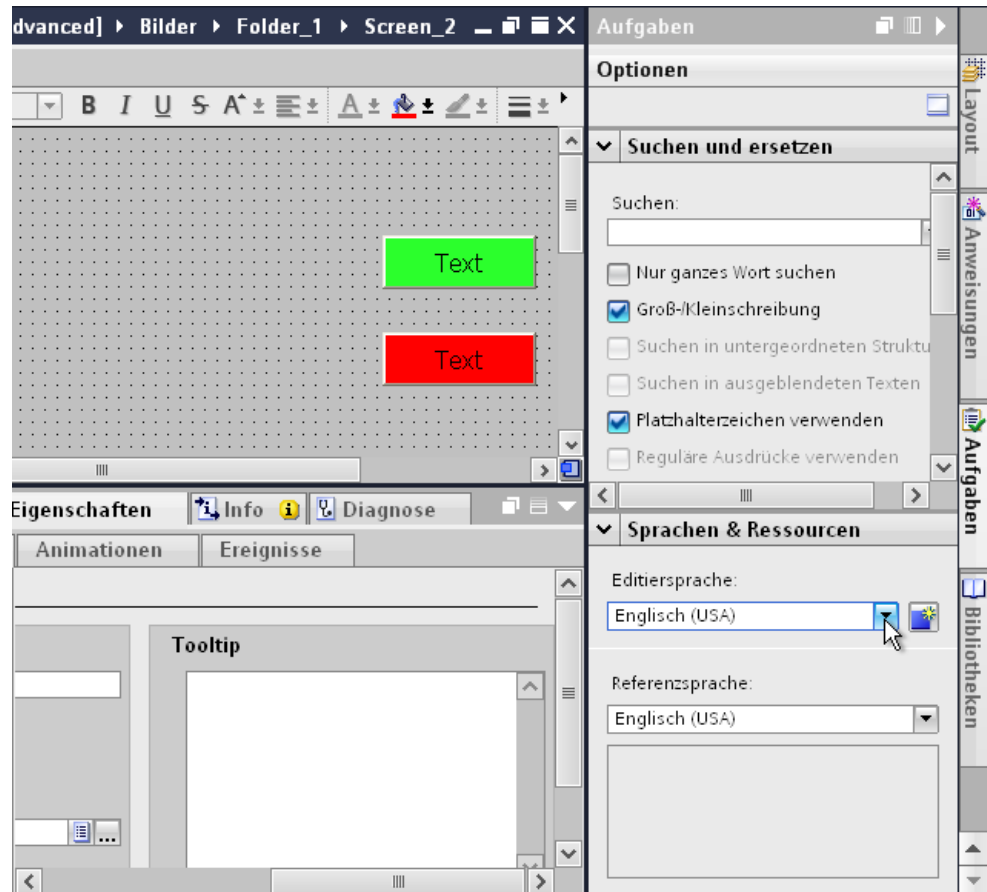
Bei der Migration sprachabhängiger Inhalte müssen Sie folgende Dinge beachten:

- Auf dem PC für die Migration muss das Betriebssystem die im Projekt verwendeten Sprachen unterstützen.
- Auf dem PC für die Migration müssen die Schriftarten installiert sein, die Sie für die Darstellung in Runtime verwenden.
- Wörterbücher werden von der Migration nicht unterstützt.

### Editiersprache integrierter Projekte nach der Migration

Bei der Migration eines integrierten Projekts bringen die zu migrierenden Projektanteile aus STEP 7 und WinCC flexible ihre jeweiligen Einstellungen für die Editiersprache mit. In WinCC gibt es nur noch eine Editiersprache für alle Projektanteile. Die Migration aktiviert für das migrierte Projekt die Editiersprache, die in STEP 7 vor der Migration eingestellt war. Wenn diese Einstellung nicht mit der Einstellung aus WinCC flexible übereinstimmt, dann sind die projektierten Texte in WinCC nicht mehr sichtbar. An den Verwendungsstellen der Texte wird kein Text angezeigt, oder es ist nur noch der Eintrag "Text" zu sehen. Um die Texte sichtbar zu machen, müssen Sie die Editiersprache wechseln. Klicken Sie am rechten Rand des TIA

Portals auf die Taskcard "Aufgaben" und wählen Sie im Bereich "Sprachen & Ressourcen" die richtige Editiersprache aus.



## Nicht unterstützte Sprachen

Die Migration von sprachabhängigen Inhalten ist abhängig davon, ob WinCC die jeweilige Sprache unterstützt.

Wenn ein Projekt nur Projektsprachen enthält, die WinCC nicht unterstützt, wird das Projekt nicht migriert.

Wenn ein Projekt unterstützte und nicht unterstützte Projektsprachen enthält, werden nur die unterstützten Sprachen migriert. Die Bearbeitungssprache und die Referenzsprache werden auf eine unterstützte Sprache gesetzt.

Folgende Sprachen werden von WinCC nicht unterstützt:

- Arabisch
- Hebräisch
- Dhivehi
- Gujara
- Kannada

- Tamil
- Telugu
- Urdu
- Punjabi
- Persisch
- Syrisch

### Siehe auch

Unterstützte Bediengeräte (WinCC flexible) (Seite 120)

## Migration von Bibliotheken (WinCC flexible)

### Einleitung

Bei der Migration von Bibliotheken müssen Sie zwei Fälle unterscheiden:

1. Migration der Projektbibliothek
2. Migration einer globalen Bibliothek

### Migration der Projektbibliothek

Eine Projektbibliothek wird gemeinsam mit den Projektdaten in der Projektdatei gespeichert. Daher wird eine Projektbibliothek mit den gleichen Einschränkungen migriert wie die Projektdaten.

### Migration einer globalen Bibliothek

Globale Bibliotheken werden von der Migration nicht unterstützt. Die im Projekt verwendeten Bibliotheksobjekte werden jedoch migriert. Die Bibliotheksobjekte werden bei der Verwendung in das Projekt kopiert und haben danach keine Verbindung mehr zur Bibliothek.

Um eine globale Bibliothek zu migrieren, müssen Sie die in der Bibliothek enthaltenen Objekte in die Projektbibliothek verschieben oder kopieren. Bei der Migration werden die Objekte dann migriert. In WinCC verschieben Sie die migrierten Objekte in eine neu zu erstellende, globale Bibliothek. Sie können sowohl einzelne Objekte als auch ganze Bibliothekskategorien verschieben oder kopieren.

### Siehe auch

Unterstützte Bediengeräte (WinCC flexible) (Seite 120)



### 4.1.6.3 Runtime-Daten migrieren (WinCC flexible)

#### Migration von Runtime-Daten (WinCC flexible)

##### Einleitung

Bei der Migration eines Projekts werden lediglich die Projektierungsdaten migriert. Die Runtime-Daten sind davon nicht betroffen. Die Runtime-Daten müssen Sie nach der Migration aktualisieren.

Die Runtime-Daten setzen sich wie folgt zusammen:

- Runtime-Projekt  
Das Runtime-Projekt enthält die übersetzten Projektdaten.
- Rezepturdaten und die Benutzerverwaltung  
Die Rezepturdaten und die Benutzerverwaltung sind Daten, die in Runtime verändert werden können.

##### Migration der Runtime-Daten

Das Runtime-Projekt aktualisieren Sie, indem Sie das Projekt in WinCC neu übersetzen und auf das Bediengerät laden.

Wenn die Rezepturdaten und die Benutzerverwaltung in Runtime geändert wurden, müssen Sie diese Daten vom Bediengerät sichern, bevor Sie das migrierte Projekt laden. Anschließend laden Sie das migrierte Projekt auf das Bediengerät. Abschließend laden Sie die gesicherten Rezepturdaten und die Benutzerverwaltung wieder auf das Bediengerät. Nähere Hinweise hierzu finden Sie im Kapitel Auto-Hotspot.

##### Siehe auch

Migration (WinCC flexible) (Seite 114)

Rezepturdaten und Benutzerverwaltung sichern (WinCC flexible) (Seite 131)

Rezepturdaten und Benutzerverwaltung wiederherstellen (WinCC flexible) (Seite 132)

#### Rezepturdaten und Benutzerverwaltung sichern (WinCC flexible)

##### Einleitung

Um die Rezepturdaten und die Benutzerverwaltung in einem migrierten Projekt weiter zu verwenden, müssen Sie diese Daten zuerst vom Bediengerät sichern. Anschließend laden Sie die Daten in das migrierte WinCC-Projekt. Zur Sicherung der Daten verwenden Sie ProSave.

## Voraussetzung

- Das WinCC flexible-Projekt läuft auf dem Bediengerät in Runtime.
- Das Bediengerät ist mit einem PC verbunden, auf dem ProSave installiert ist.

## Vorgehen

Um die Rezepturdaten und die Benutzerverwaltung zu sichern, gehen Sie folgendermaßen vor:

1. Starten Sie ProSave.
2. Wählen Sie auf der Registerkarte "Allgemein" den Gerätetyp und die Verbindungsparameter aus.
3. Aktivieren Sie die Registerkarte "Backup".
4. Wählen Sie im Feld "Datenart" den Eintrag "Rezepturen aus dem Gerätespeicher" aus. Wählen Sie nicht "Komplett Backup", da Sie sonst bei der Wiederherstellung die Rezepturdaten nicht mehr separat auswählen können.
5. Navigieren Sie im Feld "Speichern unter" zum gewünschten Ablageort und klicken Sie anschließend auf die Schaltfläche "Start Backup". Die Rezepturdaten werden gesichert.
6. Wählen Sie im Feld "Datenart" den Eintrag "Benutzerverwaltung" aus und klicken Sie auf die Schaltfläche "Start Backup". Die Benutzerverwaltung wird gesichert.

Nähere Hinweise hierzu finden Sie in der Online-Hilfe von ProSave.

## Alternative Vorgehensweise

ProSave wird standardmäßig mit WinCC flexible installiert. Auf dem Projektierungs-PC steht Ihnen innerhalb von WinCC flexible über den Menübefehl "Projekt > Transfer" der gesamte Funktionsumfang von ProSave zur Verfügung.

Alternativ sichern Sie die Rezepturdaten und die Benutzerverwaltung über das in WinCC flexible integrierte ProSave. Starten Sie WinCC flexible und wählen Sie den Menübefehl "Projekt > Transfer > Sicherung". Sichern Sie die Rezepturdaten und die Benutzerverwaltung, wie in den Schritten 4-6 beschrieben.

## Siehe auch

Migration von Runtime-Daten (WinCC flexible) (Seite 131)

Rezepturdaten und Benutzerverwaltung wiederherstellen (WinCC flexible) (Seite 132)

## Rezepturdaten und Benutzerverwaltung wiederherstellen (WinCC flexible)

### Einleitung

Um die gesicherten Rezepturdaten und die Benutzerverwaltung nach der Migration weiter zu verwenden, müssen Sie zuerst das migrierte Projekt übersetzen und auf das Bediengerät

laden. Anschließend transferieren Sie die gesicherten Daten auf das Bediengerät. Zur Wiederherstellung der Daten verwenden Sie ProSave.

### Voraussetzung

- Das migrierte Projekt ist auf das Bediengerät transferiert und läuft in Runtime.
- Das Bediengerät ist mit einem PC verbunden, auf dem ProSave installiert ist.

### Vorgehen

Um die gesicherten Rezepturdaten und die Benutzerverwaltung auf das Bediengerät zu laden, gehen Sie folgendermaßen vor:

1. Starten Sie ProSave.
2. Wählen Sie auf der Registerkarte "Allgemein" den Gerätetyp und die Verbindungsparameter aus.
3. Aktivieren Sie die Registerkarte "Restore".
4. Navigieren Sie im Feld "Öffnen von ..." zum Speicherort der gesicherten Rezepturdaten und wählen Sie die Datei aus.
5. Klicken Sie auf die Schaltfläche "Start Restore".  
Die Rezepturdaten werden auf das Bediengerät transferiert.
6. Wiederholen Sie die Schritte 4-5 zur Wiederherstellung der Benutzerverwaltung.  
Die Benutzerverwaltung wird auf das Bediengerät transferiert.

Nähere Hinweise hierzu finden Sie in der Online-Hilfe von ProSave.

### Alternative Vorgehensweise

ProSave wird standardmäßig mit WinCC installiert. Auf dem Projektierungs-PC steht Ihnen innerhalb von WinCC über den Menübefehl "Projekt > Transfer" der gesamte Funktionsumfang von ProSave zur Verfügung.

Die Rezepturdaten und die Benutzerverwaltung stellen Sie alternativ über das in WinCC integrierte ProSave wieder her. Starten Sie WinCC und wählen Sie den Menübefehl "Online > Bediengeräte Wartung > Wiederherstellen". Stellen Sie die Rezepturdaten und die Benutzerverwaltung wieder her, wie in den Schritten 4-6 beschrieben.

### Siehe auch

Migration von Runtime-Daten (WinCC flexible) (Seite 131)

Rezepturdaten und Benutzerverwaltung sichern (WinCC flexible) (Seite 131)

#### 4.1.6.4 Integrierte Projekte migrieren (WinCC flexible)

### Migration integrierter Projekte (WinCC flexible)

#### Einleitung

In einem in STEP 7 integrierten Projekt sind enthaltene Steuerungen und Bediengeräte durch die Projektierung miteinander verbunden. Auch die Projektierungsdaten von WinCC flexible und STEP 7 haben Verbindungen zueinander. Bei der Migration eines integrierten Projekts wird das komplette Projekt mit den Anteilen aus WinCC flexible und STEP 7 migriert. Die Verbindungen bleiben erhalten.

---

#### Hinweis

Bevor sie ein integriertes Projekt migrieren, sollten sie dieses Projekt in WinCC flexible übersetzen und speichern. Eine fehlerfreie Übersetzung stellt sicher, dass die Daten in WinCC flexible und in STEP 7 synchronisiert sind.

---

### Migration eines integrierten Projekts

Bei der Migration eines integrierten Projekts gelten für den WinCC flexible-Anteil die gleichen Voraussetzungen wie für die Migration eines nicht integrierten WinCC flexible-Projekts. Die im WinCC flexible-Anteil enthaltenen Objekte und Eigenschaften müssen von WinCC unterstützt werden, z. B. das Bediengerät oder der Kommunikationstreiber. An einer projektierten Verbindung muss die Eigenschaft "Online" aktiviert sein. Eine Verbindung, deren Eigenschaft "Online" deaktiviert ist, wird nicht migriert.

Neben den Voraussetzungen für den WinCC flexible-Anteil gelten weitere Voraussetzungen für den STEP 7-Anteil des integrierten Projekts. Die im STEP 7 V5.4 SP5 oder V5.5-Anteil enthaltenen Objekte und Eigenschaften müssen in STEP 7 unterstützt werden. Nähere Hinweise hierzu finden Sie in der Dokumentation zu STEP 7.

Um ein integriertes Projekt vollständig zu migrieren und nachbearbeiten zu können, müssen auf dem PC für die Migration folgende Komponenten installiert sein:

- STEP 7 V5.4 SP5 oder STEP 7 V5.5
- WinCC flexible 2008 SP2 oder WinCC flexible 2008 SP3
- STEP 7

Wenn Sie das Projekt nur im Migrationsformat speichern wollen, steht Ihnen das Migrationstool zur Verfügung. Nähere Hinweise hierzu finden Sie unter Grundlagen zur Migration (WinCC flexible) (Seite 115).

Ein integriertes Projekt wird immer vollständig migriert. Wenn Sie lediglich das darin enthaltene WinCC flexible-Projekt migrieren wollen, müssen Sie es vor der Migration aus dem STEP 7-Projekt lösen. Um das Projekt aus der Integration zu lösen, öffnen Sie das Projekt in STEP 7 V5.4 SP5 oder V5.5. Öffnen Sie im SIMATIC-Manager das WinCC flexible-Projekt. Das Projekt wird mit WinCC flexible geöffnet. Wählen Sie in WinCC flexible den Menübefehl "Projekt >

Kopiere Projekt aus STEP 7". WinCC flexible speichert eine nicht integrierte Kopie des Projekts.

## Siehe auch

Grundlagen zur Migration (WinCC flexible) (Seite 115)

## Integriertes Projekt migrieren (WinCC flexible)

### Einleitung

Bei der Migration eines integrierten Projekts werden sowohl die Anteile aus dem WinCC flexible-Projekt als auch aus dem STEP 7-Projekt migriert. Deshalb müssen Sie für die Migration die Projektdatei mit der Dateierdung "\*.s7p" wählen. Bei der Migration werden die Daten aus dem bestehenden Projekt kopiert und in ein neues Projekt migriert. Sie können nicht in ein bestehendes Projekt migrieren.

Die Migration lässt sich sowohl in der Portalansicht als auch in der Projektansicht aufrufen.

Migrieren Sie ein Projekt nur in einem neu gestartetem TIA-Portal.

Wenn Sie das Projekt nur im Migrationsformat speichern wollen, steht Ihnen das Migrationstool zur Verfügung. Nähere Hinweise hierzu finden Sie unter Grundlagen zur Migration (WinCC flexible) (Seite 115).

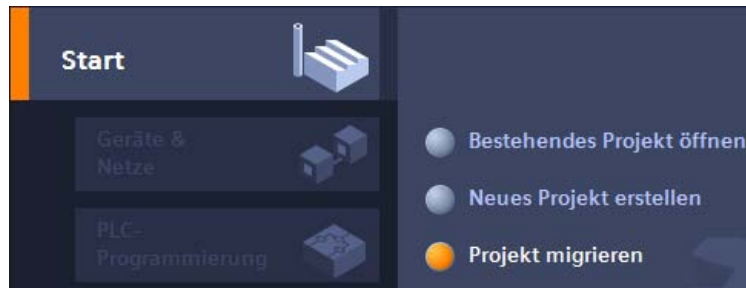
### Voraussetzung

- STEP 7 V5.4 SP5 oder STEP 7 V5.5 und alle verwendeten Optionspakete sind installiert.
- STEP 7 und alle verwendeten Optionspakete sind installiert.
- Das TIA-Portal ist neu gestartet.
- In WinCC ist kein Projekt geöffnet.
- Ein integriertes Projekt ist vorhanden.
- Das integrierte Projekt ist nicht geöffnet.

## Vorgehen

Ein integriertes Projekt migrieren Sie in der Portalansicht wie folgt:

1. Wählen Sie die Aktion "Start > Projekt migrieren".



2. Navigieren Sie im Feld "Quellpfad" zu dem Projekt, das Sie migrieren wollen.

The image shows the 'Projekt migrieren' (Migrate Project) dialog box. The title bar says 'Projekt migrieren'. Below the title bar, there is a section titled 'Wählen Sie das zu migrierende Projekt aus.' (Select the project to migrate). This section contains a 'Projektname:' field with the value 'TestMigration', a 'Quellpfad:' field with the value 'C:\Temp\TestMigration.s7p', and a checkbox labeled 'Hardwarekonfiguration ausschließen' (Exclude hardware configuration) which is currently unchecked. Below this section is a section titled 'Ziel' (Target). This section contains a 'Projektname:' field with the value 'TestMigration', a 'Zielpfad:' field with the value 'C:\Temp\_V11', an 'Autor:' field with the value 'user', and a 'Kommentar:' field which is empty. At the bottom right of the dialog box, there is a 'Migrieren' (Migrate) button.

3. Selektieren Sie die Projektdatei "\*.s7p".
4. Ändern Sie bei Bedarf die Angaben für das neu zu erstellende Projekt. Ändern Sie z. B. den Projektnamen oder den Projektpfad. In dem neuen Projekt werden die zu migrierenden Daten angelegt.
5. Um das Projekt ohne Hardwarekonfiguration zu migrieren, aktivieren Sie "Hardwarekonfiguration ausschließen".

6. Klicken Sie auf "Migrieren".  
Ein neues Projekt wird erstellt und die Migration der Daten wird gestartet:
  - Die Projektansicht wird geöffnet.
  - Der Fortschritt der Migration wird in einem Migrationsfenster angezeigt.
  - Im Inspektorfenster unter "Info > Allgemein" werden Informationen, Warnungen und Fehler zum Migrationsvorgang ausgegeben.
  - Sämtliche Informationen zur Migration werden in einer Protokolldatei gespeichert.
  - Nach Abschluss der Migration wird eine Meldung ausgegeben. In der Meldung finden Sie einen Link, über den Sie die Protokolldatei öffnen können.
7. Speichern Sie das Projekt nach Ende der Migration.

Nach Abschluss der Migration finden Sie in der Projektnavigation für jedes migrierte Bediengerät und für jede migrierte Steuerung ein neu angelegtes Gerät. Diese Geräte beinhalten die migrierten Daten.

### Migrationsprotokoll zu einem späteren Zeitpunkt öffnen

Das Migrationsprotokoll wird zusammen mit dem migrierten Projekt gespeichert. Sie können das Protokoll zu einem späteren Zeitpunkt ansehen. Die Protokolldatei öffnen Sie folgendermaßen:

1. Öffnen Sie in der Projektnavigation "Gemeinsame Daten > Protokolle".
2. Doppelklicken Sie die Protokolldatei. Das Migrationsprotokoll wird geöffnet.

### Siehe auch

Grundlagen zur Migration (WinCC flexible) (Seite 115)

### 4.1.6.5 Referenz (WinCC flexible)

### Migration der Datentypen (WinCC flexible)

#### Einleitung

Zur Harmonisierung der verwendeten Datentypen von Steuerungen und HMI-Systemen werden einige Datentypen der internen HMI-Variablen umbenannt. Die Namensgebung erfolgt IEC konform. Da sich lediglich die Namen ändern, ergeben sich für die Projektierung bei internen Variablen keine Änderungen.

Die nachfolgende Tabelle beschreibt die Abbildung der Datentypen aus WinCC flexible auf die Datentypen in WinCC.

### Migration der Datentypen

Die internen Datentypen werden bei der Migration folgendermaßen aufeinander abgebildet:

| Interne Datentypen WinCC flexible | Interne Datentypen WinCC |
|-----------------------------------|--------------------------|
| Bool                              | Bool                     |
| Char                              | SInt                     |
| Byte                              | USInt                    |
| Int                               | Int                      |
| UInt                              | UInt                     |
| Long                              | DInt                     |
| ULong                             | UDInt                    |
| Float                             | Real                     |
| Double                            | LReal                    |
| String                            | WString                  |
| DateTime                          | DateTime                 |

### Migration der externen Datentypen

Wie die Datentypen der verfügbaren Kommunikationstreiber abgebildet werden, finden Sie auf den nachfolgenden Seiten.

#### Siehe auch

- Migration (WinCC flexible) (Seite 114)
- Migration der Datentypen von Allen-Bradley DF1 (WinCC flexible) (Seite 139)
- Migration der Datentypen von Allen-Bradley Ethernet IP (WinCC flexible) (Seite 139)
- Migration der Datentypen von Mitsubishi FX (WinCC flexible) (Seite 140)
- Migration der Datentypen von Modicon Modbus (WinCC flexible) (Seite 140)
- Migration der Datentypen von Modicon Modbus TCP/IP (WinCC flexible) (Seite 141)
- Migration der Datentypen von Omron Hostlink/Multilink (WinCC flexible) (Seite 141)
- Migration der Datentypen von SIMATIC S7 200 (WinCC flexible) (Seite 142)
- Migration der Datentypen von SIMATIC S7 300/400 (WinCC flexible) (Seite 142)



## Migration der Datentypen von Allen-Bradley DF1 (WinCC flexible)

### Migration der Datentypen Allen-Bradley DF1

Die Datentypen des Kommunikationstreibers Allen-Bradley DF1 werden bei der Migration nach WinCC folgendermaßen abgebildet:

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| ASCII                      | ASCII             |
| BCD4                       | UInt              |
| BCD8                       | UDInt             |
| Bit                        | Bool              |
| Int                        | Int               |
| Long                       | DInt              |
| Real                       | Real              |
| UInt                       | UInt              |
| ULong                      | UDInt             |

#### Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 137)

## Migration der Datentypen von Allen-Bradley Ethernet IP (WinCC flexible)

### Migration der Datentypen Allen-Bradley Ethernet IP

Die Datentypen des Kommunikationstreibers Allen-Bradley Ethernet IP werden bei der Migration nach WinCC folgendermaßen abgebildet:

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| Bool                       | Bool              |
| DInt                       | DInt              |
| Int                        | Int               |
| Real                       | Real              |
| SInt                       | SInt              |
| String                     | String            |
| UDInt                      | UDInt             |
| UInt                       | UInt              |
| USInt                      | USInt             |

#### Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 137)

### Migration der Datentypen von Mitsubishi FX (WinCC flexible)

#### Migration der Datentypen Mitsubishi FX

Die Datentypen des Kommunikationstreibers Mitsubishi FX werden bei der Migration nach WinCC folgendermaßen abgebildet:

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| 12 Bit Block               | 12-Bit Block      |
| 16 Bit Block               | 16-Bit Block      |
| 20 Bit Block               | 20-Bit Block      |
| 24 Bit Block               | 24-Bit Block      |
| 28 Bit Block               | 28-Bit Block      |
| 32 Bit Block               | 32-Bit Block      |
| 4 Bit Block                | 4-Bit Block       |
| 8 Bit Block                | 8-Bit Block       |
| Bit                        | Bool              |
| Double                     | DWord             |
| IEEE-Float                 | Real              |
| String                     | String            |
| Word                       | Word              |

#### Siehe auch

Migration der Datentypen (WinCC flexible) (Seite 137)

### Migration der Datentypen von Modicon Modbus (WinCC flexible)

#### Migration der Datentypen Modicon Modbus

Der Kommunikationstreiber Modicon Modbus wird von WinCC nicht unterstützt, er wird bei der Migration durch den Treiber Modicon Modbus RTU ersetzt. Die Datentypen des Kommunikationstreibers Modicon Modbus werden bei der Migration nach WinCC folgendermaßen abgebildet:

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| +/-Double                  | +/- Double        |
| +/-Int                     | +/- Int           |
| 16 Bit Group               | 16 Bit Group      |
| ASCII                      | ASCII             |
| Bit                        | Bit               |
| Double                     | Double            |
| Float                      | Float             |
| Int                        | Int               |

**Siehe auch**

Migration der Datentypen (WinCC flexible) (Seite 137)

**Migration der Datentypen von Modicon Modbus TCP/IP (WinCC flexible)**

**Migration der Datentypen Modicon Modbus TCP/IP**

Die Datentypen des Kommunikationstreibers Modicon Modbus TCP/IP werden bei der Migration nach WinCC folgendermaßen abgebildet:

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| +/-Double                  | +/- Double        |
| +/-Int                     | +/- Int           |
| 16 Bit Group               | 16 Bit Group      |
| ASCII                      | ASCII             |
| Bit                        | Bit               |
| Double                     | Double            |
| Float                      | Float             |
| Int                        | Int               |

**Siehe auch**

Migration der Datentypen (WinCC flexible) (Seite 137)

**Migration der Datentypen von Omron Hostlink/Multilink (WinCC flexible)**

**Migration der Datentypen Omron Hostlink/Multilink**

Der Kommunikationstreiber Omron Hostlink/Multilink wird von WinCC nicht unterstützt, er wird bei der Migration durch den Treiber Omron Host Link ersetzt. Die Datentypen des Kommunikationstreibers Omron Hostlink/Multilink werden bei der Migration nach WinCC folgendermaßen abgebildet:

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| +/-DEC                     | Int               |
| +/-LDEC                    | DInt              |
| ASCII                      | String            |
| BIN                        | Bool              |
| BYTE                       | Byte              |
| DEC                        | UInt              |
| IEEE                       | Real              |
| LDEC                       | UDInt             |

**Siehe auch**

Migration der Datentypen (WinCC flexible) (Seite 137)

**Migration der Datentypen von SIMATIC S7 200 (WinCC flexible)**

**Migration der Datentypen SIMATIC S7 200**

Die Datentypen des Kommunikationstreibers SIMATIC S7 200 werden bei der Migration nach WinCC folgendermaßen abgebildet:

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| Bool                       | Bool              |
| Byte                       | Byte              |
| Char                       | Char              |
| DInt                       | DInt              |
| DWord                      | DWord             |
| Int                        | Int               |
| Real                       | Real              |
| StringChar                 | StringChar        |
| Timer                      | Timer             |
| Word                       | Word              |

**Siehe auch**

Migration der Datentypen (WinCC flexible) (Seite 137)

**Migration der Datentypen von SIMATIC S7 300/400 (WinCC flexible)**

**Migration der Datentypen SIMATIC S7 300/400**

Die Datentypen des Kommunikationstreibers SIMATIC S7 300/400 werden bei der Migration nach WinCC folgendermaßen abgebildet:

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| Bool                       | Bool              |
| Byte                       | Byte              |
| Char                       | siehe unten       |
| Counter                    | siehe unten       |
| Date                       | Date              |
| Date and Time              | Date_And_Time     |
| DInt                       | DInt              |
| DWord                      | DWord             |
| Int                        | Int               |

| Datentyp in WinCC flexible | Datentyp in WinCC |
|----------------------------|-------------------|
| Real                       | Real              |
| String                     | String            |
| StringChar                 | siehe unten       |
| Time                       | Time              |
| Time of Day                | Time_Of_Day       |
| Timer                      | siehe unten       |
| Word                       | Word              |

### Besonderheiten bei einigen Datentypen

Bei der Migration von externen Variablen, die Datentypen einer SIMATIC S7-300/400 Steuerung enthalten, müssen Sie einige Besonderheiten beachten.

#### Abbildung des S7-Datentyps "Char"

Der S7-Datentyp "Char" ist gemäß Spezifikation ein Datentyp zur Abbildung von Zeichen. Da dieser Datentyp aber oft zum Lesen und Schreiben von Zahlenwerten verwendet wurde, wird er in WinCC abgebildet auf den S7-Datentyp "Byte". Wenn dieser Fall bei der Migration eintritt, wird im Ausgabefenster eine Meldung ausgegeben.

Wenn der S7-Datentyp "Char" für Zahlenwerte verwendet wird und an der Verwendungsstelle negative Zahlen projiziert wurden, führt dies bei der Abbildung auf den S7-Datentyp "Byte" zu einem Fehler. Der S7-Datentyp "Byte" kann keine negativen Zahlen abbilden. Um den Fehler zu beheben, müssen Sie die Projektierung entsprechend anpassen. Verwenden Sie für die Verarbeitung von positiven und negativen Zahlenwerten einen vorzeichenbehafteten Datentyp, z. B. den Datentyp "Int".

Wenn der S7-Datentyp "Char" doch zur Abbildung von Zeichen genutzt wird, müssen Sie nach der Migration die Projektierung ändern. Zur Darstellung von Zeichen benutzen Sie den Datentyp "String".

Bei der Migration eines integrierten Projekts wird der Datentyp "Char" in WinCC ebenfalls auf den Datentyp "Byte" migriert. Bei einer verbundenen PLC-Variablen wird der Datentyp "Char" allerdings auf "Char" belassen. Durch die Änderung des Datentyps der HMI-Variablen wird die symbolische Adressierung der betreffenden Variablen nicht migriert. Die Variablen sind nach der Migration durch absolute Adressen miteinander verbunden und funktionieren weiterhin. Wenn Sie die symbolische Adressierung wieder herstellen wollen, müssen Sie die Projektierung nach der Migration entsprechend ändern.

#### Abbildung eines Arrays des S7-Datentyps "Char"

Ein Array vom S7-Datentyp "Char" wird bei der Migration auf ein Array des Datentyps "Byte" abgebildet.

Wenn ein Array vom S7-Datentyp "Char" für Zahlenwerte verwendet wird und an der Verwendungsstelle negative Zahlen projiziert wurden, führt dies bei der Abbildung auf ein Array vom S7-Datentyp "Byte" zu einem Fehler. Der S7-Datentyp "Byte" kann keine negativen Zahlen abbilden. Um den Fehler zu beheben, müssen Sie die Projektierung entsprechend anpassen. Verwenden Sie für die Verarbeitung von positiven und negativen Zahlenwerten einen vorzeichenbehafteten Datentyp, z. B. den Datentyp "Int".

### Abbildung des S7-Datentyps "Counter"

Eine externe Variable mit dem S7-Datentyp "Counter" mit Counter-Adresse wird auf den S7-Datentyp "Counter" abgebildet. Die Adresse wird beibehalten.

Wenn eine externe Variable mit dem S7-Datentyp "Counter" einen Datenbaustein oder einer Merker-Adresse adressiert, wird auf den S7-Datentyp "Word" abgebildet. Die Adresse wird beibehalten. Die Migration setzt die Codierung auf "SimaticBCDCounter".

Der S7-Datentyp "Counter" hat einen Wertebereich von 0-999. Bei der Versorgung über den S7-Datentyp "Word" ist eine Überschreitung des Wertebereichs auf der Steuerungsseite möglich. Stellen Sie sicher, dass der Wertebereich eingehalten wird.

Beispiel:

#### WinCC flexible

| Variable                 | S7-Datentyp | Adresse     | Kommentar                |
|--------------------------|-------------|-------------|--------------------------|
| Counter_Actual_Value     | Counter     | C10         | BCD codierter Zählerwert |
| Counter_Setpoint_Value   | Counter     | DB10.DBW200 | BCD codierter Zählerwert |
| Counter_Setpoint_Value#2 | Counter     | MW20        | BCD codierter Zählerwert |

#### WinCC

| Variable                 | S7-Datentyp | Adresse       | Codierung         | Kommentar                |
|--------------------------|-------------|---------------|-------------------|--------------------------|
| Counter_Actual_Value     | Counter     | %C10          | <Standard>        | BCD codierter Zählerwert |
| Counter_Setpoint_Value   | Word        | %DB10.%DBW200 | SimaticBCDCounter | BCD codierter Zählerwert |
| Counter_Setpoint_Value#2 | Word        | %MW20         | SimaticBCDCounter | BCD codierter Zählerwert |

### Abbildung des Datentyps "StringChar"

In WinCC gibt es keinen entsprechenden Datentyp, auf den der Datentyp "StringChar" abgebildet werden kann. Die Abbildung in WinCC ist abhängig von der Eigenschaft "Länge" des S7-Datentyps.

Eine Variable vom Datentyp "StringChar" mit der Eigenschaft "Länge" > 1 wird auf ein Array vom S7-Datentyp "Char" migriert. Die Länge des Arrays entspricht der Länge des ursprünglich projektierten Datentyps "StringChar".

Wenn die Eigenschaft "Länge" = 1 ist, wird der Datentyp in WinCC auf ein Array vom S7-Datentyp "Char" mit der Länge = 1 migriert. Der Ausdruck für ein Array mit einem Element lautet "Array[0 ..0] of Char".

### Abbildung des S7-Datentyps "Timer"

Eine externe Variable mit dem S7-Datentyp "Timer" mit Timer-Adresse wird auf den S7-Datentyp "Timer" abgebildet. Die Adresse wird beibehalten.

Wenn eine externe Variable mit dem S7-Datentyp "Timer" einen Datenbaustein oder eine Merker-Adresse adressiert, wird auf den S7-Datentyp "S5 Time" abgebildet. Die Adresse wird beibehalten.

Beispiel:

**WinCC flexible**

| Variable               | S7-Datentyp | Adresse     | Kommentar                |
|------------------------|-------------|-------------|--------------------------|
| Timer_Actual_Value     | Timer       | T10         | BCD codierter Timer-Wert |
| Timer_Setpoint_Value   | Timer       | DB10.DBW200 | BCD codierter Timer-Wert |
| Timer_Setpoint_Value#2 | Timer       | MW20        | BCD codierter Timer-Wert |

**WinCC**

| Variable               | S7-Datentyp | Adresse       | Kommentar                |
|------------------------|-------------|---------------|--------------------------|
| Timer_Actual_Value     | Timer       | %T10          | BCD codierter Timer-Wert |
| Timer_Setpoint_Value   | S5Time      | %DB10.%DBW200 | BCD codierter Timer-Wert |
| Timer_Setpoint_Value#2 | S5Time      | %MW20         | BCD codierter Timer-Wert |

**Siehe auch**

Migration der Datentypen (WinCC flexible) (Seite 137)

**4.1.7 Integrierte Projekte migrieren**

**4.1.7.1 Migration eines integrierten Projekts**

**Einleitung**

Bei der Migration eines integrierten Projekts wird das komplette Projekt mit den Anteilen aus WinCC und STEP 7 migriert. Projektierte Verbindungen zwischen Steuerung und Visualisierung bleiben erhalten.

**Migration eines integrierten Projekts**

Bei der Migration eines integrierten Projekts gelten für den STEP 7-Anteil die gleichen Voraussetzungen wie für die Migration eines nicht integrierten STEP 7-Projekts. Die im WinCC Anteil enthaltenen Objekte und Eigenschaften müssen ebenso in WinCC (TIA-Portal) unterstützt werden. Nähere Hinweise hierzu finden Sie in der Dokumentation zu WinCC.

Beachten Sie zusätzlich, dass das Ausgangsprojekt vor der Migration übersetzt werden muss.

Um ein integriertes Projekt vollständig migrieren zu können, müssen auf dem PG/PC für die Migration folgende Komponenten installiert sein:

- STEP 7 V5.4 SP5 oder STEP 7 V5.5
- WinCC V7.0 SP3 oder WinCC Flexible 2008 SP2 und SP3

Um ein integriertes Projekt vollständig nachbearbeiten zu können, müssen auf dem PC für die Nachbearbeitung folgende Komponenten installiert sein:

- STEP 7 Professional V12 SP1
- WinCC Basic, WinCC Comfort/Advanced oder WinCC Professional, in Abhängigkeit der verwendeten Komponenten

### Verwendung des Migrations-Tools

Unter den folgenden Voraussetzungen ist die Verwendung des Migrations-Tools erforderlich:

- Das Ausgangsprojekt befindet sich nicht auf demselben PG/PC wie die Installation des TIA-Portals.
- Im Ausgangsprojekt befinden sich SCADA-Geräte. Diese können ausschließlich mit dem Migrations-Tool migriert werden.
- WinCC Professional V12 und STEP 7 mit WinCC V7.0 SP3 können nicht auf demselben PG/PC installiert sein. Daher müssen integrierte Projekte mit WinCC V7.0 SP3-Anteilen mithilfe des Migrations-Tools für die Migration aufbereitet werden.

### Migration des STEP 7-Teils eines integrierten Projekts

Ein integriertes Projekt wird immer vollständig migriert. Einzelne Teile daraus können nicht alleine migriert werden. Sie können lediglich das enthaltene STEP 7-Projekt alleine migrieren, wenn Sie zuvor im SIMATIC-Manager alle HMI-Stationen löschen und anschließend in NetPro das Projekt neu übersetzen.

Alternativ können Sie das Projekt in einer Installation von STEP 7 V5.4 SP5 oder V5.5 ohne eine Installation von WinCC öffnen. Speichern Sie das Projekt dann neu ab und wählen Sie beim Speichern die Funktion "Reorganisieren". Dabei werden beim Speichern der Kopie die WinCC-Anteile automatisch entfernt.

Anschließend migrieren Sie das STEP 7-Projekt ohne das WinCC-Projekt.

### Migration eines integrierten Projekts mit der Hardware-Konfiguration

Bei integrierten Projekten werden die HMI-Geräte auch dann migriert, wenn Sie die Hardware-Konfiguration nicht in die Migration einschließen. Der STEP 7-Anteil der Hardware-Konfiguration, einschließlich der Netzkonfigurationen, Verbindungen und Alarmer, wird nur dann migriert, wenn Sie die Hardware-Konfiguration in die Migration einschließen. Andernfalls werden für die STEP 7-Geräte unspezifizierte Baugruppen angelegt, die Sie nach der Migration wieder in eine passende Baugruppe umwandeln müssen.

HMI-Baugruppen, die in einer PC-Station gesteckt sind, werden bei der Migration zu einer eigenen Station umgewandelt. Wenn Sie die Migration unter Ausschluss der Hardware-Konfiguration durchführen, dann enthält das migrierte Projekt eine nicht spezifizierte SIMATIC PC-Station und eine SIMATIC PC-Station mit den HMI-Geräten. Bezüge zu HMI-Geräten werden bei der Migration nicht übernommen. Bei eingeschlossener Hardware-Konfiguration enthält das migrierte Projekt zwei separate Stationen, die HMI-Station und die PC-Station.



### Speicherort eines integrierten WinCC-Projekts

Wenn Sie ein integriertes Projekt migrieren, muss der darin enthaltene HMI-Anteil auf demselben PG/PC liegen, wie der STEP 7-Anteil des Projekts. Liegt der HMI-Anteil auf einem anderen PG, wird nur der STEP 7-Anteil migriert.

### Nicht unterstützte Objekte

Folgende Komponenten werden von der Migration nicht unterstützt:

- STEP 7-Multi-Projekt  
Ein STEP 7-Multi-Projekt kann nicht migriert werden. Die Migration bricht den Vorgang ab.
- Central Archive Server - CAS  
Wenn in einem integrierten Projekt ein CAS enthalten ist, dann wird die Migration zwar durchgeführt, die CAS-Daten werden jedoch nicht migriert.

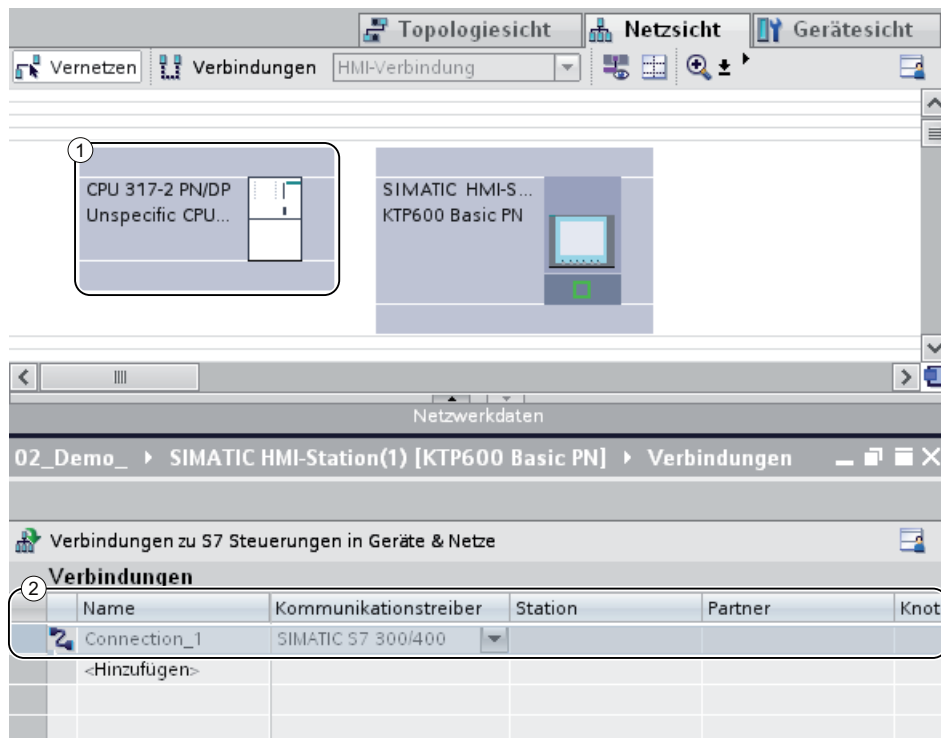
### Siehe auch

Integrierte Projekte nacharbeiten (Seite 147)

#### 4.1.7.2 Integrierte Projekte nacharbeiten

Wenn Sie ein integriertes Projekt ohne Hardware-Konfiguration migriert haben, dann werden unspezifizierte CPUs anstelle der CPUs des ursprünglichen Projekts verwendet. Da zwischen einer unspezifizierten CPU und einem HMI-Gerät keine Verbindung bestehen kann, werden Verbindungen aus dem Ausgangsprojekt ebenfalls nur unspezifiziert übernommen.

Das folgende Bild zeigt den Zustand nach einer Migration ohne Hardware-Konfiguration in einem Beispielprojekt:



- ① Die ursprüngliche CPU 317-2 PN/DP wurde bei der Migration durch eine unspezifizierte CPU ersetzt.
- ② Die Verbindung zwischen CPU und HMI-Gerät ist ebenfalls unspezifisch und muss erneuert werden.

## Vorgehen

Um ein integriertes Projekt nach der Migration weiter zu nutzen, gehen Sie folgendermaßen vor:

1. Wandeln Sie die unspezifizierten Geräte wieder in passende Geräte um.
2. Stellen Sie die integrierte HMI-Verbindung zwischen dem HMI-Gerät und der PLC wieder her.
3. Verbinden Sie alle HMI-Variablen mit der neu erstellten integrierten Verbindung.
4. Stellen Sie die Verbindung zwischen HMI-Variablen und PLC-Variablen wieder her.
5. Löschen Sie die nicht integrierte HMI-Verbindung.

Die einzelnen Schritte sind in den folgenden Kapiteln anhand eines Beispielprojekts genauer beschrieben.

## **Siehe auch**

Unspezifizierte CPUs in spezifizierte CPUs umwandeln (Seite 149)

Integrierte HMI-Verbindung erstellen (Seite 151)

HMI-Variablen neu verknüpfen (Seite 153)

Unspezifizierte Verbindung löschen (Seite 154)

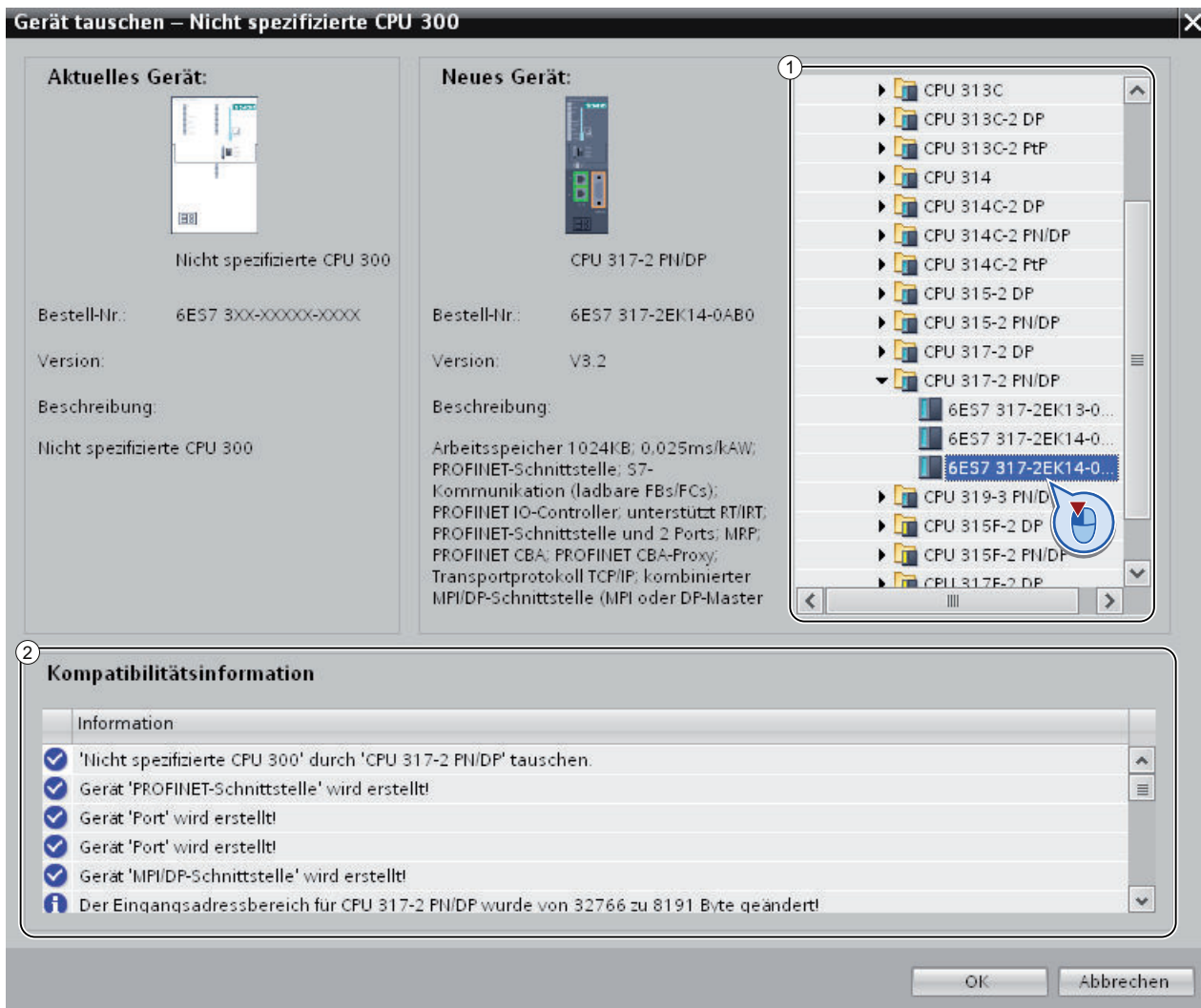
### **4.1.7.3 Unspezifizierte CPUs in spezifizierte CPUs umwandeln**

Der erste Schritt nach der Migration ohne Hardware-Konfiguration ist das Umwandeln der unspezifizierten CPUs in spezifische CPUs. Unspezifizierte CPUs sind als Platzhalter für bestimmte CPUs aus dem Hardware-Katalog zu verstehen, welche aktuell noch nicht bekannt sind. Sie können allgemeine Parameter festlegen und die CPUs bereits im Anwenderprogramm referenzieren. Das Projekt ist allerdings nicht vollständig funktionsfähig, solange die unspezifizierte CPU noch nicht spezifiziert wurde.

### Eine CPU über den Baugruppentausch spezifizieren

Um eine unspezifizierte CPU über den Baugruppentausch zu spezifizieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die unspezifizierte CPU in der Netz- oder Gerätesicht.
2. Wählen Sie im Kontextmenü den Befehl "Gerät tauschen".  
Der Dialog "Gerät tauschen" wird geöffnet.



3. Selektieren Sie in der Baumstruktur unter "Neues Gerät" die Baugruppe, durch die Sie die unspezifizierte CPU ersetzen möchten. (Bereich 1)  
Unter "Kompatibilitätsinformationen" erhalten Sie Hinweise, inwieweit die gewählte CPU mit der Konfiguration im Ausgangsprojekt kompatibel ist. (Bereich 2)
4. Klicken Sie auf "OK".
5. Führen Sie die vorangegangenen Schritte für alle unspezifizierten CPUs durch.

## Siehe auch

Integrierte HMI-Verbindung erstellen (Seite 151)

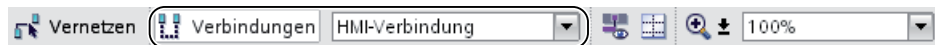
### 4.1.7.4 Integrierte HMI-Verbindung erstellen

Nachdem Sie die un spezifizierte CPU spezifiziert haben, stellen Sie die Verbindung zum HMI-Gerät her.

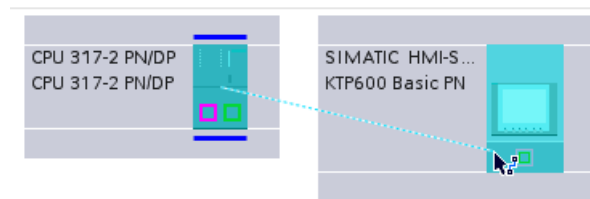
## Vorgehen

Um eine Verbindung grafisch anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Verbindungen". Sie aktivieren damit den Verbindungsmodus.

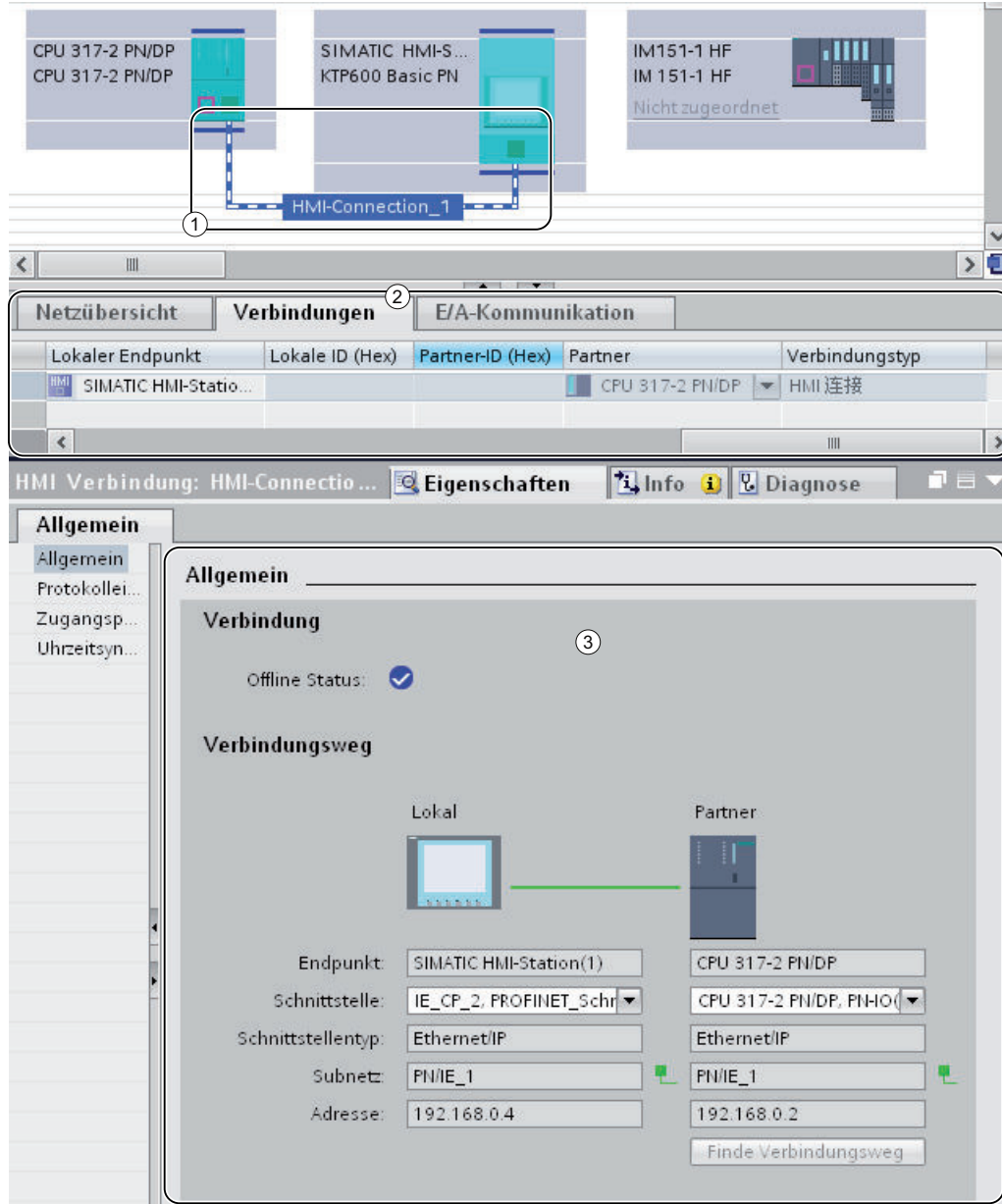


2. Wählen Sie den Verbindungstyp "HMI-Verbindung" in der danebenstehenden Klappliste. In der Netzansicht sind alle CPUs und HMI-Geräte, die für eine HMI-Verbindung in Frage kommen, farblich hervorgehoben.
3. Sie können nun den Verbindungsweg automatisch bestimmen lassen, oder explizit einen Verbindungsweg über bestimmte Schnittstellen wählen:
  - Verbindungsweg automatisch bestimmen lassen  
 Selektieren Sie die CPU, von der eine Verbindung ausgehen soll. Ziehen Sie die Maus auf die Zielkomponente. Bestätigen Sie den Verbindungsendpunkt durch erneuten Mausklick.  
 Alternativ: Selektieren Sie bei gedrückter Shift-Taste zusätzlich die Zielkomponente und wählen Sie mit der rechten Maustaste den Befehl "Neue Verbindung hinzufügen".
  - Expliziten Verbindungsweg von Schnittstelle zu Schnittstelle wählen  
 Klicken Sie auf die Subnetzschnittstelle in dem Gerät, von dem eine Verbindung ausgehen soll. Ziehen Sie anschließend den Mauszeiger bei gedrückter Maustaste auf die gewünschte Schnittstelle im Zielgerät und lassen Sie die Maustaste los.



### Ergebnis

Das folgende Bild zeigt den Zustand nach dem Anlegen der integrierten Verbindung:



- ① Eine integrierte HMI-Verbindung ist angelegt und wird in der Netzsicht hervorgehoben dargestellt.
- ② Die Verbindung wird in der Verbindungstabelle der Komponenten angezeigt.
- ③ Die Verbindung kann in den Verbindungseigenschaften bearbeitet werden.

## Siehe auch

HMI-Variablen neu verknüpfen (Seite 153)

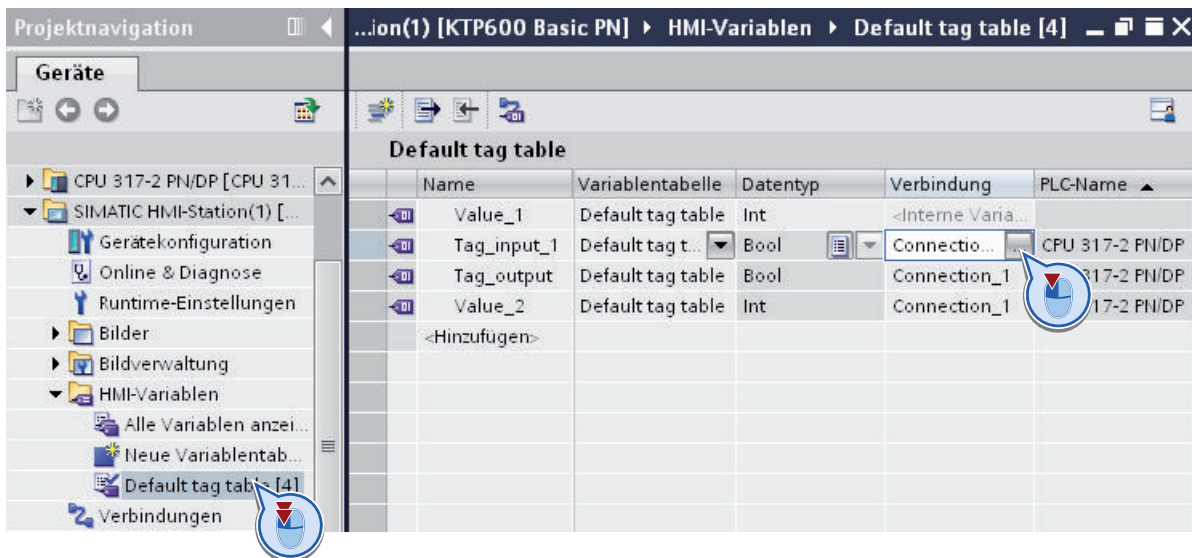
### 4.1.7.5 HMI-Variablen neu verknüpfen

Nachdem Sie eine neue HMI-Verbindung zwischen CPU und HMI-Gerät aufgebaut haben, müssen Sie der neuen Verbindung die bestehenden HMI-Variablen zuordnen. Führen Sie die folgenden Schritte für jede Zeile in der betroffenen Variablen-tabelle durch.

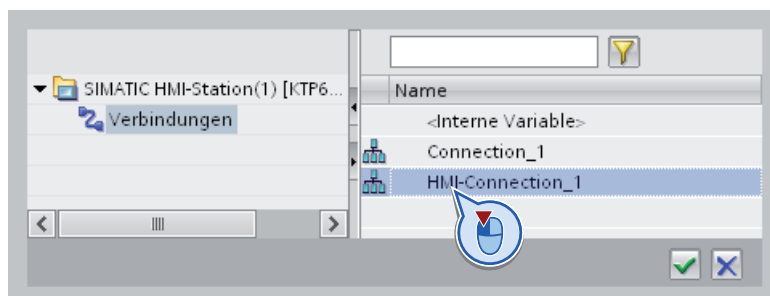
## Vorgehen

Um die HMI-Variablen neu zu verknüpfen, gehen Sie folgendermaßen vor:

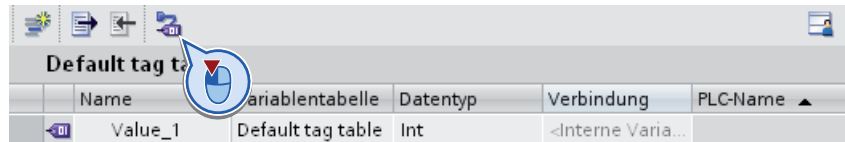
1. Navigieren Sie in der Projektnavigation zu den HMI-Variablen und doppelklicken Sie auf die entsprechende Variablen-tabelle, um diese im Arbeitsbereich anzuzeigen. Die Variablen-tabelle wird geöffnet.



2. Klicken Sie auf die Schaltfläche "..." in der Spalte "Verbindung". Ein Dialog zur Auswahl der Verbindung wird geöffnet.
3. Selektieren Sie die neu angelegte HMI-Verbindung.



4. Klicken Sie auf die Schaltfläche "✓", um die ausgewählte Verbindung zu übernehmen.
5. Klicken Sie in der Funktionsleiste auf die "Schaltfläche PLC-Variable neu verbinden."



**Siehe auch**

Unspezifizierte Verbindung löschen (Seite 154)

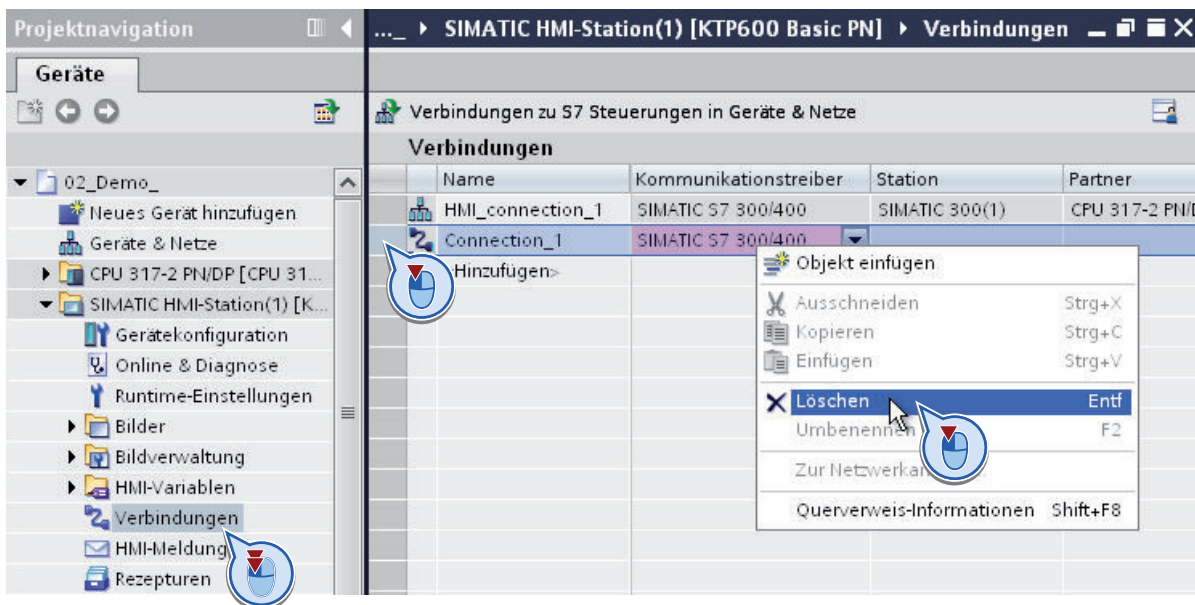
**4.1.7.6 Unspezifizierte Verbindung löschen**

Zuletzt können Sie unspezifizierte Verbindungen, die noch aus dem Ausgangsprojekt verblieben sind, entfernen.

**Vorgehen**

Um die unspezifiziert Verbindungen zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation das HMI-Gerät und doppelklicken Sie auf den Eintrag "Verbindungen".  
Die Verbindungstabelle wird geöffnet.



2. Selektieren Sie die Zeile mit der alten Verbindung in der Tabelle.
3. Wählen Sie den Befehl "Löschen" im Kontextmenü der Verbindungszeile.
4. Führen Sie die vorangegangenen Schritte für alle unspezifizierten Verbindungen des Ausgangsprojekts durch.



## 4.2 Programmierempfehlungen

### 4.2.1 Die neuen Funktionen der S7-1500-CPU im Überblick

#### Höhere Performance

Mit der S7-1500 steht Ihnen eine CPU-Familie zur Verfügung, die eine wesentlich höhere Performance bietet, als die CPUs der Baureihen S7-300/400. Bei der Programmierung mit STEP 7 V5.x waren Sie es wahrscheinlich gewohnt, mit Programmierweisen, wie z. B. der absoluten Adressierung, zu arbeiten, um eine höhere Performance der CPU und einen schlankeren Programmcode zu erreichen.

Aufgrund der hohen Performance, die die S7-1500 mit sich bringt, wurden diese Programmierweisen obsolet.

In den folgenden Absätzen möchten wir Ihnen einige neue Programmiermöglichkeiten der S7-1500 vorstellen.

#### Durchgängige Symbolik

Die S7-1500 bietet Ihnen die Möglichkeit, die Symbolik durchgängig im gesamten Projekt zu verwenden. Mit Hilfe der Autovervollständigung werden Sie beim Programmieren mit Symbolen innerhalb der Programmiereditoren kontextabhängig unterstützt. Die Datenelemente, z. B. innerhalb eines Datenbausteins, erhalten in der Deklaration einen symbolischen Namen, aber keine feste Adresse innerhalb des Datenbausteins. Dadurch können Sie beim Zugriff auf diese Datenelemente die hohe Performance der S7-1500 voll ausschöpfen. Die absoluten Adressen der Operanden müssen nicht mehr bekannt sein und Zugriffsfehler werden vermieden.

Ihr Programmcode wird durch die Symbolik verständlicher und Sie müssen weniger kommentieren. Bei einer Korrektur der Symbolik werden automatisch alle Verwendungsstellen aktualisiert.

Ein Beispiel zur Verwendung von durchgängiger Symbolik finden Sie unter: Symbolisch adressieren (Seite 157)

#### Optimierter Bausteinzugriff

Bei optimiertem Bausteinzugriff werden die deklarierten Datenelemente im verfügbaren Speicherbereich des Bausteins automatisch so angeordnet, dass seine Kapazität optimal ausgeschöpft wird. Die Daten werden auf eine Weise strukturiert und abgelegt, die für die verwendete CPU optimal ist. Die Ablage bleibt dem System überlassen. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, über den die Variable innerhalb

des Bausteins angesprochen werden kann. Dadurch steigern Sie die Performance der CPU. Zugriffsfehler, z. B. aus HMI heraus, sind nicht möglich.

| Bausteinzugriffe im Vergleich<br>Standard < > Optimierte   |  | <b>SIEMENS</b>  |
|--|--|---|
|  | Standard Bausteinzugriff<br>(S7-1200/1500 kompatibel<br>mit S7-300/400)  | Optimierter Bausteinzugriff<br>(Nur S7-1200/1500)   |
| Datenverwaltung  | Sie können die Variablen sowohl symbolisch (speicheroptimiert) als auch absolut (anwenderdefiniert) adressieren.   | Das System übernimmt die Verwaltung und Optimierung der Datenablage. Somit wird die Speicherkapazität optimal ausgeschöpft.         |
| Performance  | Der Zugriff auf eine CPU der Baureihen S7-1200/1500 erfolgt nicht immer schnellstmöglich, da die Datenablage aufgrund der absoluten Adressierung ineffektiv sein kann. | Der Zugriff erfolgt immer schnellstmöglich, da die Datenablage vom System optimiert wird und keine festen Adressen vergeben werden. |
| Fehleranfälligkeit   | Absolute Adressierung (z.B. aus HMI oder bei indirekter Adressierung) kann nach einer Änderung der festen Adresse zu Inkonsistenzen führen.                            | Zugriffsfehler, z.B. bei indirekter Adressierung oder aus HMI heraus, sind nicht möglich, da symbolisch zugegriffen wird.           |
| Remanenz   | Gültig für alle Variablen eines Datenbausteins   | Gültig für einzelne Variablen   |
| <p><b>Empfehlung:</b><br/>Um eine bestmögliche Performance zu erzielen, ist es empfehlenswert, die Bausteinzugriffsarten innerhalb Ihres Programms nicht zu mischen.</p> |  |   |

Weitere Informationen zu Bausteinen mit optimiertem Zugriff finden Sie unter "Siehe auch".

### Neue Datentypen

Die neuen Datentypen LWORD, LINT, ULINT, LTIME, LTOD, LDT und das Array (32-Bit-Grenze) bieten Ihnen eine wesentlich höhere Berechnungsgenauigkeit bei der Verwendung von mathematischen Funktionen. Im Bereich der impliziten und expliziten Datentypkonvertierung stehen Ihnen im Vergleich zu den CPUs der Baureihen S7-300/400 mehr Möglichkeiten zur Verfügung.

Weitere Informationen zu den neuen Datentypen finden Sie unter "Siehe auch".

### PLC-Datentypen

PLC-Datentypen (UDT) sind von Ihnen definierte Datenstrukturen, die Sie im Programm mehrmals verwenden können. Die Struktur eines PLC-Datentyps setzt sich aus mehreren Komponenten zusammen, die verschiedene Datentypen aufweisen können. Die Art der Komponenten legen Sie bei der Deklaration des PLC-Datentyps fest.

Den PLC-Datentyp können Sie als Basis-Datentyp für die Definition von Variablen und als Vorlage für die Erstellung von globalen Datenbausteinen verwenden. Wenn Sie nachträglich Änderungen am PLC-Datentyp vornehmen, werden diese Änderungen automatisch an allen Verwendungsstellen nachgezogen.

Sie können auch symbolisch auf einzelne Elemente eines Arrays innerhalb eines PLC-Datentyps zugreifen.

Weitere Informationen zu den neuen Datentypen finden Sie unter "Siehe auch".

Ein Beispiel zur Verwendung von PLC-Datentypen finden Sie unter: PLC-Datentypen (UDT) verwenden (Seite 166)

### Einheitliche Anweisungen in allen Programmiersprachen

Es steht Ihnen ein einheitlicher Satz von Anweisungen in allen Programmiersprachen (KOP, FUP, AWL, SCL und GRAPH) zur Verfügung.

### Slice-Zugriffe

Mit Hilfe der Slice-Zugriffe haben Sie die Möglichkeit, gezielt Bereiche innerhalb deklarierter Variablen zu adressieren. Den Zugriff auf ein einzelnes Bit können Sie bis zur Ebene der Variable symbolisch realisieren. Das einzelne Bit wird dann absolut angesprochen.

Weitere Informationen zu den Slice-Zugriffen finden Sie unter "Siehe auch".

### Indirekte Adressierung

Die indirekte Adressierung bietet die Möglichkeit, Operanden zu adressieren, deren Adresse erst zur Laufzeit berechnet wird. Innerhalb aller Programmiersprachen stehen Ihnen allgemeine Möglichkeiten, wie z. B. die indirekte Adressierung über POINTER, zur Verfügung. In der Programmiersprache SCL können Sie dazu auch die Anweisungen PEEK und POKE verwenden.

Weitere Informationen zur indirekten Adressierung finden Sie unter "Siehe auch".

### FAQs

Weitere Informationen zu den Programmierempfehlungen finden Sie unter FAQ ID: 67582299 (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=67582299&Datakey=47071380&caller=nl>)

## 4.2.2 Symbolisch adressieren

### Vorteile der symbolischen Adressierung

Mit Hilfe der durchgängigen Verwendung von aussagekräftigen Symbolen im gesamten Projekt wird der Programmcode verständlicher und leichter lesbar.

Dadurch haben Sie die folgenden Vorteile:

- Sie müssen keine umfassenden Kommentare mehr schreiben.
- Der Datenzugriff ist performanter.
- Es treten keine Fehler beim Datenzugriff auf.
- Sie müssen nicht mehr mit den absoluten Adressen arbeiten.
- Die Zuordnung des Symbols zur Speicheradresse wird von STEP 7 überwacht, d. h. wenn sich der Name oder die Adresse einer Variablen ändert, werden automatisch alle Verwendungsstellen aktualisiert.

### Programmierung in STEP 7 V5.x

In STEP 7 V5.x hatten Sie bereits die Möglichkeit, mit Hilfe von sprechenden Namen für Operanden und Bausteine, Ihr Programm aussagekräftiger zu schreiben. Dazu ordneten Sie die symbolischen Operanden den Speicheradressen und Bausteinen in der Symboltabelle zu. Damit sich eine Änderung in der Symbolik auch auf den Programmcode im Programmiereditor ausgewirkt hat, mussten Sie über die Eigenschaft "Operandenvorrang" festlegen, ob das Symbol oder der Absolutwert maßgeblich sein sollte.

Durch die symbolische Adressierung konnte die Übersichtlichkeit des Programms erhöht werden. In einigen Fällen, z. B. bei der Programmierung mit anwenderdefinierten Datentypen (UDT) konnte dies jedoch eine Beeinträchtigung der Performance nach sich ziehen.

Eine Steigerung der Performance erreichten Sie, indem Sie im UDT die Symbolik ignorierten und absolut adressierten. Dazu war es allerdings nötig, die Datenablage zu kennen. Änderungen am UDT wurden nicht automatisch nachgezogen. Mit Hilfe der absoluten Adressierung konnten Sie auch auf Teile einer Variablen zugreifen und diese bearbeiten. Der Nachteil bei einer ausschließlich absoluten Adressierung war allerdings, dass der Programmcode ab einem gewissen Umfang unübersichtlich wurde und Sie zur besseren Orientierung zusätzlich Kommentare einfügen mussten.

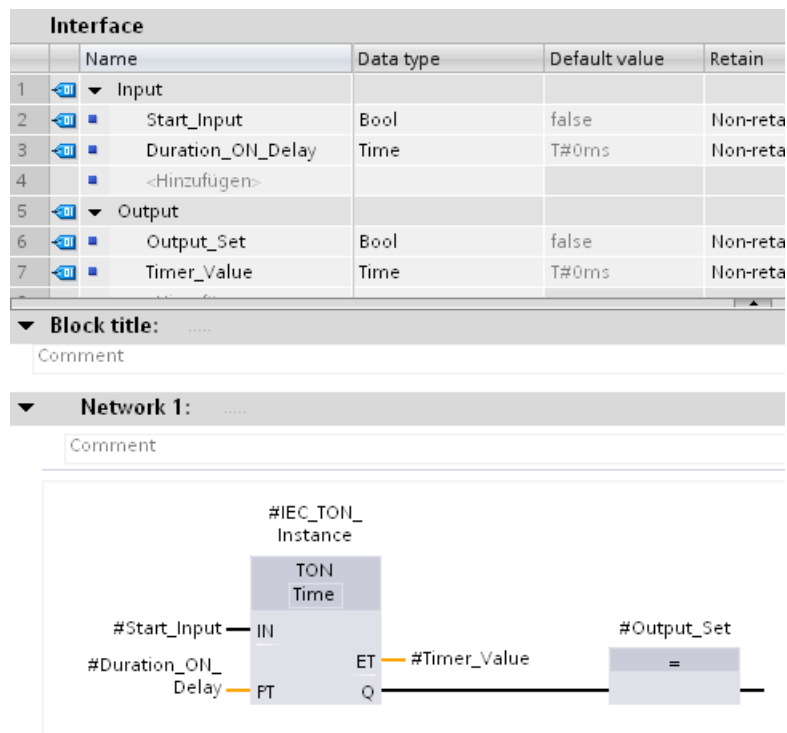
### Vorgehensweise in STEP 7 TIA-Portal

Die S7-1500-CPU bietet Ihnen eine wesentlich höhere Performance als die S7-300/400-CPU's. Um diese hohe Performance voll nutzen zu können, empfehlen wir Ihnen, für alle Bausteine den optimierten Bausteinzugriff zu aktivieren und im Programmcode symbolisch zu adressieren.

Der Programmiereditor unterstützt Sie beim Arbeiten mit Symbolen durch kontextabhängige Eingabehilfen, wie z. B. der Autovervollständigung. Mit ihrer Hilfe können Sie während des Programmierens bequem auf vorhandene Variablen oder Anweisungen zugreifen.

## Programmierbeispiel

Das folgende Beispiel zeigt, wie Sie symbolisch auf einzelne Elemente zugreifen können:



Die Variablennamen, die Sie in der Bausteinschnittstelle definiert haben, können Sie direkt an den Parametern der Anweisung TON verwenden, ohne die absolute Adresse der Variablen zu kennen.

### 4.2.3 IEC-Zeiten und -Zähler verwenden

#### Vorteile der IEC-Zeiten und -Zähler

Mit Hilfe der durchgängigen Verwendung von IEC-Zeiten und -Zählern wird Ihr Programmcode effektiver.

Dadurch haben Sie die folgenden Vorteile:

- Die Bausteine können mehrfach mit neu erzeugten Instanzdatenbausteinen aufgerufen werden.
- Die IEC-Zähler verfügen über einen großen Zählbereich.
- Die IEC-Zeiten verfügen gegenüber den S5-Zeiten über eine bessere Performance und die Zeitgenauigkeit ist höher.

### Programmierung in STEP 7 V5.x

Die S5-Zeiten und -Zähler in STEP 7 V5.x wurden über eine Nummer absolut adressiert. Durch diese Nummernabhängigkeit waren Programmbausteine mit S5-Zeiten und -Zählern nicht mehrfach verwendbar.

Der Wertebereich einer Zeit war auf maximal 9990s und der eines Zählers auf einen maximalen Grenzwert von 999 begrenzt.

### Vorgehensweise in STEP 7 TIA-Portal

Deklarieren Sie die IEC-Zeiten und -Zähler in dem Programmbaustein, in dem sie aufgerufen bzw. gebraucht werden. Die IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, oder z. B. TON\_TIME und TON\_LTIME, die Sie auch als lokale Variable in einem Baustein deklarieren können. Der IEC-Zähler ist eine Struktur vom Datentyp IEC\_SCOUNTER, IEC\_USCOUNTER, etc.

### Programmbeispiel im TIA-Portal

Das folgende Beispiel zeigt, wie Sie eine IEC-Zeit und einen IEC-Zähler als lokale Variable deklarieren können:

| Interface |               |           |               |            |                                     |                          |
|-----------|---------------|-----------|---------------|------------|-------------------------------------|--------------------------|
|           | Name          | Data type | Default value | Retain     | Visible in ...                      | Setpoint                 |
| 1         | ▶ Input       |           |               |            | <input type="checkbox"/>            | <input type="checkbox"/> |
| 2         | ▶ Output      |           |               |            | <input type="checkbox"/>            | <input type="checkbox"/> |
| 3         | ▶ InOut       |           |               |            | <input type="checkbox"/>            | <input type="checkbox"/> |
| 4         | ▼ Static      |           |               |            | <input type="checkbox"/>            | <input type="checkbox"/> |
| 5         | ▶ SwitchDelay | TON_TIME  |               | Non-ret... | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 6         | ▶ CountDB     | CTU_INT   |               | Non-retain | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

▶ Block title: .....

▼ Network 1: .....

Comment

Die Daten der IEC-Zeit TON und des IEC-Zählers CTU werden als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt.

## 4.2.4 Freigabeausgang ENO flexibel einsetzen

### Vorteile

Bei einzelnen Anweisungen und Bausteinaufrufen haben Sie die Möglichkeit, mit Hilfe des EN-/ENO-Mechanismus Laufzeitfehler zu erkennen und einen Programmabbruch zu vermeiden. Bei mathematischen Funktionen werden z. B. Überläufe über den Freigabeausgang ENO gemeldet.

In STEP 7 TIA-Portal ist in den Programmiersprachen KOP und FUP der Freigabeausgang ENO per Voreinstellung deaktiviert. Bei Bedarf können Sie den Freigabeausgang aktivieren und damit gezielt steuern, bei welchen Anweisungen Sie eine Fehlerauswertung haben möchten.

Dadurch haben Sie die folgenden Vorteile:

- Bei deaktiviertem ENO erhöht sich die Performance.
- Bei aktiviertem ENO führen Laufzeitfehler nicht zum STOP der CPU.

### Vorgehensweise in STEP 7 TIA-Portal

Um den EN-/ENO-Mechanismus einer Anweisung zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus aktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO generieren".  
Für die Anweisung wird der ENO-Wert generiert. Weitere Anweisungen werden mit dem Freigabeausgang eingefügt.

In den nachfolgenden Tabellen sind die Anweisungen mit deaktivierbarem Freigabeausgang ENO aufgeführt:

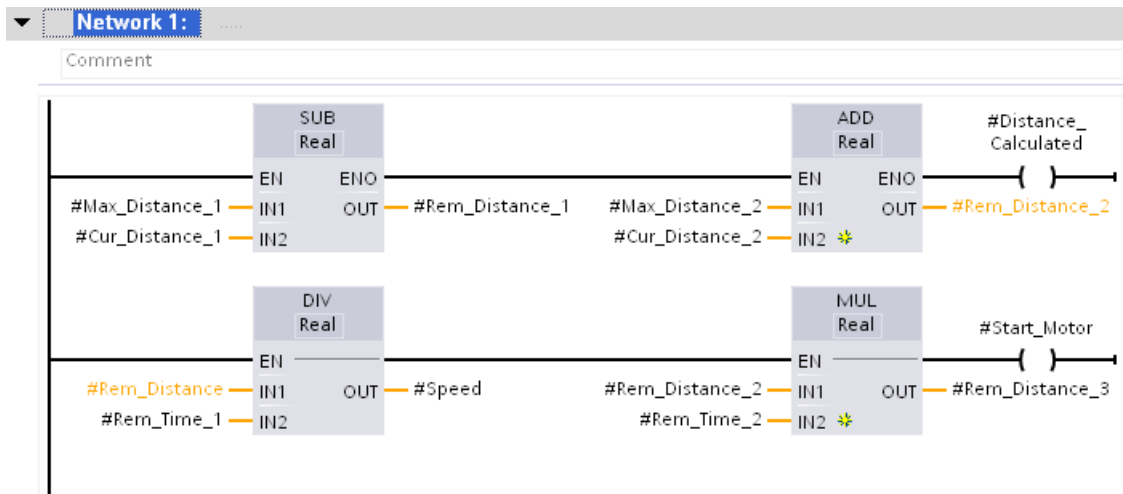
| Einfache Anweisungen     |  |
|--------------------------|--|
| Mathematische Funktionen | ADD, SUB, MUL, DIV, MOD, INC, DEC, ABS, NEG, SQR, SQRT, LN, EXP, SIN, COS, TAN, ASIN, ACOS, ATAN, FRAC, EXPT, MIN, MAX, LIMIT, CALCULATE |
| Verschieben              | MOVE, SWAP, MOVE_BLK, UMOVE_BLK, FILL_BLK, UFILL_BLK, MOVE_BLK_VARIANT   |
| Umwandler                | CONVERT, ROUND, CEIL, TRUNC, FLOOR, NORM_X, SCALE_X  |
| Wortverknüpfungen        | AND, OR, XOR, INV, DECO, ENCO  |
| Schieben und Rotieren    | SHR, SHL, ROR, ROL   |

| Erweiterte Anweisungen |  |
|------------------------|--|
| String + Char          | CONCAT, LEFT, RIGHT, MID, DELETE, INSERT, REPLACE, FIND, LEN, S_CONV |
| Datum und Uhrzeit      | T_CONV   |

Weitere Informationen zum EN-/ENO-Mechanismus in den einzelnen Programmiersprachen finden Sie unter "Siehe auch".

### Programmierbeispiel

Das folgende Beispiel zeigt, wie Sie Anweisungen mit aktiviertem und deaktiviertem Freigabeausgang ENO verwenden können:



Wenn Sie den Freigabeausgang ENO, wie bei der Anweisung SUB, aktiviert haben, werden alle nachfolgenden Anweisungen ebenfalls mit aktiviertem Freigabeausgang angelegt. Wenn es dann zu einem Arithmetikfehler während der Bearbeitung der Anweisung SUB kommt, wird die Anweisung ADD nicht ausgeführt.

Bei der Anweisung DIV im zweiten Zweig ist der Freigabeausgang ENO deaktiviert. Wenn es während der Bearbeitung zu einem Laufzeitfehler kommt, dann wird die Anweisung MUL trotzdem ausgeführt.

### 4.2.5 MOVE-Anweisungen in AWL verwenden

#### Verwendungsmöglichkeiten

Auf einer S7-1500-CPU haben Sie jetzt auch in AWL die Möglichkeit, mit MOVE-Anweisungen zu programmieren.

Dadurch haben Sie die folgenden Vorteile:

- Es ist ein einfacher Programmaufbau möglich.
- Die Performance der CPU steigt.

#### Programmierung in STEP 7 V5.x

In STEP 7 V5.x haben Sie zur Realisierung der MOVE-Funktionalitäten die Systemfunktionen "BLKMOV: Bereich kopieren" und "UBLKMOV: Bereich ununterbrechbar kopieren" verwendet.



### Vorgehensweise in STEP 7 TIA-Portal

In STEP 7 TIA-Portal stehen Ihnen die folgenden neuen MOVE-Anweisungen zur Verfügung:

- MOVE: Wert kopieren
- MOVE\_BLK: Bereich kopieren
- MOVE\_BLK\_VARIANT: Bereich kopieren
- UMOVE\_BLK: Bereich ununterbrechbar kopieren

Weitere Informationen zu den neuen MOVE-Anweisungen finden Sie unter "Siehe auch".

### Programmierbeispiel

Das folgende Beispiel zeigt die Verwendung der Anweisung "MOVE\_BLK: Bereich kopieren". Dazu wird ein Array-Bereich in einen anderen Array-Bereich kopiert:

| Data_DB |             |                      |
|---------|-------------|----------------------|
|         | Name        | Data type            |
| 1       | Static      |                      |
| 2       | Array_1     | Array [0..10] of Int |
| 3       | Array_1[0]  | Int                  |
| 4       | Array_1[1]  | Int                  |
| 5       | Array_1[2]  | Int                  |
| 6       | Array_1[3]  | Int                  |
| 7       | Array_1[4]  | Int                  |
| 8       | Array_1[5]  | Int                  |
| 9       | Array_1[6]  | Int                  |
| 10      | Array_1[7]  | Int                  |
| 11      | Array_1[8]  | Int                  |
| 12      | Array_1[9]  | Int                  |
| 13      | Array_1[10] | Int                  |
| 14      | Array_2     | Array [0..10] of Int |
| 15      | Array_2[0]  | Int                  |
| 16      | Array_2[1]  | Int                  |
| 17      | Array_2[2]  | Int                  |
| 18      | Array_2[3]  | Int                  |
| 19      | Array_2[4]  | Int                  |
| 20      | Array_2[5]  | Int                  |
| 21      | Array_2[6]  | Int                  |
| 22      | Array_2[7]  | Int                  |
| 23      | Array_2[8]  | Int                  |
| 24      | Array_2[9]  | Int                  |
| 25      | Array_2[10] | Int                  |

```
▼ Network 1: .....  
Comment  
1 | CALL MOVE_BLK  
2 | Int Uint  
3 | IN := "Data_DB".Array_1[0]  
4 | COUNT := 10  
5 | OUT := "Data_DB".Array_2[0]  
6 |  
7 |
```

Mit Hilfe der Anweisung MOVE\_BLK werden 10 Elemente aus dem "Array\_1" des Datenbausteins "Data\_DB" in das "Array\_2" des gleichen Datenbausteins kopiert.

### 4.2.6 Array-Zugriff mit variablem Index realisieren

#### Vorteile des variablen Index

Zur Adressierung der Komponenten eines Arrays können Sie als Index neben Konstanten auch Variablen vom Datentyp Ganzzahl angeben. Dabei sind Ganzzahlen mit einer Länge bis 32 Bit zulässig. Bei Verwendung von Variablen wird der Index zur Laufzeit berechnet. So können Sie z. B. in Programmschleifen bei jedem Durchlauf einen anderen Index verwenden. Sie können auch auf ein Array innerhalb eines PLC-Datentyps zugreifen.

Dadurch haben Sie die folgenden Vorteile:

- Keine Adressierung mit Adressregistern bzw. selbst aufgebaute POINTER nötig.
- Mehr Flexibilität innerhalb Ihres Programms.
- Der variable Index steht Ihnen in allen STEP 7-Programmiersprachen zur Verfügung.
- Es werden die vorhandenen Namen der Datenbausteine und Array-Variablen verwendet. Dadurch erhöht sich die Lesbarkeit des Programmcodes.
- Die Basisadresse des Arrays muss nicht bekannt sein.
- Der Programmcode ist einfacher zu erstellen.
- Der Compiler erzeugt optimierten Programmcode.

#### Vorgehensweise in STEP 7 V5.x

In STEP 7 V5.x mussten Sie zur indizierten Adressierung von Array-Elementen Adressregister mittels selbst aufgebaute POINTER verwenden. Die Programmiersprache SCL unterstützte bereits die indirekte Adressierung mit variablem Index.

### Programmierbeispiel in STEP 7 V5.x

Für das folgende AWL-Beispiel wird als Voraussetzung der Datenbaustein "Data\_classic" benötigt. Um ein Element des Arrays "Quantities" anzusprechen, müssen folgende Befehle verwendet werden:

| AWL                 | Erläuterung  |
|---------------------|--|
| AUF "Data_classic"  | // Der Datenbaustein "Data_classic" wird aufgerufen.   |
| L #index            | // Der Wert der lokalen Variablen #index wird in den Akkumulator 1 geladen.  |
| SLD 3               | // Bits 0 bis 31 des Akkumulators 1 um 3 Stellen nach links schieben.<br>// Freigewordene Bitstellen mit Nullen auffüllen. |
| LAR1                | // Adressregister 1 mit dem Inhalt des Akkumulators 1 laden.   |
| L DBW [AR1, P#10.0] | // Das mittels #index adressierte Array-Element in den Akkumulator 1 laden.<br>// P#10.0 = Basisadresse des Feldes         |

Bei dieser Vorgehensweise müssen Sie allerdings Folgendes berücksichtigen:

- Der Array-Name wird nicht verwendet. Dadurch sinkt die Lesbarkeit des Programmcodes und macht eine zusätzliche Kommentierung erforderlich.
- Die Basisadresse des Arrays (P#10.0) muss zur Adressierung bekannt sein.
- Die indizierte Adressierung muss unter Zuhilfenahme der Adressregister erfolgen.

### Programmierbeispiel in STEP 7 TIA-Portal

Für das indirekte Indizieren des Arrays "Quantities", das im Datenbaustein "Data\_DB" deklariert wurde, wird die folgende Syntax verwendet:

|                                 |                                      |
|---------------------------------|--------------------------------------|
| "Data_DB".Quantities["i"]       | // Eindimensionales Array            |
| "Data_DB".Quantities["i"].a     | // Eindimensionales Array of STRUCT  |
| "Data_DB".Quantities["i","j"]   | // Mehrdimensionales Array           |
| "Data_DB".Quantities["i","j"].a | // Mehrdimensionales Array of STRUCT |

| Teil       | Beschreibung   |
|------------|--|
| Data_DB    | Name des Datenbausteins, in dem sich das Array befindet              |
| Quantities | Variable vom Datentyp Array  |
| i, j       | PLC-Variablen vom Datentyp Ganzzahl, die als Zeiger verwendet werden |
| a          | Weitere Teilvariable der Struktur                                    |

Das folgende Beispiel zeigt das indirekte Indizieren einer Array-Komponente in AWL.

Für die Adressierung eines Array-Elements benötigen Sie nur noch eine Programmzeile:

| Adressierung in AWL            | Erläuterung   |
|--------------------------------|---|
| L "Data_DB".Quantities[#index] | // Der Wert des Array-Elements #index wird direkt aus dem Datenbaustein in den Akkumulator 1 geladen. |

Um eine bestmögliche Performance zu erzielen, beachten Sie folgende Hinweise:

- Variablen, die als Array-Index genutzt werden, deklarieren Sie als Ganzzahl kleiner gleich 32 Bit.
- Zwischenergebnisse und Array-Indizes legen Sie im temporären Lokaldatenbereich an.

### 4.2.7 PLC-Datentypen (UDT) verwenden

#### Vorteile der PLC-Datentypen

PLC-Datentypen (UDT) sind von Ihnen definierte Datenstrukturen, die Sie im Programm mehrmals verwenden können. Die Struktur eines PLC-Datentyps setzt sich aus mehreren Komponenten zusammen, die verschiedene Datentypen aufweisen können. Die Art der Komponenten legen Sie bei der Deklaration des PLC-Datentyps fest. Durch die Verwendung von PLC-Datentypen nutzen Sie die hohe Performance der S7-1500-CPU optimal aus.

In vielen Programmen werden zusammenhängende Datensätze benötigt, die von verschiedenen Programmstellen bearbeitet werden. Sie werden im laufenden Programm optimiert bzw. getauscht. Dabei handelt es sich z. B. um:

- Datensätze zur Materialverfolgung
- einen Parametersatz für eine Motoreinstellung
- verschiedene Rezepte

Dadurch haben Sie die folgenden Vorteile:

- Elemente eines PLC-Datentyps können auch indirekt adressiert werden, d.h. die Adresse ist variabel und wird erst zur Laufzeit berechnet.
- Variablen, die auf einem PLC-Datentyp basieren, erben alle Eigenschaften des PLC-Datentyps.
- Die Modifikation des PLC-Datentyps hat eine automatische Anpassung aller abgeleiteten Variablen zur Folge.
- Mithilfe der durchgängigen Symbolik wird das Programm leichter lesbar, da die Namen der einzelnen Elemente eines PLC-Datentyps im Programm angezeigt werden.
- Erhöhte Performance, da optimierter Programmcode erzeugt wird.
- Beim Bausteinaufruf kann der PLC-Datentyp als ganze Struktur übergeben werden.
- Vereinfachte Aufrufschnittstelle durch geringere zu versorgende Parameteranzahl.

### Vorgehensweise in STEP 7 V5.x

In STEP 7 V5.x hatten Sie bereits die Möglichkeit, mit Hilfe des Datentyps STRUCT oder eines PLC-Datentyps (UDT) einen Datensatz als strukturierte Variable anzulegen. Allerdings wurde die Performance durch die Nutzung der symbolischen Adressierung beeinträchtigt.

Die Deklaration in den Datenbausteinen wurde meistens als anonyme Struktur realisiert. Die Bausteine selbst wurden dann so programmiert, dass als Aktualparameter die Werte der Struktur übergeben und die berechneten Werte wieder in die Struktur kopiert wurden. So konnten Sie auch die Datenbausteinnummer übergeben und im Baustein absolut adressieren. Dabei war die Anzahl der Parameter, die Sie versorgen mussten, häufig sehr groß. Die Aktualdaten wurden in den Datenbausteinen abgelegt und die berechneten Werte an weitere Bausteine übergeben. Bei der Übergabe der Datenbausteinvariablen war allerdings keine Symbolik mehr verfügbar.

### Vorgehensweise in STEP 7 TIA-Portal

Wenn Sie in Ihrem Programm mit PLC-Datentypen arbeiten, dann können Sie diese sowohl einem Formal- als auch einem Aktualparameter zuweisen. Dadurch müssen Sie keine einzelnen Parameter mehr deklarieren. Wenn ein Baustein einen Eingangsparameter hat, der auf einem PLC-Datentyp basiert, müssen Sie eine Variable als Aktualparameter übergeben, die den gleichen PLC-Datentyp hat.

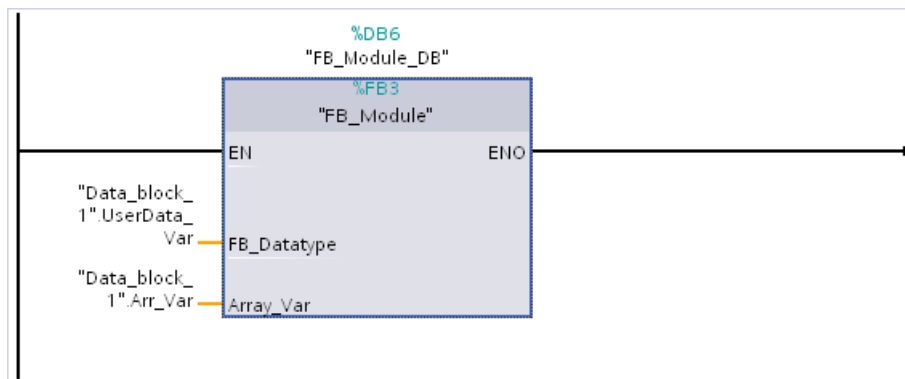
Um einen PLC-Datentyp zu verwenden, gehen Sie folgendermaßen vor:

1. Deklarieren Sie die Datensätze als PLC-Datentyp mit einem passenden Namen, z. B. "Materialverfolgung".
2. Verwenden Sie diesen Datentyp zur Deklaration eines Datenbausteins oder einer DB-Variablen.
3. Definieren Sie den Formalparameter als VAR\_IN\_OUT-/VAR\_IN- oder VAR\_OUT-Variable in der Bausteinschnittstelle.
4. Weisen Sie dem Formalparameter in der Spalte "Data type" den PLC-Datentyp zu.

### Programmierbeispiel im TIA-Portal

Das folgende Beispiel zeigt den Aufruf und die Parametrierung eines Funktionsbausteins (FB) mit zwei Formalparametern. Der Datentyp der DB-Variablen (Array [1 .. 10] of BOOL) und der PLC-Datentyp (UDT) sind identisch mit den Datentypen der FB-Parameter.

| Data_block_1 |              |                       |             |
|--------------|--------------|-----------------------|-------------|
|              | Name         | Data type             | Start value |
| 1            | Static       |                       |             |
| 2            | Arr_Var      | Array [1..10] of Bool |             |
| 3            | UserData_Var | "UserDataType"        |             |
| 4            | Var1         | Bool                  | false       |
| 5            | Var2         | Bool                  | false       |
| 6            | ByteVar1     | Byte                  | 16#0        |
| 7            | WordVar1     | Word                  | 16#0        |
| 8            | IntVar1      | Int                   | 0           |
| 9            | RealVar1     | Real                  | 0.0         |



Den PLC-Datentyp "UserData\_Type" deklarieren Sie in der Projektnavigation im Ordner "PLC-Datentypen". Anschließend verwenden Sie ihn als Datentyp für die Variable "UserData\_Var". Die beiden Variablen "UserData\_Var" und "Arr\_Var" deklarieren Sie im Datenbaustein "Data\_block\_1" und verwenden sie zur Versorgung der Parameter des Bausteins "FB\_Module".

# Erste Schritte

## 5.1 Getting Started Dokumentationen

### Einstieg in das TIA-Portal

Zum Einstieg in das TIA-Portal stehen Ihnen Getting Started Dokumentation zur Verfügung.

Die Getting Started Dokumentationen sind Anleitungen, die Schritt-für-Schritt beschreiben, wie Sie ein Projekt im TIA-Portal erstellen und bieten Ihnen die Möglichkeit, sich schnell mit dem Leistungsumfang des TIA-Portals vertraut zu machen.

### Inhalte

Die Getting Started Dokumentationen beschreiben die Erstellung eines durchgehenden Projekts für STEP7 und WinCC, das mit jedem Kapitel erweitert wird. Ausgehend von der Verwendung einfacher Basisfunktionen, nutzen Sie bei der Erstellung immer komplexere Funktionen.

Neben den Schritt-für-Schritt Anleitungen bieten die Getting Started Dokumentationen zu jedem neuen Thema auch kurze Hintergrundinformationen, welche die genutzten Funktion näher erläutern und Zusammenhänge verdeutlichen.

### Zielgruppe

Die Getting Started Dokumentationen richten sich in an Einsteiger, eignen sich aber auch für Umsteiger vorheriger Versionen von SIMATIC STEP 7 und WinCC.

### Download

Die Dokumentationen stehen ihnen kostenlos im Service&Support (<https://support.automation.siemens.com>) Portal als PDF zur Verfügung.

Über die folgenden Links können Sie die Dokumente herunterladen:

- STEP 7 Basic und WinCC Basic (V10.5) (<http://support.automation.siemens.com/WWW/view/de/40263542>)
- STEP 7 Professional und WinCC Advanced (V11) (<http://support.automation.siemens.com/WWW/view/de/28919804/133300>)
- STEP 7 Professional und WinCC Advanced (V12) ([http://www.automation.siemens.com/salesmaterial-as/interactive-manuals/getting-started\\_simatic-s7-1500/\\_content/DE/content\\_de.html](http://www.automation.siemens.com/salesmaterial-as/interactive-manuals/getting-started_simatic-s7-1500/_content/DE/content_de.html))





## Einführung in das TIA-Portal

### 6.1 Bedienoberfläche und Bedienung

#### 6.1.1 TIA-Portal starten, einstellen und beenden

##### 6.1.1.1 TIA-Portal starten und beenden

###### TIA-Portal starten

Um das TIA-Portal zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie unter Windows den Befehl "Start > Programme > Siemens Automation > TIA Portal V12".  
Das TIA-Portal wird mit den zuletzt verwendeten Einstellungen geöffnet.

###### TIA-Portal beenden

Um das TIA-Portal zu beenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Beenden".  
Wenn das Projekt nicht gespeicherte Änderungen enthält, werden Sie gefragt, ob die Änderungen gespeichert werden sollen.
  - Wählen Sie "Ja", um die Änderungen im aktuellen Projekt zu speichern und das TIA-Portal zu schließen.
  - Wählen Sie "Nein", um das TIA-Portal zu schließen, ohne die letzten Änderungen im Projekt zu speichern.
  - Wählen Sie "Abbrechen", um das Beenden abubrechen. In diesem Fall bleibt das TIA-Portal geöffnet.

##### 6.1.1.2 Übersicht der Programmeinstellungen

###### Überblick

Die folgende Tabelle zeigt die Anwendungseinstellungen, die Sie vornehmen können:

| Gruppe                   | Einstellung        | Beschreibung  |
|--------------------------|--------------------|---|
| Allgemeine Einstellungen | Benutzername       | Der Benutzername des Anwenders. Der Benutzername wird beim Anlegen eines neuen Projekts in den Projekteigenschaften hinterlegt. |
|                          | Oberflächensprache | Sprache für die Programmoberfläche  |

| Gruppe                               | Einstellung  | Beschreibung   |
|--------------------------------------|--|--|
|                                      | Mnemonik   | Legt die Mnemonik für die Programmierung fest:<br>"Deutsch" verwendet die deutsche Mnemonik, z. B. "E1.0"<br>"International" verwendet die internationale Mnemonik, z. B. "I1.0"<br>Die Unterschiede in der Mnemonik der einzelnen Befehle entnehmen Sie den Sprachbeschreibungen der jeweiligen Programmiersprache. |
|                                      | Liste zuletzt verwendeter Projekte anzeigen              | Anzahl der Einträge in der Liste der zuletzt verwendeten Projekte im Menü "Projekt"  |
|                                      | Zuletzt geöffnetes Projekt beim Start laden              | Das zuletzt geöffnete Projekt wird nach dem Start des TIA-Portals automatisch geöffnet.  |
|                                      | Abgeschnittene Texte komplett anzeigen                   | Texte, die aufgrund ihrer Länge abgeschnitten werden, werden in einem Tooltip angezeigt.   |
|                                      | Tooltips anzeigen (Kontextsensitive Hilfe ist verfügbar) | Tooltips werden angezeigt und Sie erhalten kontextbezogene Hilfe. Ist die Funktion deaktiviert, können Sie den Tooltip über <F1> öffnen.   |
|                                      | Kaskade in den Tooltips automatisch öffnen               | Die Tooltips klappen nach kurzer Zeit automatisch auf und eine Kaskade mit weiterer Hilfe wird angezeigt. Ist die Option deaktiviert, müssen die Tooltips manuell aufgeklappt werden.  |
| Auf Standard zurücksetzen            | Alle Anwendungseinstellungen                             | Alle Änderungen, die Sie im TIA-Portal nach der Installation durchgeführt haben, werden rückgängig gemacht.  |
|                                      | Layout der Editoren                                      | Setzt das komplette Layout der Anwendung auf den Auslieferungszustand zurück.  |
|                                      | Alle Meldefenster anzeigen                               | Alle Meldefenster, deren Erscheinen manuell unterdrückt wurde, werden wieder angezeigt.  |
| Startansicht                         | Zuletzt verwendete Ansicht                               | Startet das Programm immer in der zuletzt verwendeten Ansicht. Dies kann entweder die Portalansicht oder die Projektansicht sein.  |
|                                      | Portalansicht  | Startet das TIA-Portal immer in der Portalansicht, unabhängig davon, in welcher Ansicht zuletzt gearbeitet wurde.  |
|                                      | Projektansicht   | Startet das TIA-Portal immer in der Projektansicht, unabhängig davon, in welcher Ansicht zuletzt gearbeitet wurde.   |
| Ansicht für Objekte in der Übersicht | Details  | Wenn mehrere Ansichten zur Verfügung stehen, dann wird standardmäßig die Detailansicht gestartet, z. B. im Übersichtsfenster.  |
|                                      | Liste  | Wenn mehrere Ansichten zur Verfügung stehen, dann wird standardmäßig die Listenansicht gestartet, z. B. im Übersichtsfenster.  |
|                                      | Symbole  | Wenn mehrere Ansichten zur Verfügung stehen, dann wird standardmäßig die Symbolansicht gestartet, z. B. im Übersichtsfenster.  |
| Speichereinstellungen                | Zuletzt verwendeten Speicherort verwenden                | Beim ersten Speichern eines Projekts ist der zuletzt verwendete Pfad zum Speichern voreingestellt.   |
|                                      | Standardeinstellung für Speicherort festlegen            | Ermöglicht das Festlegen von Speicherpfaden für: <ul style="list-style-type: none"> <li>• Projekte</li> <li>• Bibliotheken</li> </ul>  |
| Datenaustausch                       | Speicherort für Datenimport                              | An diesem Speicherpfad wird standardmäßig nach Importdateien gesucht.  |
|                                      | Speicherort für Datenexport                              | Dieser Speicherpfad ist für den Datenexport voreingestellt.  |

| Gruppe | Einstellung                      | Beschreibung   |
|--------|----------------------------------|--|
|        | Speicherort für Support Packages | Nach dem Laden von Support Packages werden diese im angegebenen Speicherpfad abgelegt und können dann von dort aus installiert werden. |
|        | Speicherort für Protokolldateien | Protokolldateien werden am hier angegebenen Ort abgelegt.  |

## Siehe auch

TIA-Portal starten und beenden (Seite 171)

Bedienoberflächenlayout zurücksetzen (Seite 206)

Einstellungen ändern (Seite 175)

Anzeige von Tooltips und Tooltip-Kaskaden konfigurieren (Seite 227)

### 6.1.1.3 Übersicht der Skript- und Texteditor-Einstellungen

#### Überblick

Die folgende Tabelle zeigt die Einstellungen für Skript- und Texteditoren, die Sie vornehmen können:

| Gruppe        | Einstellung                 | Beschreibung  |
|---------------|-----------------------------|---|
| Schriftart    | Schriftart und Schriftgröße | Stellt die Schriftart und Schriftgröße für den Text in Texteditoren ein.  |
| Schriftfarben | Farbeeinstellungen          | Sie können die Farben für einzelne Textelemente in den Texteditoren aus den jeweiligen Klapplisten wählen. Es stehen Einstellmöglichkeiten für die folgenden Textelemente zur Verfügung: <ul style="list-style-type: none"> <li>• Text</li> <li>• Schlüsselwörter</li> <li>• Kommentare</li> <li>• Übersetzbare Kommentare</li> <li>• Anweisungen</li> <li>• Skripte</li> <li>• Standardfunktionen</li> <li>• Systemfunktionen</li> <li>• String-Konstanten</li> <li>• Symbolische Konstanten</li> <li>• Variablen</li> <li>• Objektmodelle</li> <li>• Formalparameter</li> </ul> |
|               | Auf Standard zurücksetzen   | Setzt alle Schriftfarben in Editoren auf Werkseinstellungen zurück.   |
| Tabulatoren   | Tabulatorweite              | Stellt die Weite von Tabulatoren ein.   |
|               | Tabulatoren verwenden       | Aktiviert die Verwendung von Tabulatoren.   |

| Gruppe                            | Einstellung                     | Beschreibung  |
|-----------------------------------|---------------------------------|---|
|                                   | Leerzeichen verwenden           | Verwendet anstatt von Tabulatoren Leerzeichen.  |
| Einrückung                        | Einzug zu Beginn eines Absatzes | Legt fest, ob zu Beginn eines neuen Absatzes ein Einzug geschaltet werden soll. Folgende Auswahlmöglichkeiten stehen zur Verfügung: <ul style="list-style-type: none"> <li>• Keine<br/>Zu Beginn eines Absatzes wird in Editoren kein Einzug verwendet.</li> <li>• Absatz<br/>Die erste Zeile eines Absatzes in Editoren erhält einen automatischen Einzug.</li> <li>• Smart<br/>Der Programmcode wird erkannt und die Absätze werden zur Verbesserung der Lesbarkeit der Syntax automatisch eingerückt.</li> </ul> |
| Ansicht                           | Zeilennummern anzeigen          | Zeigt die Zeilennummer links neben dem Text an.   |
|                                   | Leerzeichen anzeigen            | Zeigt Steuerzeichen innerhalb eines Texts an.   |
| AWL (Anweisungsliste)             | Schriftart und Schriftgröße     | Stellt die Schriftart und die Schriftgröße für AWL-Programmcode ein.  |
| SCL (Structured Control Language) | Schriftart und Schriftgröße     | Stellt die Schriftart und die Schriftgröße für SCL ein.   |
|                                   | Tabulatorweite                  | Stellt die Weite von Tabulatoren in SCL-Programmen ein.   |
|                                   | Zeileneinzug                    | Erstellt SCL-Programme automatisch mit syntaxgerechter Einrückung.  |
|                                   | Zeilennummern anzeigen          | Zeigt in SCL-Programmen Zeilennummern an.   |

**Siehe auch**

Einstellungen ändern (Seite 175)

**6.1.1.4 Übersicht der Druckeinstellungen**

**Überblick**

Die folgende Tabelle zeigt die Einstellungen für den Druck, die Sie vornehmen können:

| Gruppe    | Einstellung                                | Beschreibung  |
|-----------|--|---|
| Allgemein | Tabellendaten immer in Wertepaaren drucken | Tabellen werden nicht in tabellarischer Form, sondern als Liste gedruckt. Zu jeder Spalte werden die entsprechenden Werte aufgelistet.<br><br>Aktivieren Sie diese Option beispielsweise, wenn Sie eine Tabelle drucken möchten, die zu groß für die Druckfläche ist. |
|           | Daten immer in Tabellen drucken            | Alle Parameter von Technologieobjekten werden in Form einer Tabelle gedruckt.   |
|           | Maskengrafiken drucken, wenn möglich       | Wenn der verwendete Editor diese Funktion unterstützt, werden die Inhalte des Editors nicht nur als Tabelle, sondern als vollständige Grafik wie auf dem Bildschirm gedruckt.   |

| Gruppe                 | Einstellung              | Beschreibung   |
|------------------------|--------------------------|--|
| Hardware-Konfiguration | Aktive grafische Sicht   | Die Grafiken der Netz- und Gerätesicht werden mit ausgedruckt.   |
|                        | Aktive Tabelle           | Eine zu einem Editor gehörende Tabelle wird beim Drucken des Editors mit ausgegeben.                                     |
| PLC-Programmierung     | Zoomfaktor               | Legt fest, in welcher Größe Bausteine ausgedruckt werden sollen.   |
|                        | mit Schnittstelle        | Die Schnittstellen von Bausteinen werden mit ausgedruckt.  |
|                        | mit Kommentaren          | Kommentare zu Bausteinen werden mit ausgedruckt.   |
|                        | mit Zeilennummern        | Bei textuellen Programmiersprachen werden die Zeilennummern des Programmcodes mit ausgedruckt.                           |
| HMI-Bilder             | Tab-Reihenfolge anzeigen | Im Ausdruck wird die Reihenfolge angegeben, in der die Runtime Objekte mit Hilfe des Tabulators angewählt werden können. |

**Siehe auch**

Einstellungen ändern (Seite 175)

**6.1.1.5 Einstellungen ändern****Vorgehen**

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein", um die in den vorangegangenen Kapiteln beschriebenen Einstellungen zu ändern. Oder klicken Sie auf einen der weiteren Einträge in der Bereichsnavigation, um Einstellungen für Ihre installierten Produkte zu ändern.
3. Ändern Sie die Einstellungen.

**Ergebnis**

Die Änderung wird sofort übernommen und muss nicht explizit gespeichert werden.

**Siehe auch**

Übersicht der Programmeinstellungen (Seite 171)

Übersicht der Skript- und Texteditor-Einstellungen (Seite 173)

Übersicht der Druckeinstellungen (Seite 174)

## 6.1.2 Aufbau der Bedienoberfläche

### 6.1.2.1 Ansichten

#### Ansichten

Es stehen Ihnen drei verschiedene Ansichten für Ihr Automatisierungsprojekt zur Verfügung:

- Die Portalansicht ist eine aufgabenorientierte Sicht auf die Projektaufgaben.
- Die Projektansicht ist eine Sicht auf die Bestandteile des Projekts und auf die zugehörigen Arbeitsbereiche und Editoren.
- Die Bibliotheksansicht (Seite 181) zeigt die Elemente der Projektbibliothek und der geöffneten globalen Bibliotheken.

Sie können über einen Link zwischen den beiden Ansichten wechseln.

### 6.1.2.2 Portalansicht

#### Funktion der Portalansicht

Die Portalansicht bietet eine aufgabenorientierte Sicht der Werkzeuge. Hier können Sie schnell entscheiden, was Sie tun möchten und das Werkzeug für die jeweilige Aufgabe aufrufen. Falls erforderlich, wird für die ausgewählte Aufgabe automatisch zur Projektansicht (Seite 178) gewechselt.

## Aufbau der Portalansicht

Das folgende Bild zeigt ein Beispiel für die Komponenten der Portalansicht:



- ① Portale für die verschiedenen Aufgaben
- ② Aktionen zum selektierten Portal
- ③ Auswahlfenster zur selektierten Aktion
- ④ Zur Projektansicht wechseln
- ⑤ Anzeige des aktuell geöffneten Projekts

## Portale

Die Portale stellen die grundlegenden Funktionen für die einzelnen Aufgabengebiete zur Verfügung. Welche Portale Ihnen in der Portalansicht angeboten werden, hängt von den installierten Produkten ab.

### **Aktionen zum selektierten Portal**

Abhängig vom ausgewählten Portal werden Ihnen hier die Aktionen angeboten, die Sie in diesem Portal ausführen können. Der Aufruf der Hilfe steht Ihnen in jedem Portal kontextsensitiv zur Verfügung.

### **Auswahlfenster zur selektierten Aktion**

Das Auswahlfenster steht Ihnen in allen Portalen zur Verfügung. Der Inhalt des Fensters passt sich der aktuellen Auswahl an.

### **Zur Projektansicht wechseln**

Mithilfe des Links "Projektansicht" können Sie zur Projektansicht wechseln.

### **Anzeige des aktuell geöffneten Projekts**

An dieser Stelle erhalten Sie die Information, welches Projekt aktuell geöffnet ist.

### **Siehe auch**

Projektnavigation (Seite 181)  
Grundlagen zum Arbeitsbereich (Seite 184)  
Inspektorfenster (Seite 192)  
Grundlagen zu den Task Cards (Seite 194)  
Detailansicht (Seite 198)

## **6.1.2.3 Projektansicht**

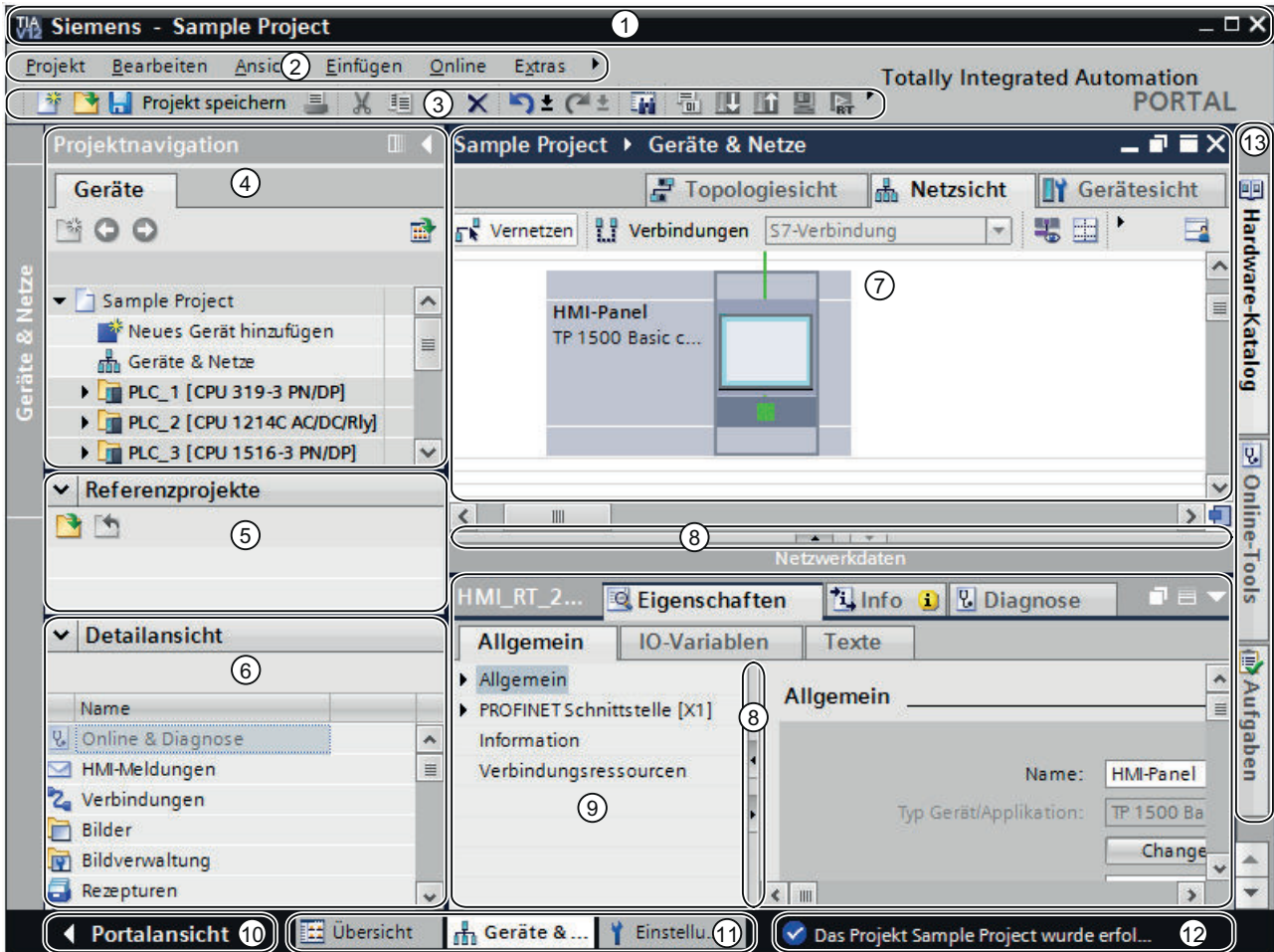
### **Funktion der Projektansicht**

Die Projektansicht ist eine strukturierte Sicht auf alle Bestandteile des Projekts.



## Aufbau der Projektansicht

Das folgende Bild zeigt beispielhaft die Komponenten der Projektansicht:



- ① Titelleiste
- ② Menüleiste
- ③ Funktionsleiste
- ④ Projektnavigation (Seite 181)
- ⑤ Referenzprojekte (Seite 196)
- ⑥ Detailansicht (Seite 198)
- ⑦ Arbeitsbereich (Seite 194)
- ⑧ Fensterteiler
- ⑨ Inspektorfenster (Seite 192)
- ⑩ Zur Portalansicht (Seite 176) wechseln
- ⑪ Editorleiste
- ⑫ Statusleiste mit Fortschrittsanzeige
- ⑬ Task Cards (Seite 194)

### **Titelleiste**

In der Titelleiste wird der Name des Projekts angezeigt.

### **Menüleiste**

In der Menüleiste finden Sie alle Befehle, die Sie für Ihre Arbeit benötigen.

### **Funktionsleiste**

Die Funktionsleiste stellt Ihnen oft benötigte Befehle über Schaltflächen zur Verfügung. Dadurch können Sie auf diese Befehle schneller zugreifen.

### **Fensterleiter**

Fensterleiter trennen einzelne Elemente der Programmoberfläche. Mit den Pfeilen auf den Fensterleitern lassen sich die angrenzenden Teile der Programmoberfläche ein- und ausblenden.

### **Zur Portalansicht wechseln**

Mithilfe des Links "Portalansicht" können Sie zur Portalansicht wechseln.

### **Editorleiste**

In der Editorleiste werden die geöffneten Editoren angezeigt. Falls Sie viele Editoren geöffnet haben, werden sie gruppiert dargestellt. Über die Editorleiste können Sie schnell zwischen den geöffneten Elementen wechseln.

### **Statusleiste mit Fortschrittsanzeige**

In der Statusleiste finden Sie die Fortschrittsanzeige für aktuell laufende Hintergrundprozesse. Hierzu gehört auch eine Fortschrittsleiste, die den Fortschritt grafisch anzeigt. Verweilen Sie etwas länger mit dem Mauszeiger über der Fortschrittsleiste, so wird ein Tooltip eingeblendet, der Ihnen weitere Informationen über die laufenden Hintergrundprozesse gibt. Sie können die Hintergrundprozesse über die Schaltfläche neben der Fortschrittsleiste abbrechen.

Falls zurzeit keine Hintergrundprozesse ausgeführt werden, wird in der Statusleiste die zuletzt generierte Meldung angezeigt.

### **Siehe auch**

Grundlagen zum Arbeitsbereich (Seite 184)

#### 6.1.2.4 Bibliotheksansicht

##### Funktion der Bibliotheksansicht

Die Bibliotheksansicht bietet Ihnen eine Übersicht über die Elemente der Projektbibliothek und der geöffneten globalen Bibliotheken. Sie können über die Task Card "Bibliotheken" zur Bibliotheksansicht wechseln.

Siehe auch: Übersicht über die Bibliotheksansicht (Seite 318)

#### 6.1.2.5 Projektnavigation

##### Funktion der Projektnavigation

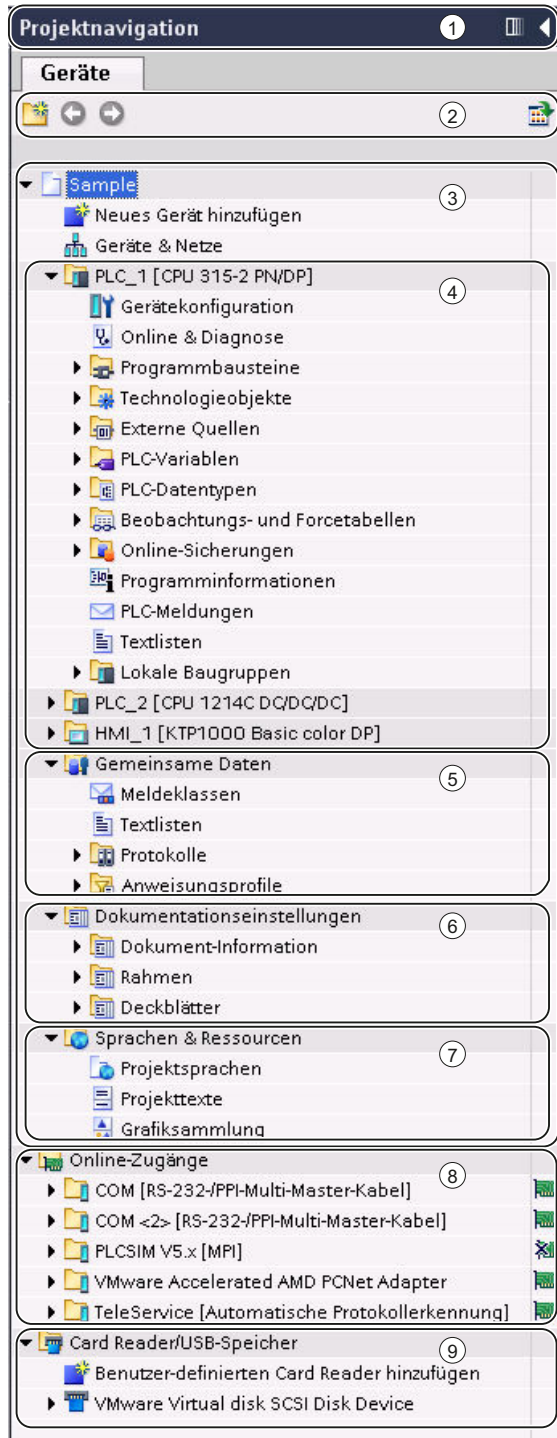
Über die Projektnavigation haben Sie Zugang zu allen Komponenten und Projektdaten. Sie können in der Projektnavigation z. B. folgende Aktionen durchführen:

- Neue Komponenten hinzufügen
- Bestehende Komponenten bearbeiten
- Die Eigenschaften bestehender Komponenten abfragen und verändern

Sie können die Objekte der Projektnavigation entweder mit der Maus selektieren oder über die Tastatur, in dem Sie den Anfangsbuchstaben des gewünschten Objekts eingeben. Wenn mehrere Objekte mit dem gleichen Buchstaben beginnen, wird das nächst untere Objekt selektiert. Die Projektnavigation muss das aktuelle Oberflächenelement sein, damit Sie die Objekte über ihre Anfangsbuchstaben auswählen können.

### Aufbau der Projektnavigation

Das folgende Bild zeigt beispielhaft die Komponenten der Projektnavigation:



- ① Titelleiste
- ② Funktionsleiste
- ③ Projekt
- ④ Geräte
- ⑤ Gemeinsame Daten
- ⑥ Dokumentationseinstellungen
- ⑦ Sprachen & Ressourcen
- ⑧ Online-Zugänge
- ⑨ Card Reader/USB-Speicher

## Titelleiste

In der Titelleiste der Projektnavigation finden Sie die Schaltflächen für das manuelle und automatische Zuklappen der Projektnavigation. Nach dem manuellen Zuklappen positioniert sich die Schaltfläche "Reduzieren" am linken Rand. Sie verändert sich von einem Pfeil-nach-links in einen Pfeil-nach-rechts und dient dem erneuten Aufklappen der Projektnavigation. Über die Schaltfläche "Automatisch reduzieren" können Sie die Projektnavigation automatisch zuklappen lassen, wenn Sie sie nicht benötigen.

Siehe auch: Arbeitsbereich maximieren und minimieren (Seite 186)

## Funktionsleiste

Über die Funktionsleiste der Projektnavigation können Sie folgende Aktionen durchführen:

- Einen neuen Anwenderordner erstellen, z. B. zum Gruppieren von Bausteinen im Ordner "Programmbausteine".
- Vorwärts zur Quelle eines Links und zurück zum Link navigieren  
Für Links in der Projektnavigation stehen zwei Schaltflächen zur Verfügung, über die Sie vom Link zur Quelle und zurück navigieren können.
- Übersicht zum selektierten Objekt im Arbeitsbereich einblenden  
Wenn Sie die Übersicht einblenden, werden die unterlagerten Objekte und Aktionen der Elemente in der Projektnavigation ausgeblendet.

## Projekt

Im Ordner "Projekt" finden Sie alle projektrelevanten Objekte und Aktionen, z. B.:

- Geräte
- Sprachen & Ressourcen
- Online-Zugänge

## Gerät

Für jedes Gerät im Projekt gibt es einen eigenen Ordner mit dessen projektinternem Namen. Innerhalb dieser Ordner sind Objekte und Aktionen strukturiert, die zum Gerät gehören.

### **Gemeinsame Daten**

In diesem Ordner finden Sie Daten, die Sie geräteübergreifend verwenden können, wie z. B. gemeinsame Meldeklassen, Protokolle, Skripte und Textlisten.

### **Dokumentationseinstellungen**

In diesem Ordner legen Sie das Layout für die spätere, gedruckte Projektdokumentation fest.

### **Sprachen & Ressourcen**

In diesem Ordner legen Sie die Projektsprachen und die Projekttexte fest.

### **Online-Zugänge**

In diesem Ordner finden Sie alle Schnittstellen des PG/PC, auch wenn diese nicht zur Kommunikation mit einer Baugruppe verwendet werden.

### **Card Reader/USB-Speicher**

In diesem Ordner werden alle an den PG/PC angeschlossenen Kartenleser und andere USB-Speichermedien verwaltet.

### **Siehe auch**

- Portalansicht (Seite 176)
- Projektansicht (Seite 178)
- Grundlagen zum Arbeitsbereich (Seite 184)
- Inspektorfenster (Seite 192)
- Grundlagen zu den Task Cards (Seite 194)
- Detailansicht (Seite 198)

## **6.1.2.6 Arbeitsbereich**

### **Grundlagen zum Arbeitsbereich**

#### **Funktion des Arbeitsbereichs**

Innerhalb des Arbeitsbereichs werden die Objekte angezeigt, die Sie zur Bearbeitung öffnen. Diese Objekte sind z. B.:

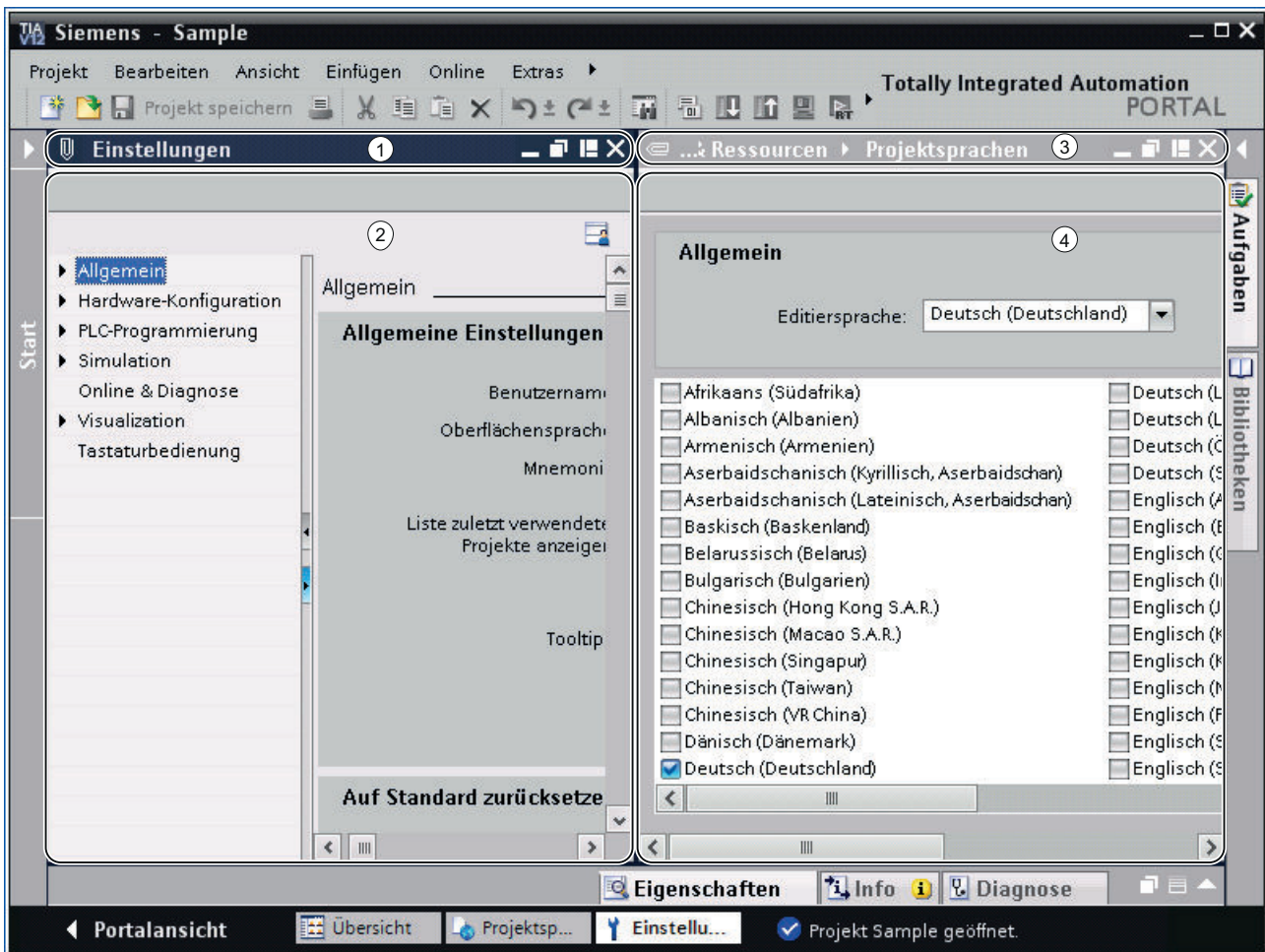
- Editoren und Sichten
- Tabellen

Sie können mehrere Objekte öffnen, von denen jedoch normalerweise nur eins im Arbeitsbereich zu sehen ist. Alle weiteren Objekte werden in der Editorleiste als Register

angezeigt. Wenn Sie für bestimmte Aufgaben zwei Objekte gleichzeitig sehen möchten, können Sie den Arbeitsbereich vertikal oder horizontal teilen oder Elemente des Arbeitsbereichs abdocken. Falls Sie kein Objekt geöffnet haben, ist der Arbeitsbereich leer.

### Aufbau des Arbeitsbereichs

Das folgende Bild zeigt beispielhaft einen vertikal geteilten Arbeitsbereich:



- ① Titelleiste linker Editor
- ② Arbeitsbereich linker Editor
- ③ Titelleiste rechter Editor
- ④ Arbeitsbereich rechter Editor

## Siehe auch

- Arbeitsbereich maximieren und minimieren (Seite 186)
- Arbeitsbereich teilen (Seite 188)
- Elemente des Arbeitsbereichs abdocken (Seite 188)
- Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 189)
- Elemente des Arbeitsbereichs minimieren und maximieren (Seite 191)
- Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 192)
- Anordnung von Editoren und Tabellen sichern (Seite 205)
- Speichern des Bedienoberflächenlayouts (Seite 203)

## Arbeitsbereich maximieren und minimieren

Sie haben die Möglichkeit, den Arbeitsbereich so anzupassen, dass Ihnen eine möglichst große Arbeitsfläche zur Verfügung steht. Dazu können Sie folgende Funktion nutzen:

- **Arbeitsbereich maximieren**  
Sie können mit einem Klick die Task Cards, die Projektnavigation und das Inspektorfenster schließen. Dadurch wird die Fläche des Arbeitsbereichs vergrößert. Sie können den Arbeitsbereich jederzeit wieder minimieren, um zur vorherigen Ansicht zurückzukehren.
- **Task Cards, Projektnavigation und Inspektorfenster automatisch zuklappen**  
Sie können für die Task Cards, die Projektnavigation und das Inspektorfenster die Option "Automatisch reduzieren" verwenden. Dadurch klappen diese Fenster automatisch zu, wenn Sie sie nicht benötigen.

## Arbeitsbereich maximieren und minimieren

Um den Arbeitsbereich zu maximieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie ein Element, z. B. einen Editor oder eine Tabelle.  
Das Element wird im Arbeitsbereich angezeigt.
2. Klicken Sie in der Titelleiste des Elements auf die Schaltfläche "Maximieren".  
Die Task Cards, die Projektnavigation und das Inspektorfenster werden zugeklappt und der Arbeitsbereich wird mit maximaler Größe dargestellt.

Um den Arbeitsbereich wieder zu minimieren, gehen Sie folgendermaßen vor:



1. Klicken Sie in der Titelleiste des angezeigten Elements auf die Schaltfläche "Einbetten". Sie erhalten wieder die Ansicht wie vor dem Maximieren des Arbeitsbereichs. Das heißt, wenn die Task Cards, die Projektnavigation oder das Inspektorfenster davor aufgeklappt waren, werden sie wieder aufgeklappt.

### **Task Cards, Projektnavigation und Inspektorfenster automatisch zuklappen**

Um die Task Cards automatisch zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste der Task Cards auf "Automatisch reduzieren". Die Task Cards klappen zu, wenn Sie mit der Maus auf eine Fläche außerhalb der Task Cards klicken.
2. Um die Task Cards zu verwenden, klicken Sie auf die zugeklappten Task Cards.
3. Die Task Cards klappen auf und Sie können sie verwenden. Die Option "Automatisch reduzieren" bleibt dabei aktiviert.

Um die Projektnavigation automatisch zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste der Projektnavigation auf "Automatisch reduzieren". Die Projektnavigation klappt zu, wenn Sie mit der Maus auf eine Fläche außerhalb der Projektnavigation klicken.
2. Um die Projektnavigation zu verwenden, klicken Sie auf die zugeklappte Projektnavigation. Die Projektnavigation klappt auf und Sie können sie verwenden. Die Option "Automatisch reduzieren" bleibt dabei aktiviert.

Um das Inspektorfenster automatisch zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Inspektorfensters auf "Automatisch reduzieren". Das Inspektorfenster klappt zu, wenn Sie mit der Maus auf eine Fläche außerhalb des Inspektorfensters klicken.
2. Um das Inspektorfenster zu verwenden, klicken Sie auf das zugeklappte Inspektorfenster. Das Inspektorfenster klappt auf und Sie können es verwenden. Die Option "Automatisch reduzieren" bleibt dabei aktiviert.

Um das automatische Zuklappen wieder zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie im jeweiligen Fenster erneut auf "Dauerhaft erweitern". Die Option "Automatisch reduzieren" wird deaktiviert und das Fenster bleibt aufgeklappt.

### **Siehe auch**

Grundlagen zum Arbeitsbereich (Seite 184)

Arbeitsbereich teilen (Seite 188)

Elemente des Arbeitsbereichs abdocken (Seite 188)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 189)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 191)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 192)

Anordnung von Editoren und Tabellen sichern (Seite 205)

## Arbeitsbereich teilen

Sie können den Arbeitsbereich vertikal oder horizontal teilen.

## Vorgehen

Um den Arbeitsbereich vertikal oder horizontal zu teilen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Editorbereich vertikal teilen" oder "Editorbereich horizontal teilen".  
Das angeklickte Element und das nächste Element der Editorleiste werden nebeneinander oder untereinander angezeigt.

---

### Hinweis

Wenn kein Element im Arbeitsbereich geöffnet ist, stehen Ihnen die Funktionen "Editorbereich vertikal teilen" und "Editorbereich horizontal teilen" nicht zur Verfügung

---

## Siehe auch

- Grundlagen zum Arbeitsbereich (Seite 184)
- Arbeitsbereich maximieren und minimieren (Seite 186)
- Elemente des Arbeitsbereichs abdocken (Seite 188)
- Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 189)
- Elemente des Arbeitsbereichs minimieren und maximieren (Seite 191)
- Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 192)
- Anordnung von Editoren und Tabellen sichern (Seite 205)

## Elemente des Arbeitsbereichs abdocken

Sie können folgende Elemente vom Arbeitsbereich abdocken und in einem eigenen Fenster öffnen:

- Editoren
- Tabellen
- Einstellungsfenster
- Task Cards
- Inspektorfenster

Abgedockte Elemente können Sie jederzeit wieder in den Arbeitsbereich einbetten.

---

#### **Hinweis**

##### **Eigenschaften von Elementen in einem abgedockten Fenster**

Die Eigenschaften von Elementen, die Sie in einem abgedockten Fenster selektiert haben, werden nur dann im Inspektorfenster angezeigt, wenn das Inspektorfenster ebenfalls abgedockt ist.

---

### **Elemente des Arbeitsbereichs abdocken**

Um Elemente aus dem Arbeitsbereich abzudocken, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Elements auf die Schaltfläche "Ablösen".  
Das Element wird aus dem Arbeitsbereich ausgelöst und als eigenes Fenster angezeigt. Sie können das Fenster nun frei platzieren. Wenn Sie das Fenster minimieren, können Sie es über die Editorleiste wiederherstellen.

### **Elemente im Arbeitsbereich einbetten**

Um Elemente wieder im Arbeitsbereich einzubetten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Elements auf die Schaltfläche "Einbetten".  
Das Element wird wieder im Arbeitsbereich angezeigt.

### **Siehe auch**

Grundlagen zum Arbeitsbereich (Seite 184)

Arbeitsbereich maximieren und minimieren (Seite 186)

Arbeitsbereich teilen (Seite 188)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 189)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 191)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 192)

Anordnung von Editoren und Tabellen sichern (Seite 205)

### **Gruppierte Elemente des Arbeitsbereichs verwenden**

Wenn Sie mehr als fünf gleichartige Elemente öffnen, z. B. Editoren oder Tabellen, werden diese in der Editorleiste gruppiert. Sie können diese Gruppen folgendermaßen verwenden:

- Einzelne Elemente einer Gruppe anzeigen
- Alle Elemente einer Gruppe in eigenen Fenstern anzeigen
- Alle angezeigten Elemente einer Gruppe in den Arbeitsbereich einbetten
- Alle angezeigten Elemente minimieren
- Alle Elemente einer Gruppe schließen

### Einzelne Elemente einer Gruppe anzeigen

Um einzelne Elemente einer Gruppe anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste auf die Gruppe, in der sich das Element befindet, das Sie anzeigen möchten.  
Eine Liste mit allen verfügbaren Elementen der Gruppe wird angezeigt.
2. Klicken Sie auf das Element, das Sie anzeigen möchten.

### Alle Elemente einer Gruppe in eigenen Fenstern anzeigen

Um alle Elemente einer Gruppe in eigenen Fenstern anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste mit der rechten Maustaste auf die Gruppe, deren Elemente Sie anzeigen möchten.
2. Wählen Sie im Kontextmenü "Gruppe wiederherstellen".  
Alle Elemente der Gruppe werden in eigenen Fenstern angezeigt, die sich überlagern. Verschieben Sie die Fenster, um die einzelnen Elemente zu sehen oder wählen Sie ein Element über die Gruppe in der Editorleiste.

### Alle angezeigten Elemente einer Gruppe in den Arbeitsbereich einbetten

Um alle in eigenen Fenstern angezeigten Elemente einer Gruppe wieder in den Arbeitsbereich einzubetten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste mit der rechten Maustaste auf die Gruppe, deren Elemente Sie einbetten möchten.
2. Wählen Sie im Kontextmenü "Gruppe einbetten".  
Alle Elemente der Gruppe werden wieder in den Arbeitsbereich eingebettet.

### Alle angezeigten Elemente minimieren

Um alle Elemente einer Gruppe zu minimieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste mit der rechten Maustaste auf die Gruppe, deren Elemente Sie minimieren möchten.
2. Wählen Sie im Kontextmenü "Gruppe minimieren".  
Alle Elemente der Gruppe werden minimiert. Die minimierten Elemente sind dennoch geöffnet und können in der Editorleiste über die Gruppe schnell wieder maximiert werden.

### Alle Elemente einer Gruppe schließen

Um alle Elemente einer Gruppe zu schließen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste mit der rechten Maustaste auf die Gruppe, deren Elemente Sie schließen möchten.
2. Wählen Sie im Kontextmenü "Gruppe schließen".  
Alle Elemente der Gruppe werden geschlossen. Die Gruppe wird entfernt.

## Siehe auch

- Grundlagen zum Arbeitsbereich (Seite 184)
- Arbeitsbereich maximieren und minimieren (Seite 186)
- Arbeitsbereich teilen (Seite 188)
- Elemente des Arbeitsbereichs abdocken (Seite 188)
- Elemente des Arbeitsbereichs minimieren und maximieren (Seite 191)
- Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 192)
- Anordnung von Editoren und Tabellen sichern (Seite 205)

## Elemente des Arbeitsbereichs minimieren und maximieren

Sie können die geöffneten Elemente des Arbeitsbereichs, z. B. Editoren oder Tabellen, bei Bedarf minimieren. Ein minimiertes Element ist dennoch geöffnet und kann über die Editorleiste schnell wieder maximiert werden.

## Elemente des Arbeitsbereichs minimieren

Um Elemente des Arbeitsbereichs zu minimieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Elements auf die Schaltfläche "Minimieren".  
Das Element wird minimiert und ist weiterhin über die Editorleiste verfügbar.

Um alle Elemente gleichzeitig zu minimieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Alle minimieren".

## Elemente des Arbeitsbereichs maximieren

Um Elemente im Arbeitsbereich wieder zu maximieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Editorleiste auf das gewünschte Element.  
Das Element wird wieder maximiert und im Arbeitsbereich angezeigt.

## Siehe auch

- Grundlagen zum Arbeitsbereich (Seite 184)
- Arbeitsbereich maximieren und minimieren (Seite 186)
- Arbeitsbereich teilen (Seite 188)
- Elemente des Arbeitsbereichs abdocken (Seite 188)
- Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 189)
- Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 192)
- Anordnung von Editoren und Tabellen sichern (Seite 205)

### Zwischen den Elementen des Arbeitsbereichs wechseln

Sie können jederzeit zwischen den Elementen des Arbeitsbereichs wechseln.

### Zwischen den Elementen des Arbeitsbereichs wechseln

Um zum vorherigen oder nächsten Editor zu wechseln, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Nächster Editor" oder "Vorheriger Editor".  
Der nächste oder der vorherige Editor wird angezeigt.

### Siehe auch

Grundlagen zum Arbeitsbereich (Seite 184)

Arbeitsbereich maximieren und minimieren (Seite 186)

Arbeitsbereich teilen (Seite 188)

Elemente des Arbeitsbereichs abdocken (Seite 188)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 189)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 191)

Anordnung von Editoren und Tabellen sichern (Seite 205)

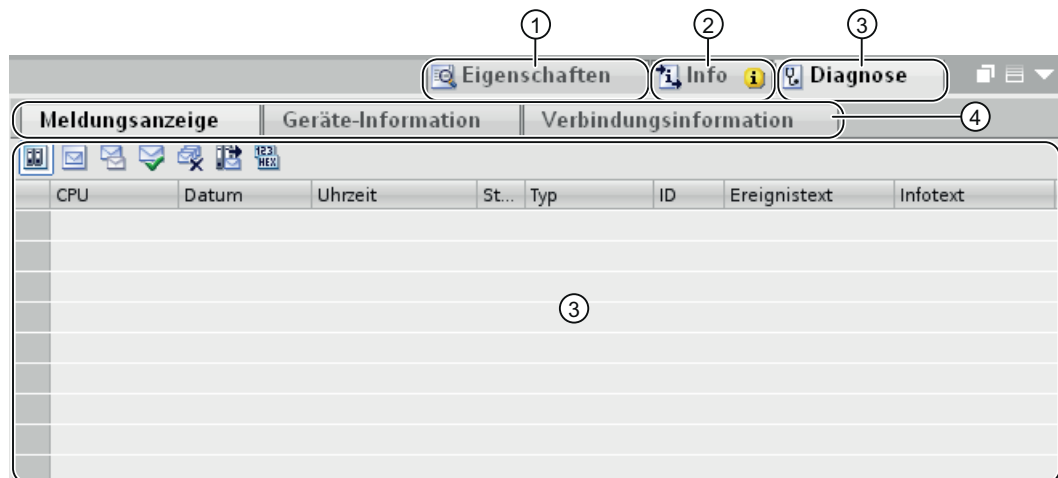
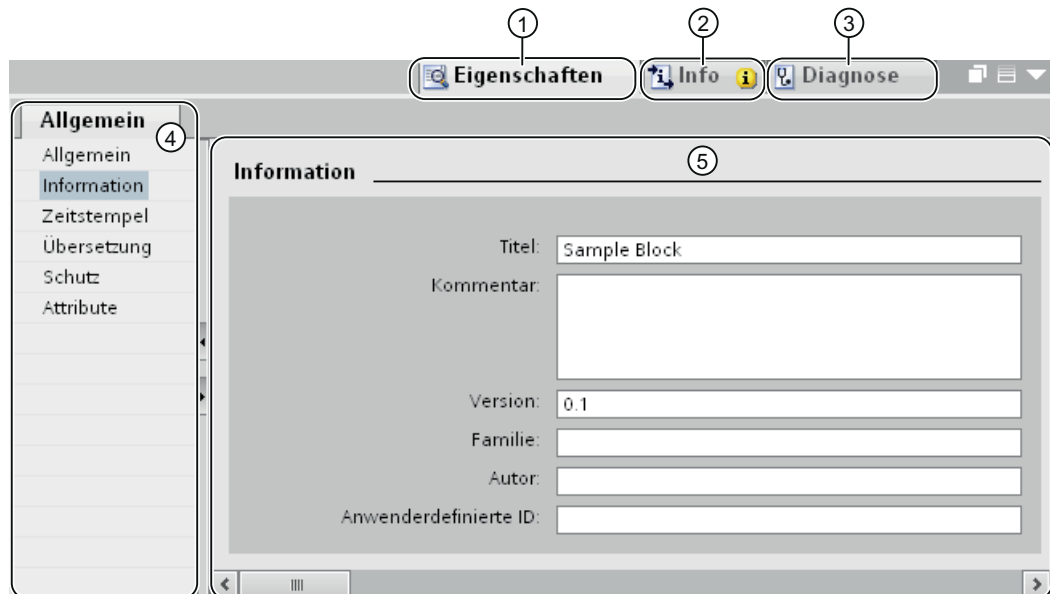
#### 6.1.2.7 Inspektorfenster

### Funktion des Inspektorfensters

Im Inspektorfenster werden zusätzliche Informationen zu einem selektierten Objekt oder zu ausgeführten Aktionen angezeigt.

## Aufbau des Inspektorfensters

Die folgenden Bilder zeigen die Komponenten des Inspektorfensters:



- ① Register "Eigenschaften"
- ② Register "Info"
- ③ Register "Diagnose"
- ④ Navigation innerhalb der Register:
  - Bereichsnavigation im Register "Eigenschaften"
  - untergeordnete Register in den Registern "Info" und "Diagnose"

### Register "Eigenschaften"

In diesem Register werden die Eigenschaften des selektierten Objekts angezeigt. Editierbare Eigenschaften können Sie hier verändern.

### Register "Info"

In diesem Register werden weitere Informationen zum selektierten Objekt und Meldungen zu den ausgeführten Aktionen, z. B. Übersetzen, angezeigt.

### Register "Diagnose"

In diesem Register erhalten Sie Informationen zu Systemdiagnose-Ereignissen, projektierten Meldungseignissen und zur Verbindungsdiagnose.

### Navigation innerhalb der Register

Über die Bereichsnavigation und die untergeordneten Register können Sie innerhalb der Register die gewünschten Informationen anzeigen.

### Siehe auch

- Projektnavigation (Seite 181)
- Grundlagen zum Arbeitsbereich (Seite 184)
- Portalansicht (Seite 176)
- Projektansicht (Seite 178)
- Grundlagen zu den Task Cards (Seite 194)
- Detailansicht (Seite 198)

## 6.1.2.8 Task Cards

### Grundlagen zu den Task Cards

#### Funktion der Task Cards

Abhängig vom bearbeiteten oder selektierten Objekt stehen Ihnen Task Cards zur Verfügung, über die Sie weitere Aktionen ausführen können, z. B.:

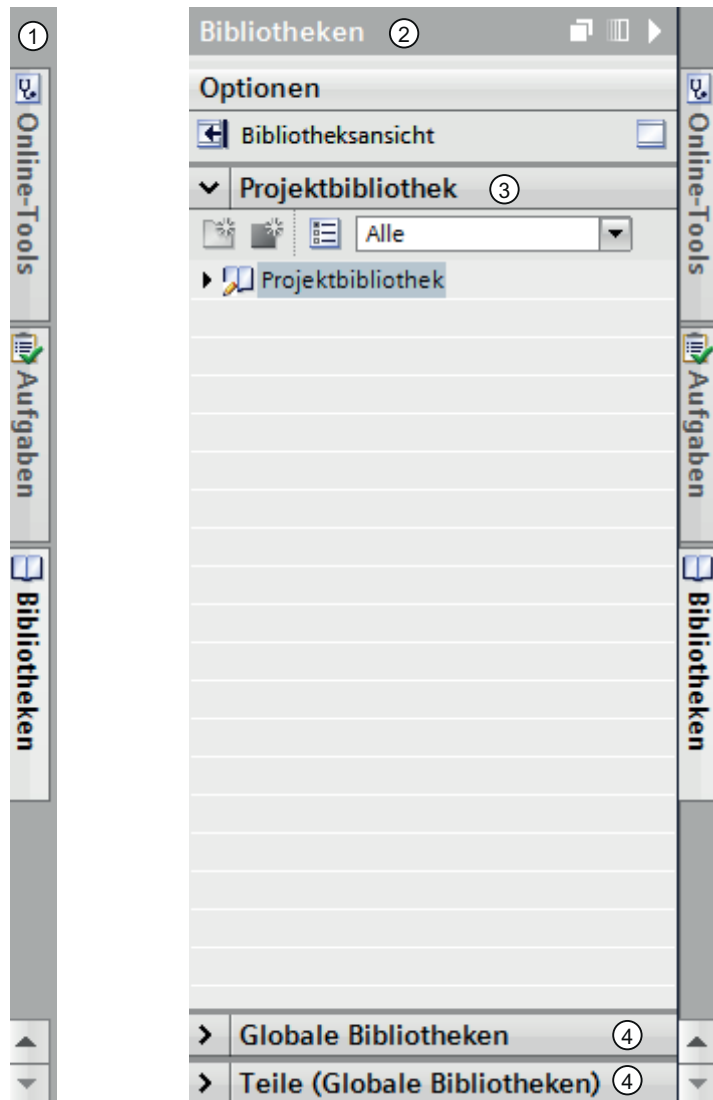
- Objekte aus einer Bibliothek oder aus dem Hardware-Katalog auswählen
- Objekte im Projekt suchen und ersetzen
- Vordefinierte Objekte in den Arbeitsbereich ziehen

Die verfügbaren Task Cards finden Sie in einer Leiste am rechten Bildschirmrand. Sie können sie jederzeit auf- und zuklappen. Welche Task Cards Ihnen zur Verfügung stehen, hängt von den installierten Produkten ab. Komplexere Task Cards sind in Paletten unterteilt, die Sie ebenfalls auf- und zuklappen können.



## Aufbau der Task Cards

Das folgende Bild zeigt beispielhaft die Leiste mit den Task Cards:



- ① Task Cards geschlossen
- ② Task Card geöffnet
- ③ Geöffnete Palette einer Task Card
- ④ Geschlossene Palette einer Task Card

## Siehe auch

- Palettenmodus ändern (Seite 196)
- Projektnavigation (Seite 181)
- Grundlagen zum Arbeitsbereich (Seite 184)
- Inspektorfenster (Seite 192)
- Portalansicht (Seite 176)
- Projektansicht (Seite 178)
- Detailansicht (Seite 198)

## Palettenmodus ändern

Sie können zwischen zwei Palettenmodi wählen:

- Einzelpalettenmodus:  
Es ist immer nur eine Palette geöffnet. Wenn Sie eine weitere Palette öffnen, wird die bisher geöffnete Palette automatisch geschlossen.
- Mehrpalettenmodus:  
Sie können mehrere Paletten gleichzeitig öffnen.

## Vorgehen

Um den Palettenmodus zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb einer Task Card über den Paletten auf die Schaltfläche "Palettenmodus ändern".

## Siehe auch

- Grundlagen zu den Task Cards (Seite 194)

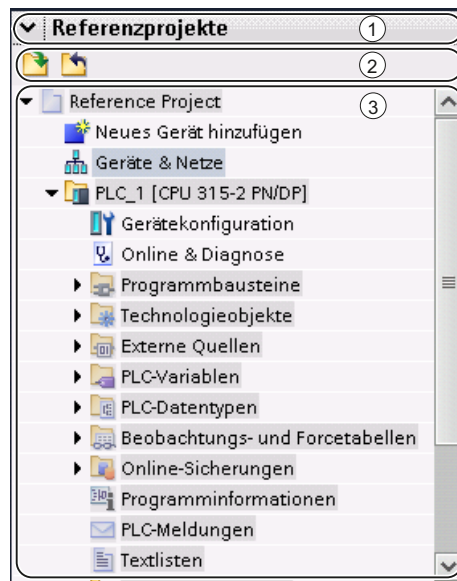
### 6.1.2.9 Referenzprojekte

#### Funktion von Referenzprojekten

In der Palette "Referenzprojekte" können Sie zusätzlich zum aktuellen Projekt weitere Projekte öffnen. Diese Referenzprojekte werden schreibgeschützt geöffnet und können nicht bearbeitet werden. Sie können die Objekte eines Referenzprojekts aber in Ihr aktuelles Projekt ziehen und dort weiter bearbeiten. Zusätzlich können Sie die Objekte eines Referenzprojekts mit den Objekten Ihres aktuellen Projekts vergleichen.

#### Aufbau der Palette "Referenzprojekte"

Das folgende Bild zeigt den Aufbau der Palette "Referenzprojekte":



- ① Titelleiste
- ② Funktionsleiste
- ③ Geöffnete Referenzprojekte

### Titelleiste

In der Titelleiste der Palette "Referenzprojekte" finden Sie den Pfeil für das Schließen der Palette. Nach dem Schließen verändert er sich von einem Pfeil-nach-unten in einen Pfeil-nach-rechts und dient dem Öffnen der Palette.

### Funktionsleiste

In der Funktionsleiste finden Sie die Schaltflächen für das Öffnen und Schließen von Referenzprojekten.

### Geöffnete Referenzprojekte

Geöffnete Referenzprojekte werden mit ihren Objekten und ihrer hierarchischen Struktur schreibgeschützt angezeigt.

### Siehe auch

Grundlagen zu Referenzprojekten (Seite 252)

Referenzprojekt öffnen und schließen (Seite 252)

### 6.1.2.10 Detailansicht

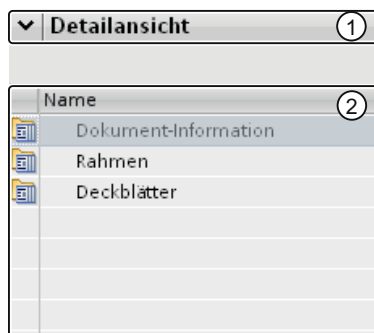
#### Funktion der Detailansicht

In der Detailansicht werden bestimmte Inhalte eines selektierten Objekts im Übersichtsfenster oder in der Projektnavigation angezeigt. Mögliche Inhalte sind beispielsweise Textlisten oder Variablen.

Inhalte von Ordnern werden dagegen nicht angezeigt. Verwenden Sie für die Anzeige von Ordnerinhalten entweder die Projektnavigation oder das Übersichtsfenster.

#### Aufbau der Detailansicht

Das folgende Bild zeigt ein Beispiel für die Detailansicht:



- ① Titelleiste
- ② Inhalte des selektierten Objekts

#### Titelleiste

In der Titelleiste der Detailansicht finden Sie den Pfeil für das Schließen der Detailansicht. Nach dem Schließen verändert er sich von einem Pfeil-nach-unten in einen Pfeil-nach-rechts und dient dem Öffnen der Detailansicht.

#### Objekte

Die angezeigten Inhalte variieren abhängig vom selektierten Objekt. Sie können die Objektinhalte per Drag & Drop direkt aus der Detailansicht an die gewünschte Verwendungsstelle ziehen.

## Siehe auch

- Projektnavigation (Seite 181)
- Grundlagen zum Arbeitsbereich (Seite 184)
- Inspektorfenster (Seite 192)
- Grundlagen zu den Task Cards (Seite 194)
- Portalansicht (Seite 176)
- Projektansicht (Seite 178)

### 6.1.2.11 Übersichtsfenster

## Übersichtsfenster

### Funktionen des Übersichtsfensters

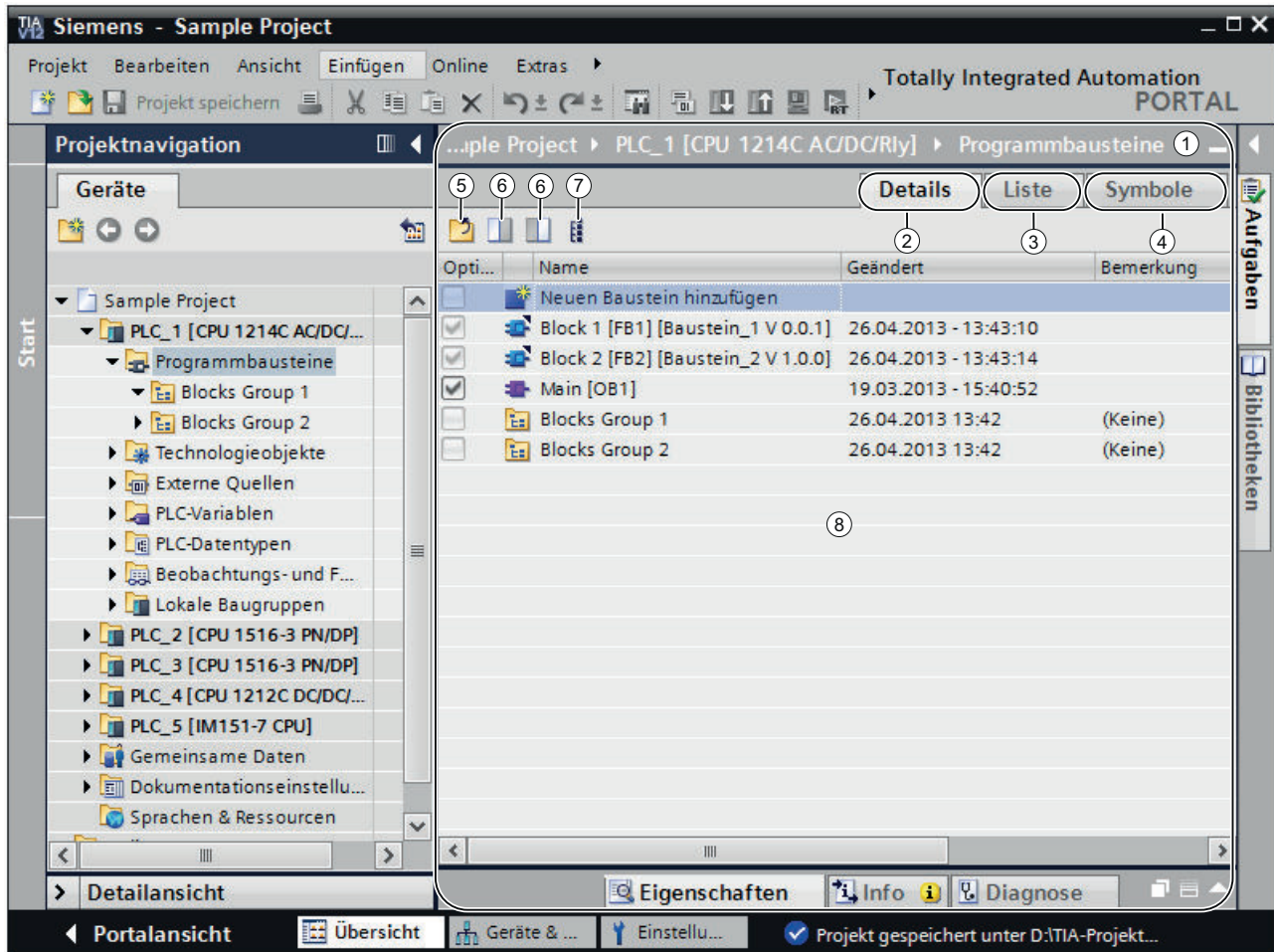
Das Übersichtsfenster dient zur Ergänzung der Projektnavigation. Das Übersichtsfenster zeigt die Inhalte des aktuell in der Projektnavigation selektierten Ordners an.

Außerdem können Sie im Übersichtsfenster folgende Aktionen durchführen:

- Objekte öffnen
- Eigenschaften von Objekten im Inspektorfenster anzeigen und bearbeiten
- Objekte umbenennen
- Objektspezifische Aktionen über das Kontextmenü aufrufen
- Objekte in einer Gegenüberstellung vergleichen
- Verschiedene Objekt-Operationen durchführen, wie z. B. Objekte per Drag & Drop aus der Bibliothek einfügen, Objekte verschieben, kopieren, einfügen und löschen

### Aufbau des Übersichtsfensters

Das folgende Bild zeigt die Komponenten des Übersichtsfensters:



- ① Übersichtsfenster
- ② Umschalten in die Detailansicht
- ③ Umschalten in die Listenansicht
- ④ Umschalten in die Symbolansicht
- ⑤ Wechseln zur übergeordneten Ebene
- ⑥ Teilen des Übersichtsfensters in zwei Seiten. Es wird entweder die rechte oder linke Hälfte des Übersichtsfensters synchronisiert. Durch erneutes Klicken wird die Teilung wieder aufgehoben.
- ⑦ Es werden alle Bausteine innerhalb des Ordners "Programmbausteine" angezeigt, auch wenn diese sich in unterlagerten Gruppen befinden. Diese Option ist nur in der Detailsicht verfügbar.
- ⑧ Inhalte des Objekts, welches in der Projektnavigation selektiert ist.

## Darstellungsformen des Übersichtsfensters

Die Inhalte des Übersichtsfensters können folgendermaßen dargestellt werden:

- **Detailansicht**  
Die Objekte werden in einer Liste mit zusätzlichen Informationen, wie z. B. dem Änderungsdatum, angezeigt.
- **Listenansicht**  
Die Objekte werden in einer einfachen Liste angezeigt.
- **Symbolansicht**  
Die Objekte werden symbolisch nach Kategorien geordnet angezeigt.

## Siehe auch

Objekte im Übersichtsfenster vergleichen (Seite 201)

Detailansicht des Übersichtsfensters sortieren (Seite 202)

Übersicht über die Bibliotheksansicht (Seite 318)

## Objekte im Übersichtsfenster vergleichen

Im Übersichtsfenster können Sie die Inhalte zweier Ordner oder Objekte gegenüberstellen und vergleichen. Das Übersichtsfenster wird in der Mitte geteilt und Sie können auf der linken und rechten Seite jeweils unterschiedliche Inhalte anzeigen.

Darüber hinaus ist es möglich, Objekte zwischen den geteilten Fenstern per Drag & Drop zu verschieben. Auf diese Weise können Sie beispielsweise Inhalte von einem Ordner in einen anderen verschieben.

## Vorgehen

Um das Übersichtsfenster in der Mitte zu teilen, oder um die Teilung wieder aufzuheben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Linke Seite synchronisieren" oder "Rechte Seite synchronisieren", um das Übersichtsfenster zu teilen. Dabei wird entweder die linke oder die rechte Seite des Übersichtsfensters mit den Inhalten des selektierten Objekts in der Projektnavigation synchronisiert.
2. Um die Teilung wieder aufzuheben, klicken Sie erneut auf das zuvor gewählte Symbol.

## Siehe auch

Übersichtsfenster (Seite 199)

### Detailansicht des Übersichtsfensters sortieren

In der Detailansicht des Übersichtsfensters stehen Ihnen einige Möglichkeiten zur Verfügung, die Darstellung anzupassen:

- Einfügen zusätzlicher Spalten  
Einige Spalten sind standardmäßig ausgeblendet, um die Übersichtlichkeit zu erhöhen. Nicht angezeigte Spalten blenden Sie bei Bedarf ein. Welche Spalten zur Verfügung stehen, ist abhängig vom gewählten Objekt.
- Anzeigen der Programmbausteine in flacher Hierarchie  
Inhalte des Ordners "Programmbausteine" lassen sich in einer flachen Hierarchie anzeigen. Alle Bausteine werden auf einmal angezeigt, auch wenn sich diese in verschiedenen Gruppen befinden.
- Sortieren der Tabellenspalten  
Einzelne Spalten der Tabelle lassen sich auf- oder absteigend sortieren.

### Spalten ein- oder ausblenden

Um zusätzliche Spalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Titelleiste der Tabelle.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen" und selektieren Sie die Spalten, die angezeigt werden sollen.

### Programmbausteine in flacher Hierarchie anzeigen

Um die Inhalte des Ordners "Programmbausteine" in einer flachen Hierarchie anzuzeigen, gehen Sie folgendermaßen vor:

Selektieren Sie den Ordner "Programmbausteine" in der Projektnavigation.

Klicken Sie in der Funktionsleiste auf das Symbol "Unterlagerte Elemente anzeigen".

Alle Bausteine werden auf einmal in der Tabelle angezeigt, auch wenn diese sich in unterlagerten Gruppen befinden.

### Tabelle auf- oder absteigend sortieren

Um die Tabelle nach einer Spalte auf- oder absteigend zu sortieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Tabellenkopf einer Spalte, wenn Sie diese aufsteigend sortieren möchten.
2. Klicken Sie erneut auf den Tabellenkopf derselben Spalte, um die Spalte absteigend zu sortieren.
3. Klicken Sie ein drittes Mal auf den Tabellenkopf derselben Spalte, um die Sortierung wieder aufzuheben.

### Siehe auch

Übersichtsfenster (Seite 199)



## 6.1.2.12 Bedienoberflächenlayout

### Speichern des Bedienoberflächenlayouts

#### Möglichkeiten das Bedienoberflächenlayout zu sichern

Wenn Sie eine Änderung an der Bedienoberfläche vornehmen, dann bleibt diese auch nach einem Neustart des TIA-Portals erhalten. Zu einer Änderung am Bedienoberflächenlayout zählt beispielsweise das Verschieben eines Fensters oder das Anpassen der Größe eines Editors.

Neben dem automatischen Speichern des Bedienoberflächenlayouts haben Sie die Möglichkeit bestimmte Anordnungen manuell zu sichern:

- **Speichern des Fensterlayouts**  
Sie können die Anordnungen der Fenster und Editoren des TIA-Portals manuell speichern und zu einem späteren Zeitpunkt wiederherstellen. Fünf Fensterlayouts lassen sich über eine Tastenkombination aufrufen.  
Verwenden Sie diese Funktion beispielsweise, wenn Sie mit einem Notebook arbeiten, welches Sie bei Bedarf um einen externen Monitor ergänzen. So können Sie sich jeweils ein Fensterlayout für den mobilen Einsatz auf dem Notebook-Display und eines für die Arbeit im Büro mit einem externen Monitor anlegen.
- **Speichern der Anordnung innerhalb von Editoren**  
Bei einigen Editoren können Sie die Darstellung anpassen. Sie können z. B. die Breite von Tabellen anpassen oder einzelnen Tabellenspalten ein- und ausblenden.

#### Siehe auch

Fensterlayout speichern (Seite 203)

Fensterlayout laden (Seite 204)

Fensterlayouts verwalten (Seite 205)

Anordnung von Editoren und Tabellen sichern (Seite 205)

Bedienoberflächenlayout zurücksetzen (Seite 206)

Grundlagen zum Arbeitsbereich (Seite 184)

#### Fensterlayout speichern

Sie können das aktuelle Fensterlayout speichern, um es zu einem späteren Zeitpunkt in derselben Form wieder aufzurufen.

#### Vorgehen

Um ein Fensterlayout zu speichern, gehen Sie folgendermaßen vor:

1. Ordnen Sie alle Fenster so an, wie Sie diese speichern möchten.
2. Wählen Sie im Menü "Fenster" den Befehl "Fensterlayout speichern unter".  
Der Dialog "Fensterlayout speichern" wird geöffnet.

3. Geben Sie in das Feld "Name" einen Namen für das Fensterlayout ein.
4. Geben Sie eine Beschreibung des Fensterlayouts in das Feld "Beschreibung" ein, um das Fensterlayout später leichter identifizieren zu können.
5. Klicken Sie auf die Schaltfläche "Speichern".

## Ergebnis

Das neue Fensterlayout wird an letzter Position nach den bereits vorhandenen Fensterlayouts gespeichert. Die ersten fünf Fensterlayouts können über eine Tastenkombination aufgerufen werden.

## Siehe auch

Speichern des Bedienoberflächenlayouts (Seite 203)

## Fensterlayout laden

Wenn Sie bereits ein Fensterlayout gespeichert haben, können Sie dieses laden und so Ihre Arbeitsumgebung schnell an die jeweiligen Gegebenheiten anpassen. Die ersten fünf Fensterlayouts können Sie im Schnellzugriff über das Menü "Fenster" oder über eine Tastenkombination laden.

Wenn Sie ein Fensterlayout geladen haben und anschließend Änderungen an der Anordnung der Fenster vorgenommen haben, können Sie das ursprünglich gespeicherte Fensterlayout wiederherstellen.

## Fensterlayouts 1 bis 5 über einen Schnellzugriff laden

Um eines von den ersten fünf gespeicherten Fensterlayouts zu laden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" ein Fensterlayout aus oder wählen Sie die Tastenkombination <Alt+Shift+[1 ... 5]>.

## Weitere Fensterlayouts laden

Um ein Fensterlayout zu wählen, welches sich nicht unter den ersten fünf Fensterlayouts befindet, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Weitere Fensterlayouts".  
Der Dialog "Fensterlayouts verwalten" wird geöffnet.
2. Selektieren Sie das gewünschte Fensterlayout.
3. Klicken Sie auf "OK".

## Fensterlayout wiederherstellen

Um zum gespeicherten Fensterlayout zurückzukehren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Fensterlayout wiederherstellen" oder wählen Sie die Tastenkombination <Alt+Shift+0>.

## Siehe auch

Speichern des Bedienoberflächenlayouts (Seite 203)

## Fensterlayouts verwalten

Sie können bei bereits vorhandenen Fensterlayouts die folgenden Aktionen durchführen:

- Reihenfolge der Fensterlayouts ändern  
Die Reihenfolge der Fensterlayouts ist entscheidend, da sich die ersten fünf Fensterlayouts direkt über das Menü "Fenster" und eine Tastenkombination aufrufen lassen.
- Wählen eines Fensterlayouts  
Wenn sich ein Fensterlayout nicht unter den ersten fünf Fensterlayouts befindet, können Sie es über den Dialog "Fensterlayouts verwalten" aufrufen.
- Löschen von Fensterlayouts

## Vorgehen

Um die vorhandenen Fensterlayouts zu verwalten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Fenster" den Befehl "Fensterlayouts verwalten".  
Der Dialog "Fensterlayouts verwalten" wird geöffnet.
2. Selektieren Sie das Fensterlayout, welches Sie verändern möchten.
3. Klicken Sie auf das Symbol "aufwärts " oder "abwärts", um das Fensterlayout nach vorne oder nach hinten zu bringen.
4. Klicken Sie auf das Symbol "löschen", um das selektierte Fensterlayout zu löschen.
5. Klicken Sie auf "OK".  
Das selektierte Fensterlayout wird aktiviert.

## Siehe auch

Speichern des Bedienoberflächenlayouts (Seite 203)

## Anordnung von Editoren und Tabellen sichern

Sie haben die Möglichkeit, Editoren und Tabellen an Ihre Bedürfnisse anzupassen. So können Sie z. B. in Tabellen die Spalten ausblenden, die Sie nicht benötigen. Die von Ihnen angepasste Ansicht können Sie anschließend sichern.

## Vorgehen

Um die Anordnung von Editoren und Tabellen des Arbeitsbereichs zu sichern, gehen Sie folgendermaßen vor:

1. Passen Sie den Editor oder die Tabelle nach Ihren Bedürfnissen an.
2. Klicken Sie innerhalb des Editors oder der Tabelle auf die Schaltfläche "Anordnung merken".

## Ergebnis

Die Anordnung wird gesichert. Wenn Sie den Editor oder die Tabelle erneut öffnen, wird diese Anordnung verwendet.

## Siehe auch

Grundlagen zum Arbeitsbereich (Seite 184)

Arbeitsbereich maximieren und minimieren (Seite 186)

Arbeitsbereich teilen (Seite 188)

Elemente des Arbeitsbereichs abdocken (Seite 188)

Gruppierte Elemente des Arbeitsbereichs verwenden (Seite 189)

Elemente des Arbeitsbereichs minimieren und maximieren (Seite 191)

Zwischen den Elementen des Arbeitsbereichs wechseln (Seite 192)

Speichern des Bedienoberflächenlayouts (Seite 203)

## Bedienoberflächenlayout zurücksetzen

Jede Änderung, die Sie am Layout der Bedienoberfläche vornehmen, wird gespeichert. Die Änderungen stehen somit auch nach dem Neustart des TIA-Portals zur Verfügung. Ändern Sie beispielsweise die Höhe und Breite eines Texteditors, oder verändern die Aufteilung einer Tabelle, so bleiben Ihre Änderungen dauerhaft erhalten und Sie müssen Ihre individuellen Anpassungen nicht erneut vornehmen.

In manchen Fällen kann es jedoch sinnvoll sein, die ursprünglichen Layout-Einstellungen wiederherzustellen, z. B. wenn ein weiterer Nutzer eine andere Aufteilung der Bedienoberfläche bevorzugt.

## Vorgehen

Um die Einstellungen der Bedienoberfläche auf Standardeinstellungen zurückzusetzen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein".
3. Klicken Sie unter "Auf Standard zurücksetzen > Editoren-Layout" auf die Schaltfläche "Auf Standard zurücksetzen".

## Ergebnis

Die Standardeinstellungen für die Bedienoberfläche werden wiederhergestellt.

## Siehe auch

Übersicht der Programmeinstellungen (Seite 171)

Speichern des Bedienoberflächenlayouts (Seite 203)

## 6.1.3 Tastaturbedienung im TIA-Portal

### 6.1.3.1 Bedienung des TIA-Portals mit der Tastatur

Sie können sich durch das TIA-Portal mit Hilfe der Tastatur bewegen, wenn Sie zum Beispiel gerade keine Maus zur Verfügung haben. Viele Funktionen sind zusätzlich über eine Tastenkombination zu erreichen. Eine Übersicht aller Tastenkombinationen finden Sie in den Einstellungen des TIA-Portals.

In den folgenden Kapiteln erfahren Sie, wie Sie sich mit Hilfe der Tastatur im TIA-Portal bewegen, Objekte bearbeiten und das TIA-Portal an Ihre Bedürfnisse anpassen können.

## Siehe auch

Übersicht aller Tastenkombinationen anzeigen (Seite 207)

### 6.1.3.2 Übersicht aller Tastenkombinationen anzeigen

Sie können sich eine Übersicht aller Tastenkombinationen anzeigen lassen.

## Vorgehen

Um eine Übersicht aller möglichen Tastenkombinationen anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Die Einstellungen des TIA-Portals werden angezeigt.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Tastaturbedienung".  
Sie sehen eine Übersicht aller Tastenkombinationen, die für die aktuell installierten Produkte gültig ist.

### 6.1.3.3 Grundfunktionen des TIA-Portals

Im Folgenden erfahren Sie, wie Sie grundlegende Funktionen des TIA-Portals ausschließlich mit der Tastatur nutzen können.

### Grundfunktionen des TIA-Portals mit der Tastatur nutzen

Die folgende Tabelle zeigt, wie sie mit Tastenkombinationen auf Grundfunktionen des TIA-Portals zugreifen können:

| Funktion  | Tastenkombination | Menübefehl             |
|---|-------------------|------------------------|
| Zwischen Projektansicht und Portalansicht wechseln  | <Alt+F7>          |                        |
| Hilfe öffnen<br>Wenn Sie Hilfe zum TIA-Portal benötigen, drücken Sie <F1>.                                | <F1>              | Hilfe > Hilfe anzeigen |
| Aktuelle Aktion abbrechen   | <Esc>             |                        |
| Suchen  | <Strg+F>          |                        |
| Objekt ersetzen   | <Strg+H>          |                        |
| Weitersuchen<br>Wenn Sie eine Suche gestartet haben, können Sie mit <F3> zur nächsten Fundstelle springen | <F3>              |                        |
| Objekt drucken  | <Strg+P>          | Projekt > Drucken      |

### Menüs bedienen

Die folgende Tabelle zeigt, wie Sie sich mit Hilfe der Tastatur durch Menüs bewegen können:

| Funktion  | Tastenkombination                                  |
|---|--|
| Tastaturbedienung im Menü starten<br>Das Menü erreichen Sie mit der <Alt>-Taste und können anschließend weiter mit den Pfeiltasten durch das Menü navigieren. Die Auswahl eines Menübefehls bestätigen Sie mit <Return>.  | <Alt>  |
| Direkt zu einem bestimmten Menü springen<br>Sie können direkt zu einem einzelnen Menüeintrag springen, indem Sie die <Alt>-Taste gedrückt halten. Bei jedem Menüeintrag wird ein Buchstabe unterstrichen. Drücken Sie zusätzlich zur <Alt>-Taste den unterstrichenen Buchstaben.  | <Alt+unterstrichener Buchstabe im jeweiligen Menü> |
| Kontextmenü eines Objekts öffnen<br>Mit der Kontextmenütaste (auf Microsoft Windows kompatiblen Tastaturen) öffnen Sie das Kontextmenü des selektierten Objekts. Alternativ benutzen Sie <Shift+F10>, wenn Sie keine Microsoft Windows kompatible Tastatur verwenden. Sie können mit den Pfeiltasten durch das Kontextmenü navigieren und einen Menübefehl mit <Return> wählen. | <Kontextmenütaste><br>Alternativ: <Shift+F10>      |

### Ausklappbare Elemente bedienen

Die folgende Tabelle zeigt, wie Sie ausklappbare Elemente mit der Tastatur bedienen können:

| Funktion   | Tastenkombination   |
|--|---------------------|
| Ordner in einer Baumstruktur aufklappen<br>Mit <Pfeil-nach-rechts> klappen Sie beispielsweise einen Ordner in der Projektnavigation auf. | <Pfeil-nach-rechts> |
| Ordner in einer Baumstruktur schließen<br>Mit <Pfeil-nach-links> klappen Sie beispielsweise einen Ordner in der Projektnavigation zu.    | <Pfeil-nach-links>  |

| Funktion   | Tastenkombination            |
|--|------------------------------|
| Klappliste öffnen<br>Klapplisten öffnen Sie mit <F4> und können anschließend mit den Pfeiltasten durch die Klappliste navigieren. Bestätigen Sie am Schluss Ihre Auswahl mit <Return>. | <F4>                         |
| Autovervollständigung öffnen   | <Strg+Leertaste><br><Strg+I> |
| Objektauswahl anzeigen   | <Strg+J>                     |

### 6.1.3.4 Nutzen projektbezogener Funktionen

#### Projekt bearbeiten

| Funktion                              | Tastenkombination | Menübefehl                |
|---------------------------------------|-------------------|---------------------------|
| Projekt öffnen                        | <Strg+O>          | Projekt > Öffnen          |
| Projekt schließen                     | <Strg+W>          | Projekt > Schließen       |
| Projekt speichern                     | <Strg+S>          | Projekt > Speichern       |
| Projekt unter anderem Namen speichern | <Strg+Shift+S>    | Projekt > Speichern unter |
| Projekt löschen                       | <Strg+E>          | Projekt > Projekt löschen |
| Projekt drucken                       | <Strg+P>          | Projekt > Drucken         |
| Letzte Aktion rückgängig machen       | <Strg+Z>          | Bearbeiten > Rückgängig   |
| Letzte Aktion wiederholen             | <Strg+Y>          | Bearbeiten > Wiederholen  |

#### Hilfe aufrufen

| Funktion       | Tastenkombination    | Menübefehl             |
|----------------|----------------------|------------------------|
| Hilfe aufrufen | <F1> oder <Shift+F1> | Hilfe > Hilfe anzeigen |

### 6.1.3.5 Anordnung von Fenstern

Im Folgenden erfahren Sie, wie Sie mit der Tastatur einzelne Fenster des TIA-Portals öffnen und schließen können oder wie Sie mit Hilfe der Tastatur mit gespeicherten Fensterlayouts arbeiten können.

#### Fenster öffnen und schließen

Die folgende Tabelle zeigt, wie Sie mit Tastenkombinationen einzelne Fenster öffnen und schließen können:

| Funktion                           | Tastenkombination | Menübefehl                  |
|------------------------------------|-------------------|-----------------------------|
| Projektnavigation öffnen/schließen | <Strg+1>          | Ansicht > Projektnavigation |
| Detailansicht öffnen/schließen     | <Strg+4>          | Ansicht > Detailansicht     |

| Funktion  | Tastenkombination | Menübefehl                 |
|---|-------------------|----------------------------|
| Übersicht öffnen/schließen  | <Strg+2>          | Ansicht > Übersicht        |
| Task Card öffnen/schließen  | <Strg+3>          | Ansicht > Task Card        |
| Bibliotheken öffnen   | <Strg+Shift+L>    |                            |
| Hardware-Katalog öffnen<br>Wenn Sie sich in der Geräte- oder Netzsicht befinden, wird der Hardware-Katalog geöffnet.      | <Strg+Shift+C>    |                            |
| Inspektorfenster öffnen/schließen   | <Strg+5>          | Ansicht > Inspektorfenster |
| Register "Eigenschaften" im Inspektorfenster öffnen   | <Strg+6>          |                            |
| Register "Info" im Inspektorfenster öffnen  | <Strg+7>          |                            |
| Register "Diagnose" im Inspektorfenster öffnen  | <Strg+8>          |                            |
| Referenzprojekte anzeigen oder ausblenden   | <Strg+9>          |                            |
| Bildschirmtastatur anzeigen<br>Sie können eine Tastatur am Bildschirm anzeigen, z. B. für die Verwendung an Touchscreens. | <Strg+Shift+K>    |                            |
| Alle Editoren schließen   | <Strg+Shift+F4>   | Fenster > Alle schließen   |

### Gespeicherte Fensterlayouts verwenden

Sie können sich individuelle Fensteranordnungen speichern und diese zu einem späteren Zeitpunkt wiederherstellen. Die folgende Tabelle zeigt, wie Sie mit Hilfe von Tastenkombinationen auf gespeicherte Fensterlayouts zugreifen:

| Funktion  | Tastenkombination                       | Menübefehl                               |
|---|---|--|
| Aktives Fensterlayout wiederherstellen<br>Wenn Sie ein gespeichertes Fensterlayout verwenden und Sie haben zwischenzeitlich Änderungen an der Programmoberfläche vorgenommen, dann können Sie mit <Alt+Shift+0> den Ursprungszustand des aktiven Fensterlayouts wiederherstellen. | <Shift+Alt+0>                           | Fenster > Fensterlayout wiederherstellen |
| Fensterlayout laden<br>Sie können mit <Alt+Shift+[Nummer des Fensterlayouts]> eines von den ersten fünf gespeicherten Fensterlayouts aktivieren.  | <Shift+Alt+[Nummer des Fensterlayouts]> | Fenster > Fensterlayout 1 bis 5          |

#### 6.1.3.6 Navigation durch die Programmoberfläche

Das TIA-Portal untergliedert sich in verschiedene Oberflächenbereiche, wie z. B. einzelne Fenster, Funktionsleisten und Editoren. Wenn Sie innerhalb eines Oberflächenbereichs mit der Tastatur arbeiten möchten, müssen Sie diesen zunächst in den Fokus nehmen. Im Folgenden erfahren Sie, wie Sie mit der Tastatur einzelne Oberflächenbereiche fokussieren können. Außerdem erfahren Sie, wie Sie sich mit der Tastatur innerhalb eines Oberflächenbereichs im TIA-Portal bewegen.



## Wechseln zwischen Oberflächenbereichen und Editoren

Die folgende Tabelle zeigt, wie Sie zwischen einzelnen Oberflächenbereichen des TIA-Portals wechseln:

| Funktion  | Tastenkombination   |
|---|---|
| <p>Wechsel zwischen den Oberflächenbereichen im Uhrzeigersinn</p> <p>Mit der Taste &lt;F6&gt; können Sie zwischen den einzelnen Oberflächenbereichen des TIA-Portals im Uhrzeigersinn wechseln. Der Oberflächenbereich, der aktuell im Fokus steht, ist mit einer blauen Titelleiste gekennzeichnet. Befinden Sie sich beispielsweise in der Projektnavigation und drücken die Taste &lt;F6&gt;, dann springen Sie in den aktuell geöffneten Editor. Drücken Sie erneut &lt;F6&gt; und die Task Cards sind im Fokus.</p> <p>Drücken Sie hingegen &lt;Shift+F6&gt;, um entgegen dem Uhrzeigersinn zwischen den Arbeitsbereichen zu wechseln.</p> | <F6>  |
| <p>Wechsel zwischen den Oberflächenbereichen entgegen dem Uhrzeigersinn</p> <p>Mit &lt;Shift+F6&gt; wechseln Sie entgegen dem Uhrzeigersinn zwischen den Oberflächenbereichen des TIA-Portals.</p>  | <Shift+F6>  |
| <p>Wechsel zum nächsten geöffneten Editor</p> <p>Mit &lt;Strg+Alt+Pfeil-nach-rechts&gt; wechseln Sie zum nächsten geöffneten Editor. Die geöffneten Editoren sehen Sie in der Editorleiste.</p>   | <Strg+Alt+Pfeil-nach-rechts><br>Alternative: <Strg+F6>      |
| <p>Wechseln zum vorherigen geöffneten Editor</p> <p>Mit &lt;Strg+Alt+Pfeil-nach-links&gt; wechseln Sie zum zuletzt geöffneten Editor.</p>   | <Strg+Alt+Pfeil-nach-links><br>Alternative: <Strg+Shift+F6> |
| <p>Zum übergeordneten Teil des Oberflächenbereichs springen</p> <p>Mit &lt;Shift+Esc&gt; springen Sie zum übergeordneten Teil der Programmoberfläche. Wenn Sie beispielsweise ein Gerät in der Projektnavigation selektiert haben und Sie drücken &lt;Shift+Esc&gt;, dann wird die gesamte Projektnavigation in den Fokus genommen.</p>   | <Shift+Esc><br>Alternativ: <Alt+Pfeil-nach-oben>            |
| <p>Zum untergeordneten Teil des Oberflächenbereichs springen</p> <p>Mit &lt;Return&gt; wird der untergeordnete Teil der Programmoberfläche in den Fokus genommen. Wenn Sie z. B. gerade die Eigenschaften eines Geräts im Inspektorfenster geöffnet haben, um das Gerät zu parametrieren, dann gelangen Sie mit &lt;Return&gt; eine Ebene tiefer in der Programmoberfläche. Im Anschluss können Sie mit der Tabulator-Taste bis zum gewünschten Parameter navigieren.</p>   | <Return><br>Alternativ: <Alt+Pfeil-nach-unten>              |

## Navigation innerhalb von Oberflächenbereichen und Editoren

Die folgende Tabelle zeigt, wie Sie sich innerhalb eines Oberflächenbereichs mit Hilfe der Tastatur bewegen:

| Funktion  | Tastenkombination |
|---|-------------------|
| <p>Springen zum nächsten Element innerhalb eines Oberflächenbereichs</p> <p>Mit der Tabulator-Taste können Sie von einem zum nächsten Element innerhalb eines Arbeitsbereichs springen. Wenn Sie beispielsweise die Eigenschaften eines Geräts geöffnet haben und möchten von einem Eingabefeld zum nächsten springen, drücken Sie die Tabulator-Taste. Dabei werden eventuelle Änderungen am aktuellen Eingabefeld übernommen.</p> | <Tab>             |
| <p>Springen zum vorherigen Element innerhalb eines Oberflächenbereichs</p> <p>Mit &lt;Shift+Tab&gt; können Sie zum voranstehenden Element innerhalb eines Arbeitsbereichs springen, beispielsweise zu einem vorhergehenden Textfeld. Dabei werden eventuelle Änderungen am aktuellen Eingabefeld übernommen.</p>  | <Shift+Tab>       |

| Funktion   | Tastenkombination |
|--|-------------------|
| Wechsel zum nächsten Register innerhalb eines Oberflächenbereichs<br>Wenn ein Oberflächenbereich in einzelne Register unterteilt ist, können Sie zwischen den Registern mit der Tastenkombination <Strg+Tab> umschalten. Wenn Sie sich beispielsweise im Inspektorfenster im Register "Eigenschaften" befinden und Sie möchten in das Register "Info" springen, drücken Sie die Tastenkombination <Strg+Tab>.  | <Strg+Tab>        |
| Zum vorhergehenden Register wechseln<br>Mit <Strg+Shift+Tab> wechseln Sie zum zuletzt geöffneten Register innerhalb eines Oberflächenbereichs.   | <Strg+Shift+Tab>  |
| Springen zur Funktionsleiste eines Editors<br>Mit der Taste <Alt+F10> springen Sie zur Funktionsleiste eines Editors. Wenn Sie beispielsweise die Druckvorschau geöffnet haben und Sie möchten in der Funktionsleiste zur nächsten Seite des Ausdrucks wechseln, drücken Sie <Alt+F10>. Anschließend navigieren Sie mit den Pfeiltasten zum entsprechenden Symbol in der Funktionsleiste und bestätigen die Auswahl mit <Return>.  | <Alt+F10>         |
| Pfeile auf dem Fensterteiler bedienen, um Oberflächenelemente ein- oder auszublenden<br>Die Tabelle im Arbeitsbereich ist minimierbar und maximierbar. Navigieren Sie zunächst in den Arbeitsbereich und nehmen Sie mit Hilfe der Tabulator-Taste einen der kleinen Pfeile auf der Trennlinie oberhalb der Tabelle in den Fokus. Die Pfeile liegen im Fokus, sobald diese blau hinterlegt sind. Anschließend drücken Sie die Leertaste, um die Tabelle zu minimieren oder zu maximieren. | <Leertaste>       |

### 6.1.3.7 Anpassen von Editoren

Im Folgenden erfahren Sie, wie Sie mit Hilfe der Tastatur die Anordnung der Editoren ändern können. Außerdem erfahren Sie, wie Sie die Darstellungsgröße und den Ausschnitt innerhalb eines grafischen Editors wählen können.

#### Anordnen und anpassen von Editoren

Die folgende Tabelle zeigt, wie Sie geöffnete Editoren untereinander oder nebeneinander anordnen können, oder wie Sie einen geöffneten Editor schließen können:

| Funktion  | Tastenkombination | Menübefehl                                |
|---|-------------------|---|
| Aktiven Editor schließen  | <Strg+F4>         |   |
| Editorbereich vertikal teilen<br>Haben Sie beispielsweise das Übersichtsfenster und die Netzsicht geöffnet und möchten diese nebeneinander darstellen, drücken Sie die Taste <F12>.           | <F12>             | Fenster > Editorbereich vertikal teilen   |
| Editorbereich horizontal teilen<br>Sie können zwei geöffnete Editoren im Arbeitsbereich untereinander darstellen.   | <Strg+F12>        | Fenster > Editorbereich horizontal teilen |
| Fensterteilung aufheben<br>Wenn Sie zwei Editoren im geteilten Modus untereinander oder nebeneinander im Arbeitsbereich anzeigen, können Sie die Teilung mit <Alt+Shift+F12> wieder aufheben. | <Alt+Shift+F12>   | Fenster > Teilung aufheben                |

## Darstellung in einem Editor anpassen

Die folgende Tabelle zeigt, wie Sie die Darstellung innerhalb grafischer Editoren vergrößern oder verkleinern können und wie Sie den Ausschnitt eines Editors verschieben:

| Funktion   | Tastenkombination                                   |
|--|---|
| Darstellung in einem Editor stufenweise vergrößern<br>Mit <Strg> und der Taste <Plus> auf dem Ziffernblock der Tastatur vergrößern Sie die Darstellung des Editors.    | <Strg+Plus><br>Alternativ: <Strg+aufwärts scrollen> |
| Darstellung in einem Editor stufenweise verkleinern<br>Mit <Strg> und der Taste <Minus> auf dem Ziffernblock der Tastatur verkleinern Sie die Darstellung des Editors. | <Strg+Minus><br>Alternativ: <Strg+abwärts scrollen> |
| 100%-Ansicht im Editor einstellen<br>In einem grafischen Editor vergrößern oder verkleinern Sie die aktuelle Ansicht auf 100%, wenn sie <Strg+0> drücken.              | <Strg+0>  |
| Ausschnitt des Editors verschieben<br>Wenn Sie die Leertaste gedrückt halten, können Sie den angezeigten Ausschnitt eines Editors mit der Maus verschieben.            | <Leertaste>   |

### 6.1.3.8 Bearbeiten von Objekten

#### Objekte selektieren

Die folgende Tabelle zeigt, wie Sie einzelne Objekte, wie z. B. Geräte in der Projektnavigation, selektieren können:

| Funktion   | Tastenkombination | Menübefehl |
|--|-------------------|------------|
| Ein links, rechts, oberhalb oder unterhalb liegendes Objekt selektieren  | <Pfeiltasten>     |            |
| Zum ersten Objekt innerhalb des aktuellen Oberflächenbereichs springen<br>Das erste Objekt innerhalb des aktuell im Fokus liegenden Oberflächenbereichs wird selektiert. In der Projektnavigation wäre dies z. B. der Projektknoten ganz oben. | <Pos1>            |            |
| Zum letzten Objekt innerhalb des aktuellen Oberflächenbereichs springen<br>Das letzte Objekt innerhalb des aktuell im Fokus liegenden Oberflächenbereichs wird selektiert, z. B. der letzte Eintrag in der Projektnavigation.                  | <Ende>            |            |

| Funktion   | Tastenkombination                   | Menübefehl                   |
|--|-------------------------------------|------------------------------|
| Alle Objekte eines Bereichs selektieren<br>Alle Objekte, die innerhalb des aktuell im Fokus liegenden Arbeitsbereichs liegen, werden selektiert.   | <Strg+A>                            | Bearbeiten > Alles auswählen |
| Mehrere Objekte auswählen<br>Wenn Sie mehrere Objekte selektieren möchten, die nicht unmittelbar nebeneinander liegen, müssen Sie zunächst mit <Strg+Pfeiltasten> den Fokus (graue Umrandung eines Objekts) auf das nächste, gewünschte Objekt verschieben. Die aktuelle Selektion bleibt dabei erhalten. Im Anschluss drücken Sie die Leertaste, um das neu fokussierte Objekt ebenfalls zu selektieren. Diesen Vorgang wiederholen Sie bis alle gewünschten Objekte selektiert sind. | <Strg+Pfeiltasten> +<br><Leertaste> |                              |

### Objekte bearbeiten

Die folgende Tabelle zeigt eine Übersicht aller Tastenkombinationen, die für das Bearbeiten von Objekten notwendig sind:

| Funktion  | Tastenkombination                    | Menübefehl                |
|---|--------------------------------------|---------------------------|
| Neues Objekt einfügen<br>Abhängig vom Kontext, in dem Sie sich gerade befinden, wird ein neues Objekt eingefügt. Wenn Sie sich beispielsweise in der Gerätesicht befinden, wird der Dialog "Gerät hinzufügen" geöffnet, um ein neues Gerät anzulegen. | <Strg+N>                             |                           |
| Objekt öffnen   | <Return>                             |                           |
| Objekt umbenennen   | <F2>                                 | Bearbeiten > Umbenennen   |
| Objekt kopieren   | <Strg+C><br>Alternativ: <Strg+Einf>  | Bearbeiten > Kopieren     |
| Objekt ausschneiden   | <Strg+X><br>Alternativ: <Shift+Entf> | Bearbeiten > Ausschneiden |
| Objekt einfügen   | <Strg+V><br>Alternativ: <Shift+Einf> | Bearbeiten > Einfügen     |
| Objekt löschen  | <Entf>                               | Bearbeiten > Löschen      |
| Objekt übersetzen   | <Strg+B>                             | Bearbeiten > Übersetzen   |
| Eigenschaften eines Objekts öffnen<br>Viele Objekte im TIA-Portal verfügen über editierbare Eigenschaften. Drücken Sie die Tastenkombination <Alt+Enter>, um die Eigenschaften eines Objekts anzuzeigen.  | <Alt+Return>                         | -                         |

#### 6.1.3.9 Textbearbeitung

Im Folgenden erfahren Sie, wie Sie Textbearbeitungsfunktionen ausschließlich mit der Tastatur nutzen können.

## Texte editieren

Die folgende Tabelle zeigt grundlegende Editierfunktionen für Text:

| Funktion  | Tastenkombination |
|---|-------------------|
| Einfüge- bzw. Überschreibmodus umschalten   | <Einfg>           |
| Editiermodus beenden  | <Esc>             |
| Löschen   | <Entf>            |
| Löschen von Zeichen   | <Backspace>       |
| Eingabe in einem Eingabefeld bestätigen und Eingabefeld verlassen   | <Return>          |
| Zeilenumbruch in einem mehrzeiligen Eingabefeld<br>In einem mehrzeiligen Eingabefeld müssen Sie die <Shift>-Taste gedrückt halten, um einen Zeilenumbruch zu erzeugen.                                    | <Shift+Return>    |
| Eingaben in einem Eingabefeld zurücksetzen<br>Wenn Sie sich in einem Eingabefeld befinden und Sie drücken <Esc>, dann verlassen Sie das Eingabefeld und durchgeführte Änderungen werden nicht übernommen. | <Esc>             |

## Navigation innerhalb eines Textbereichs

Die folgende Tabelle zeigt, wie Sie mit der Tastatur innerhalb eines Textbereichs navigieren:

| Funktion  | Tastenkombination |
|---|-------------------|
| Zum Zeilenanfang springen   | <Pos1>            |
| Zum Zeilenende springen   | <Ende>            |
| Zum Textanfang springen   | <Strg+Pos1>       |
| Zum Textende springen   | <Strg+Ende>       |
| Zur vorhergehenden Seite springen   | <Bild Auf>        |
| Zur nachfolgenden Seite springen  | <Bild Ab>         |
| Eingabe in einem Eingabefeld bestätigen und Eingabefeld verlassen   | <Return>          |
| Zeilenumbruch in einem mehrzeiligen Eingabefeld   | <Shift+Return>    |
| Eingaben in einem Eingabefeld zurücksetzen<br>Wenn Sie sich in einem Eingabefeld befinden und Sie drücken <Esc>, dann verlassen Sie das Eingabefeld und durchgeführte Änderungen werden nicht übernommen. | <Esc>             |

## Text selektieren

Die folgende Tabelle zeigt, wie Sie mit der Tastatur Text selektieren:

| Funktion  | Tastenkombination                                    |
|---|--|
| Auswahl auf das links oder rechts liegende Wort erweitern<br>Der Text oder die bestehende Textauswahl wird bis zum Ende des Wortes ausgewählt. Wenn Sie sich am Anfang oder Ende eines Wortes befinden, wird das vorhergehende oder nachfolgende Wort ausgewählt. | <Strg+Shift+Pfeil-nach-links oder Pfeil-nach-rechts> |
| Auswahl bis zum Zeilenanfang erweitern  | <Shift+Pos1>   |
| Auswahl bis zum Zeilenende erweitern  | <Shift+Ende>   |

| Funktion   | Tastenkombination |
|--|-------------------|
| Auswahl bis zum Textanfang erweitern<br>Der Text wird bis zum Anfang oder bis zum Ende hin ausgewählt. | <Strg+Shift+Pos1> |
| Auswahl bis zum Textende erweitern<br>Der Text wird bis zum Anfang oder bis zum Ende hin ausgewählt.   | <Strg+Shift+Ende> |

### 6.1.3.10 Bearbeiten von Tabellen

Im Folgenden erfahren Sie, wie Sie mit der Tastatur in Tabellen navigieren, einzelne Felder bearbeiten und Teile von Tabellen selektieren können.

#### Allgemeine Tastaturbedienung in Tabellen

Die folgende Tabelle zeigt, wie Sie Tabellen ausschließlich mit Hilfe der Tastatur bearbeiten können:

| Funktion   | Tastenkombination  |
|--|--------------------|
| Zelle in den Editiermodus setzen   | <F2> oder <Return> |
| Eingabe bestätigen und Editiermodus beenden  | <Return>           |
| Editieren abbrechen und Änderungen verwerfen   | <Esc>              |
| Klappliste in einer Zelle öffnen<br>Mit <F4> öffnen Sie die Klappliste. Mit den Pfeiltasten wählen Sie anschließend den gewünschten Eintrag und bestätigen die Auswahl mit <Return>. | <F4>               |
| Klappliste in einer Zelle schließen und Änderungen verwerfen   | <Esc>              |

#### In Tabellen navigieren

Die folgende Tabelle zeigt, wie Sie sich mit Hilfe der Tastatur innerhalb einer Tabelle bewegen:

| Funktion                                    | Tastenkombination       |
|---|-------------------------|
| Zur nächsten Zelle                          | <Pfeiltasten>           |
| Zur nächsten editierbaren Zelle nach rechts | <Tab>                   |
| Zur nächsten editierbaren Zelle nach links  | <Shift+Tab>             |
| Eine Bildschirmseite nach oben              | <Bild Auf>              |
| Eine Bildschirmseite nach unten             | <Bild Ab>               |
| Zur ersten Zelle in der Zeile               | <Pos1>                  |
| Zur letzten Zelle in der Zeile              | <Ende>                  |
| Zur ersten Zelle der Tabelle                | <Strg+Pos1>             |
| Zur letzten Zelle der Tabelle               | <Strg+Ende>             |
| Zur obersten Zelle in der Spalte            | <Strg+Pfeil-nach-oben>  |
| Zur untersten Zelle in der Spalte           | <Strg+Pfeil-nach-unten> |

### Bereiche in Tabellen selektieren

Die folgende Tabelle zeigt, wie Sie mit der Tastatur Bereiche innerhalb einer Tabelle selektieren können:

| Funktion   | Tastenkombination              |
|--|--------------------------------|
| Spalte selektieren                                     | <Strg+Leertaste>               |
| Zeile selektieren                                      | <Shift+Leertaste>              |
| Alle Zellen selektieren                                | <Strg+A>                       |
| Selektion um eine Zelle erweitern                      | <Shift+Pfeiltasten>            |
| Selektion eine Seite nach oben erweitern               | <Shift+Bild Auf>               |
| Selektion eine Seite nach unten erweitern              | <Shift+Bild Ab>                |
| Selektion bis zur ersten Zeile erweitern               | <Strg+Shift+Pfeil-nach-oben>   |
| Selektion bis zur letzten Zeile erweitern              | <Strg+Shift+Pfeil-nach-unten>  |
| Selektion bis zur ersten Zelle in der Zeile erweitern  | <Strg+Shift+Pfeil-nach-links>  |
| Selektion bis zur letzten Zelle in der Zeile erweitern | <Strg+Shift+Pfeil-nach-rechts> |

### 6.1.3.11 Nutzen von Online-Funktionen

#### Online-Funktionen mit der Tastatur steuern

Die folgende Tabelle zeigt eine Übersicht der Tastenkombinationen, mit denen Sie die Online-Funktionen des TIA-Portals nutzen können:

| Funktion  | Tastenkombination | Menübefehl                               |
|---|-------------------|--|
| Online-Verbindung herstellen  | <Strg+K>          | Online > Online verbinden                |
| Online-Verbindung trennen   | <Strg+M>          | Online > Online-Verbindung trennen       |
| Projektdatei in das Gerät laden   | <Strg+L>          | Online > Laden in Gerät                  |
| Erreichbare Teilnehmer anzeigen<br>Es wird ein Dialog geöffnet, in dem alle Geräte angezeigt werden, die mit der PG/PC-Schnittstelle des PG/PCs verbunden sind.                             | <Strg+U>          | Online > Erreichbare Teilnehmer anzeigen |
| CPU starten<br>Die CPU wird in den Betriebszustand "RUN" versetzt. Die CPU muss dazu online verbunden sein.   | <Strg+Shift+E>    | Online > CPU starten                     |
| CPU stoppen<br>Die CPU wird in den Betriebszustand "STOP" versetzt. Die CPU muss dazu online verbunden sein.  | <Strg+Shift+Q>    | Online > CPU stoppen                     |
| Simulation starten<br>Die Hardware und Software des Projektes kann in einer simulierten Online-Umgebung getestet werden, ohne dass die Baugruppen tatsächlich online verbunden sein müssen. | <Strg+Shift+X>    | Online > Simulation > Starten            |

### 6.1.3.12 Bildschirmtastatur verwenden

#### Einführung

Für die Bedienung des TIA-Portals steht Ihnen zusätzlich die Microsoft Bildschirmtastatur zur Verfügung.

#### Bildschirmtastatur anzeigen

Um die Bildschirmtastatur anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Ansicht" den Befehl "Bildschirmtastatur".

#### Bildschirmtastatur beenden

Um die Bildschirmtastatur zu beenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Datei" der Bildschirmtastatur den Befehl "Beenden".

## 6.1.4 Betriebssystemabhängige Besonderheiten

### 6.1.4.1 Einfluss von Benutzerrechten

#### Einschränkungen bei limitierten Benutzerrechten

Die Software bietet einige Funktionen, welche direkten Zugriff auf die Hardware des PG/PC und damit auf das installierte Betriebssystem erfordern. Um den vollen Funktionsumfang nutzen zu können, muss die Software eng mit dem Betriebssystem zusammenarbeiten. Für eine reibungslose Interaktion sollten Sie daher mit ausreichenden Benutzerrechten am Betriebssystem angemeldet sein.

Insbesondere Funktionen, die eine Online-Verbindung erfordern oder die Einstellungen von Schnittstellenkarten ändern, sind möglicherweise nicht vollständig nutzbar, wenn Sie mit eingeschränkten Benutzerrechten arbeiten.



## Erkennen von eingeschränkten Funktionen

Funktionen, die spezielle Rechte erfordern, können Sie folgendermaßen erkennen:

- Ein Schild-Symbol wird neben der Funktion angezeigt.



Die Funktion ist bedienbar, wird allerdings durch die Benutzerkontensteuerung reglementiert.

- Ein Feld ist grau und nicht bedienbar.  
Für das Feld benötigen Sie zwingend Administrationsrechte. In manchen Betriebssystemumgebungen können Sie kurzfristig Administrationsrechte durch die Eingabe eines Administratorkeywords erlangen.

---

### Hinweis

Ist ein Feld grau, ist dies nicht in jedem Fall durch fehlende Rechte bedingt. Beachten Sie zusätzliche Informationen in den Tooltip-Kaskaden, um die Bedingungen für das Bearbeiten zu erfahren.

---

## 6.1.4.2 Benutzerrechte erweitern

### Beschränkungen durch Benutzerrechte aufheben

Bestimmte Funktionen sind möglicherweise nicht bedienbar, wenn Sie nicht mit ausreichenden Benutzerrechten am Betriebssystem angemeldet sind. Um etwaige Beschränkungen aufzuheben, gibt es die folgenden Möglichkeiten:

- Freigeben von erweiterten Rechten mit Hilfe der Windows-Benutzerkontensteuerung
- Anmelden am Betriebssystem mit Administrationsrechten
- Nutzen vorübergehender Administrationsrechte

### Erweiterte Rechte mit Hilfe der Windows-Benutzerkontensteuerung freigeben

Um eine Funktion zu bedienen, die mit dem Schild-Symbol der Windows-Benutzerkontensteuerung versehen ist, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Feld oder die Schaltfläche mit dem Schild-Symbol.  
Die Sicherheitsabfrage der Windows-Benutzerkontensteuerung wird geöffnet.
2. Folgen Sie den Anweisungen der Windows-Benutzerkontensteuerung und geben Sie auf Nachfrage gegebenenfalls ein Administratorkeyword ein.

Die Funktion ist nun einmalig ohne Einschränkungen nutzbar.

### Am Betriebssystem mit Administrationsrechten anmelden

Um eine Funktion zu benutzen, die aufgrund fehlender Benutzerrechte deaktiviert ist, gehen Sie folgendermaßen vor:

1. Schließen Sie die Software.
2. Melden Sie sich am Betriebssystem ab.
3. Melden Sie sich mit Administrationsrechten am Betriebssystem an.
4. Starten Sie die Software erneut.

### Nutzen vorübergehender Administrationsrechte

Um vorübergehend Administrationsrechte zu erhalten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Einstellungen ändern". Die Schaltfläche ist in Dialogen zu finden, die das vorübergehende Zuweisen von Administrationsrechten erlauben. Ein Dialog des Betriebssystems zur Eingabe eines Administratorkennworts wird geöffnet.
2. Geben Sie ein Administratorkennwort ein.

Die Einstellungen können vorübergehend geändert werden. Bei erneutem Aufruf des Dialogs müssen Sie das Vorgehen wiederholen.

---

#### Hinweis

Diese Funktion wird nicht von jedem Betriebssystem unterstützt. Wenn keine Schaltfläche "Einstellungen ändern" vorhanden ist oder wenn die Schaltfläche gegraut ist, dann müssen Sie sich stattdessen am Betriebssystem mit Administrationsrechten anmelden.

---

## 6.2 Hilfe zum Informationssystem

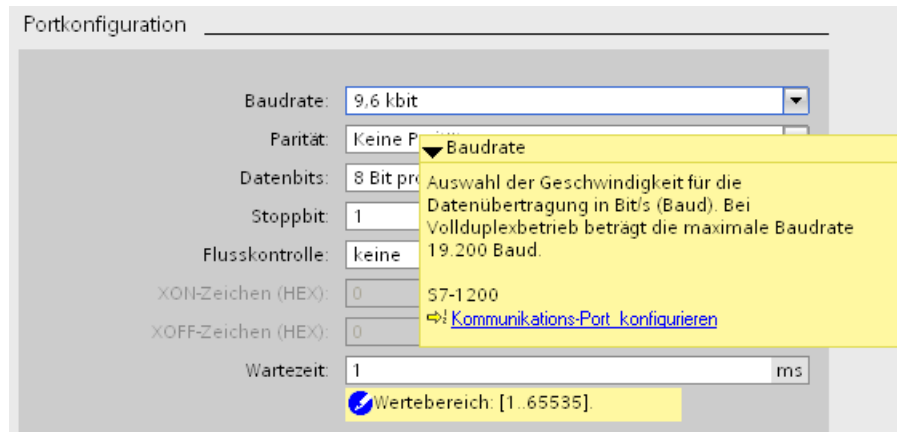
### 6.2.1 Allgemeines zum Informationssystem

#### Schnelle Antworten auf Ihre Fragen

Zur Lösung Ihrer Aufgaben steht Ihnen eine umfassende Hilfe zur Verfügung, die grundlegende Konzepte, Handlungen und Funktionen beschreibt. Während der Arbeit mit dem Programm erhalten Sie zusätzlich folgende Unterstützung:

- Roll-out für korrekte Eingaben in Dialogfelder
- Tooltip für die Informationen zu Oberflächenelementen, wie z. B. Eingabefeldern, Schaltflächen und Symbolen. Tooltips sind zum Teil durch Kaskaden mit genaueren Informationen ergänzt.
- Hilfe zum aktuellen Kontext, z. B. zu Menübefehlen erhalten Sie, wenn Sie die Tasten <F1> oder <Shift+F1> drücken.

Das folgende Bild zeigt ein Beispiel für einen Tooltip mit Kaskade (oben) und für einen Roll-out (unten):



## Hilfe





In der Hilfe sind Konzepte, Handlungen und Funktionen beschrieben. Außerdem sind Referenzinformationen und Beispiele enthalten. Die Hilfe wird in einem eigenen Fenster geöffnet.

Auf der linken Seite des Hilfefensters wird ein Navigationsbereich angezeigt. Den Navigationsbereich können Sie auch ausblenden, um Platz auf dem Bildschirm zu schaffen. Im Navigationsbereich stehen Ihnen die folgenden Funktionen zur Verfügung:

- Inhaltsverzeichnis
- Suche im Index
- Volltextsuche über die gesamte Hilfe
- Favoriten

## Kennzeichnung der Themen in der Hilfe nach dem Informationstyp

Die Hilfethemen sind abhängig vom Informationstyp mit unterschiedlichen Symbolen gekennzeichnet.

| Symbol  | Informationstyp    | Erklärung   |
|---|--------------------|---|
|  | Handlungsanweisung | Beschreibt, wie Sie schrittweise vorgehen, um eine bestimmte Aufgabe durchzuführen.             |
|  | Beispiel           | Enthält ein konkretes Anwendungsbeispiel, um die Aufgabe zu erläutern.                          |
|  | Sachverhalt        | Enthält Hintergrundinformationen, die Sie wissen müssen, um eine Aufgabe durchführen zu können. |
|  | Referenz           | Enthält umfassende Referenzinformationen zum Nachschlagen.                                      |

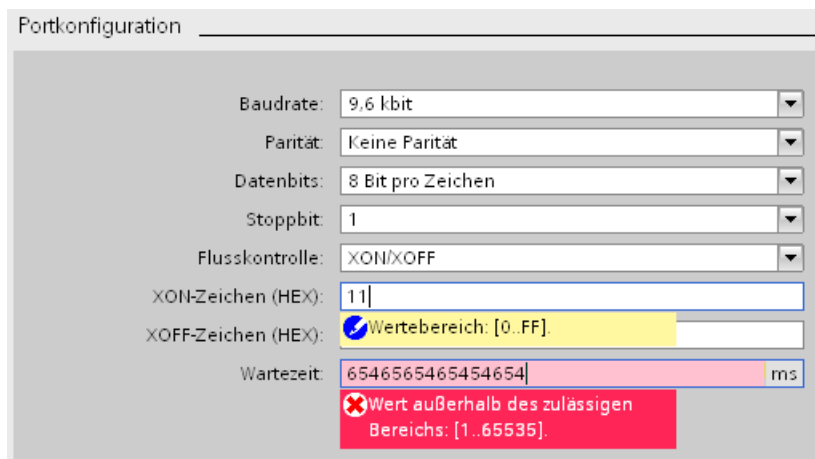
### Kennzeichnung der Themen in der Hilfe nach dem Zielsystem

Je nachdem, welche Produkte installiert sind, enthält die Hilfe möglicherweise Kapitel, die nur für bestimmte Geräte gültig sind. Um solche Kapitel auf den ersten Blick zu erkennen, finden Sie einen entsprechenden Hinweis in Klammern innerhalb des Inhaltsverzeichnisses. Die Suchergebnisse in der Volltextsuche und im Index sind auf die gleiche Weise gekennzeichnet, wenn sie nur für bestimmte Geräte gültig sind.

### Roll-out

Einige Eingabefelder bieten eine Information, die "herausrollt" und Ihnen hilft, gültige Parameter und Werte einzugeben. Das Roll-out informiert Sie über die zulässigen Wertebereiche und Datentypen der Eingabefelder.

Das folgende Bild zeigt ein Roll-out (gelb) und eine Roll-out-Fehlermeldung (rot), die Sie auf einen ungültigen Wert hinweist:



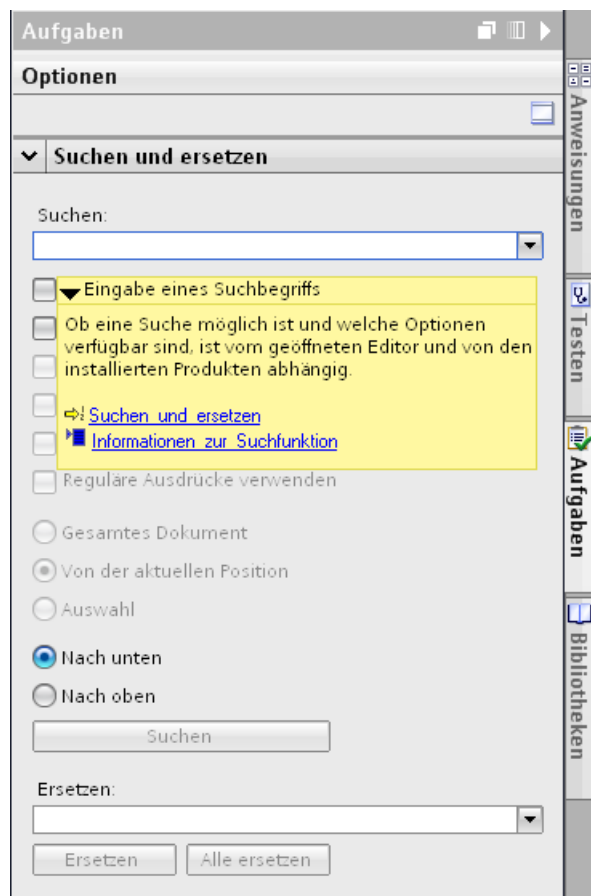
### Tooltip

Oberflächenelemente bieten Ihnen einen Tooltip zur leichteren Identifikation.

Tooltips, die links ein Pfeilsymbol haben, enthalten noch zusätzliche Informationen in Tooltip-Kaskaden. Verweilen Sie etwas länger mit dem Mauszeiger über dem Tooltip oder klicken Sie auf das Pfeilsymbol, so wird diese Information angezeigt. Die automatische Anzeige von Tooltip-Kaskaden kann deaktiviert werden.

Sind in der Hilfe weiterführende Informationen, erscheint ein Link zum entsprechenden Hilfethema in der Kaskade. Wenn Sie auf den Link klicken, dann wird das passende Thema in der Hilfe geöffnet.

Das folgende Bild zeigt einen Tooltip mit geöffneter Kaskade:



## Siehe auch

Anzeige von Tooltips und Tooltip-Kaskaden konfigurieren (Seite 227)

## 6.2.2 Hilfe öffnen

### Hilfe öffnen

Sie haben folgende Möglichkeiten, um die Hilfe zu öffnen:

1. Wählen Sie im Menü "Hilfe" den Befehl "Hilfe anzeigen" oder drücken Sie <F1>, um die passende Hilfe zum jeweiligen Kontext anzuzeigen.

oder

1. Klicken Sie auf den Link innerhalb einer Tooltip-Kaskade, um direkt zu einer weiterführenden Stelle in der Hilfe zu gelangen.

### 6.2.3 Hilfe nach Stichwörtern durchsuchen

#### Suche nach Stichwörtern im Hilfetext

Um die Hilfethemen nach vordefinierten Stichwörtern zu durchsuchen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Hilfe auf die Schaltfläche "Inhaltsverzeichnis ein-/ausblenden", um das Inhaltsverzeichnis einzublenden.  
Das Inhaltsverzeichnis wird eingeblendet und die Register "Index", "Suchen" und "Favoriten" sind sichtbar.
2. Öffnen Sie das Register "Index".
3. Geben Sie den Suchbegriff in das Eingabefeld ein oder wählen Sie den Suchbegriff aus der Stichwortliste aus.
4. Klicken Sie auf "Anzeigen".

### 6.2.4 Im Volltext suchen

#### Im Volltext suchen

Um den gesamten Text der Hilfe nach bestimmten Wörtern zu durchsuchen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Hilfe auf die Schaltfläche "Inhaltsverzeichnis ein-/ausblenden", um das Inhaltsverzeichnis einzublenden.  
Das Inhaltsverzeichnis wird eingeblendet und die Register "Index", "Suchen" und "Favoriten" sind sichtbar.
2. Öffnen Sie das Register "Suchen".
3. Geben Sie den Suchbegriff im Eingabefeld ein.
4. Verfeinern Sie Ihre Suche gegebenenfalls durch zusätzliche Kriterien:
  - Wählen Sie "Vorherige Ergebnisse suchen", um einen weiteren Suchdurchlauf nur innerhalb Ihrer letzten Suchergebnisse zu starten.
  - Wählen Sie "Ähnliche Wörter suchen", um auch solche Wörter zu finden, die von Ihrem Suchbegriff geringfügig abweichen.
  - Wählen Sie "Nur Titel suchen", um ausschließlich Ergebnisse zu erhalten, die Ihren Suchbegriff im Titel enthalten. Der Inhalt der Hilfethemen wird bei der Suche nicht berücksichtigt.

5. Klicken Sie auf die Pfeiltaste rechts des Suchfeldes, um logische Verknüpfungen zu benutzen. Die folgenden logischen Verknüpfungen stehen zur Verfügung:
  - Verbinden Sie zwei oder mehrere Suchbegriffe mit dem Operator "AND", um nur Hilfethemen zu finden, die alle gesuchten Wörter im Text enthalten.
  - Verbinden Sie zwei oder mehrere Suchbegriffe mit dem Operator "OR", um nur Hilfethemen zu finden, die ein oder mehrere Suchbegriffe im Text enthalten.
  - Verbinden Sie zwei oder mehrere Suchbegriffe mit dem Operator "NEAR", um nur Hilfethemen zu finden, in denen die Suchbegriffe in unmittelbarem Zusammenhang (acht Wörter) stehen.
  - Stellen Sie einem Wort den Operator "NOT" voran, um Hilfethemen aus der Suche auszuschließen, die dieses Wort enthalten.
6. Klicken Sie auf "Themen auflisten", um die Suche zu starten.  
Die Ergebnisse werden nun mit Titel, Position und Rang aufgelistet. In der Spalte "Position" wird das Kapitel angezeigt, in dem das gefundene Hilfe-Thema zu finden ist. Die Sortierung nach Rang erfolgt entsprechend der Position der gefundenen Hilfethemen im Inhaltsverzeichnis und nach der Häufigkeit der Treffer in den Hilfethemen.

## 6.2.5 Favoriten verwenden

### Favoriten verwenden

Sie können einzelne Hilfethemen als Favoriten abspeichern. Dadurch ersparen Sie sich eine erneute Suche nach dem Hilfethema.

### Favoriten speichern:

Um eine Seite als Favorit abzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Hilfethema oder das Kapitel, das Sie als Favorit speichern möchten.
2. Klicken Sie in der Funktionsleiste der Hilfe auf die Schaltfläche "Inhaltsverzeichnis ein-/ausblenden", um das Inhaltsverzeichnis einzublenden.  
Das Inhaltsverzeichnis wird eingeblendet und die Register "Index", "Suchen" und "Favoriten" sind sichtbar.
3. Öffnen Sie das Register "Favoriten".
4. Klicken Sie auf die Schaltfläche "Hinzufügen".  
Das Hilfethema oder das Kapitel wird als Favorit gespeichert und steht Ihnen auch beim erneuten Öffnen der Hilfe zur Verfügung.

### Favoriten aufrufen:

Um eine Seite aus den Favoriten abzurufen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Hilfe auf die Schaltfläche "Inhaltsverzeichnis ein-/ausblenden", um das Inhaltsverzeichnis einzublenden.  
Das Inhaltsverzeichnis wird eingeblendet und die Register "Index", "Suchen" und "Favoriten" sind sichtbar.
2. Öffnen Sie das Register "Favoriten".
3. Wählen Sie das Thema aus der Liste aus, das Sie öffnen möchten.
4. Klicken Sie auf die Schaltfläche "Anzeigen".

### Favoriten löschen

Um einen Eintrag aus den Favoriten zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Hilfe auf die Schaltfläche "Inhaltsverzeichnis ein-/ausblenden", um das Inhaltsverzeichnis einzublenden.  
Das Inhaltsverzeichnis wird eingeblendet und die Register "Index", "Suchen" und "Favoriten" sind sichtbar.
2. Öffnen Sie das Register "Favoriten".
3. Wählen Sie das Thema aus der Liste aus, das Sie entfernen möchten.
4. Klicken Sie auf die Schaltfläche "Entfernen".

## 6.2.6 Hilfethemen drucken

### Informationen drucken

Sie können entweder alle Inhalte der Hilfe oder einzelne Themen daraus drucken.

### Vorgehen

Um die Themen festzulegen, die Sie drucken möchten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Drucken-Dialog anzeigen".  
Das Inhaltsverzeichnis wird in einem eigenen Fenster geöffnet.
2. Aktivieren Sie im Dialog "Hilfethemen drucken" die Optionskästchen für die Ordner und Hilfethemen, die gedruckt werden sollen.
3. Klicken Sie auf die Schaltfläche "Drucken", um die markierten Informationen zu drucken.  
Der Dialog "Drucken" wird geöffnet.
4. Wählen Sie den Drucker aus, auf dem Sie die Hilfethemen ausdrucken möchten.
5. Klicken Sie auf "Eigenschaften", falls Sie weitere Druckereinstellungen angeben möchten.
6. Bestätigen Sie Ihre Eingaben mit "OK".  
Die Hilfethemen werden auf dem ausgewählten Drucker gedruckt.



## 6.2.7 Anzeige von Tooltips und Tooltip-Kaskaden konfigurieren

### Konfigurationsmöglichkeiten für Tooltips und Tooltip-Kaskaden

Sie können die Anzeige von Tooltips und Tooltip-Kaskaden an Ihre Bedürfnisse anpassen. Sie haben folgende Einstellungsmöglichkeiten:

- Anzeige abgeschnittener Texten ein- oder ausblenden  
Manchmal können Texte zu lang für ein Textfeld sein. Die Texte werden dann in einem Tooltip vollständig angezeigt, wenn Sie mit der Maus über dem Textfeld verweilen. Sie können diese Funktion aktivieren oder deaktivieren.
- Tooltips aktivieren oder deaktivieren  
Tooltips geben genauere Informationen zu einem Oberflächenelement. Außerdem ermöglichen Sie die Anzeige von Tooltip-Kaskaden. Wenn Sie die Tooltips deaktivieren, werden daher auch die Kaskaden mit kontextbezogener Hilfe nicht mehr angezeigt. Sie haben jedoch die Möglichkeit, den Tooltip für das gerade aktive Oberflächenelement manuell anzuzeigen, indem Sie <F1> drücken.
- Das automatische Öffnen von Tooltip-Kaskaden aktivieren oder deaktivieren  
Nach einer kurzen Verweildauer mit der Maus über einem Tooltip werden eventuell vorhandene Kaskaden automatisch angezeigt. Sie können die automatische Anzeige der Kaskaden aktivieren oder deaktivieren. Bei deaktivierter automatischer Anzeige müssen Sie die Kaskade gegebenenfalls manuell öffnen. Klicken Sie dazu auf das Pfeilsymbol innerhalb des Tooltips.

### Vorgehen

Um Anzeige von Tooltips und Tooltip-Kaskaden zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein".
3. Aktivieren oder deaktivieren Sie die einzelnen Optionskästchen im Bereich "Tooltips" entsprechend Ihren Bedürfnissen. Das Optionskästchen "Kaskade in den Tooltips automatisch öffnen" ist nur bedienbar, wenn Sie die Anzeige der Tooltips aktiviert haben.

### Siehe auch


Allgemeines zum Informationssystem (Seite 220)


## 6.2.8 Sicherheitshinweise


### Sicherheitshinweise

In der Hilfe finden Sie Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne

Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

|   |
|---|
|  <b>GEFAHR</b>                                     |
| bedeutet, dass Tod oder schwere Körperverletzung eintreten wird, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden. |

|   |
|---|
|  <b>WARNUNG</b>                                    |
| bedeutet, dass Tod oder schwere Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden. |

|   |
|---|
|  <b>VORSICHT</b>   |
| mit Warndreieck bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden. |

|  |
|--|
| <b>ACHTUNG</b>   |
| ohne Warndreieck bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden. |

---

**Hinweis**

bedeutet, dass ein unerwünschtes Ergebnis oder Zustand eintreten kann, wenn der entsprechende Hinweis nicht beachtet wird.

---


Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

**Qualifiziertes Personal**

Das zugehörige Gerät/System darf nur in Verbindung mit dieser Dokumentation eingerichtet und betrieben werden. Inbetriebsetzung und Betrieb eines Gerätes/Systems dürfen nur von qualifiziertem Personal vorgenommen werden. Qualifiziertes Personal im Sinne der sicherheitstechnischen Hinweise dieser Dokumentation sind Personen, die die Berechtigung haben, Geräte, Systeme und Stromkreise gemäß den Standards der Sicherheitstechnik in Betrieb zu nehmen, zu erden und zu kennzeichnen.

## Bestimmungsgemäßer Gebrauch

Beachten Sie Folgendes:

|   |
|---|
|  <b>WARNUNG</b>  |
| Das Gerät darf nur für die im Katalog und in der technischen Beschreibung vorgesehenen Einsatzfälle und nur in Verbindung mit von Siemens empfohlenen bzw. zugelassenen Fremdgeräten und -komponenten verwendet werden. Der einwandfreie und sichere Betrieb des Produktes setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung und Montage sowie sorgfältige Bedienung und Instandhaltung voraus. |

## Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

## Haftungsausschluss

Wir haben den Inhalt der Hilfe auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Hilfe werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

## 6.2.9 Individuelle Dokumentation zusammenstellen

### Individualisierte Dokumentation

Im Service- und Support-Bereich der Siemens-Internetseite können Sie sich eine individuelle und an Ihre Bedürfnisse angepasste Dokumentation zusammenstellen. Dafür stehen Ihnen alle konfigurierbaren Handbücher und Betriebsanleitungen des Service- und Support-Bereichs zur Verfügung. Daraus entnehmen Sie die für Sie interessanten Teile und kombinieren diese zu einer persönlichen Dokumentation in einer Bibliothek. Die Dokumentation strukturieren Sie mit Hilfe von Ordnern in der Bibliothek. Die Ordner werden später zu den einzelnen Kapiteln der individuellen Dokumentation.

Ihre persönliche Bibliothek rufen Sie hier auf (<https://www.automation.siemens.com/mdm/?guiLanguage=de>).

### Voraussetzung

- Die verwendeten Handbücher oder Betriebsanleitungen müssen konfigurierbar sein. Konfigurierbare Handbücher erkennen Sie am Namenszusatz "konfigurierbar".
- Um den vollen Funktionsumfang nutzen zu können, müssen Sie im Siemens Support Portal registriert und angemeldet sein.

### **Dokumentation in verschiedenen Sprachen**

Die zusammengestellte Dokumentation können Sie bei Bedarf in die Sprachen Englisch, Französisch, Spanisch, Italienisch und Chinesisch umschalten. So haben Sie beispielsweise die Möglichkeit relevante Informationen für ein bestimmtes Projekt zusammenzutragen und diese anderssprachigen Kollegen zur Verfügung stellen.

### **Exportfunktion innerhalb der Dokumentation**

In Ihrer Bibliothek können Sie an jeder beliebigen Stelle Teile in unterschiedlichen Formaten (PDF, XML, RTF) exportieren.

### **Hilfe zum Erstellen der Dokumentation**

Weitere Hilfe zum Erstellen und zur Nutzung der individuellen Dokumentation erhalten Sie auf der Service- und Support-Seite ([https://www.automation.siemens.com/mdm/help/de/mdm\\_reference\\_manual\\_de-DE.htm](https://www.automation.siemens.com/mdm/help/de/mdm_reference_manual_de-DE.htm)).

# Projekte bearbeiten

## 7.1 Grundlagen zu Projekten

### Einführung

Projekte dienen zur geordneten Ablage der Daten und Programme, die bei der Erstellung einer Automatisierungslösung entstehen. Die in einem Projekt zusammengefassten Daten enthalten insbesondere:

- Konfigurationsdaten über den Hardware-Aufbau und Parametrierungsdaten für Baugruppen
- Projektierungsdaten für die Kommunikation über Netze
- Projektierungsdaten für die Geräte
- Protokolle für wichtige Ereignisse im Lebenszyklus des Projekts

### Projekthierarchie

Daten werden in einem Projekt in Form von Objekten abgelegt. Die Objekte sind innerhalb eines Projekts in einer Baumstruktur (Projekthierarchie) angeordnet.

Die Basis der Projekthierarchie bilden die Geräte und Stationen und deren zugehörige Projektierungsdaten und Programme.

Zusätzlich werden in der Projektnavigation z. B. gemeinsame Daten des Projekts und die Online-Zugänge dargestellt.

### Siehe auch

Neues Projekt erstellen (Seite 232)

Projekte öffnen (Seite 233)

Projekte speichern (Seite 237)

Projekte löschen (Seite 239)

Protokolle verwenden (Seite 231)

## 7.2 Protokolle verwenden

Bei einigen Operationen innerhalb des TIA-Portals werden im Hintergrund automatisch Protokolle angelegt, die Änderungen am Projekt dokumentieren. Protokolle werden beispielsweise bei der Migration von Projekten und Programmen oder der Aktualisierung von Instanzen aus der Bibliothek automatisch angelegt.

Protokolle werden in der Projektnavigation im Ordner "Gemeinsame Daten" angezeigt. Sie werden zusammen mit dem Projekt im Projektordner abgespeichert und sind daher unabhängig vom verwendeten PG/PC lesbar, sobald Sie das Projekt geöffnet haben.

Neben der Anzeige im TIA-Portal ist es möglich, die Protokolle auszudrucken.

### Protokolle öffnen

Um ein Protokoll aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Gemeinsame Daten > Protokolle" in der Projektnavigation.
2. Doppelklicken Sie das gewünschte Protokoll in der Liste.  
Die Inhalte des Protokolls werden im Arbeitsbereich angezeigt.

### Protokoll löschen

Um ein Protokoll zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Protokoll in der Projektnavigation.
2. Drücken Sie die Taste <Entf>.  
Das selektierte Protokoll wird aus dem Projektverzeichnis gelöscht und aus der Projektnavigation entfernt.

## 7.3 Projekte anlegen und verwalten

### 7.3.1 Neues Projekt erstellen

#### Vorgehen

Um ein neues Projekt zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Neu".  
Der Dialog "Neues Projekt erstellen" wird geöffnet.
2. Geben Sie den gewünschten Projektnamen und Pfad ein oder übernehmen Sie die vorgeschlagenen Daten.
3. Klicken Sie auf die Schaltfläche "Erstellen".

#### Ergebnis

Das neue Projekt wird erstellt und in der Projektnavigation angezeigt.

## Siehe auch

Grundlagen zu Projekten (Seite 231)

Projekte öffnen (Seite 233)

Projekte speichern (Seite 237)

Projekte löschen (Seite 239)

## 7.3.2 Projekte öffnen

Im TIA-Portal lassen sich alle Projekte aus der aktuellen und früheren Version öffnen.

Projekte aus dem TIA-Portal V12 müssen erst in das aktuelle Projektformat hochgerüstet werden, ehe Sie damit im TIA-Portal V12 SP1 weiterarbeiten können. Sie erhalten daher beim Öffnen die Aufforderung, das Projekt hochzurüsten. Nach dem Hochrüsten des V12-Projekts auf die aktuelle Produktversion lässt sich dieses nicht mehr im TIA-Portal V12 öffnen. Eine Kopie des Originalprojekts bleibt jedoch erhalten.

Projekte aus der Version V11.x können Sie wie gewohnt öffnen. Der Funktionsumfang des TIA-Portals ist dann jedoch auf die Funktionalität des TIA-Portal V11.x beschränkt. Um mit dem vollen Funktionsumfang der aktuellen TIA-Portal-Version weiterzuarbeiten, rüsten Sie das Projekt manuell hoch.

## Vorgehen

Um ein bestehendes Projekt zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Öffnen".  
Der Dialog "Projekt öffnen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird darin angezeigt.
2. Selektieren Sie ein Projekt aus der Liste und klicken Sie auf "Öffnen".
3. Falls das gewünschte Projekt nicht in der Liste vorhanden ist, klicken Sie auf die Schaltfläche "Durchsuchen". Navigieren Sie zum gewünschten Projektordner und öffnen Sie die Projektdatei.  
Projekte des TIA-Portals V12.x besitzen die Endung ".ap12". Ältere Projekte des TIA-Portals haben die Endung ".ap[Versionsnummer]". Für Projekte des TIA-Portals V11.x ist die Endung also beispielsweise ".ap11".

## Ergebnis

Das Projekt wird in der Projektansicht geöffnet.

## Siehe auch

- Hinweise zur Kompatibilität (Seite 234)
- Projekte hochrüsten (Seite 235)
- Grundlagen zu Projekten (Seite 231)
- Neues Projekt erstellen (Seite 232)
- Projekte speichern (Seite 237)
- Projekte löschen (Seite 239)

### 7.3.3 Hinweise zur Kompatibilität

Mit dem TIA-Portal können Sie auch Projekte öffnen, die nicht mit derselben Version oder mit einem anderen Installationsumfang erstellt worden sind. Im Folgenden erfahren Sie, was Sie in diesem Fall beachten müssen.

#### Öffnen von Projekten aus älteren Versionen des TIA-Portals

Projekte aus früheren Versionen des TIA-Portals können geöffnet werden. Jedoch sind einige Besonderheiten bei Projekten aus früheren Versionen des TIA-Portals zu beachten:

- Projekte aus dem TIA-Portal V10.0 und V10.5  
Projekte aus den Versionen V10.0 oder V10.5 können geöffnet werden. Dabei werden Sie in das aktuelle Projektformat konvertiert.
- Projekte aus dem TIA-Portal V11.x  
Projekte aus der Version V11.x werden geöffnet und bleiben unverändert. Der Funktionsumfang ist auf die Möglichkeiten des TIA-Portals V11.x beschränkt. Somit bleiben die Projekte rückwärts kompatibel und können weiterhin mit der früheren Version des TIA-Portals bearbeitet werden. Um ein Projekt, welches Sie einmal mit der aktuellen Version des TIA-Portals gespeichert haben, im TIA-Portal V11 weiterzubearbeiten, muss sich das TIA-Portal V11 auf dem neuesten Stand befinden. Das bedeutet, es müssen alle Service Packs und alle weiteren Updates installiert sein.

Wenn Sie in der aktuellen Version des TIA-Portals ein Projekt aus einer älteren Version ab TIA-Portal V11 öffnen, können Sie dem Projekt Komponenten hinzufügen, die für diese ältere Version im Rahmen eines Hardware Support Packages (HSP) nachgeliefert wurden. Das Projekt gilt dann immer noch als kompatibel mit dieser älteren Version des TIA-Portals. Es muss lediglich ebenfalls das Hardware-Support-Package in der älteren Version installiert sein, um mit dem Projekt dort weiterzuarbeiten.

Wenn Sie den vollen Funktionsumfang der aktuellen Version in einem Projekt aus dem TIA-Portal V11.x nutzen möchten, müssen Sie das Projekt hochrüsten. Beim Hochrüsten wird das Projekt in das aktuelle Projektformat konvertiert und der volle Funktionsumfang der aktuellen Version wird nutzbar.



## Öffnen von Projekten aus neueren Versionen des TIA-Portals

Möchten Sie ein Projekt aus einer neueren Version öffnen, ist dies unter den folgenden Voraussetzungen möglich:


- Das Projekt ist mit Version des TIA-Portals erstellt worden, die neuer ist als V12 SP1, beispielsweise einem neueren Service Pack.
- Das Projekt enthält keine Daten, welche mit der aktuellen Installation inkompatibel sind.

## Kompatibilität von Projekten der aktuellen TIA-Portal-Version

Beachten Sie, dass Projekte, die mit dem TIA-Portal V12 SP1 gespeichert wurden, aufgrund der erweiterten Funktionalität gegenüber älteren Versionen, nicht rückwärtskompatibel sind. Projekte, die mit dem TIA-Portal V12 SP1 gespeichert wurden, können ausschließlich mit dem TIA-Portal V12 SP1 oder neuer geöffnet werden.

## Öffnen von Projekten, die mit Zusatzprodukten erstellt wurden

Enthält das zu öffnende Projekt Daten, die mit einer optionalen Software erstellt worden sind, das entsprechende Software-Produkt ist jedoch nicht installiert, so können folgende Fälle eintreten:

- Es fehlen ausschließlich Software-Komponenten, die nicht zwingend erforderlich sind:  
Ein Dialog erscheint, in dem die fehlenden Software-Komponenten aufgelistet sind. Nach dem Öffnen des Projekts werden die Eigenschaften des Projekts angezeigt. An dieser Stelle haben Sie die Möglichkeit, fehlende Produkte nachzuinstallieren. Auch ohne die Nachinstallation der fehlenden Produkte stehen alle im Projekt enthaltenen Geräte zur Verfügung. Sie können jedoch nur die Geräte bearbeiten, die von der aktuell installierten Software unterstützt werden.  
Geräte, deren Bearbeitung aufgrund von fehlender Software nicht unterstützt wird, sind mit folgendem Symbol in der Projektnavigation gekennzeichnet:  

- Mindestens ein Software-Paket wird für das Öffnen des Projekts zwingend benötigt:  
Ein Dialog erscheint, in dem die fehlenden Software-Komponenten aufgelistet sind. Die zwingend erforderlichen Pakete sind gekennzeichnet. Das Projekt lässt sich nur öffnen, wenn Sie die fehlenden Komponenten installieren.

## Siehe auch

Projekte öffnen (Seite 233)

Projekte hochrüsten (Seite 235)

## 7.3.4 Projekte hochrüsten

Sie können Projekte aus der Version V11.x des TIA-Portals auch in der aktuellen Version öffnen und bearbeiten. Der Funktionsumfang bleibt dabei allerdings auf die Möglichkeiten aus Version V11.x beschränkt. Somit ist die Rückwärtskompatibilität des Projekts gewährleistet und das Projekt lässt sich auch weiterhin mit der früheren Version des TIA-Portals bearbeiten.

Wenn Sie den vollen Funktionsumfang der aktuellen Version des TIA-Portals nutzen möchten, müssen Sie das Projekt hochrüsten.

Projekte aus dem TIA-Portal V12 müssen Sie in jedem Fall hochrüsten, um damit im TIA-Portal V12 SP1 weiterzuarbeiten. Sie erhalten beim Öffnen eines V12-Projekts eine entsprechende Aufforderung.

Wenn Sie globale Bibliotheken aus dem TIA-Portal V11.x weiterverwenden möchten, müssen Sie diese ebenfalls hochrüsten. Dies geschieht nicht automatisch mit dem Hochrüsten des Projekts.

## Vorgehen

Um ein Projekt hochzurüsten, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Projekt aus einer früheren Version des TIA-Portals.
2. Wählen Sie im Menü "Projekt" den Befehl "Hochrüsten".  
Es erscheint eine Sicherheitsabfrage.
3. Bestätigen Sie die Sicherheitsabfrage mit "Ja".

---

### Hinweis

#### Hochrüsten von know-how-geschützten Bausteinen

Erst nach einmaligem Öffnen des Bausteins mit dem Passwort wird der Baustein hochgerüstet und lässt sich laden. Öffnen Sie daher know-how-geschützte Bausteine nach dem Hochrüsten des Projekts einmal, um die Bausteine ebenfalls hochzurüsten. Wenn Sie viele know-how-geschützte Bausteine mit demselben Passwort geschützt haben, können Sie diese alle auf einmal selektieren und öffnen.

---

## Ergebnis

Das ursprüngliche Projekt wurde geschlossen und bleibt im Originalzustand gespeichert. Vom ursprünglichen Projekt wurde eine neue Version erzeugt. Die neue Version des Projekts ist geöffnet.

## Siehe auch

Hinweise zur Kompatibilität (Seite 234)

Projekte öffnen (Seite 233)

Bibliotheken aus älteren Versionen hochrüsten (Seite 325)

### 7.3.5 Eigenschaften des Projekts anzeigen

Sie können sich die Eigenschaften eines Projekts anzeigen lassen. Zu den Eigenschaften gehören die folgenden Punkte:

- **Meta-Daten zum Projekt**  
Hier erhalten Sie beispielsweise Informationen zum Erstellzeitpunkt, zum Autor, den Speicherpfad, der Projektgröße, Hinweise zum Copyright, zu den verwendeten Projektsprachen, usw. Viele der Attribute sind änderbar.
- **Projektverlauf**  
Der Projektverlauf enthält eine Übersicht mit wichtigen Ereignissen im Lebenszyklus eines Projekts. Hier können Sie z. B. sehen, mit welcher Version des TIA-Portals ein Projekt erstellt wurde und ob es zwischenzeitlich in eine Version konvertiert wurde. Ist ein Projekt beispielsweise durch eine Migration entstanden, wird dies in der Projektverlaufstabelle ebenfalls mit dem Datum und der Uhrzeit des Migrationszeitpunkts angezeigt. Wenn zu einem Ereignis ein Protokoll angelegt wurde, können Sie außerdem das Protokoll direkt aufrufen.
- **Support-Packages im Projekt**  
Es wird eine Übersicht zusätzlicher Software angezeigt, die notwendig ist, um alle Geräte des Projekts bearbeiten zu können. Außerdem werden installierte GSD-Dateien (Gerätebeschreibungsdateien für weitere Geräte im Hardware-Katalog) aufgelistet.
- **Softwareprodukte im Projekt**  
Sie können eine Übersicht aller installierten und für das Projekt notwendigen Softwareprodukte anzeigen lassen.

#### Vorgehen

Um die Projekteigenschaften anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das geöffnete Projekt.
2. Wählen Sie im Kontextmenü des Projekts den Befehl "Eigenschaften".  
Der Dialog mit den Eigenschaften des Projekts wird geöffnet.
3. Wählen Sie in der Bereichsnavigation die gewünschten Projekteigenschaften aus, die angezeigt werden sollen.

### 7.3.6 Projekte speichern

Sie können ein Projekt zu jedem beliebigen Zeitpunkt entweder unter dem gleichen oder einem anderen Namen speichern. Das Speichern ist auch dann möglich, wenn das Projekt noch fehlerhafte Elemente enthält.

### Projekt speichern

Um ein Projekt zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Speichern".  
Alle Änderungen im Projekt werden unter dem aktuellen Projektnamen gespeichert. Wenn Sie ein Projekt aus einer früheren Version des TIA-Portals bearbeiten, bleibt die Dateiendung des Projekts ebenfalls erhalten und Sie können das Projekt weiterhin in der früheren Version des TIA-Portals bearbeiten.

### Projekt speichern unter

Um das Projekt unter einem anderen Namen zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Speichern unter".  
Der Dialog "Aktuelles Projekt speichern unter" wird geöffnet.
2. Wählen Sie im Feld "Speichern in" den gewünschten Projektordner.
3. Geben Sie im Feld "Dateiname" den neuen Projektnamen ein.
4. Bestätigen Sie Ihre Eingabe mit "Speichern".  
Das Projekt wird unter dem neuen Namen gespeichert und geöffnet.

### Siehe auch

- Grundlagen zu Projekten (Seite 231)
- Neues Projekt erstellen (Seite 232)
- Projekte öffnen (Seite 233)
- Projekte löschen (Seite 239)
- Projekte hochrüsten (Seite 235)
- Hinweise zur Kompatibilität (Seite 234)

## 7.3.7 Projekte schließen

### Vorgehen

Um ein Projekt zu schließen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Schließen".  
Falls Sie seit dem letzten Speichern Änderungen am Projekt vorgenommen haben, wird eine Meldung eingeblendet.
2. Geben Sie an, ob die Änderungen gespeichert werden sollen.

## 7.3.8 Projekte löschen

---

### Hinweis

Beim Löschen werden die gesamten Projektdaten vom Speichermedium entfernt.

---

### Voraussetzung

Das zu löschende Projekt ist nicht geöffnet.

### Vorgehen

Um ein bestehendes Projekt zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Projekt löschen".  
Der Dialog "Projekt löschen" wird geöffnet und die Liste der zuletzt verwendeten Projekte wird darin angezeigt.
2. Selektieren Sie ein Projekt aus der Liste.  
Wenn das gewünschte Projekt nicht in der Liste vorhanden ist, klicken Sie auf die Schaltfläche "Durchsuchen". Navigieren Sie zum gewünschten Projektordner und öffnen Sie die Projektdatei.
3. Klicken Sie auf die Schaltfläche "Löschen".
4. Bestätigen Sie die Sicherheitsabfrage mit "Ja", um das Projekt endgültig zu löschen.

### Ergebnis

Der komplette Projektordner wird aus dem Dateisystem gelöscht.

### Siehe auch

- Grundlagen zu Projekten (Seite 231)
- Neues Projekt erstellen (Seite 232)
- Projekte öffnen (Seite 233)
- Projekte speichern (Seite 237)

## 7.3.9 Mit mehrsprachigen Projekten arbeiten

### 7.3.9.1 Grundlagen zu Projekttexten

#### Texte in unterschiedlichen Sprachen im Projekt

Wenn Sie beim Bearbeiten eines Projekts Texte eingeben, werden Sie dies in der Regel in Ihrer Muttersprache tun. Geben Sie das Projekt im Anschluss an einen anderen Bearbeiter ab, der diese Sprache nicht beherrscht, benötigt er eine Übersetzung aller relevanten Texte in einer für ihn verständlichen Sprache. Alle Texte sind daher übersetzbar. So können Sie sicherstellen, dass jeder, der zu einem späteren Zeitpunkt mit den Texten im Projekt konfrontiert wird, diese Texte in der von ihm gewünschten Sprache angezeigt bekommt.

#### Projektsprache

Als Projektsprachen werden alle Sprachen bezeichnet, in denen ein Projekt später bearbeitet werden soll. Alle Texte können auf Grundlage der Editiersprache in die verschiedenen Projektsprachen übersetzt werden. Welche Sprachen im Projekt verfügbar sind, legen Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen" fest.

#### Editiersprache

Jedes Projekt besitzt eine sogenannte Editiersprache. Wenn Sie Texte eingeben, werden diese immer in der Editiersprache angelegt. Achten Sie daher darauf, dass die eingestellte Editiersprache der Sprache entspricht, in der Sie Texte eingeben. So vermeiden Sie Probleme beim späteren Übersetzen der Texte.

Die Editiersprache ist unabhängig von der Sprache der Benutzeroberfläche. Sie können beispielsweise Englisch als Oberflächensprache eingestellt haben, als Editiersprache allerdings Italienisch verwenden. Machen Sie Texteingaben, so werden diese in diesem Fall in der Projektsprache "Italienisch" angelegt, obwohl die Benutzeroberfläche des TIA-Portals englisch angezeigt wird.

Die Editiersprache stellen Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projektsprachen > Editiersprache" ein.

#### Referenzsprache

Die Referenzsprache dient als Vorlage zur Übersetzung. In der Task Card "Aufgaben > Sprachen und Ressourcen" wird zu jedem Eingabefeld der Text in der Referenzsprache eingeblendet. So wissen Sie, welcher Text in ein Textfeld gehört, auch wenn in der aktuell eingestellten Editiersprache noch kein Text eingetragen ist.

## Anwendertexte und Systemtexte

Um die Übersichtlichkeit zu verbessern, wird zwischen Anwendertexten und Systemtexten unterschieden:

- Anwendertexte sind Texte, die vom Nutzer angelegt worden sind.
- Systemtexte sind Texte, die automatisch in Abhängigkeit von der Konfiguration im Projekt angelegt werden.

Die Projekttexte verwalten Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".

## Beispiele für mehrsprachige Projekttexte

Sie können z. B. die folgenden Projekttexte mehrsprachig verwalten:

- Bausteintitel und Bausteinkommentare
- Netzwerktitel und Netzwerkkommentare
- Kommentare in Tabellen
- Meldungstexte
- Bedienerrelevante Texte
- Textlisten
- Beschriftungen von Schaltflächen
- Anzeigenamen von Rezepturen

## Übersetzen von Texten

Um Texte zu übersetzen, gibt es drei unterschiedliche Vorgehensweisen.

- Alle im Projekt verwendeten Texte tabellarisch übersetzen  
Sie können die Übersetzungen für die einzelnen Projektsprachen direkt in die Tabelle "Projekttexte" eingeben. Diese finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".
- Einzelnen Objekten zugeordnete Texte im Inspektorfenster eingeben  
Im Inspektorfenster können Sie die Texte übersetzen, die den aktuell selektierten Objekten zugeordnet sind. In einer Tabelle werden Spalten für alle verfügbaren Projektsprachen angezeigt. Dort können Sie die Übersetzungen für den jeweiligen Text eingeben.
- Texte mithilfe von Referenztexten übersetzen  
Für kleinere Textmengen können Sie die Editiersprache wechseln. Alle Textfelder sind anschließend wieder mit den Standardwerten befüllt und können in der aktuellen Sprache ausgefüllt werden. Zur Orientierung, was Sie zuletzt in das Feld eingegeben haben, können Sie sich den Text in der Referenzsprache anzeigen lassen. Dazu verwenden Sie die Task Card "Aufgaben" und öffnen die Palette "Sprachen & Ressourcen".
- Texte exportieren und extern übersetzen  
Bei großen Textmengen können Sie Texte in eine Office Open XML Datei exportieren und diese mit einem gängigen Tabellenkalkulationsprogramm übersetzen. Die übersetzte Liste importieren Sie anschließend wieder in das TIA-Portal.

---

### Hinweis

#### Verwenden von asiatischen Projektsprachen

Ostasiatische Projektsprachen werden unter Windows XP nur dann korrekt angezeigt, wenn unter Windows XP Professional in der Systemsteuerung unter "Regions- und Sprachoptionen > Sprachen" die Option "Dateien für ostasiatische Sprachen installieren" aktiviert ist.

---

### Siehe auch

Übersicht der Programmeinstellungen (Seite 171)

Einstellungen ändern (Seite 175)

Anwendungsbeispiele für mehrsprachige Projekte (Seite 247)

### 7.3.9.2 Projektsprachen auswählen

Alle Texte innerhalb eines Projekts können in derselben Sprache dargestellt werden, die Sie auch für Ihre Softwareoberfläche ausgewählt haben. Dafür ist es notwendig, dass alle Projektttexte in der entsprechenden Sprache vorliegen. Die im Projekt zur Verfügung stehenden Projektsprachen können Sie selbst festlegen.

### Voraussetzung

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.

### Vorgehen

Um die Projektsprachen auszuwählen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die darunter liegenden Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projektsprachen".  
Im Arbeitsbereich sehen Sie eine Liste von Sprachen, die Sie auswählen können.
3. Aktivieren Sie die gewünschten Sprachen.

### Ergebnis

Alle Texte können in den aktivierten Sprachen angezeigt werden, sofern bereits eine Übersetzung in den Sprachen vorliegt.



### 7.3.9.3 Editiersprache festlegen

Alle Texte im Projekt werden bei der Eingabe in der so genannten Editiersprache angelegt. Wenn Sie die Editiersprache ändern, werden alle zukünftigen Texteingaben in der neuen Editiersprache gespeichert.

#### Voraussetzung

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.

#### Vorgehen

Um die Editiersprache zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projektsprachen".  
Im Arbeitsbereich werden die Einstellmöglichkeiten zu den Projektsprachen angezeigt.
3. Wählen Sie unter "Allgemein > Editiersprache" die Editiersprache aus.

### 7.3.9.4 Alle Projekttexte tabellarisch übersetzen

Sie können sämtliche Projekttexte, die im aktuell geöffneten Projekt verwendet werden, in einer Liste anzeigen und bearbeiten. Anwender- und Systemtexte sind zur besseren Übersicht in zwei unterschiedlichen Listen getrennt. Beide Listen enthalten für jede Projektsprache jeweils eine Spalte, in die Sie Übersetzungen von Texten eintragen können.

#### Voraussetzung

- Sie befinden sich in der Projektansicht.
- Sie haben mindestens eine weitere Projektsprache gewählt.

#### Vorgehen

Um Texte in der projektweiten Liste zu übersetzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die darunter liegenden Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte".  
Im Arbeitsbereich erscheint eine Liste mit den Anwendertexten im Projekt.
3. Klicken Sie auf "Systemtexte", wenn Sie statt der Anwendertexte die Liste mit Systemtexten bearbeiten möchten.

4. Schaffen Sie Übersichtlichkeit in der Liste, wenn die Menge an Texten zu groß wird.
  - Um identische Texte zu gruppieren und sie auf einmal zu übersetzen, klicken Sie in der Funktionsleiste auf die Schaltfläche "Gruppierung ein/aus".
  - Um Texte mit fehlenden Übersetzungen auszublenden, klicken Sie in der Funktionsleiste auf die Schaltfläche "Leere Texte ausfiltern ein/aus".
  - Um die angezeigten Projektttexte weiter auf bestimmte Geräte einzugrenzen, wählen Sie in der Klappliste die Geräte aus, zu denen Sie Projektttexte anzeigen möchten.
5. Geben Sie die Übersetzung für die Projektttexte in der jeweiligen Spalte ein.

#### 7.3.9.5 Einzelnen Objekten zugehörige Texte übersetzen

Wenn Sie die Texte einzelner Objekte bearbeiten wollen, wäre es zu umständlich die passenden Texte in der Tabelle mit allen Projektttexten herauszusuchen. Daher gibt es im Inspektorfenster eine Tabelle, in der ausschließlich die Texte angezeigt werden, die den aktuell selektierten Objekten zugeordnet sind. In der Tabelle können Sie fehlende Übersetzungen für einzelne Projektsprachen ergänzen oder bereits vorhandene Texte ändern.

#### Voraussetzung

Für die zu übersetzenden Texte muss mindestens in einer Projektsprache bereits ein Text eingegeben sein.

#### Vorgehen

Um die Texte der aktuell selektierten Objekte zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Objekte, deren Texte Sie bearbeiten möchten.
2. Öffnen Sie das Register "Eigenschaften" im Inspektorfenster.
3. Öffnen Sie das unterlagerte Register "Texte" im Inspektorfenster.  
Eine Tabelle mit allen Texten, die zu den selektierten Objekten gehören, wird angezeigt. Sie enthält jeweils eine Spalte für die aktuell eingestellte Editiersprache und die Referenzsprache sowie weitere Spalten für die zusätzlichen Projektsprachen.
4. Ergänzen oder ändern Sie in der Tabelle die Einträge für die einzelnen Projektsprachen.

#### Siehe auch

Anwendungsbeispiele für mehrsprachige Projekte (Seite 247)

#### 7.3.9.6 Texte mithilfe von Referenztexten übersetzen

#### Einführung

Nach dem Wechseln der Editiersprache werden alle Texte in Eingabefeldern in der neuen Editiersprache angezeigt. Sind für die neu eingestellte Sprache noch keine Übersetzungen vorhanden, sind die Eingabefelder leer oder mit den Standardwerten befüllt.

Wenn Sie in ein Eingabefeld Text eingeben, wird dieser in der aktuellen Editiersprache gespeichert. Im Anschluss existieren für dieses Eingabefeld Texte in zwei Projektsprachen - in der ehemaligen Editiersprache und in der aktuellen Editiersprache. Auf diese Weise ist es möglich, Texte in mehreren Projektsprachen anzulegen.

Sie können sich bei einem Eingabefeld bereits bestehende Übersetzungen in anderen Projektsprachen anzeigen lassen. Diese dienen als Vergleich für die Texteingabe in der aktuellen Editiersprache und werden als Referenzsprache bezeichnet.

---

**Hinweis**

Die Anzeige von Referenztexten ist abhängig von den installierten Produkten und wird nicht von jedem Editor unterstützt.

---

**Voraussetzung**

Für ein Eingabefeld existiert mindestens eine Übersetzung in einer weiteren Projektsprache.

**Vorgehen**

Um sich für ein Eingabefeld die Übersetzung in einer Referenzsprache anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Task Card "Aufgaben" die Palette "Sprachen & Ressourcen".
2. Wählen Sie aus der Klappliste "Referenzsprache" eine Referenzsprache aus.

**Ergebnis**

Die Referenzsprache ist voreingestellt. Wenn Sie in ein Textfeld klicken, werden in der Task Card "Aufgaben > Sprachen & Ressourcen" Übersetzungen angezeigt, die bereits in anderen Projektsprachen vorhanden sind.

**7.3.9.7 Projekttexte exportieren und importieren**

Sie können Projekttexte zur Übersetzung exportieren und anschließend wieder importieren. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann in Microsoft Excel oder in vielen anderen Tabellenkalkulationsprogrammen bearbeitet werden.

Sie haben folgende Möglichkeiten für den Export:

- Einzelne Projekttexte exportieren
- Alle Anwendertexte oder Systemtexte auf einmal exportieren  
Der Export kann in diesem Fall zusätzlich nach Kategorien eingegrenzt werden.

---

### Hinweis

#### Zeilenbeschränkung bei Microsoft Excel

Beachten Sie, dass Tabellenkalkulationsprogramme möglicherweise nur eine bestimmte Anzahl an Zeilen verarbeiten können. Microsoft Excel 2003 unterstützt beispielsweise maximal 65536 Zeilen. Neuere Versionen von Microsoft Excel unterstützen erheblich mehr Zeilen.

---

### Einzelne Projekttexte exportieren

Um einzelne Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Sprachen & Ressourcen". Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte". Der Projekttexte-Editor wird geöffnet.
3. Wählen Sie das Register "Anwendertexte" oder "Systemtexte" im Editor, je nachdem, welche Texte Sie exportieren möchten.
4. Selektieren Sie die Projekttexte, die Sie exportieren möchten.
5. Klicken Sie in der Werkzeugleiste des Editors auf das Symbol "Projekttexte exportieren". Der Dialog "Exportieren" wird geöffnet.
6. Wählen Sie aus der Klappliste "Quellsprache" die Sprache, aus der Sie die Übersetzung vornehmen möchten.
7. Wählen Sie aus der Klappliste "Zielsprache" die Sprache, in die Sie die Texte übersetzen möchten. In der Klappliste stehen die bereits festgelegten Projektsprachen zur Verfügung. Fehlt die gewünschte Sprache, legen Sie diese zunächst im Projektsprachen-Editor fest.
8. Geben Sie im Eingabefeld "Zu exportierende Datei auswählen" einen Speicherpfad und einen Dateinamen für die Exportdatei an.
9. Klicken Sie auf "Exportieren".

### Alle System- oder Anwendertexte exportieren

Um alle Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Werkzeuge" den Befehl "Projekttexte exportieren". Der Dialog "Exportieren" wird geöffnet.
2. Wählen Sie aus der Klappliste "Quellsprache" die Sprache, aus der Sie die Übersetzung vornehmen möchten.
3. Wählen Sie aus der Klappliste "Zielsprache" die Sprache, in die Sie die Texte übersetzen möchten. In der Klappliste stehen die bereits festgelegten Projektsprachen zur Verfügung. Fehlt die gewünschte Sprache, legen Sie diese zunächst im Projektsprachen-Editor fest.

4. Aktivieren Sie unter "Inhalte auswählen" das Optionskästchen "Anwendertexte", um Anwendertexte zu exportieren. Wählen Sie "Systemtexte", um Systemtexte zu exportieren. Aktivieren Sie beide Optionskästchen, um sowohl Anwendertexte als auch Systemtexte zu exportieren.
5. Wählen Sie unter "Inhalte auswählen" die gewünschten Textkategorien für die Anwendertexte oder die Systemtexte aus.
6. Geben Sie im Eingabefeld "Exportdatei" einen Dateinamen für die Exportdatei an.
7. Wählen Sie im Eingabefeld "Pfad" den Pfad im Dateisystem aus, an dem die Exportdatei gespeichert werden soll.
8. Klicken Sie auf "Exportieren".

### Projekttexte importieren

Um eine Datei mit Projekttexten zu importieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Werkzeuge" den Befehl "Projekttexte importieren". Der Dialog "Importieren" wird geöffnet.
2. Wählen Sie aus dem Feld "Zu importierende Datei auswählen" den Pfad und den Dateinamen der Importdatei aus.
3. Aktivieren Sie das Optionskästchen "Basissprache importieren", wenn Sie in der Exportdatei Änderungen an der Basissprache vorgenommen haben und die Einträge im Projekt mit den Änderungen überschreiben möchten.
4. Klicken Sie auf "Importieren".

### Siehe auch

Anwendungsbeispiele für mehrsprachige Projekte (Seite 247)

### 7.3.9.8 Anwendungsbeispiele für mehrsprachige Projekte

#### Einleitung

Gehen wir davon aus, Sie arbeiten in einem Team mit Kollegen, die jeweils Englisch, Französisch oder Deutsch sprechen. Mit Hilfe des TIA-Portals haben Sie ein Projekt angelegt und bereits eine funktionierende Konfiguration erstellt.

Um das Projekt für die anderen Kollegen des Teams nachvollziehbar zu halten, möchten Sie alle Geräte in den Sprachen Englisch und Deutsch mit Kommentaren versehen. Zunächst möchten Sie die Kommentare in Deutsch eingeben. Anschließend möchten Sie aus Zeit- und Kostengründen ein externes Übersetzungsbüro beauftragen, um die Texte dort in einem Tabellenkalkulationsprogramm ins Englische übersetzen zu lassen.

Zusätzlich möchten Sie einen einzelnen Kommentar zu einem bestimmten Gerät auch in Französisch vorhalten, damit der französischsprachige Kollege dieses Gerät weiterbearbeiten kann.

Im Folgenden ist beispielhaft beschrieben, wie Sie dieses Ziel mit den Mitteln des TIA-Portals erreichen können.

### Das Projekt ins Englische übersetzen

Um die Kommentare zunächst auf Deutsch einzugeben und anschließend ins Englische übersetzen zu lassen, gehen Sie folgendermaßen vor:

1. Stellen Sie die Editiersprache auf "Deutsch" ein und befüllen Sie alle Kommentarfelder mit den jeweiligen Texten auf Deutsch.  
Bei dem ausgewählten Gerät für den französischsprachigen Kollegen geben Sie beispielsweise zunächst auf Deutsch "Unser neues Gerät" ein.  
Alle Kommentare sind nun in deutscher Sprache hinterlegt.
2. Exportieren Sie alle Anwendertexte in eine Office Open XML Datei mit der Endung ".xlsx".
3. Lassen Sie die in der Datei enthaltenen Anwendertexte in einem gängigen Tabellenkalkulationsprogramm, wie z. B. Microsoft Excel, ins Englische übersetzen.
4. Importieren Sie die Datei nach der Übersetzung in das TIA-Portal.  
Alle Texte stehen nun in Deutsch und Englisch zur Verfügung.

### Einzelnes Kommentarfeld auf Französisch übersetzen

Um den Inhalt eines einzelnen Kommentarfeldes auf Französisch zu übersetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Kommentarfeld zu dem Gerät, welches für die Weiterbearbeitung durch den französischsprachigen Kollegen vorgesehen ist.
2. Öffnen Sie die Palette "Sprachen & Ressourcen" in der Task-Card "Aufgaben".
3. Stellen Sie in der Palette "Sprachen & Ressourcen" als Editiersprache "Französisch" ein.  
Als Referenzsprache stellen Sie beispielsweise "Englisch" ein.  
Da im Französischen noch keine Übersetzung gespeichert ist, ist das Kommentarfeld leer.  
In der Palette "Sprachen & Ressourcen" wird jedoch die englische Übersetzung "Our new device" als Referenz angezeigt.
4. Orientieren Sie sich am englischen Referenztext und geben Sie in das Kommentarfeld "Notre nouvel appareil" ein.  
Der Kommentar zu diesem Gerät liegt nun in den Sprachen Deutsch, Englisch und Französisch vor.

### Siehe auch

Grundlagen zu Projekttexten (Seite 240)

Projekttexte exportieren und importieren (Seite 245)

Einzelnen Objekten zugehörige Texte übersetzen (Seite 244)

## 7.3.10 Projekte archivieren und dearchivieren

### 7.3.10.1 Arbeiten mit Projektarchiven

#### Archivierung und Weitergabe von Projekten

Wenn Sie längere Zeit mit einem Projekt arbeiten, kann dies insbesondere bei umfangreichen Hardware-Aufbauten zu großen Dateien führen. Daher können Sie die Projektgröße reduzieren, wenn Sie ein Projekt beispielsweise auf einer externen Festplatte archivieren möchten, oder wenn Sie ein Projekt per E-Mail verschicken möchten und daher auf eine geringe Dateigröße angewiesen sind.

#### Möglichkeiten zur Reduktion der Projektgröße

Es gibt zwei Möglichkeiten, die Projektgröße zu reduzieren:

- **TIA-Portal-Projektarchive**  
TIA-Portal-Projektarchive sind komprimierte Dateien, die jeweils ein ganzes Projekt inklusive der gesamten Ordnerstruktur des Projekts enthalten. Bevor das Projektverzeichnis zur Archivdatei gepackt wird, werden sämtliche Dateien auf ihre wesentlichen Bestandteile reduziert, um die Projektgröße zusätzlich zu verringern. Projektarchive eignen sich daher gut zum Versand per E-Mail.  
Projektarchive eines Projekts, die mit dem TIA-Portal V11.x erstellt wurden, haben die Dateiendung ".zap11". Projekte, die mit dem TIA-Portal V12.x erstellt wurden, haben die Dateiendung ".zap12".  
Um ein Projektarchiv zu öffnen, müssen Sie das Projekt dearchivieren. Dabei wird die Archivdatei an einem von Ihnen gewählten Ort zur ursprünglichen Projektverzeichnisstruktur mit den darin enthaltenen Projektdateien entpackt.
- **Minimiertes TIA-Portal-Projekt**  
Sie können auf die zusätzliche Komprimierung in einer Archivdatei verzichten und stattdessen eine Kopie des Projektverzeichnisses erstellen. Die enthaltenen Dateien werden auf die wesentlichen Bestandteile des Projekts reduziert. Somit wird der Speicherplatzbedarf minimiert. Der volle Funktionsumfang des Projekts bleibt erhalten und Sie können das Projekt wie gewohnt öffnen.  
Ein minimiertes TIA-Portal-Projekt ist besonders gut zur Archivierung geeignet, z. B. auf einem externen Datenträger.

#### Siehe auch

Projekte dearchivieren (Seite 251)

Projekte archivieren (Seite 249)

### 7.3.10.2 Projekte archivieren

Sie können den Speicherplatzbedarf des gerade geöffneten Projekts reduzieren, indem Sie das Projekt zu einer komprimierten Datei packen, oder indem Sie die Projektdateien auf ihre

wesentlichen Bestandteile reduzieren. Beides erreichen Sie mit der Archivieren-Funktion des TIA-Portals.

---

#### **Hinweis**

Beim Archivieren wird der zuletzt gespeicherte Stand des Projekts verwendet. Speichern Sie daher das Projekt ab bevor Sie die Archivieren-Funktion nutzen. So sind auch Ihre letzten Änderungen im archivierten Projekt enthalten.

---

## **Vorgehen**

Um ein Projekt zu archivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie den Befehl "Archivieren" im Menü "Projekt".  
Der Dialog "Aktuelles Projekt archivieren unter..." wird geöffnet.
2. Wählen Sie das Verzeichnis, an dem Sie die Archivdatei oder das neue Projektverzeichnis speichern möchten.
3. Wählen Sie aus der Klappliste "Dateityp" den Dateityp aus:
  - TIA-Portal-Projektarchiv, wenn Sie eine komprimierte Datei des Projekts erstellen möchten.
  - TIA-Portal-Projekt minimiert, wenn Sie lediglich eine Kopie des Projektverzeichnisses mit minimiertem Speicherplatzbedarf erstellen möchten.
4. Geben Sie einen Dateinamen in das Feld "Dateiname" ein, wenn Sie eine Archivdatei erstellen. Wenn Sie ein minimiertes Projektverzeichnis erstellen, geben Sie stattdessen den Namen des neu zu erstellenden Projektverzeichnisses in das Feld "Dateiname" ein.
5. Klicken Sie auf "Speichern".

## **Ergebnis**

Wenn Sie ein Projektarchiv erstellt haben, wird eine komprimierte Datei mit der Endung ".zap11" (für Projekte, die mit der Version V11.x des TIA-Portals erstellt wurden) oder ".zap12" (für Projekte, die mit der Version V12.x des TIA-Portals erstellt wurden) generiert. Die Datei enthält das komplette Projektverzeichnis. Die einzelnen Dateien des Projekts wurden zusätzlich auf die wesentlichen Bestandteile reduziert, um Speicherplatz zu sparen.

Wenn Sie ein minimiertes TIA-Portal-Projekt erstellt haben, wird lediglich eine Kopie des ursprünglichen Projektverzeichnisses am gewünschten Ort erstellt. Die darin enthaltenen Dateien wurden auf die wesentlichen Bestandteile reduziert, um Speicherplatz zu sparen.

## **Siehe auch**

Arbeiten mit Projektarchiven (Seite 249)

Projekte dearchivieren (Seite 251)



### 7.3.10.3 Projekte dearchivieren

Projekte, die mit der Archivieren-Funktion des TIA-Portals gepackt wurden, können Sie wieder entpacken. Dabei wird die enthaltene Projektverzeichnisstruktur inklusive aller Projektdateien wiederhergestellt.

#### Voraussetzung

Es darf kein Projekt geöffnet sein.

#### Vorgehen

Um ein Projektarchiv zu entpacken, gehen Sie folgendermaßen vor:

1. Wählen Sie den Befehl "Dearchivieren" im Menü "Projekt".  
Der Dialog "Archiviertes Projekt dearchivieren" wird geöffnet.
2. Wählen Sie das Projektarchiv aus.
3. Klicken Sie auf Öffnen.
4. Der Dialog "Ordner suchen" wird geöffnet.
5. Wählen Sie das Zielverzeichnis aus, an dem das archivierte Projekt entpackt werden soll.
6. Klicken Sie auf "OK".

#### Ergebnis

Das Projekt wird im gewählten Verzeichnis entpackt und sofort geöffnet.

#### Siehe auch

Arbeiten mit Projektarchiven (Seite 249)

## 7.4 Referenzprojekte verwenden

### 7.4.1 Grundlagen zu Referenzprojekten

#### Einführung

Sie haben die Möglichkeit zusätzlich zum aktuellen Projekt weitere Projekte als Referenz zu öffnen. Diese Referenzprojekte können Sie folgendermaßen verwenden:

- Sie können einzelne Objekte aus einem Referenzprojekt in das aktuelle Projekt ziehen und anschließend bearbeiten.
- Sie können bestimmte Objekte, z. B. Codebausteine, aus einem Referenzprojekt schreibgeschützt öffnen. Dies ist jedoch nicht für alle Elemente möglich.
- Sie können Geräte des Referenzprojekts mithilfe eines Offline/Offline-Vergleichs mit Geräten aus dem aktuellen Projekt vergleichen.

Beachten Sie, dass die Referenzprojekte schreibgeschützt sind. Sie können die Objekte eines Referenzprojekts also nicht verändern.

#### Siehe auch

Referenzprojekte vergleichen (Seite 253)

Referenzprojekt öffnen und schließen (Seite 252)

Referenzprojekte (Seite 196)

### 7.4.2 Referenzprojekt öffnen und schließen

#### Referenzprojekt öffnen

Um ein Referenzprojekt zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation in der Palette "Referenzprojekte" in der Funktionsleiste auf "Referenzprojekt öffnen".  
Der Dialog "Referenzprojekt öffnen" wird geöffnet.
2. Navigieren Sie zum gewünschten Projektordner und öffnen Sie die Projektdatei. Projekte des TIA-Portals V12.x besitzen die Endung ".ap12". Ältere Projekte des TIA-Portals haben die Endung ".ap[Versionsnummer]".
3. Klicken Sie auf "Öffnen".  
Das gewählte Projekt wird schreibgeschützt als Referenzprojekt geöffnet.

### Referenzprojekt schließen

Um ein Referenzprojekt zu schließen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation in der Palette "Referenzprojekte" das Referenzprojekt, das Sie schließen möchten.
2. Klicken Sie in der Funktionsleiste auf "Referenzprojekt schließen".  
Das gewählte Referenzprojekt wird geschlossen.

### Siehe auch

Grundlagen zu Referenzprojekten (Seite 252)

Referenzprojekte vergleichen (Seite 253)

Referenzprojekte (Seite 196)

## 7.4.3 Referenzprojekte vergleichen

### Einführung

Sie können Geräte aus Referenzprojekten sowohl mit Geräten aus dem aktuellen Projekt als auch aus dem gleichen oder einem anderen Referenzprojekt oder auch aus einer Bibliothek vergleichen.

---

#### Hinweis

Beachten Sie folgende Hinweise:

- Sie können für die Vergleichsobjekte keine Aktionen festlegen, da die Referenzprojekte schreibgeschützt sind.
  - Sie können für die Vergleichsobjekte einen Detailvergleich durchführen, wenn der Typ des Vergleichsobjekts einen Detailvergleich generell erlaubt.
  - Sie können auch beim Vergleichen von Referenzprojekten jederzeit zwischen automatischem und manuellem Vergleich wechseln.
- 

### Vorgehen

Um die Objekte eines Referenzprojekts mit den Gerätedaten des aktuellen Projekts zu vergleichen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Gerät, dessen Daten Sie mit den Daten eines Referenzprojekts vergleichen möchten und das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Öffnen Sie in der Projektnavigation die Palette "Referenzprojekte".

4. Selektieren Sie das Gerät eines Referenzprojekts, das Sie mit den Gerätedaten aus dem aktuellen Projekt vergleichen möchten.
5. Ziehen Sie das Gerät aus dem Referenzprojekt in die rechte Drop-Fläche des Vergleichseditors.  
Den Status der Objekte können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Wenn Sie ein Objekt selektieren, werden die Eigenschaften des Objekts und des entsprechenden Objekts des zugeordneten Geräts übersichtlich im Eigenschaftsvergleich angezeigt.  
Sie können jederzeit aus dem aktuellen Projekt, einer Bibliothek oder aus einem Referenzprojekt andere Geräte in die Drop-Flächen ziehen und somit einen neuen Vergleich starten. Dabei ist es unerheblich, welches Gerät Sie in welche Drop-Fläche ziehen.

### Siehe auch

- Grundlagen zu Referenzprojekten (Seite 252)
- Referenzprojekte (Seite 196)
- Referenzprojekt öffnen und schließen (Seite 252)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Vergleichseditor verwenden (Seite 264)

## 7.5 Projektdaten bearbeiten

### 7.5.1 Projektdaten übersetzen und laden

#### 7.5.1.1 Projektdaten übersetzen

### Allgemeine Informationen zum Übersetzen von Projektdaten

#### Übersetzen von Projektdaten

Beim Übersetzen werden die Projektdaten so umgewandelt, dass sie vom Gerät lesbar sind. Hardware-Konfigurationsdaten und Programmdateien können getrennt oder gemeinsam übersetzt werden. Sie können die Projektdaten gleichzeitig für ein oder mehrere Zielsysteme übersetzen.

Folgende Projektdaten müssen vor dem Laden übersetzt werden:

- Hardware-Projektdaten, z. B. Konfigurationsdaten der Geräte oder Netze und Verbindungen
- Software-Projektdaten, z. B. Programmbausteine oder Prozessbilder

---

### **Hinweis**

Während ein Gerät übersetzt wird, kann kein weiterer Übersetzungsvorgang gestartet werden. Beachten Sie dabei, dass ein Übersetzungsvorgang nicht nur manuell durch Sie, sondern bei Bedien- und Beobachtengeräten auch automatisch angestoßen werden kann.

---

## **Umfang der Übersetzung**

Wenn Sie Projektdaten übersetzen, haben Sie abhängig vom Gerät folgende Möglichkeiten:

- Hardware und Software (nur Änderungen)
- Hardware (nur Änderungen)
- Software (nur Änderungen)
- Software (Bausteine komplett übersetzen)
- Software (Speicherreserve zurücksetzen)

## **Siehe auch**

Projektdaten übersetzen (Seite 255)

## **Projektdaten übersetzen**

Das folgende Kapitel beschreibt den allgemeinen Vorgang des Übersetzens von Projektdaten innerhalb der Projektnavigation. Wie das Übersetzen bestimmter Objekte funktioniert und welche Besonderheiten es dabei zu beachten gilt, finden Sie in der Hilfe zum Produkt.

## **Vorgehen**

Um Projektdaten zu übersetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die Geräte, deren Projektdaten Sie übersetzen möchten.
2. Wählen Sie im Kontextmenü im Untermenü "Übersetzen" die gewünschte Option.

---

### **Hinweis**

Beachten Sie, dass es vom selektierten Gerät abhängt, welche Optionen Ihnen zur Verfügung stehen.

---

Die Projektdaten werden übersetzt. Ob das Übersetzen erfolgreich durchgeführt wurde, können Sie im Inspektorfenster unter "Info > Übersetzen" kontrollieren.

## **Siehe auch**

Allgemeine Informationen zum Übersetzen von Projektdaten (Seite 254)

### 7.5.1.2 Projektdaten laden

#### Allgemeine Informationen zum Laden

##### Einführung

Für das Einrichten Ihres Automatisierungssystems ist es notwendig, dass Sie die Projektdaten, die Sie offline erzeugt haben, in die angeschlossenen Geräte laden. Diese Projektdaten entstehen z. B. bei der Konfiguration von Hardware, Netzen und Verbindungen, bei der Programmierung des Anwenderprogramms oder beim Anlegen von Rezepturen. Beim erstmaligen Laden werden die Projektdaten vollständig geladen. Bei weiteren Ladevorgängen werden nur noch Änderungen geladen. Sie können die Projektdaten in Geräte und Memory Cards laden.

---

##### Hinweis

Während ein Gerät übersetzt wird, kann kein Ladevorgang gestartet werden. Beachten Sie dabei, dass ein Übersetzungsvorgang nicht nur manuell durch Sie, sondern bei Bedien- und Beobachtengeräten auch automatisch angestoßen werden kann.

---

Abhängig vom Objekt, das Sie laden möchten, haben Sie folgende Optionen:

- **Hardware und Software**  
Sowohl Hardware-Konfiguration als auch Software werden in das Ziel geladen.
- **Hardwarekonfiguration**  
Es wird nur die Hardware-Konfiguration in das Ziel geladen.
- **Software (nur Änderungen)**  
Es werden nur die Objekte in das Ziel geladen, die online und offline unterschiedlich sind.
- **PLC-Programm ins Gerät laden und zurücksetzen**  
Es werden alle Bausteine in das Ziel geladen und alle Werte auf ihre Startwerte zurückgesetzt. Beachten Sie, dass dies auch für die remanenten Werte der Fall ist.

Sie können aber auch Projektdaten, die sich bereits in einem Gerät befinden, wieder in Ihr Projekt laden. Dabei haben Sie folgende Möglichkeiten:

- **Laden eines kompletten Geräts**  
Das Gerät wird mit allen relevanten Daten in das Projekt geladen.
- **Laden von Bausteinen und Parameter**  
Es werden nur die Bausteine und die Parameter aus dem Gerät in das Projekt geladen.

In beiden Fällen werden beim Laden alle Instanzen von Bibliotheks-Typen wieder mit der passenden Version des Typen in der Projektbibliothek verbunden. Wenn zu einer geladenen Instanz noch kein passender Typ oder nicht die richtige Version des Typen in der Projektbibliothek vorhanden ist, wird der Typ oder die Version in der Projektbibliothek ergänzt.

## Siehe auch

- Projektdaten in ein Gerät laden (Seite 257)
- Projektdaten auf eine Memory Card laden (Seite 258)
- Projektdaten von einem Gerät laden (Seite 259)

## Projektdaten in ein Gerät laden

Das folgende Kapitel beschreibt den allgemeinen Vorgang des Ladens von Projektdaten in ein Gerät. Wie das Laden bestimmter Objekte funktioniert und welche Besonderheiten es dabei zu beachten gilt, finden Sie in der Hilfe zum Produkt.

## Voraussetzung

- Die Projektdaten sind konsistent.
- Jedes zu ladende Gerät ist über einen Online-Zugang erreichbar.

## Vorgehen

Um Projektdaten in ausgewählte Geräte zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein oder mehrere Geräte.
2. Klicken Sie mit der rechten Maustaste auf ein selektiertes Element.  
Das Kontextmenü wird geöffnet.
3. Wählen Sie im Kontextmenü im Untermenü "Laden in Gerät" die gewünschte Option.

---

### **Hinweis**

Beachten Sie, dass es vom selektierten Gerät abhängt, welche Optionen Ihnen zur Verfügung stehen.

---

Wenn erforderlich, werden die Projektdaten übersetzt.

- Falls Sie bereits eine Online-Verbindung hergestellt hatten, wird der Dialog "Vorschau laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
- Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet und Sie müssen zunächst die Schnittstellen auswählen, über die Sie die Online-Verbindung zum Gerät herstellen möchten.  
Siehe auch: Auto-Hotspot

4. Kontrollieren Sie die Meldungen im Dialog "Vorschau laden" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".

---

**Hinweis**

Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

---

Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

5. Klicken Sie auf die Schaltfläche "Laden".  
Der Ladevorgang wird durchgeführt. Anschließend wird der Dialog "Ergebnisse laden" geöffnet. In diesem Dialog können Sie prüfen, ob der Ladevorgang erfolgreich war, und eventuell weitere Aktionen auswählen.
6. Klicken Sie auf die Schaltfläche "Fertig stellen".

## Ergebnis

Die gewählten Projektdaten wurden in die Geräte geladen.

## Siehe auch

- Allgemeine Informationen zum Laden (Seite 256)
- Projektdaten auf eine Memory Card laden (Seite 258)
- Projektdaten von einem Gerät laden (Seite 259)

## Projektdaten auf eine Memory Card laden

### Voraussetzung

- Die Memory Card wird angezeigt.
- Siehe auch: Auf Memory Cards zugreifen (Seite 312)

### Vorgehen

Um Projektdaten auf eine Memory Card zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation die Projektdaten auf die Memory Card, die Sie laden möchten.  
Der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
2. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".  
Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.



3. Klicken Sie auf die Schaltfläche "Laden".  
Der Ladevorgang wird durchgeführt. Anschließend wird der Dialog "Ergebnisse laden" geöffnet. In diesem Dialog können Sie prüfen, ob der Ladevorgang erfolgreich war, und eventuell weitere Aktionen auswählen.
4. Klicken Sie auf die Schaltfläche "Fertig stellen".

## Siehe auch

- Allgemeine Informationen zum Laden (Seite 256)
- Projektdaten in ein Gerät laden (Seite 257)
- Projektdaten von einem Gerät laden (Seite 259)

## Projektdaten von einem Gerät laden

Das folgende Kapitel beschreibt den allgemeinen Vorgang des Ladens von Projektdaten von einem Gerät. Welche Projektdaten Sie von einem Gerät laden können, ist von den installierten Produkten abhängig.

Prinzipiell haben Sie beim Laden von Projektdaten von einem Gerät in Ihr Projekt folgende Möglichkeiten:

- **Gerät in PG/PC laden**  
Mit dieser Option können Sie mit einem leeren Projekt starten und vorhandene Projektdaten direkt von einem Gerät laden.
- **Laden von Gerät**  
Es werden nur bestimmte Projektdaten aus dem Gerät in das Projekt geladen. Welche Projektdaten geladen werden können, finden Sie in der Hilfe zum Produkt.

In beiden Fällen werden beim Laden alle Instanzen von Bibliotheks-Typen wieder mit der passenden Version des Typen in der Projektbibliothek verbunden. Wenn zu einer geladenen Instanz noch kein passender Typ oder nicht die richtige Version des Typen in der Projektbibliothek vorhanden ist, wird der Typ oder die Version in der Projektbibliothek ergänzt.

## Voraussetzung

- Ein Projekt ist geöffnet.
- Die zu ladende Hardware-Konfiguration und Software muss mit dem TIA-Portal kompatibel sein. Stellen Sie die Kompatibilität insbesondere dann sicher, wenn die Daten auf dem Gerät mit einer früheren Programmversion oder mit einer anderen Projektierungssoftware erstellt wurden.

## Gerät in PG/PC laden

Um das komplette Gerät in Ihr Projekt zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation den Projektnamen.  
Im Menü "Online" wird der Befehl "Gerät in PG/PC laden" aktiviert.
2. Wählen Sie im Menü "Online" den Befehl "Gerät in PG/PC laden".  
Der Dialog "Gerät in PG/PC laden" wird geöffnet.

3. Wählen Sie in der Klappliste "Typ der PG/PC-Schnittstelle" den Typ der Schnittstelle aus, die Sie für den Ladevorgang verwenden möchten.
4. Wählen Sie in der Klappliste "PG/PC-Schnittstelle" die Schnittstelle aus, die verwendet werden soll.
5. Klicken Sie rechts neben der Klappliste "PG/PC-Schnittstelle" auf die Schaltfläche "Schnittstelle konfigurieren", um die Einstellungen für die gewählte Schnittstelle anzupassen.  
Siehe auch: Auto-Hotspot
6. Wählen Sie in der Tabelle der erreichbaren Teilnehmer den Teilnehmer aus, von dem Sie die Projektdaten laden möchten.
7. Klicken Sie auf "Laden".  
Abhängig vom ausgewählten Gerät erscheint ein Dialog, in dem Sie weitere Informationen eingeben müssen, wie z. B. die Position auf dem Baugruppenträger.  
Die Projektdaten des Geräts werden in das Projekt geladen. Sie können es offline bearbeiten und anschließend erneut in das Gerät laden.

## Laden von Gerät

Um nur bestimmte Projektdaten von einem Gerät in Ihr Projekt zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zum Gerät her, von dem Sie Projektdaten laden möchten.  
Siehe auch: Auto-Hotspot
2. Selektieren Sie in der Projektnavigation ein Element, das das Laden von Projektdaten erlaubt.  
Im Menü "Online" wird der Befehl "Laden von Gerät" aktiv.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".  
Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet.
4. Kontrollieren Sie die Meldungen im Dialog "Vorschau für das Laden von Gerät" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".  
Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".  
Der Ladevorgang wird durchgeführt.

## Siehe auch

Allgemeine Informationen zum Laden (Seite 256)

Projektdaten in ein Gerät laden (Seite 257)

Projektdaten auf eine Memory Card laden (Seite 258)

## 7.5.2 Projektdaten vergleichen

### 7.5.2.1 Grundlagen zum Projektdatenvergleich

#### Funktion

Sie können Projektdaten des gleichen Typs miteinander vergleichen, um mögliche Unterschiede zu ermitteln. Grundsätzlich stehen Ihnen die folgenden Vergleichsarten zur Verfügung:

- **Online/Offline-Vergleich**  
Bei dieser Vergleichsart werden die Objekte eines Geräts mit den Objekten eines Projekts verglichen. Dazu muss eine Online-Verbindung zum Gerät aufgebaut werden.
- **Offline/Offline-Vergleich**  
Bei dieser Vergleichsart können Sie Objekte vergleichen, die aus Projekten oder Bibliotheken stammen. Dabei können Sie entscheiden, ob der Vergleich automatisch für alle Objekte durchgeführt wird oder ob Sie einzelne Objekte manuell vergleichen möchten.
- **Detailvergleich**  
Für einige Objekte, z. B. Bausteine, können Sie über den Online/Offline- und Offline/Offline-Vergleich hinaus einen Detailvergleich durchführen. Dabei werden die verglichenen Objekte nebeneinander geöffnet und die Unterschiede werden gekennzeichnet.

Ein einfacher Online/Offline-Vergleich findet bereits statt, wenn Sie eine Online-Verbindung herstellen. Dabei werden vergleichsfähige Objekte in der Projektnavigation mit Symbolen gekennzeichnet, die das Vergleichsergebnis repräsentieren.

Der normale Online/Offline- und Offline/Offline-Vergleich wird im Vergleichseditor durchgeführt. Im Vergleichseditor können Sie auch Aktionen für nicht identische Objekte auswählen.

---

#### Hinweis

- Nicht alle Objekte ermöglichen jede Vergleichsart. Für welche Projektdaten Sie welchen Vergleich durchführen können, ist von den installierten Produkten abhängig.
  - Übersetzen Sie Ihr Anwenderprogramm, bevor Sie einen Vergleich oder Detailvergleich starten. Nach jeder Änderung des Programms während eines Vergleichs ist dies zu wiederholen, bevor Sie das Vergleichsergebnis aktualisieren. Damit stellen Sie sicher, dass der Vergleich den aktuellen Status anzeigt.
- 

#### Siehe auch

Online/Offline-Vergleich durchführen (Seite 262)

Offline/Offline-Vergleich durchführen (Seite 262)

Vergleichseditor verwenden (Seite 264)

Detailvergleich durchführen (Seite 269)

### 7.5.2.2 Online/Offline-Vergleich durchführen

#### Voraussetzung

Die Projektnavigation ist geöffnet.

#### Vorgehen

Um einen Online/Offline-Vergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Online/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Online".
3. Falls Sie bisher noch keine Online-Verbindung zu diesem Gerät hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Verbinden".  
Die Online-Verbindung wird hergestellt und der Vergleichseditor geöffnet.

#### Ergebnis

Alle Objekte, die online und offline vorhanden sind, werden angezeigt. Den Status der Objekte können Sie an den Symbolen im Vergleichseditor und in der Projektnavigation ablesen. Im Vergleichseditor können Sie nun abhängig vom Status der Objekte bestimmte Aktionen für die Objekte festlegen.

#### Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 261)

Offline/Offline-Vergleich durchführen (Seite 262)

Vergleichseditor verwenden (Seite 264)

Detailvergleich durchführen (Seite 269)

### 7.5.2.3 Offline/Offline-Vergleich durchführen

Beim Offline/Offline-Vergleich können Sie die Projektdaten zweier Geräte innerhalb eines Projekts oder aus verschiedenen Projekten oder aus der Bibliothek vergleichen. Dabei können Sie entscheiden, ob der Vergleich automatisch für alle Objekte durchgeführt wird oder ob Sie einzelne Objekte manuell vergleichen möchten.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

#### Voraussetzung

Die Projektnavigation ist geöffnet.

### **Automatischen Offline/Offline-Vergleich durchführen**

Um einen automatischen Offline/Offline-Vergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläch im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.  
Alle vorhandenen Objekte der ausgewählten Geräte werden abhängig von den Einstellungen des Vergleichseditors angezeigt. Den Status der Objekte können Sie an den Symbolen im Vergleichseditor ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen.

### **Manuellen Offline/Offline-Vergleich durchführen**

Um einen manuellen Offline/Offline-Vergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich.
5. Selektieren Sie die Objekte, die Sie miteinander vergleichen möchten.  
Der Eigenschaftsvergleich wird angezeigt. Den Status der Objekte können Sie an den Symbolen ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen.

### **Siehe auch**

Grundlagen zum Projektdatenvergleich (Seite 261)

Online/Offline-Vergleich durchführen (Seite 262)

Vergleichseditor verwenden (Seite 264)

Detailvergleich durchführen (Seite 269)

### 7.5.2.4 Vergleichseditor verwenden

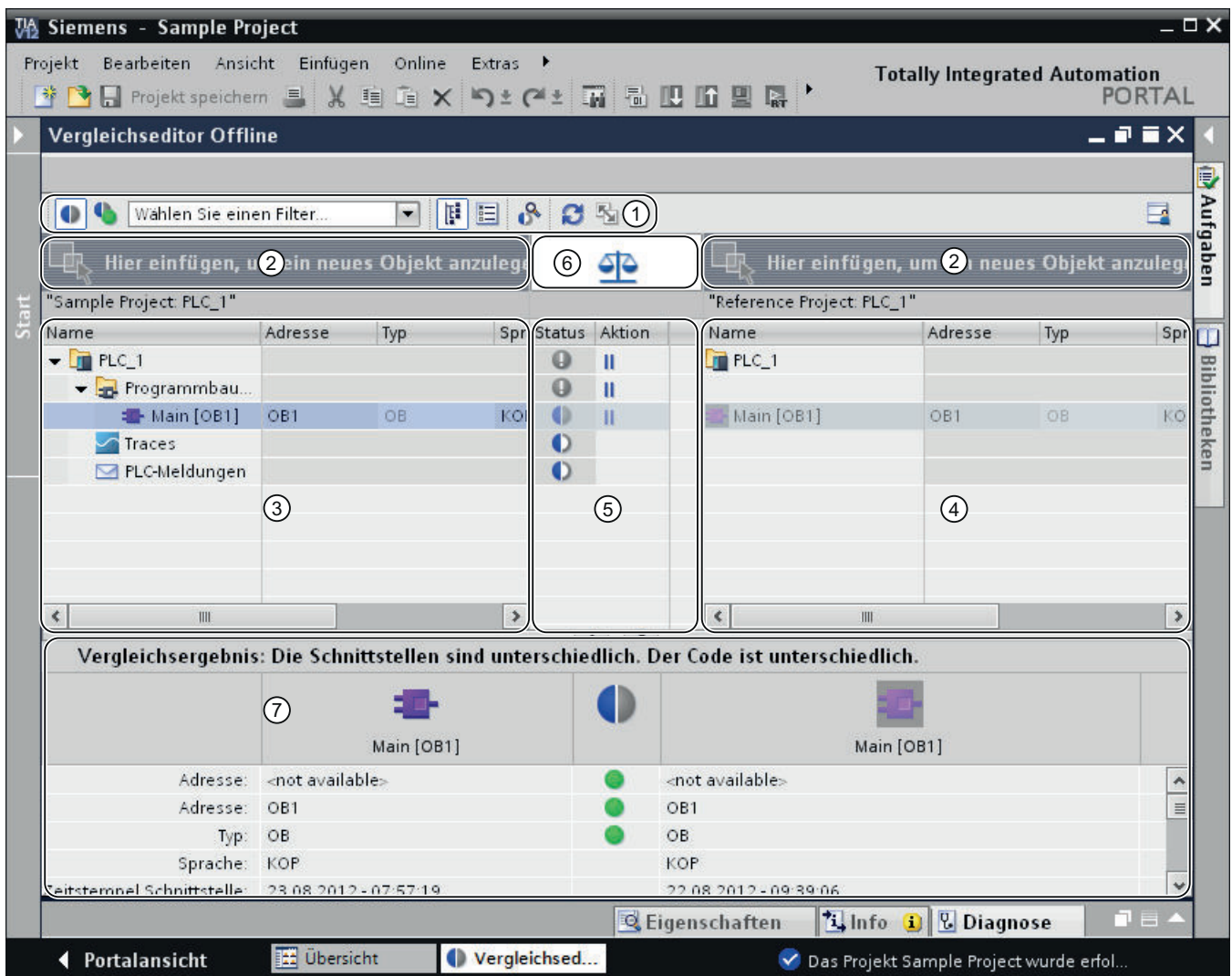
#### Übersicht über den Vergleichseditor

#### Funktion

Der Vergleichseditor stellt die Ergebnisse eines Online/Offline- und Offline/Offline-Vergleichs in einer tabellarischen Übersicht dar. Zusätzlich können Sie festlegen, welche Aktionen für nicht identische Vergleichsobjekte ausgeführt werden sollen.

#### Komponenten des Vergleichseditors

Das folgende Bild zeigt die Komponenten des Vergleichseditors am Beispiel eines manuellen Offline/Offline-Vergleichs:



① Funktionsleiste des Vergleichseditors

- ② Drop-Flächen (nur Offline/Offline-Vergleich)
- ③ Linke Vergleichstabelle
- ④ Rechte Vergleichstabelle
- ⑤ Status- und Aktionsbereich
- ⑥ Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich (nur Offline/Offline-Vergleich)
- ⑦ Eigenschaftsvergleich

### **Funktionsleiste des Vergleichseditors**

Über die Funktionsleiste haben Sie Zugriff auf folgende Funktionen des Vergleichseditors:

- Nur unterschiedliche Objekte anzeigen  
Sie können identische Objekte ausblenden, um die Übersichtlichkeit zu verbessern.
- Identische und unterschiedliche Objekte anzeigen  
Sie können identische Objekte einblenden, wenn Sie den Vergleich vollständig angezeigt bekommen möchten.
- Umfang des Vergleichs  
Sie können festlegen, welche Objekte verglichen werden sollen.
- Detailvergleich starten  
Sie können für Objekte einen Detailvergleich starten, um sich die einzelnen Unterschiede anzeigen zu lassen. Diese Funktion steht jedoch nicht für alle Objekte zur Verfügung.
- Vergleichsergebnisse aktualisieren  
Nachdem Sie Objekte geändert haben, können Sie mit dieser Funktion die Vergleichsergebnisse aktualisieren.
- Nicht identische Objekte synchronisieren  
Sie können nicht identische Objekte durch bestimmte Aktionen synchronisieren.
- Ansicht ändern  
Sie können zwischen einer hierarchischen und einer flachen Ansicht wählen. In der hierarchischen Ansicht werden die Geräte in ihrer Struktur dargestellt, in der flachen Ansicht werden die Objekte der Geräte ohne Struktur aufgelistet.

### **Drop-Flächen**

Bei einem Offline/Offline-Vergleich können Sie in die Drop-Flächen die Geräte ziehen, die Sie vergleichen möchten. Die zu vergleichenden Geräte können dabei aus einem Projekt, aus Referenzprojekten, aus der Projektbibliothek oder aus globalen Bibliotheken stammen. Beachten Sie jedoch, dass Sie komplette Bibliotheken nur in die rechte Drop-Fläche ziehen können.

### **Vergleichstabellen**

In den Vergleichstabellen werden die Objekte der Geräte angezeigt, die miteinander verglichen werden.

Die folgende Tabelle zeigt die Bedeutung der Spalten der Vergleichstabellen:

| Spalte                     | Erläuterung  |
|----------------------------|--|
| Name                       | Name des Vergleichsobjekts   |
| Kommentar                  | Kommentar zum Vergleichsobjekt   |
| Titel                      | Titel des Vergleichsobjekts  |
| Adresse                    | Adresse des Vergleichsobjekts  |
| Typ                        | Typ des Vergleichsobjekts  |
| Sprache                    | Programmiersprache, die für das Vergleichsobjekt eingestellt ist.          |
| Zeitstempel Schnittstelle  | Zeitpunkt der letzten Änderung der Bausteinschnittstelle                   |
| Zeitstempel Code           | Zeitpunkt der letzten Änderung des Quellcodes                              |
| Autor                      | Name des Autors des Vergleichsobjekts                                      |
| Version                    | Version des Vergleichsobjekts  |
| Familie                    | Name der Objektfamilie   |
| Ladespeicher               | Speicherauslastung des Ladespeichers des Vergleichsobjekts                 |
| Arbeitsspeicher            | Speicherauslastung des Arbeitsspeichers des Vergleichsobjekts              |
| Datum der letzten Änderung | Zeitpunkt der letzten Änderung   |
| Programmsignatur           | Programmsignatur des Vergleichsobjekts (SIMATIC Safety)                    |
| Interface (Prüfsumme)      | Prüfsumme der Bausteinschnittstelle des Vergleichsobjekts (SIMATIC Safety) |

Standardmäßig sind nicht alle Spalten eingeblendet. Sie können jedoch wie in allen Tabelleneditoren die Spalten nach Bedarf ein- oder ausblenden und nach den einzelnen Spalten sortieren.




### Status- und Aktionsbereich

Der Status- und Aktionsbereich bietet Ihnen folgende Möglichkeiten:




- Bei einem Offline/Offline-Vergleich können Sie zwischen automatischem und manuellem Vergleich umschalten.
- Sie können sich die Ergebnisse eines automatischen Vergleichs ansehen. Die Ergebnisse werden mit Symbolen angezeigt.
- Sie können Aktionen für nicht identische Objekte festlegen.

### Status- und Aktionssymbole









Die folgende Tabelle zeigt die Symbole für die Vergleichsergebnisse eines Online/Offline-Vergleichs:

| Symbol  | Erläuterung  |
|---|--|
|  | Ordner enthält Objekte, deren Online- und Offline-Version verschieden sind |
|  | Vergleichsergebnis ist nicht bekannt                                       |
|  | Online- und Offline-Version des Objekts sind identisch                     |







| Symbol  | Erläuterung  |
|---|--|
|  | Online- und Offline-Version des Objekts sind verschieden |
|  | Objekt nur offline vorhanden                             |
|  | Objekt nur online vorhanden                              |

Die folgende Tabelle zeigt die Symbole für die Vergleichsergebnisse eines Offline/Offline-Vergleichs:

| Symbol  | Erläuterung   |
|---|---|
|    | Ausgangsprogramm  |
|    | Verglichene Version   |
|    | Ordner enthält Objekte, deren verglichenen Versionen verschieden sind |
|    | Ergebnis des Offline/Offline-Vergleichs ist nicht bekannt             |
|   | Die verglichenen Versionen des Objekts sind identisch                 |
|  | Die verglichenen Versionen des Objekts sind verschieden               |
|  | Objekt nur im Ausgangsprogramm vorhanden                              |
|  | Objekt nur in der verglichenen Version vorhanden                      |

Die folgende Tabelle zeigt die Symbole für die möglichen Aktionen:

| Symbol  | Erläuterung  |
|---|--|
|  | Keine Aktion   |
|  | Objekt der verglichenen Version mit dem Objekt aus dem Ausgangsprogramm überschreiben  |
|  | Objekt des Ausgangsprogramms mit dem Objekt aus der verglichenen Version überschreiben |
|  | Unterschiedliche Aktionen für die Vergleichsobjekte innerhalb des Ordners              |

## Eigenschaftsvergleich

Der Eigenschaftsvergleich vergleicht die Eigenschaften der selektierten Vergleichsobjekte. Das Ergebnis wird mit Symbolen angezeigt. Beim manuellen Vergleich erfolgt nur ein Eigenschaftsvergleich, sodass der Status- und Aktionsbereich leer bleibt. Beim automatischen Vergleich können Sie den Eigenschaftsvergleich zusätzlich zum Vergleich in den Vergleichstabellen verwenden.

## Siehe auch

- Grundlagen zum Projektdatenvergleich (Seite 261)
- Ansicht ändern (Seite 273)
- Online/Offline-Vergleich durchführen (Seite 262)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Ansicht im Vergleichseditor filtern (Seite 268)
- Vergleichsergebnisse aktualisieren (Seite 270)
- Nicht identische Objekte synchronisieren (Seite 271)

## Ansicht im Vergleichseditor filtern

Um die Übersichtlichkeit des Vergleichseditors zu verbessern, können Sie die Ansicht über folgende Filtermechanismen einschränken:

- **Identische Vergleichsobjekte ausblenden**  
Sie können Vergleichsobjekte, deren Online/Offline- bzw. Offline/Offline-Versionen identisch sind, ausblenden. Umgekehrt können Sie solche Vergleichsobjekte, wenn Sie ausgeblendet sind, jederzeit wieder einblenden.
- **Angezeigte Objekte**  
Sie können festlegen, für welche Objekte die Vergleichsergebnisse angezeigt werden sollen.

## Voraussetzung

Der Vergleichseditor ist geöffnet.

### **Identische Vergleichsobjekte ausblenden**

Um identische Objekte auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Nur unterschiedliche Objekte anzeigen".  
Es werden nur die Elemente angezeigt, die online und offline unterschiedlich sind.

### **Identische Vergleichsobjekte einblenden**

Um die identischen Objekte wieder einzublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Identische und unterschiedliche Objekte anzeigen".  
Es werden alle Elemente angezeigt.

### **Angezeigte Objekte auswählen**

Um die Objekte auszuwählen, für die die Vergleichsergebnisse angezeigt werden sollen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Pfeilschaltfläche der Klappliste.
2. Wählen Sie das gewünschte Objekt aus.

### **Siehe auch**

- Ansicht ändern (Seite 273)
- Online/Offline-Vergleich durchführen (Seite 262)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Übersicht über den Vergleichseditor (Seite 264)
- Vergleichsergebnisse aktualisieren (Seite 270)
- Nicht identische Objekte synchronisieren (Seite 271)

### **Detailvergleich durchführen**

---

#### **Hinweis**

Nicht alle Objekte ermöglichen einen Detailvergleich. Für welche Projektdaten Sie einen Detailvergleich durchführen können, ist von den installierten Produkten abhängig.

---

## Vorgehen

Um einen Detailvergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Führen Sie zunächst einen Online/Offline- oder einen Offline/Offline-Vergleich durch. Der Vergleichseditor wird geöffnet.

---

### Hinweis

Einen Detailvergleich können Sie nur für Objekte durchführen, die sowohl in der linken als auch in der rechten Vergleichstabelle vorhanden sind.

---

2. Selektieren Sie im Vergleichseditor das Objekt, für das Sie einen Detailvergleich durchführen möchten.
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten".

## Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 261)

Online/Offline-Vergleich durchführen (Seite 262)

Offline/Offline-Vergleich durchführen (Seite 262)

Ansicht ändern (Seite 273)

## Vergleichsergebnisse aktualisieren

Sobald Sie ein Objekt verändern, sind die Vergleichsergebnisse nicht mehr gültig und müssen aktualisiert werden.

---

### Hinweis

Bei einem Online/Offline-Vergleich kann es vorkommen, dass durch Änderungen im Gerät der Vergleichseditor vom System automatisch aktualisiert wird, falls Objekte des Vergleichs von der Änderung betroffen sind. Dies kann folgende Auswirkungen haben:

- Eventuell werden einige der von Ihnen festgelegten Aktionen ungültig, z. B. falls das Objekt im Gerät nicht mehr existiert. Objekte mit solchen ungültigen Aktionen werden hervorgehoben, damit Sie eine neue, gültige Aktion festlegen können.
  - Möglicherweise wird auch Ihre Auswahl aufgehoben, die Sie vor dem automatischen Aktualisieren getroffen haben.
- 

## Voraussetzung

Der Vergleichseditor ist geöffnet.

## Vorgehen

Um die Vergleichsergebnisse zu aktualisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Anzeige aktualisieren". Die Vergleichsergebnisse werden aktualisiert.

---

### **Hinweis**

Beachten Sie, dass Ihnen die Schaltfläche "Anzeige aktualisieren" nicht zur Verfügung steht, während der Vergleichseditor Inhalte lädt oder synchronisiert.

---

## Siehe auch

- Ansicht ändern (Seite 273)
- Online/Offline-Vergleich durchführen (Seite 262)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Übersicht über den Vergleichseditor (Seite 264)
- Ansicht im Vergleichseditor filtern (Seite 268)
- Nicht identische Objekte synchronisieren (Seite 271)

## Nicht identische Objekte synchronisieren

### Aktionen festlegen

Wenn Sie einen Vergleich durchgeführt haben, können Sie im Vergleichseditor festlegen, welche Aktionen auf nicht identische Objekte angewendet werden sollen. Für identische Objekte können Sie keine Aktionen wählen.

Bei einem Online/Offline-Vergleich sind nur Synchronisationsaktionen in eine Richtung zulässig, damit die Programmkonsistenz erhalten bleibt. So können Sie z. B. mehrere Bausteine in ein Gerät oder von einem Gerät laden, aber Sie können in einer Synchronisationsaktion keine gemischten Ladeaktionen durchführen. Dabei bestimmt die erste Aktion, die Sie im Vergleichseditor einstellen, die Synchronisationsrichtung. Stellen Sie also z. B. bei einem Baustein ein, dass der Offline-Baustein in das Gerät geladen werden soll, können Sie über eine Synchronisierung die anderen Objekte auch nur in das Gerät laden. Um doch wieder Objekte aus dem Gerät zu laden, stellen Sie zunächst die Option "Keine Aktion"

ein, anschließend können Sie die Aktion wieder nach Bedarf einstellen. Oder Sie führen einen neuen Vergleich durch.

---

### Hinweis

Beachten Sie beim Festlegen von Aktionen folgende CPU-spezifischen Besonderheiten:

- S7-300/400:
    - Sie können Aktionen für den Ordner "Programmbausteine", für Ordner, die Sie selbst erstellt haben, oder für einzelne Bausteine festlegen.
    - Das Laden von SCL- und GRAPH-Bausteinen aus dem Gerät in das Offline-Projekt ist nicht möglich.
  - S7-1200/1500:
    - Sie können Aktionen für den Ordner "Programmbausteine", für Ordner, die Sie selbst erstellt haben, oder für einzelne Bausteine festlegen. Wenn Sie einen Online/Offline-Vergleich durchgeführt haben und als Aktion das Laden in das Gerät wählen, wird ein konsistentes Laden durchgeführt. Wenn Sie dagegen als Aktion das Objekt vom Gerät in das Projekt laden, können auch einzelne Bausteine geladen werden.
    - Das Laden von SCL-Bausteinen aus dem Gerät in das Offline-Projekt ist nicht möglich.
- 

### Voraussetzung

Der Vergleichseditor ist geöffnet.

### Vorgehen

Um für ein nicht identisches Objekt eine Aktion auszuwählen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Status- und Aktionsbereich in der Spalte "Aktion" doppelt in die Zelle des Objekts, für das Sie eine Aktion festlegen möchten.  
Die Zelle ändert sich in eine Klappliste.
2. Klicken Sie in die Klappliste.
3. Wählen Sie die gewünschte Aktion aus.  
Für das Objekt wird beim nächsten Synchronisieren die eingestellte Aktion ausgeführt.  
Wenn Sie die zuvor eingestellte Auswahl der Aktion unbeabsichtigt geändert haben, können Sie die Änderung bis zur nächsten Synchronisation rückgängig machen.
4. Um die zuvor eingestellte Auswahl der Aktion wiederherzustellen, klicken Sie mit der rechten Maustaste auf das Objekt oder den Ordner.
5. Wählen Sie im Kontextmenü den Befehl "Letzte Auswahl wieder herstellen".

### Siehe auch

Übersicht über den Vergleichseditor (Seite 264)

Ansicht im Vergleichseditor filtern (Seite 268)

Vergleichsergebnisse aktualisieren (Seite 270)

Objekte synchronisieren (Seite 273)

## **Objekte synchronisieren**

Durch das Synchronisieren werden die Aktionen ausgeführt, die Sie für nicht identische Objekte festgelegt haben. Beachten Sie dabei jedoch, dass Sie bei einem Online/Offline-Vergleich in einer Synchronisationsaktion nur Aktionen in eine Richtung ausführen können.

### **Voraussetzung**

- Der Vergleichseditor ist geöffnet.
- Die gewünschten Aktionen sind ausgewählt.

### **Vorgehen**

Um die Objekte zu synchronisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Aktionen ausführen".

### **Ergebnis**

Die Aktionen, die Sie für die Objekte festgelegt haben, werden ausgeführt.

### **Siehe auch**

Übersicht über den Vergleichseditor (Seite 264)  
Ansicht im Vergleichseditor filtern (Seite 268)  
Vergleichsergebnisse aktualisieren (Seite 270)  
Aktionen festlegen (Seite 271)

### **Ansicht ändern**

Sie können zwischen einer hierarchischen und einer flachen Ansicht wählen. In der hierarchischen Ansicht werden die Geräte in ihrer Struktur dargestellt, in der flachen Ansicht werden die Objekte der Geräte ohne Struktur aufgelistet.

### Hierarchische Ansicht einstellen

Um die hierarchische Ansicht einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Vergleichseditors auf die Schaltfläche "Hierarchisch anzeigen".

### Flache Ansicht einstellen

Um die flache Ansicht einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Vergleichseditors auf die Schaltfläche "Flach anzeigen".

### Siehe auch

Grundlagen zum Projektdatenvergleich (Seite 261)

Übersicht über den Vergleichseditor (Seite 264)

Ansicht im Vergleichseditor filtern (Seite 268)

Detailvergleich durchführen (Seite 269)

Vergleichsergebnisse aktualisieren (Seite 270)

Nicht identische Objekte synchronisieren (Seite 271)

## 7.5.3 Projektdaten schützen

### 7.5.3.1 Schutzkonzept für Projektdaten

#### Einführung

Sie haben die Möglichkeit Ihre Projektdaten vor unberechtigtem Zugriff zu schützen. Dazu gehören z. B.:

- Zugriffsschutz für Geräte
- Kopier- und Anzeigeschutz von Objekten
- Einschränkungen beim Ausdruck von know-how-geschützten Objekten

Wenn Sie ein Objekt mit Know-how-Schutz versehen, dann bleibt dieser Schutz auch nach dem Einfügen in eine Bibliothek bestehen. Beachten Sie, dass nicht für alle Objekte jeder Schutzmechanismus zur Verfügung steht. Wie Sie bestimmte Objekte schützen können, finden Sie in der Hilfe zum Produkt.

#### Zugriffsrechte für Geräte zurücknehmen

Wenn Sie eine Funktion ausführen möchten, die über die Schutzstufe des Geräts passwortgeschützt ist, erhalten Sie einen Dialog zur Passwortabfrage. Bei korrekter Eingabe



des Passworts können Sie die gewünschte Funktion ausführen. Das Zugriffsrecht auf das Gerät bleibt bestehen, bis Sie das TIA-Portal schließen.

Wenn Sie den Passwortschutz bei geöffnetem TIA-Portal wieder aktivieren möchten, können Sie die Zugriffsrechte für ein Gerät explizit wieder zurücknehmen. Dadurch können bestimmte Funktionen für das geschützte Gerät erst nach einer erneuten Eingabe des korrekten Passworts ausgeführt werden. Für welche Funktionen ein Passwort eingegeben werden muss, legen Sie bei der Parametrierung der Schutzstufe des Geräts fest.

## **Siehe auch**

Projektdaten drucken (Seite 293)

### **7.5.3.2 Zugriffsrechte für Geräte zurücknehmen**

#### **Voraussetzung**

- Für das Gerät wurde eine Schutzstufe eingestellt.
- Eine geschützte Funktion für das Gerät wurde über eine Passworteingabe freigegeben.

#### **Vorgehen**

Um die Zugriffsrechte für das Gerät wieder zurückzunehmen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Gerät, für das Sie die Zugriffsrechte zurücknehmen möchten.
2. Wählen Sie im Menü "Online" den Befehl "Zugriffsrechte löschen".

#### **Ergebnis**

Die Zugriffsrechte werden zurückgenommen und für alle passwortgeschützten Funktionen des Geräts wird ab jetzt wieder der Dialog zur Passworteingabe geöffnet. Die Funktion kann nur durch die Eingabe des korrekten Passworts ausgeführt werden.

Falls für das Gerät eine Online-Verbindung besteht, wird die Online-Verbindung getrennt.

## **Siehe auch**

Schutzkonzept für Projektdaten (Seite 274)

## 7.5.4 Projektinhalte drucken

### 7.5.4.1 Projektdokumentation drucken

#### Dokumentationsfunktion

##### Einführung

Nach dem Erstellen eines Projekts lassen sich die Inhalte in übersichtlicher Form drucken. Sie können entweder das gesamte Projekt drucken oder einzelne Objekte. Ein sinnvoll strukturierter Ausdruck erleichtert sowohl die weitere Bearbeitung des Projekts als auch Service-Arbeiten. Der Ausdruck kann ebenso als Präsentation für Ihre Kunden dienen oder als komplette Anlagendokumentation.

Sie können das Projekt in Form von standardisierten Schaltbüchern aufbereiten und mit einem einheitlichen Layout ausdrucken. Den Umfang des Drucks können Sie eingrenzen. Entweder Sie drucken das gesamte Projekt, einzelnen Objekten mit ihren Eigenschaften oder einen kompakten Überblick des Projekts. Außerdem können Sie die Inhalte eines geöffneten Editors drucken.

#### Den Ausdruck mit Rahmen und Deckblättern verbessern

Sie haben die Möglichkeit, das Aussehen des Drucks nach eigenen Vorgaben zu gestalten, z. B. um Ihr eigenes Firmenlogo einzufügen oder um das Cooperate Design Ihres Unternehmens in der Projektdokumentation umzusetzen. Sie können beliebig viele Design-Varianten als Rahmen und Deckblätter anlegen. Die Rahmen und Deckblätter werden in der Projektnavigation unter dem Eintrag "Dokumentationseinstellungen" abgelegt und sind Teil des Projekts. Innerhalb der Rahmen und Deckblätter können Sie Platzhalter für Daten aus zuvor hinterlegten Dokument-Informationen einfügen. Diese werden beim Druck automatisch mit den entsprechenden Meta-Daten befüllt.

Möchten Sie auf die freie Gestaltung verzichten, stehen fertige Rahmen und Deckblätter zur Verfügung. Darunter befinden sich auch Vorlagen nach dem ISO-Standard für technische Produktdokumentation.

#### Modularer Aufbau eines Ausdrucks

Ein Ausdruck besteht allgemein aus folgenden Bestandteilen:

- Deckblatt (nur beim Druck aus der Projektnavigation)
- Inhaltsverzeichnis (nur beim Druck aus der Projektnavigation)
- Name und Pfad eines Objekts innerhalb der Projektnavigation
- Objektdaten

Der Ausdruck des Deckblattes oder des Inhaltsverzeichnisses kann im Dialog "Drucken" deaktiviert werden.

## Siehe auch

- Rahmen anlegen (Seite 282)
- Deckblatt anlegen (Seite 283)
- Deckblätter und Rahmen bearbeiten (Seite 284)
- Dokument-Informationen eingeben (Seite 280)
- Druckfunktion für Modulbeschriftungsstreifen (Seite 295)

## Ausdruck von Projekthinhalten

### Verfügbarkeit der Druckfunktion

Folgende Inhalte können gedruckt werden:

- Ein gesamtes Projekt in der Projektnavigation
- Ein oder mehrere einem Projekt zugehörige Objekte in der Projektnavigation
- Inhalte eines Editors
- Tabellen
- Bibliotheken
- Diagnosesicht des Inspektorfensters

In folgenden Bereichen ist der Druck nicht möglich:

- Portalansicht
- Detailsicht
- Übersichtsfenster
- Vergleichseditor
- Alle Register des Inspektorfensters außer der Diagnosesicht
- Alle Task Cards außer der Bibliotheken
- Überwiegende Zahl der Dialoge
- Nicht zum Projekt gehörige Eigenschaften und Geräte des PG/PC, wie z. B. die Online-Zugänge und angeschlossene Card Reader.

### Umfang des Ausdrucks

Um drucken zu können, muss mindestens ein druckbares Element selektiert sein.

Wird ein selektiertes Objekt gedruckt, so werden alle unterlagerten Objekte mitgedruckt. Wird beispielsweise ein Gerät in der Projektnavigation selektiert, werden alle dazugehörigen Daten ebenfalls mit ausgedruckt. Wenn Sie das gesamte Projekt in der Projektnavigation für den Druck auswählen, dann werden alle Projekthinhalte außer der grafischen Sichten gedruckt. Diese müssen separat gedruckt werden. Einträge der Projektnavigation, die nicht zum Projekt gehören, können nicht gedruckt werden. Dazu zählen beispielsweise die Online-Zugänge sowie die angeschlossenen Card-Reader und USB-Speicher.

Beim Drucken von Tabelleninhalten werden alle Zeilen einer Tabelle gedruckt, in denen eine Zelle markiert ist. Um eine oder mehrere Tabellenspalten zu drucken, müssen die gewünschten Spalten markiert sein. Sind keine einzelnen Zellen oder Spalten markiert, wird die gesamte Tabelle gedruckt.

### Beschränkungen beim Druck

Generell können alle Objekte gedruckt werden, die auch an der Benutzeroberfläche sichtbar gemacht werden können. Umgekehrt heißt dies, dass Objekte, auf die Sie keinen Zugriff haben, auch nicht gedruckt werden können. Mögliche Gründe für das Scheitern eines Drucks können beispielsweise sein:

- Es ist keine gültige Lizenz zur Anzeige eines Objekts vorhanden.
- Es liegt keine Gerätebeschreibung für ein Objekt vor.
- Eine notwendige Softwarekomponente zur Anzeige eines Objekts ist nicht installiert.

### Siehe auch

Projektdaten drucken (Seite 293)

### Einstellungen für den Druck ändern

### Einstellungen für den Druck ändern

Sie können generelle Einstellungen für den Druck vornehmen, welche auch nach dem Schließen und erneuten Öffnen des TIA-Portals erhalten bleiben. Einige Einstellungen sind abhängig von den installierten Produkten. Folgende Einstellungen sind jedoch immer möglich:

### Tabellendaten immer in Wertpaaren drucken

Ist diese Funktion aktiviert, werden Tabellen nicht in tabellarischer Form gedruckt, sondern als Paar zwischen Schlüssel und Wert.

Beispiel:

| Objektname | Eigenschaft 1 | Eigenschaft 2 |
|------------|---------------|---------------|
| Objekt A   | Wert A1       | Wert A2       |
| Objekt B   | Wert B1       | Wert B2       |

In diesem Fall sieht der Ausdruck wie folgt aus:

#### Objekt A

Eigenschaft 1: Wert A1

Eigenschaft 2: Wert A2

#### Objekt B

Eigenschaft 1: Wert B1

Eigenschaft 2: Wert B2

## Maskeneditoren drucken

- Daten immer in Tabellen drucken  
Alle Parameter von Technologieobjekten werden in Form einer Tabelle gedruckt.
- Maskengrafiken drucken, wenn möglich  
Wenn der verwendete Editor diese Funktion unterstützt, werden die Inhalte des Editors nicht als Tabelle, sondern als vollständige Grafik wie auf dem Bildschirm gedruckt.

## Vorgehen

Um die Druckeinstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie die Gruppe "Allgemein".
3. Aktivieren Sie im Bereich "Druckeinstellungen" die gewünschten Voreinstellungen.  
Die Änderungen werden sofort übernommen und bleiben auch nach dem Schließen des TIA-Portals projektübergreifend erhalten.

## Siehe auch

Übersicht der Druckeinstellungen (Seite 174)

## Festlegen des Drucklayouts

### Drucklayout festlegen

Wenn Sie nicht auf vorgefertigte Druckvorlagen zurückgreifen möchten, können Sie selbst ein Deckblatt oder das Layout der einzelnen Seiten festlegen. Ihre Entwürfe werden zusammen mit dem jeweiligen Projekt gespeichert.

Unter der Gruppe "Dokument-Information" in der Projektnavigation finden Sie Ihre Entwürfe für das Deckblatt und Vorlagen für das Seitenlayout. Darüber hinaus finden Sie dort unter dem Eintrag "Dokument-Information" Meta-Daten zum Projekt. Beim anschließenden Druck können Sie im Dialog "Drucken" aus den gespeicherten Deckblättern, Vorlagen für das Seitenlayout und den Meta-Daten ein individuelles Erscheinungsbild zusammenstellen.

### Deckblatt gestalten

Das Deckblatt lässt sich individuell gestalten. Sie können eine Hintergrundgrafik einfügen und Platzhalter für Text auf der Seite vorsehen. Die Platzhalter werden beim Druck automatisch mit Daten aus Dokument-Informationen befüllt.

Die Deckblätter befinden sich in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Deckblätter".

## Inhaltsseite gestalten

Die laufenden Seiten eines Ausdrucks können folgende Elemente enthalten:

- Rahmen mit statischem Inhalt, z. B. Firmenlogo
- Platzhalter für Text, z. B. den Namen des Projekts, die Seitennummer oder den Startzeitpunkt des Drucks  
Etliche Werte für die einzelnen Platzhalter können in einer Dokument-Information festgelegt werden. Andere Werte, wie z. B. der Projektname, sind vorgegeben und werden automatisch beim Druck eingefügt.
- Fußnote  
Die Fußnote wird immer unterhalb des Bereichs für den Inhalt ausgegeben.
- Bereich für den Inhalt  
Sie können einen Bereich festlegen, innerhalb dessen der Druckinhalt eingebettet werden soll.

Die Gestaltung der Inhaltsseiten wird in Rahmen gespeichert. Die einzelnen Rahmen befinden sich in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Rahmen".

## Dokument-Informationen eingeben

Sie können zu jedem Projekt Meta-Daten in Dokument-Informationen eingeben. Außerdem werden in den Dokument-Informationen ein Rahmen für den Druck und ein Deckblatt festgelegt. Sie können bei Bedarf verschiedene Dokument-Informationen anlegen, um beim Druck schnell zwischen verschiedenen Dokument-Informationen mit unterschiedlichen Informationen, Rahmen, Deckblättern, Seitengrößen und Seitenausrichtungen wechseln zu können. Dies ist beispielsweise auch dann hilfreich, wenn Sie in verschiedenen Sprachen drucken möchten und für jede Sprache unterschiedliche Dokument-Informationen vorgesehen haben.

Im Dokumentationseditor können Sie Platzhalter auf dem Deckblatt oder im Rahmen der laufenden Seiten festlegen. Diese Platzhalter können beim Druck automatisch mit den Meta-Daten aus Dokument-Informationen befüllt werden.

Die verschiedenen Dokument-Informationen sind somit ein Bestandteil der Druckfunktion und bestimmen das Drucklayout sowie die Druckinhalte.

## Vorgehen

Um Meta-Daten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unter "Dokument-Information > Dokument-Information" auf "Neue Dokument-Information hinzufügen", um neue Dokument-Informationen anzulegen.  
Die neuen Dokument-Informationen werden angelegt und sofort geöffnet.
2. Geben Sie im Feld "Name" einen Namen für das Set ein.
3. Befüllen Sie die einzelnen Felder mit den Meta-Daten zum Projekt.

## Deckblätter und Rahmen verwalten

### Einsetzen von Deckblättern und Rahmen

#### Nutzen von Deckblättern

Sie können dem Druck Ihrer Anlagendokumentation ein professionelles Aussehen verleihen, indem Sie ihr ein Deckblatt voranstellen. Das Deckblatt können Sie frei gestalten oder auf vorgefertigte Deckblätter zurückgreifen. Vorgefertigte Deckblätter lassen sich anpassen und ebenfalls wieder als Vorlage ablegen.

Deckblätter können in globalen Bibliotheken gespeichert werden und stehen so projektübergreifend zur Verfügung.

Deckblätter sind immer als rechte Druckseite vorgesehen.

#### Nutzen von Rahmen

Die laufenden Seiten der Anlagendokumentation können Sie in einen stets gleichbleibenden Seitenrahmen einbetten. Der Rahmen kann Platzhalter für Meta-Daten zum Projekt enthalten, die in Dokument-Informationen gespeichert sind. Er kann aber auch frei gestaltbare grafische Elemente enthalten.

Sie können eigene Rahmen anlegen, aber auch auf vorgefertigte Seitenrahmen zurückgreifen. Vorgefertigte Seitenrahmen können Sie anpassen und im Anschluss selbst wieder als neuen Rahmen abspeichern.

Rahmen können genauso wie Deckblätter in globalen Bibliotheken gespeichert werden und stehen so projektübergreifend zur Verfügung.

Rahmen sind immer als rechte Druckseiten vorgesehen.

## Deckblätter und Vorlagen in der Projektnavigation

Zum Projekt gehörige Deckblätter und Rahmen werden in der Projektnavigation unter dem Eintrag "Dokument-Information" abgelegt. Darunter liegen jeweils Ordner für Rahmen und Deckblätter.

Folgende Operationen stehen Ihnen für Deckblätter und Rahmen in der Projektnavigation zur Verfügung:

- Erstellen eigener Unterordner
- Kopieren und Einfügen
- Einfügen von Deckblättern und Rahmen aus der Systembibliothek "Dokumentationsvorlagen"
- Kopieren von Deckblättern und Vorlagen in eine globale Bibliothek

## Deckblätter und Vorlagen in Bibliotheken

In der Systembibliothek "Dokumentationsvorlagen" sind einige Deckblätter und Vorlagen gespeichert, die in jedem Projekt zur Verfügung stehen. Die Deckblätter und Vorlagen lassen

sich von dort per Drag & Drop in die Projektnavigation ziehen. In der Projektnavigation können Sie die Deckblätter und Vorlagen anschließend auf das Projekt anpassen.

Deckblätter und Vorlagen lassen sich von der Projektnavigation in eine globale Bibliothek ziehen. Danach stehen sie in jedem Projekt zur Verfügung.

## Siehe auch

Grundlagen zu Bibliotheken (Seite 313)

Übersicht über die Task Card "Bibliotheken" (Seite 315)

Deckblätter und Rahmen gestalten (Seite 284)

Vorgefertigte Rahmen und Deckblätter nutzen (Seite 283)

## Rahmen anlegen

Sie können zu jedem Projekt eine beliebige Anzahl von Rahmen anlegen. Die Rahmen werden in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Rahmen" abgelegt. Jeder Dokument-Information können Sie einen Rahmen zuweisen. Wenn Sie sich beim Druck für eine Dokument-Information entscheiden, wird der zugehörige Rahmen verwendet.

## Vorgehen

Um einen neuen Rahmen anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Rahmen" auf den Eintrag "Neuen Rahmen hinzufügen".  
Der Dialog "Rahmen anlegen" wird geöffnet.
2. Geben Sie einen Namen für den Rahmen im Feld "Name" ein.
3. Wählen Sie die Papiergröße aus der Klappliste "Papierformat".
4. Wählen Sie in der Klappliste "Orientierung", ob die Seite im Hoch- oder Querformat angelegt werden soll.

Klicken Sie auf die Schaltfläche "Hinzufügen".

## Ergebnis

Ein neuer Rahmen wird angelegt. Der Rahmen wird anschließend automatisch im Dokumentationseditor geöffnet und kann dort bearbeitet werden.

## Siehe auch

Deckblätter und Rahmen bearbeiten (Seite 284)

Deckblatt anlegen (Seite 283)



## Deckblatt anlegen

Sie können zu jedem Projekt eine beliebige Anzahl von Deckblättern für den Ausdruck anlegen. Die Deckblätter werden in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Deckblätter" abgelegt. Jeder Dokument-Information können Sie ein Deckblatt zuweisen. Wenn Sie sich beim Druck für bestimmte Dokument-Informationen entscheiden, wird das zugehörige Deckblatt verwendet.

## Vorgehen

Um ein neues Deckblatt anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unterhalb der Gruppe "Dokument-Information > Deckblätter" auf den Eintrag "Neues Deckblatt hinzufügen". Der Dialog "Neues Deckblatt hinzufügen" wird geöffnet.
2. Geben Sie einen Namen für das Deckblatt im Feld "Name" ein.
3. Wählen Sie die Papiergröße aus der Klappliste "Papierformat".
4. Wählen Sie in der Klappliste "Orientierung", ob die Seite im Hoch- oder Querformat angelegt werden soll.

Klicken Sie auf die Schaltfläche "Hinzufügen".

## Ergebnis

Ein neues Deckblatt wird angelegt. Das Deckblatt wird anschließend automatisch im Dokumentationseditor geöffnet und kann dort bearbeitet werden.

## Siehe auch

Deckblätter und Rahmen bearbeiten (Seite 284)

Rahmen anlegen (Seite 282)

## Vorgefertigte Rahmen und Deckblätter nutzen

Mit dem TIA-Portal werden bereits einige Rahmen und Deckblätter mitgeliefert. Diese können Sie nach Ihren Wünschen ändern.

## Vorgehen

Um mitgelieferte Rahmen und Deckblätter einzufügen und zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Task Card "Bibliotheken" die Palette "Globale Bibliotheken".
2. Öffnen Sie im Ordner "Kopiervorlagen" den Ordner "Cover Pages" für Deckblätter, oder "Frames" für Rahmen.

3. Ziehen Sie aus einem der beiden Ordner ein Deckblatt oder einen Rahmen per Drag & Drop in die Projektnavigation in einen der folgenden Ordner:

- Für Rahmen: "Dokument-Information > Rahmen"
- Für Deckblätter: "Dokument-Information > Deckblätter".

Der vorgefertigte Rahmen oder das vorgefertigte Deckblatt kann jetzt im Projekt verwendet werden.

4. Doppelklicken Sie in der Projektnavigation auf den neuen Eintrag, um den Rahmen oder das Deckblatt zu bearbeiten.

### Siehe auch

Einsetzen von Deckblättern und Rahmen (Seite 281)

Deckblätter und Rahmen bearbeiten (Seite 284)

### Deckblätter und Rahmen gestalten

#### Deckblätter und Rahmen bearbeiten

Der Dokumentationseditor ist ein grafischer Editor und dient der Gestaltung von Rahmen und Deckblättern für die Anlagendokumentation. Im Dokumentationseditor können Sie Bilder oder Textelemente auf den Rahmen und Deckblättern platzieren. Die Textelemente sind entweder statisch, oder sie werden beim Druck automatisch mit den Daten aus der Dokument-Informationen befüllt, die Sie im Druck-Dialog gewählt haben.

### Vorgehen

Um ein Deckblatt oder einen Rahmen im Dokumentationseditor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation in der Gruppe "Dokument-Information > Rahmen" oder "Dokument-Information > Deckblätter" auf den Eintrag für ein bereits vorhandenes Deckblatt oder einen Rahmen.  
Der Dokumentationseditor wird geöffnet.
2. Gestalten sie das Deckblatt oder den Rahmen entsprechend Ihren Vorstellungen.
3. Schließen Sie den Dokumentationseditor.  
Die Änderungen am Deckblatt oder am Rahmen werden automatisch übernommen.

### Siehe auch

Deckblatt anlegen (Seite 283)

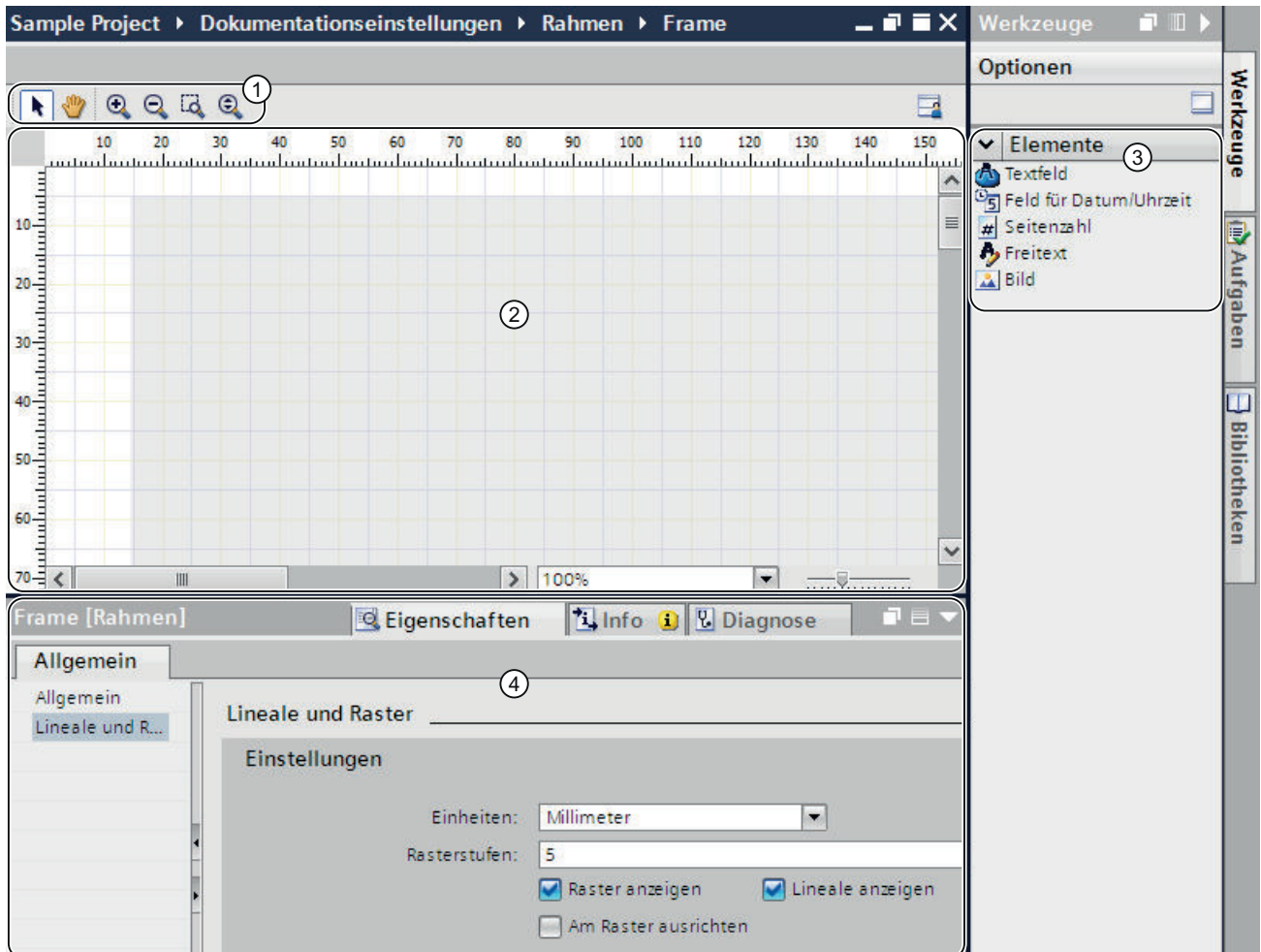
Rahmen anlegen (Seite 282)

Allgemeine Bedienung des Dokumentationseditors (Seite 285)

## Allgemeine Bedienung des Dokumentationseditors

### Komponenten des Dokumentationseditors

Das folgende Bild zeigt einen Überblick über die Komponenten des Dokumentationseditors:



- ① Funktionsleiste  
In der Funktionsleiste finden Sie folgende Werkzeuge (von links nach rechts):
  - Pfeil-Werkzeug  
Ermöglicht das Selektieren eines Objekts.
  - Navigationswerkzeug  
Erlaubt das Verschieben des Seitenausschnitts.
  - Vergrößern-Schaltfläche  
Vergrößert die Seitendarstellung schrittweise.
  - Verkleinern-Schaltfläche  
Verkleinert die Seitendarstellung schrittweise.
  - Zoomfaktor auswählen  
Passt die Seitengröße an den mit dem Lasso-Zoom-Werkzeug gewählten Arbeitsbereich an.
  - Dynamischer Zoom  
Passt die Seitenbreite an den Arbeitsbereich an.
- ② Arbeitsbereich  
Im Arbeitsbereich können Sie das Deckblatt oder den Rahmen gestalten.
- ③ Task Card "Werkzeuge"  
In der Task Card "Werkzeuge" stehen die verschiedenen Typen von Platzhaltern, die Sie auf dem Deckblatt oder dem Rahmen verwenden können, zur Verfügung. Die Platzhalter können per Drag & Drop im Arbeitsbereich platziert werden.
- ④ Eigenschaften im Inspektorfenster  
Im Register "Eigenschaften" des Inspektorfensters können Sie die Eigenschaften des aktuell selektierten Objekts anzeigen und ändern. Sie können beispielsweise die Eigenschaften der Seite ändern, Text formatieren, die Position von Objekten auf der Seite genau festlegen, usw.

## Bedienung im Dokumentationseditor

Der Dokumentationseditor besitzt die folgenden grundlegenden Funktionen:

- **Drag & Drop-Funktionalität**  
Der Dokumentationseditor ist ein grafischer Editor. Das bedeutet, Sie können Objekte frei mit der Maus platzieren. Im Arbeitsbereich wird ein Abbild der Seite gezeigt, welches identisch mit dem späteren Drucklayout ist.  
Möchten Sie Objekte auf der Seite selektieren, um sie zu verschieben oder die Eigenschaften der Objekte zu ändern, muss das Pfeil-Werkzeug in der Funktionsleiste aktiviert sein.
- **Zoomfunktion**  
Mit Hilfe der Zoomfunktion können Sie die Darstellungsgröße der Seite ändern. Sie haben zwei Möglichkeiten, um die Seitengröße anzupassen:
  - **Über die Symbole in der Funktionsleiste**  
Aktivieren Sie das Lupen-Symbol "Zoomfaktor vergrößern" oder "Zoomfaktor verkleinern" in der Funktionsleiste des Dokumentationseditors. Klicken Sie anschließend auf die Seite, um sie stufenweise zu vergrößern oder zu verkleinern.  
Um einen bestimmten Bereich zu vergrößern, wählen Sie das Werkzeug "Zoomfaktor auswählen" und ziehen Sie mit der Maus einen Rahmen um den Bereich, den Sie in den Fokus rücken möchten.  
Um den Arbeitsbereich stufenlos zu vergrößern oder zu verkleinern, verwenden Sie das Werkzeug "Dynamischer Zoom". Klicken Sie auf eine Stelle im Arbeitsbereich und ziehen Sie mit gedrückter Maustaste die Maus nach oben, um die Seitendarstellung zu vergrößern. Um die Seitendarstellung zu verkleinern, ziehen Sie die Maus nach unten.
  - **Über die Zoom-Leiste**  
Über die Zoomleiste in der rechten unteren Ecke des Arbeitsbereichs können Sie ebenfalls die Darstellungsgröße ändern. Wählen Sie einen Prozentwert aus der Klappliste aus oder geben Sie einen Prozentwert ein. Alternativ können Sie die Darstellungsgröße über den Schieberegler steuern.
- **Navigation über die Seite**  
Neben dem Bildlauf gibt es die Möglichkeit, mit dem Navigationswerkzeug den Seitenausschnitt zu verändern. Um den Ausschnitt mit dem Navigationswerkzeug zu ändern, aktivieren Sie das Hand-Symbol in der Funktionsleiste. Klicken Sie anschließend mit der Maus auf die Seite, halten Sie die Maustaste gedrückt und verschieben Sie die Seite in die gewünschte Position.

## Positionierungshilfsmittel nutzen und anpassen

Für die Positionierung von Elementen auf der Seite stehen Ihnen verschiedene Hilfsmittel zur Verfügung:

- **Lineale**  
An den Seitenrändern des Arbeitsbereichs sind Lineale angebracht.
- **Seitenraster**  
Der Seitenhintergrund im Arbeitsbereich ist mit einem Raster hinterlegt.

Sie können die Positionierungshilfsmittel im Inspektorfenster unter "Eigenschaften > Lineale und Raster" ein- und ausblenden oder anpassen. Sie haben folgende Einstellmöglichkeiten:

- Einheiten:  
Legen Sie die Maßeinheit für das Raster und für die Lineale fest.
- Rasterstufen:  
Legen Sie die Weite des Rasters fest.
- Raster anzeigen:  
Bestimmen Sie, ob das Raster angezeigt oder ausgeblendet werden soll.
- Am Raster ausrichten:  
Legen Sie fest, ob Objekte automatisch am Raster ausgerichtet werden sollen. Ist die Option aktiviert, bekommen die Rasterlinien eine "magnetische" Funktion.
- Lineale anzeigen:  
Bestimmen Sie, ob die Lineale angezeigt werden sollen.

### Siehe auch

Deckblätter und Rahmen bearbeiten (Seite 284)

Druckfläche festlegen (Seite 288)

Platzhalter für Meta-Daten einfügen (Seite 289)

### Druckfläche festlegen

Innerhalb eines Rahmens ist ein Bereich für die eigentlichen Druckinhalte vorgesehen. Die Projektdaten werden dann innerhalb des festgelegten und stets gleichbleibenden Bereichs innerhalb des Rahmens eingefügt. Sie können die Größe des Druckbereichs anpassen.

### Voraussetzung

Ein Rahmen ist im Dokumentationseditor geöffnet.

### Vorgehen

Um einen Bereich für die Druckinhalte festzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf etwas dunkler gehaltene Fläche innerhalb der Seitendarstellung im Dokumentationseditor, um den Bereich für die Druckinhalte zu selektieren. Die Eigenschaften der Druckfläche werden im Inspektorfenster geöffnet.
2. Geben Sie im Inspektorfenster die Position der Druckfläche auf der X- und der Y-Achse ein.
3. Geben Sie im Inspektorfenster die Breite und die Höhe der Druckfläche in cm an.

Alternativ können Sie die Breite und Position des Druckfeldes in der grafischen Darstellung der Seite ändern. Ziehen Sie dazu mit der Maus die Ränder der Druckfläche in die gewünschte Größe und Position.

## Siehe auch

Rahmen anlegen (Seite 282)

Allgemeine Bedienung des Dokumentationseditors (Seite 285)

## Platzhalter für Meta-Daten einfügen

Sie können auf dem Deckblatt und in einem Rahmen Platzhalter vorsehen. Die Platzhalter werden beim Druck automatisch mit Meta-Daten aus Dokument-Informationen befüllt, sofern es sich um Platzhalter für Text handelt. Alternativ können Sie unveränderliche Daten, wie beispielsweise Freitext oder ein Bild einfügen.

Alle Elemente sind in nummerierten Ebenen angeordnet. Wenn Objekte sich überlagern, können sie festlegen, in welcher Reihenfolge diese angeordnet sind.

## Typen von Platzhaltern

Folgende Typen von Platzhaltern stehen zur Verfügung:

- **Textfeld**  
Das Textfeld steht als Platzhalter für ein Textelement aus einer Dokument-Information. Welcher Text aus einer Dokument-Information automatisch beim Druck eingesetzt werden soll, stellen Sie in den Eigenschaften des Textfeldes ein.
- **Feld für Datum und Uhrzeit**  
Anstelle des Platzhalters wird beim Druck ein Datum und eine Uhrzeit eingefügt. Dies kann beispielsweise das Erstellungsdatum sein, oder auch der Zeitpunkt der letzten Änderung am Projekt. In den Eigenschaften im Inspektorfenster legen Sie fest, um welches Datum oder welche Uhrzeit es sich handelt.
- **Seitenzahl**  
Beim Druck wird automatisch die entsprechende Seitenzahl eingesetzt.
- **Freitext**  
In den Eigenschaften des Textfeldes können Sie frei wählbaren Text eingeben. Der Text ist statisch und wird nicht von der beim Druck gewählten Dokument-Information beeinflusst.
- **Bild**  
Wählen Sie die Bild-Datei in den Eigenschaften des Platzhalters im Inspektorfenster aus. Bilder in den Formaten BMP, JPEG, PNG, EMF oder GIF sind möglich.

## Voraussetzung

Ein Deckblatt oder ein Rahmen ist im Dokumentationseditor geöffnet.

## Vorgehen

Um Platzhalter für Meta-Daten auf dem Deckblatt oder in einen Rahmen einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie aus der Task Card "Werkzeuge > Elemente" per Drag & Drop ein Feld in den Arbeitsbereich des Dokumentationseditors.  
Der Platzhalter wird eingefügt. Die Eigenschaften des Platzhalters werden im Inspektorfenster angezeigt und können dort bearbeitet werden.
2. Wählen Sie aus der Klappliste "Text" im Inspektorfenster unter "Eigenschaften > Allgemein > Textfeld" das Meta-Datum aus, welches beim Druck eingefügt werden soll. Oder geben Sie alternativ Freitext ein, oder wählen Sie ein Bild, je nachdem um welche Art von Platzhalter es sich handelt.
3. Legen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Position und Größe" die Position des Platzhalters auf der X- und Y-Achse fest und geben Sie die Breite und Höhe des Textfeldes in cm ein. Bestimmen Sie im Feld "Ebene" die Reihenfolge der Objekte, falls diese sich überlagern. Je kleiner der Wert, desto weiter hinten befindet sich ein Objekt.
4. Wählen Sie im Inspektorfenster unter "Eigenschaften > Ansicht" die Schriftformatierung und die Orientierung des Textes sowie die Ausrichtung der Schrift. Bei Bildern können Sie diese Einstellung nicht vornehmen.

## Siehe auch

Allgemeine Bedienung des Dokumentationseditors (Seite 285)

## Druckvorschau anzeigen

## Druckvorschau erstellen

## Druckvorschau erstellen

Sie können eine Vorschau des Ausdrucks erstellen. Ebenso wie beim tatsächlichen Druck sind dafür Dokument-Informationen wählbar. So sehen Sie in der Vorschau bereits den gewählten Rahmen und gegebenenfalls das Deckblatt. Die Einstellungen bleiben für den späteren Druck erhalten.



## Vorgehen

Um eine Druckvorschau zu erstellen und den Umfang des späteren Ausdrucks festzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Druckvorschau".  
Der Dialog "Druckvorschau" wird geöffnet.
2. Wählen Sie das Rahmen-Layout, das Sie für den Ausdruck verwenden möchten.
  - Wählen Sie in der Klappliste "Dokument-Information" die Dokumentations-Informationen aus, die für den späteren Druck herangezogen werden sollen.
  - Aktivieren Sie das Optionskästchen "Deckblatt drucken", um das Deckblatt zu drucken, welches im gewählten Set von Dokument-Informationen festgelegt ist.
  - Aktivieren Sie das Optionskästchen "Inhaltsverzeichnis drucken", um dem Ausdruck ein Inhaltsverzeichnis voranzustellen.

Die Optionskästchen zum Drucken des Deckblatts und des Inhaltsverzeichnisses sind nur aktivierbar, wenn Sie den Ausdruck über die Projektnavigation gestartet haben.
3. Wählen Sie unter "Objekte/Bereich drucken" aus, was gedruckt werden soll. Die Auswahl ist nur möglich, wenn Sie den Druck aus einem Editor heraus gestartet haben, der diese Funktion unterstützt.
  - Wählen Sie "Alle", um alle Inhalte des Editors auszudrucken.
  - Wählen Sie "Auswahl", um nur die aktuell im Editor selektierten Objekte zur drucken.
4. Wählen Sie den Druckumfang unter "Eigenschaften" aus.
  - Wählen Sie "Alle", um alle Konfigurationsdaten der selektierten Objekte gedruckt werden sollen.
  - Wählen Sie "Sichtbar", um alle gerade am Bildschirm sichtbaren Informationen innerhalb eines Editors zu drucken. Die Option ist nur wählbar, wenn Sie den Druckauftrag aus einem Editor heraus gestartet haben, der diese Funktion unterstützt.
  - Wählen Sie "Kompakt", um eine verkürzte Ausgabe der Projektdaten auszudrucken.
5. Klicken Sie auf "Vorschau", um die Vorschau zu erzeugen.  
Im Arbeitsbereich wird eine Druckvorschau erstellt.

---

### **Hinweis**

#### **Wartezeit bei umfangreichen Dokumenten**

Bei besonders umfangreichen Projekten kann das Erstellen der Druckvorschau bis zu mehreren Minuten dauern. Sie können auf ausreichend leistungsfähigen Systemen zwischenzeitlich normal weiterarbeiten. In der Statusleiste wird der Fortschritt der Druckvorschau angezeigt.

---

## Siehe auch

Bedienung innerhalb der Druckvorschau (Seite 292)

## Bedienung innerhalb der Druckvorschau

### Funktionen innerhalb der Druckvorschau

Die Druckvorschau zeigt ein genaues Abbild des späteren Drucks an. Über die Symbole in der Funktionsleiste lässt sich die Anzeige der Druckvorschau ändern. Folgende Funktionen stehen zur Verfügung (von links nach rechts):

- Navigationsmodus  
Erlaubt das Verschieben des Seitenausschnitts.  
Um den Ausschnitt mit dem Navigationswerkzeug zu ändern, aktivieren Sie das Pfeil-Symbol. Klicken Sie anschließend mit der Maus auf die Seite, halten Sie die Maustaste gedrückt und verschieben Sie die Seite in die gewünschte Position.
- Zoomfunktion
  - "Vergrößern" und "Verkleinern"  
Vergrößert oder verkleinert die Seitendarstellung.  
Um die Darstellung schrittweise zu vergrößern oder zu verkleinern, aktivieren Sie das entsprechende Symbol. Klicken Sie anschließend auf die Seite, um sie stufenweise zu vergrößern oder zu verkleinern.  
Um einen bestimmten Bereich zu vergrößern, aktivieren Sie das Symbol "Vergrößern" und ziehen Sie mit der Maus einen Rahmen um den Bereich, den Sie in den Fokus rücken möchten.  
Um einen Bereich auszuwählen, der in den Fokus gerückt werden soll, aktivieren Sie das Symbol "Vergrößern/Verkleinern mit Rechteck". Ziehen Sie anschließend mit der Maus einen Rahmen um den Bereich, den Sie in den Fokus rücken möchten.  
Um dynamisch durch die Seite zu zoomen, aktivieren Sie das Symbol "Dynamisch vergrößern/verkleinern". Ziehen Sie anschließend mit gedrückter Maustaste die Maus nach unten über die Seite, um die Darstellung zu verkleinern. Ziehen Sie stattdessen nach oben, um die Darstellung zu vergrößern.
  - Prozentwert in der Klappliste  
Legt die Darstellungsgröße der Seite in Prozent fest.  
Geben Sie einen Prozentwert ein oder wählen Sie einen Prozentwert aus der Klappliste aus. Alternativ wählen Sie die Optionen "An Seite anpassen" aus der Klappliste, um die Seitengröße an den Arbeitsbereich anzupassen. Oder wählen Sie "An Breite anpassen", um die Seitenbreite an den Arbeitsbereich anzupassen.

- "Vorwärts" und "Rückwärts"  
Jede Änderung des Seitenausschnitts, der Seitenzahl oder der Darstellungsgröße wird in einer Historie im Hintergrund gespeichert. Mit der Schaltfläche "Vorwärts" oder "Rückwärts" können Sie jeweils zur vorigen oder zur nächsten Ansicht zurückkehren.
- Seitennavigation
  - "Erste Seite"  
Springt zurück zur ersten Seite.
  - "Vorherige Seite "  
Geht eine Seite zurück.
  - Eingabefeld "Seitenzahl"  
Zeigt die aktuelle Seite an. Um direkt zu einer Seite zu springen, geben Sie die gewünschte Seitenzahl ein.
  - "Nächste Seite"  
Geht zur nächsten Seite.
  - "Letzte Seite"  
Springt zur letzten Seite.

## Siehe auch

Druckvorschau erstellen (Seite 290)

## Projektdaten drucken

Wenn Sie Projektdaten ausdrucken möchten, stehen Ihnen zwei Möglichkeiten zur Verfügung:

- Sofort-Druck mit Standardeinstellungen über die Schaltfläche "Drucken" in der Funktionsleiste.  
Die Schaltfläche ist nur aktiv, wenn ein druckbares Objekt selektiert ist.
- Ausdruck über den Menübefehl "Projekt > Drucken" mit erweiterten Einstellmöglichkeiten.  
Sie können beispielsweise einen anderen Drucker wählen, bestimmte Dokumentations-Informationen wählen, oder bestimmen, ob ein Deckblatt und Inhaltsverzeichnis gedruckt werden wollen. Außerdem können Sie den Druckumfang festlegen oder sich vor dem Ausdruck eine Druckvorschau anzeigen lassen.

## Voraussetzung

- Mindestens ein Drucker ist konfiguriert.
- Die zu druckenden Objekte sind nicht geschützt.  
Sind Objekte geschützt, dann ist der Druckumfang eingeschränkt. Heben Sie den Know-How-Schutz auf, um die Objekte vollständig zu drucken.

## Projektdaten drucken

Um Daten des aktuellen Projekts oder das gesamte Projekt mit erweiterten Einstellmöglichkeiten auszudrucken, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das gesamte Projekt, um das ganze Projekt auszudrucken. Um nur einzelne, einem Projekt zugehörige Elemente zu drucken, selektieren Sie diese in der Projektnavigation.
2. Wählen Sie im Menü "Projekt" den Befehl "Drucken".  
Der Dialog "Drucken" wird geöffnet.
3. Wählen Sie im Feld "Name" den Drucker aus.
4. Klicken Sie auf "Erweitert", um die Windows-Einstellungen des Druckers zu bearbeiten.
5. Wählen Sie das Rahmen-Layout, das Sie für den Ausdruck verwenden möchten.
  - Wählen Sie in der Klappliste "Dokument-Information" die Dokument-Informationen aus. Der in den Dokument-Informationen gespeicherte Rahmen wird für den Druck verwendet. Alle Platzhalter innerhalb des gewählten Rahmens werden mit den Meta-Daten aus den ausgewählten Dokument-Informationen befüllt.
  - Aktivieren Sie das Optionskästchen "Deckblatt drucken", um das Deckblatt zu drucken, welches in den gewählten Dokument-Informationen hinterlegt ist.
  - Aktivieren Sie das Optionskästchen "Inhaltsverzeichnis drucken", um dem Ausdruck ein Inhaltsverzeichnis voranzustellen.

Die Optionskästchen zum Drucken des Deckblatts und des Inhaltsverzeichnisses sind nur aktivierbar, wenn Sie den Ausdruck über die Projektnavigation gestartet haben.
6. Wählen Sie unter "Objekte/Bereich drucken" aus, was gedruckt werden soll. Die Auswahl ist nur möglich, wenn Sie den Druck aus einem Editor heraus gestartet haben, der diese Funktion unterstützt.
  - Wählen Sie "Alle", um alle Inhalte des Editors auszudrucken.
  - Wählen Sie "Auswahl", um nur die aktuell im Editor selektierten Objekte zu drucken.
7. Wählen Sie den Druckumfang unter "Eigenschaften" aus.
  - Wählen Sie "Alle", um alle Konfigurationsdaten der selektierten Objekte zu drucken.
  - Wählen Sie "Sichtbar", um alle gerade am Bildschirm sichtbaren Informationen innerhalb eines Editors zu drucken. Die Option ist nur wählbar, wenn Sie den Druckauftrag aus einem Editor heraus gestartet haben.
  - Wählen Sie "Kompakt", um eine verkürzte Ausgabe der Projektdaten auszudrucken.
8. Klicken Sie auf "Vorschau", um vorab eine Druckvorschau zu erzeugen.  
Im Arbeitsbereich wird eine Druckvorschau erstellt.
9. Klicken Sie auf "Drucken", um den Ausdruck zu starten.

---

### Hinweis

#### Umfang des Dialogs "Drucken"

Die angebotenen Optionen im Dialog "Drucken" variieren abhängig von den zu druckenden Elementen.

---

## Ergebnis

Die Projektdaten werden im Hintergrund für den Druck vorbereitet und anschließend auf dem ausgewählten Drucker gedruckt. Der Fortschritt des Drucks wird in der Statusleiste angezeigt. Während der Druck vorbereitet wird können Sie normal weiterarbeiten.

Die Ergebnisse des Drucks sowie mögliche Fehler und Warnungen werden nach Abschluss des Druckauftrages im Inspektorfenster unter "Info" aufgelistet.

## Druckauftrag abbrechen

Um einen laufenden Druckauftrag abzubrechen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Statusleiste auf das rote Kreuz neben der Fortschrittsanzeige für den Druck.  
Der Ausdruck wird zeitnah abgebrochen.

## Siehe auch

Schutzkonzept für Projektdaten (Seite 274)

Zugriffsrechte für Geräte zurücknehmen (Seite 275)

Ausdruck von Projektinhalten (Seite 277)

Deckblätter und Rahmen gestalten (Seite 284)

### 7.5.4.2 Modulbeschriftungsstreifen drucken

#### Druckfunktion für Modulbeschriftungsstreifen

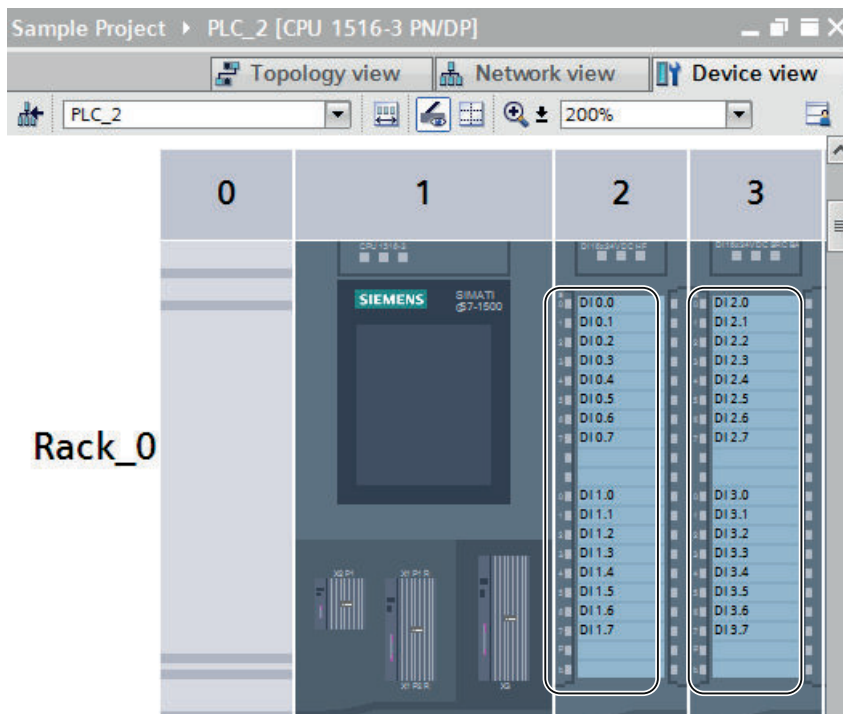
#### Drucken von Beschriftungsstreifen für Hardware-Module

Mithilfe des TIA-Portals können Sie Beschriftungsstreifen für die Module Ihres Projekts drucken. Die Beschriftungsstreifen sind passgenau für die jeweils verwendeten Module und können mit den folgenden Informationen bedruckt werden:

- Symbolischer Name des Ein- oder Ausgangs
- Absolute Adresse des Ein- oder Ausgangs
- Symbolischer Name und zusätzlich die absolute Adresse des Ein- oder Ausgangs. Die Reihenfolge ist einstellbar.

In der Gerätesicht werden die Module grafisch dargestellt. Wenn Sie die Vergrößerung in der Gerätesicht auf mindestens 200% einstellen, werden dort die Beschriftungen für die einzelnen Module sichtbar. Der Ausdruck auf dem Beschriftungsstreifen entspricht der Darstellung der Beschriftung in der Gerätesicht.

Das folgende Bild zeigt beispielhaft zwei Module in der Gerätesicht, an denen die Beschriftung der Ein- und Ausgänge sichtbar ist:



### Export und Weiterbearbeitung als Microsoft Word-Datei

Bevor Sie drucken können, werden die Beschriftungstreifen zunächst als Microsoft DOCX-Datei exportiert. Diese lässt sich mit gängigen Textverarbeitungsprogrammen, wie z. B. Microsoft Word 2010, weiterbearbeiten. Die einzelnen Beschriftungstreifen sind in der DOCX-Datei als Tabelle repräsentiert. So können Sie den Text oder die Hintergründe der einzelnen Zellen beliebig formatieren.

Standardmäßig wird die Laufweite des Texts innerhalb der Tabelle so angepasst, dass Texte nicht abgeschnitten werden. Wenn Sie verhindern möchten, dass der Text dadurch zu sehr gedehnt oder gestaucht wird, ändern Sie die Laufweite des Textes in den Eigenschaften der Tabellenzelle.

### Druckmedien

Die Beschriftungstreifen können Sie entweder auf vorgefertigte Druckbögen oder auf gewöhnlichem DIN-A4-Papier drucken. Aus den vorgefertigten Druckbögen können Sie die einzelnen Beschriftungstreifen herausbrechen und in die vorgesehenen Beschriftungsfelder Ihrer Module stecken. Wenn Sie auf gewöhnlichem Papier drucken, schneiden Sie die einzelnen Beschriftungstreifen selbst aus. Dabei helfen Ihnen Schnittmarken, die automatisch auf dem Ausdruck angebracht werden.

Da sich die Papiereinzüge an verschiedenen Druckern geringfügig unterscheiden, kann der Ausdruck unter Umständen leicht verschoben auf dem Papier sein. Beim Druck der Beschriftungstreifen auf die vorgefertigten Druckbögen kommt es jedoch auf millimetergenauen Druck an, da ansonsten der Text nicht passgenau innerhalb des Stanzbereichs liegt. Außerdem kann bei ungenauem Druck die Beschriftung eines Ein- oder Ausgangs nicht mehr deckungsgleich mit den Kanalstatusanzeigen des Moduls sein. Daher können Sie im TIA-Portal einen Wert für die Verschiebung Ihres Druckers eingeben, um einen

präzisen Druck zu gewährleisten. Wie Sie den geeigneten Wert für die Verschiebung Ihres Druckers ermitteln, erfahren Sie im Kapitel "Verschiebung des Druckbereichs ermitteln (Seite 299)".

## Siehe auch

- Beschriftungsstreifen drucken (Seite 297)
- Verschiebung des Druckbereichs ermitteln (Seite 299)
- Dokumentationsfunktion (Seite 276)

## Beschriftungsstreifen drucken

Sie können für die Module in Ihrem Projekt Beschriftungsstreifen drucken, wenn für die verwendeten Module das Anbringen von Beschriftungsstreifen vorgesehen ist. Die Beschriftungsstreifen werden zunächst in ein Microsoft Word DOCX-Dateien exportiert. Für jede Baugruppenfamilie wird eine eigene DOCX-Datei erstellt (z.B. für alle gewählten S7-1500-Module). Im Anschluss können Sie die Beschriftungsstreifen Ihren Wünschen entsprechend umgestalten. Der Druck erfolgt schließlich aus dem Textverarbeitungsprogramm heraus.

Ist der Druck von Beschriftungsstreifen nicht vorgesehen, dann ist der entsprechende Kontextmenübefehl inaktiv. Wenn Sie mehrere Module für den Druck von Beschriftungsstreifen ausgewählt haben und mindestens eines der Module unterstützt den Druck von Beschriftungsstreifen nicht, wird eine Meldung im Inspektorfenster im Register "Info > Allgemein" angezeigt. Darin werden alle nicht unterstützten Module aufgelistet und der Export der Druckdatei wird für die unterstützten Module fortgesetzt.

## Voraussetzung

- Die gewählten Module müssen den Druck von Beschriftungsstreifen unterstützen.
- Ein Textverarbeitungsprogramm muss installiert sein, das Microsoft Word DOCX-Dateien unterstützt, z. B. Microsoft Word 2010.
- Sie benötigen die vorgefertigten Beschriftungsstreifen für Ihre Module oder handelsübliches DIN-A4-Papier.

## Vorgehen

Um Beschriftungsstreifen für Hardware-Module zu drucken, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation die Module aus, für die Sie Beschriftungsstreifen drucken möchten.
  - Sie können ein oder mehrere Stationen auswählen, um für alle darin gesteckten Module Beschriftungsstreifen auszudrucken.
  - Oder wählen Sie die gewünschten Module unterhalb der Stationen im Ordner "Lokale Baugruppen" aus.
2. Klicken Sie mit der rechten Maustaste auf eines der Geräte und wählen Sie im Kontextmenü den Befehl "Beschriftungsstreifen exportieren".  
Der Dialog "Beschriftungsstreifen exportieren" wird geöffnet.

3. Wählen Sie im Bereich "Inhalt des Beschriftungsstreifens" aus, welche Daten auf den Beschriftungsstreifen gedruckt werden sollen:
  - Wählen Sie "Symbolische Namen", um den symbolischen Namen des Ein- oder Ausgangs zu drucken (entspricht den Inhalten der Spalte "Name" in der IO-Variablen-tabelle).
  - Wählen Sie "Absolute Adresse", um die absolute Adresse des jeweiligen Ein- oder Ausgangs zu drucken (entspricht den Inhalten der Spalte "Adresse" in der IO-Variablen-tabelle).
  - Wählen Sie "Absolute und symbolische Adresse" oder "Symbolische und absolute Adresse", um beide Adressen zu drucken. Die Druckreihenfolge entspricht der genannten Reihenfolge.
4. Wählen Sie im Bereich "Papiersorte" aus, auf welches Papier Sie später drucken möchten.
  - Wählen Sie "Druck auf einen SIEMENS-Beschriftungsbogen", wenn Sie auf einem vorgefertigten Beschriftungsbogen für Ihre Module drucken möchten.
  - Wählen Sie "Druck auf handelsübliches Papier", wenn Sie auf normales DIN-A4-Papier drucken möchten.
5. Wählen Sie im Bereich "Verschiebung des Druckbereichs" bei Bedarf Korrekturwerte für Ihren Drucker zur korrekten Ausrichtung des Druckbereichs. Dies ist nur notwendig, wenn Sie auf vorgefertigte Beschriftungsstreifen drucken.
  - Geben Sie in das Feld "Verschiebung vertikal" einen Korrekturwert in Millimeter ein. Ein negativer Wert verschiebt den Druckbereich nach oben. Ein positiver Wert verschiebt den Druckbereich nach unten.
  - Geben Sie in das Feld "Verschiebung horizontal" einen Korrekturwert in Millimeter ein. Ein negativer Wert verschiebt den Druckbereich nach links. Ein positiver Wert verschiebt den Druckbereich nach rechts.
6. Wählen Sie im Feld "Pfad" einen Pfad aus, an dem die exportierten DOCX-Dateien abgelegt werden sollen.
7. Klicken Sie auf die Schaltfläche "Export", um den Export in eine DOCX-Datei zu starten. Die DOCX-Dateien werden erstellt.
8. Öffnen Sie die DOCX-Dateien mit einem gängigen Textverarbeitungsprogramm, z. B. mit Microsoft Word 2010, und ändern Sie bei Bedarf die Gestaltung der Beschriftungsstreifen.
9. Drucken Sie die Beschriftungsstreifen aus Ihrem Textverarbeitungsprogramm heraus aus. Verwenden Sie dazu das Papier, das Sie im Export-Dialog angegeben haben.
10. Brechen Sie die Beschriftungsstreifen an den vorgesehenen Stanzungen aus, wenn Sie das vorgefertigte Papier verwendet haben. Bei handelsüblichem DIN-A4-Papier schneiden Sie die Beschriftungsstreifen aus.

## Siehe auch

Verschiebung des Druckbereichs ermitteln (Seite 299)



## Verschiebung des Druckbereichs ermitteln

Wenn Sie einen vorgefertigten Beschriftungsbogen verwenden, muss dieser passgenau bedruckt werden, damit der Text genau an den vorgestanzten Beschriftungsstreifen ausgerichtet ist und später zu den Kanalstatusanzeigen des Moduls passt. Die Papiereinzüge variieren jedoch leicht von Drucker zu Drucker. Deshalb müssen Sie im TIA-Portal gegebenenfalls einen passenden Korrekturwert für Ihren Drucker eingeben. Der Druckbereich wird dann in der exportierten DOCX-Datei so verschoben, dass der Druck zu den vorgefertigten Beschriftungsbögen passt.

Die Einstellungen zur Verschiebung des Druckbereichs werden für den jeweiligen Windows-Benutzer gespeichert. Wenn Sie sich unter einem anderen Benutzernamen unter Windows anmelden, müssen Sie die Korrekturwerte erneut eingeben.

Im Folgenden erfahren Sie, wie Sie den Korrekturwert für Ihren Drucker ermitteln.

## Voraussetzung

- Sie benötigen einen vorgefertigten Beschriftungsbogen.
- Sie müssen den gleichen Drucker zur Verfügung haben, auf dem später der Ausdruck erfolgen soll. Der Drucker muss druckbereit für den Druck auf handelsüblichem DIN-A4-Papier sein.

## Vorgehen

Um den Korrekturwert für Ihren Drucker zu ermitteln, gehen Sie folgendermaßen vor:

1. Drucken Sie einen Beschriftungsbogen auf gewöhnlichem DIN-A4-Papier aus, wie im Kapitel "Beschriftungsstreifen drucken (Seite 297)" beschrieben.
2. Vergleichen Sie den Ausdruck auf dem DIN-A4-Papier mit dem vorgefertigten Beschriftungsbogen.
3. Wenn der Druckbereich verschoben ist, müssen Sie Korrekturwerte verwenden.
  - Messen Sie mithilfe eines Lineals den horizontalen Versatz zum vorgefertigten Beschriftungsbogen. Dieser wird im Export-Dialog für den Druck später in das Feld "Verschiebung horizontal" eingetragen. Ist der Druckbereich nach rechts verschoben, muss ein negativer Korrekturwert eingetragen werden. Ist der Druckbereich nach links verschoben, muss ein positiver Korrekturwert eingetragen werden.
  - Messen Sie mithilfe eines Lineals den vertikalen Versatz zum vorgefertigten Beschriftungsbogen. Dieser wird im Export-Dialog für den Druck später in das Feld "Verschiebung vertikal" eingetragen. Ist der Druckbereich nach unten verschoben, muss ein negativer Korrekturwert eingetragen werden. Ist der Druckbereich nach oben verschoben, muss ein positiver Korrekturwert eingetragen werden.

## 7.6 Aktionen rückgängig machen und wiederholen

### 7.6.1 Grundlagen zum Rückgängig machen und Wiederholen von Aktionen

#### Funktion

Sie können durchgeführte Aktionen jederzeit rückgängig machen. Dazu wird jede Aktion, die Sie durchführen, in einem Aktionenstapel gespeichert. Beim Rückgängig machen wird der Stapel von oben nach unten abgearbeitet. Wenn Sie also eine Aktion rückgängig machen, die im Stapel weiter unten liegt, dann werden auch alle darüber liegenden Aktionen automatisch rückgängig gemacht.

Sie können rückgängig gemachte Aktionen so lange wiederholen, bis Sie eine neue Aktion ausführen. Wenn Sie eine neue Aktion ausführen, ist das Wiederholen rückgängig gemachter Aktionen nicht mehr möglich.

#### Besonderheiten beim Rückgängig machen

Es gibt einige Aktionen, die den Aktionenstapel leeren. Sie können diese Aktionen und auch die vor diesen Aktionen durchgeführten Aktionen nicht rückgängig machen. Folgende Aktionen leeren den Aktionenstapel:

- Speichern
- Projektverwaltung (neues Projekt anlegen, Projekt öffnen, Projekt schließen, Projekt löschen)
- Übersetzen
- Bausteine wiederherstellen
- Online-Verbindung herstellen
- Laden
- Auf Memory Cards schreiben

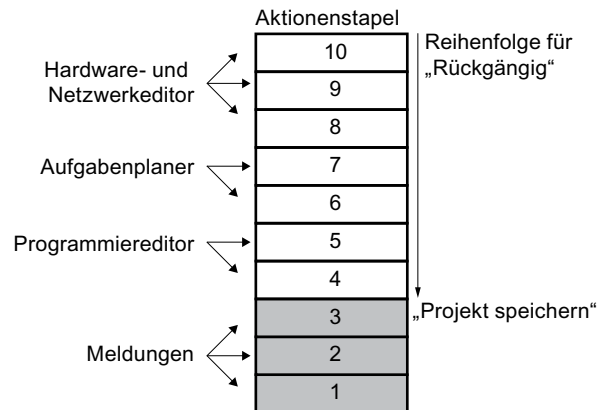
#### Anzeige des Aktionenstapels

Sobald Sie eine Aktion durchführen, die rückgängig gemacht werden kann, wird in der Funktionsleiste die Schaltfläche "Rückgängig" aktiv. Diese Schaltfläche ist zweigeteilt und über den Pfeil-nach-unten können Sie eine Klappliste öffnen. In dieser Klappliste werden Ihnen alle Aktionen des Aktionenstapels angezeigt, die Sie rückgängig machen können. Falls Sie Aktionen in einem anderen als dem aktuell angezeigten Editor ausgeführt hatten, wird auch der entsprechende Editor als Zwischenüberschrift angezeigt. Dadurch können Sie immer erkennen, an welcher Stelle sich das Rückgängig machen auswirkt. Die Zwischenüberschriften werden aus der Liste entfernt, wenn für den entsprechenden Editor keine Aktionen mehr rückgängig gemacht werden können.

Rückgängig gemachte Aktionen werden in den Aktionenstapel für das Wiederholen eingefügt. Hier können Sie die rückgängig gemachten Aktionen wiederholen. Die Anzeige der Aktionen, die Sie wiederholen können, ist analog zur Anzeige der Aktionen, die Sie rückgängig machen können.

### Beispiel für das Rückgängig machen von Aktionen

Das folgende Bild zeigt das Rückgängig machen von Aktionen aus verschiedenen Editoren und Tabellen:



In diesem Beispiel können Sie die Aktionen 1 bis 3 nicht rückgängig machen, da das Projekt gespeichert wurde. Die Aktionen 4 bis 10 können Sie in Pfeilrichtung rückgängig machen. Das heißt, Sie müssen Aktion 10 zuerst rückgängig machen. Nachdem Sie Aktion 8 rückgängig gemacht haben, können Sie anschließend nicht die Aktion 5 rückgängig machen. Zunächst müssen Sie die Aktionen 7 und 6 rückgängig machen. Als Letztes können Sie Aktion 4 rückgängig machen. Sie haben aber die Möglichkeit mehrere Aktionen in einem Schritt rückgängig zu machen, in dem Sie eine untere Aktion des Aktionsstapels rückgängig machen. Alle darüber stehenden Aktionen werden automatisch rückgängig gemacht.

Das gleiche Prinzip gilt auch für das Wiederholen von Aktionen.

### Siehe auch

Aktion rückgängig machen (Seite 301)

Aktion wiederholen (Seite 303)

## 7.6.2 Aktion rückgängig machen

Sie haben beim Rückgängig machen von Aktionen folgende Möglichkeiten:

- Nur die letzte Aktion rückgängig machen  
Es wird nur die zuletzt durchgeführte Aktion rückgängig gemacht.
- Eine beliebige Anzahl von Aktionen rückgängig machen  
Es werden mehrere Aktionen des Aktionsstapels in einem Schritt rückgängig gemacht.

### Nur die letzte Aktion rückgängig machen

Um die zuletzt durchgeführte Aktion rückgängig zu machen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Rückgängig".
  - Falls die Aktion nicht in dem aktuell angezeigten Editor durchgeführt wurde, wird eine Sicherheitsabfrage eingeblendet.
  - Falls für das Rückgängig machen ein Editor geöffnet werden muss, der ein geschütztes Objekt enthält, müssen Sie das Passwort für das Objekt eingeben.
2. Bestätigen Sie ggf. die Sicherheitsabfrage mit "Ja".
3. Geben Sie ggf. das Passwort ein.  
Der Editor, in dem die Aktion durchgeführt wurde, wird angezeigt und die Aktion wird rückgängig gemacht.

### Eine beliebige Anzahl von Aktionen rückgängig machen

Um mehrere Aktionen des Aktionenstapels in einem Schritt rückgängig zu machen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste neben der Schaltfläche "Rückgängig" auf den Pfeil-nach-unten.  
Eine Klappliste wird geöffnet, in der alle Aktionen aufgeführt werden, die Sie rückgängig machen können. Aktionen anderer Editoren werden durch die Editornamen als Zwischenüberschriften gekennzeichnet.
2. Klicken Sie auf die Aktion, die Sie rückgängig machen möchten.  
Die gewählte Aktion und alle über der gewählten Aktion stehenden Aktionen werden rückgängig gemacht. Falls für das Rückgängig machen ein Editor geöffnet werden muss, der ein geschütztes Objekt enthält, müssen Sie das Passwort für das Objekt eingeben.
3. Geben Sie ggf. die erforderlichen Passwörter ein.  
Die Editoren, in denen die Aktionen durchgeführt wurden, werden angezeigt und die Aktionen werden rückgängig gemacht.

### Siehe auch

Grundlagen zum Rückgängig machen und Wiederholen von Aktionen (Seite 300)

Aktion wiederholen (Seite 303)

### 7.6.3 Aktion wiederholen

Sie haben die Möglichkeit, eine rückgängig gemachte Aktion zu wiederholen, sodass Sie den Stand vor der Aktion "Rückgängig" erhalten. Dies ist jedoch nur so lange möglich, bis Sie eine neue Aktion durchführen. Beim Wiederholen von Aktionen haben Sie folgende Möglichkeiten:

- Nur die zuletzt rückgängig gemachte Aktion wiederholen  
Es wird nur die zuletzt rückgängig gemachte Aktion wiederholt.
- Eine beliebige Anzahl von rückgängig gemachten Aktionen wiederholen  
Es werden mehrere rückgängig gemachten Aktionen des Aktionenstapels in einem Schritt wiederholt.

#### Nur die zuletzt rückgängig gemachte Aktion wiederholen

Um die zuletzt rückgängig gemachte Aktion zu wiederholen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Wiederholen".
  - Falls die Aktion nicht in dem aktuell angezeigten Editor wiederholt wird, wird eine Sicherheitsabfrage eingeblendet.
  - Falls für das Wiederholen ein Editor geöffnet werden muss, der ein geschütztes Objekt enthält, müssen Sie das Passwort für das Objekt eingeben.
2. Bestätigen Sie ggf. die Sicherheitsabfrage mit "Ja".
3. Geben Sie ggf. das Passwort ein.  
Der Editor, in dem die Aktion rückgängig gemacht wurde, wird angezeigt und die Aktion wird wiederholt.

#### Eine beliebige Anzahl von rückgängig gemachten Aktionen wiederholen

Um mehrere rückgängig gemachte Aktionen des Aktionenstapels in einem Schritt zu wiederholen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste neben der Schaltfläche "Wiederholen" auf den Pfeil-nach-unten.  
Eine Klappliste wird geöffnet, in der alle Aktionen aufgeführt werden, die Sie rückgängig gemacht haben. Aktionen anderer Editoren werden durch die Editornamen als Zwischenüberschriften gekennzeichnet.
2. Klicken Sie auf die Aktion, die Sie wiederholen möchten.  
Die gewählte Aktion und alle über der gewählten Aktion stehenden Aktionen werden wiederholt. Falls für das Wiederholen ein Editor geöffnet werden muss, der ein geschütztes Objekt enthält, müssen Sie das Passwort für das Objekt eingeben.
3. Geben Sie ggf. die erforderlichen Passwörter ein.  
Die Editoren, in denen die Aktionen rückgängig gemacht wurden, werden angezeigt und die Aktionen werden wiederholt.

#### Siehe auch

Grundlagen zum Rückgängig machen und Wiederholen von Aktionen (Seite 300)

Aktion rückgängig machen (Seite 301)

## 7.7 In Projekten suchen und ersetzen

### 7.7.1 Informationen zur Suchfunktion

#### Suchen und Ersetzen

Sie haben die Möglichkeit, nach Texten innerhalb von Editoren zu suchen. Die Suchfunktion findet alle Texte innerhalb des gerade geöffneten Editors, die den Suchbegriff enthalten. Die Ergebnisse werden nacheinander im geöffneten Editor selektiert.

Weiterhin haben Sie folgende Möglichkeiten:

- Verfeinern der Suche durch Zusatzoptionen
- Ersetzen gefundener Texte

Welche Zusatzoptionen verfügbar sind und welche Art von Texten gesucht werden können, ist von den installierten Produkten und dem geöffneten Editor abhängig.

#### Siehe auch

Suchen und ersetzen (Seite 304)

### 7.7.2 Suchen und ersetzen

#### Suche benutzen

Mit Hilfe der Funktion "Suchen und ersetzen" haben Sie die Möglichkeit Texte innerhalb eines Editors zu suchen oder zu ersetzen.

#### Zusatzoptionen für die Suche

Sie können die Suche verfeinern, indem Sie eine der folgenden Zusatzoptionen wählen:

- Nur ganzes Wort suchen  
Es werden nur ganze Wörter gesucht. Zusammengesetzte Wörter, die den Suchbegriff als Bestandteil eines Wortes enthalten, werden ignoriert.
- Groß-/Kleinschreibung  
Groß- und Kleinschreibung werden bei Ihrer Suche berücksichtigt.
- Suchen in untergeordneten Strukturen  
Es werden auch Texte gesucht, die in einem anderen Objekt enthalten sind.

- Suchen in ausgeblendeten Texten  
Es werden auch Texte bei der Suche berücksichtigt, die einem anderen Text zwar zugeordnet, aber aktuell ausgeblendet sind.
- Platzhalterzeichen verwenden  
Geben Sie einen Stern als Platzhalter für beliebig viele Zeichen ein. Beispiel: Sie suchen nach allen Wörtern, die mit "Gerät" beginnen. Geben Sie also "Gerät\*" in das Suchfeld ein. Geben Sie hingegen ein Fragezeichen als Platzhalter ein, wenn Sie nur ein einzelnes Zeichen auslassen möchten.
- Reguläre Ausdrücke verwenden (nur für die Suche in Skripten)  
Ein regulärer Ausdruck ist eine Zeichenkette, die der Beschreibung von Mengen dient und als Filter benutzt wird. Auf diese Weise können komplexe Suchmuster realisiert werden.

Welche Zusatzoptionen zur Verfügung stehen, ist von den installierten Produkten und dem geöffneten Editor abhängig.

## Suche starten

Um die Funktion "Suchen und ersetzen" zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Bearbeiten" den Befehl "Suchen und ersetzen" oder öffnen Sie in der Task Card "Aufgaben" die Palette "Suchen und ersetzen".  
Die Palette "Suchen und ersetzen" wird geöffnet.
2. Geben Sie einen Begriff in die Klappliste "Suchen" ein.  
Alternativ können Sie aus der Klappliste einen der letzten Suchbegriffe auswählen.
3. Aktivieren Sie die gewünschten Zusatzoptionen für die Suche.
4. Wählen Sie mit Hilfe der Optionsfelder den Ausgangspunkt für die Suche und die Suchrichtung.
  - Wählen Sie "Gesamtes Dokument", wenn Sie unabhängig von der aktuellen Auswahl den gesamten Editor durchsuchen möchten.
  - Wählen Sie "Von der aktuellen Position", wenn Sie die Suche von der aktuellen Auswahl ausgehend starten möchten.
  - Wählen Sie "Auswahl", wenn Sie nur innerhalb der aktuellen Selektion suchen möchten.
  - Wählen Sie "Nach unten", um den Editor von oben nach unten oder von links nach rechts zu durchsuchen.
  - Wählen Sie "Nach oben", um den Editor von unten nach oben oder von rechts nach links zu durchsuchen.
5. Klicken Sie auf "Suchen".  
Der erste Suchtreffer wird im Editor markiert.
6. Um den nächsten Suchtreffer anzuzeigen, klicken Sie erneut auf "Suchen".  
Der nächste Suchtreffer wird im Editor markiert. Wiederholen Sie gegebenenfalls diesen Schritt solange bis Sie zum letzten Treffer gelangt sind.

## Ersetzen des Suchbegriffs

Sie haben die Möglichkeit, einzelne Suchtreffer zu ersetzen oder automatisch alle gefundenen Texte ersetzen zu lassen, wenn der jeweilige Editor diese Funktionalität unterstützt. Um Begriffe zu ersetzen, gehen Sie folgendermaßen vor:

1. Geben Sie einen Begriff in die Klappliste "Suchen" ein.  
Alternativ können Sie aus der Klappliste einen der letzten Suchbegriffe auswählen.
2. Aktivieren Sie die gewünschten Zusatzoptionen für die Suche.
3. Klicken Sie auf die Schaltfläche "Suchen", um einen Suchlauf nach dem eingegebenen Begriff durchzuführen.  
Der erste Suchtreffer wird im Editor angezeigt.
4. Geben Sie den Text, durch den Sie den Suchtreffer ersetzen möchten, in die Klappliste "Ersetzen" ein.  
Alternativ können Sie aus der Klappliste einen der zuletzt eingegebenen Texte auswählen.
5. Um den markierten Suchtreffer durch den vorgegebenen Text zu ersetzen, klicken Sie auf die Schaltfläche "Ersetzen".  
Der gefundene Text wird ersetzt und der nächste Suchtreffer wird im Editor markiert. Wiederholen Sie diesen Schritt solange, bis Sie alle gewünschten Suchtreffer ersetzt haben. Um zum nächsten Treffer zu springen ohne das markierte Wort zu ersetzen, klicken Sie anstatt auf die Schaltfläche "Ersetzen" auf "Suchen".
6. Um alle Suchtreffer automatisch auf einmal zu ersetzen, klicken Sie auf die Schaltfläche "Alle ersetzen".

### Siehe auch

Informationen zur Suchfunktion (Seite 304)

## 7.8 Mit Textlisten arbeiten

### 7.8.1 Textlisten

#### Einführung

Texte, die in Meldungen referenziert werden sollen, können Sie zentral verwalten. Alle Texte sind in so genannten Textlisten gespeichert. Jede Textliste hat einen eindeutigen Namen, mit dem ihr Inhalt aufgerufen wird. Jedem Text in einer Textliste ist ein Wertebereich zugeordnet. Tritt ein Wert aus einem Wertebereich auf, wird der zugehörige Text aufgerufen.

Alle Texte können in sämtliche Projektsprachen übersetzt werden. Dafür stehen Ihnen zwei Möglichkeiten zur Verfügung:



- Sie können die Übersetzung der Texte in eine Liste eingeben. Die Liste finden Sie in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte".
- Sie können alle Texte in eine Datei im Office Open XML Format exportieren und die Übersetzung in einem Tabellenkalkulationsprogramm eingeben. Anschließend können die Übersetzungen wieder importiert werden.

Die Übersetzung der Texte in die weiteren Projektsprachen erfolgt im Rahmen der Projekttexte. Im Textlisten-Editor verwalten Sie lediglich die Zuordnung der einzelnen Texte zu einer Textliste.

Jedes Gerät im Projekt kann über eigene Textlisten verfügen. In der Projektnavigation sind diese daher unterhalb der Geräte angeordnet. Darüber hinaus gibt es Textlisten, die für alle Geräte gültig sind. Diese sind in der Projektnavigation unter "Gemeinsame Daten > Textlisten" zu finden.

### Anwenderdefinierte und systemdefinierte Textlisten

Es gibt zwei Arten von Textlisten:

- **Anwenderdefinierte Textlisten**  
Anwenderdefinierte Textlisten können Sie selbst anlegen und mit Texten befüllen, d. h. Sie können Wertebereiche und dazu gehörende Texte selbst festlegen. Bei anwenderdefinierten Textlisten beginnt der Name der Textliste standardmäßig mit "USER". Sie können diese Bezeichnung allerdings beliebig ändern.
- **Systemdefinierte Textlisten**  
Systemdefinierte Textlisten werden vom System vorgegeben. Es handelt sich immer um Texte, die Geräte betreffen. Sie werden automatisch angelegt, sobald Sie ein Gerät in das Projekt einfügen. Bei Systemmeldungen beginnt der Name der Textliste mit "SYSTEM". Der Name der Textliste sowie die darin enthaltenen Wertebereiche sind nicht veränderbar. Sie können nur Texte bearbeiten, die einzelnen Wertebereichen zugeordnet sind.

| Anwenderdefinierte Textlisten   | Systemdefinierte Textlisten  |
|---|--|
| Eine anwenderdefinierte Textliste kann nur einem Gerät zugeordnet sein.       | Systemdefinierte Textlisten können sowohl einem Gerät als auch dem ganzen Projekt zugeordnet sein. |
| Sie können neue Textlisten anlegen und vorhandene löschen.                    | Sie können keine neuen Textlisten anlegen und keine Textlisten löschen.                            |
| Sie können in den Textlisten Wertebereiche hinzufügen und löschen.            | Sie können in den Textlisten keine Wertebereiche hinzufügen oder löschen.                          |
| Sie können sowohl die Wertebereiche als auch die zugehörigen Texte festlegen. | Sie können nur den zu einem Wertebereich gehörigen Text bearbeiten.                                |

### Gerätespezifische und geräteübergreifende Textlisten

Gerätespezifische Textlisten betreffen nur ein Gerät des Projekts und sind deshalb nur für dieses gültig. In der Projektnavigation sind sie daher unterhalb eines Gerätes angeordnet. Gerätespezifische Textlisten können anwenderdefiniert oder auch vom System vorgegeben sein.

Sind systemdefinierte Textlisten für mehrere Geräte allgemeingültig oder keinem Gerät eindeutig zuzuordnen, werden diese in der Projektnavigation unter "Gemeinsame Daten"

zusammengefasst. Diese Textlisten stehen für alle Geräte zur Verfügung. Geräteübergreifende Textlisten sind immer vom System vorgegeben und dienen ausschließlich den Systemdiagnosemeldungen. Sie können daher keine anwenderdefinierten Textlisten unter "Gemeinsame Daten" ablegen.

### Siehe auch

Projekttexte exportieren und importieren (Seite 245)

## 7.8.2 Anwenderdefinierte Textlisten anlegen

### Textlisten anlegen

Sie können anwenderdefinierte Textlisten für einzelne Geräte neu anlegen.

### Voraussetzung

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.
- Das Projekt enthält mindestens ein Gerät.

### Vorgehen

Um anwenderdefinierte Textlisten anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Pfeil links neben einem Gerät. Die unterhalb des Gerätes angeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Textlisten".  
Alle Textlisten, die dem Gerät zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
3. Doppelklicken Sie in die erste freie Zeile der Tabelle.  
Eine neue anwenderdefinierte Textliste wird erstellt.
4. Tragen Sie in der Spalte "Name" einen Namen für Ihre neue Textliste ein.
5. Wählen Sie aus der Klappliste in der Spalte "Auswahl", ob Sie die Wertebereiche dezimal, binär oder in Bit angeben möchten. Abhängig vom Gerät sind an dieser Stelle gegebenenfalls weitere Optionen möglich.
6. Geben Sie in der Spalte "Bemerkung" eine Bemerkung ein.  
Eine neue anwenderdefinierte Textliste wurde angelegt und Sie können als nächstes die Wertebereiche und Texte eingeben.

### 7.8.3 Anwenderdefinierte Textlisten bearbeiten

#### Anwenderdefinierte Textlisten bearbeiten

Sie können Wertebereiche und dazugehörigen Texte in anwenderdefinierte Textlisten eintragen. Anwenderdefinierte Textlisten befinden sich in der Projektnavigation immer unterhalb eines Geräts.

#### Voraussetzung

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.
- Das Projekt enthält mindestens ein Gerät

#### Anwenderdefinierte Textlisten mit Wertebereichen und Texten ergänzen

Um anwenderdefinierte Textlisten mit Wertebereichen und Texten zu ergänzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Pfeil links neben einem Gerät. Die unterhalb angeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Textlisten". Alle Textlisten, die dem Gerät zugeordnet sind, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
3. Selektieren Sie eine Textliste in der Tabelle. Im Arbeitsbereich werden die Inhalte der selektierten Textliste geöffnet. Dort können Sie Wertebereich eintragen und den einzelnen Wertebereichen Texte zuordnen.
4. Geben Sie die gewünschten Wertebereiche in die Spalten "Bereich von" und "Bereich bis" ein. Der Eintrag muss dem gewählten Zahlenformat für die Textliste entsprechen.
5. Geben Sie zu jedem Wertebereich einen Text in die Spalte "Eintrag" ein.

### 7.8.4 Systemdefinierte Textlisten bearbeiten

#### Systemdefinierte Textlisten bearbeiten

In systemdefinierten Textlisten können Sie lediglich die einzelnen, einem Wertebereich zugeordneten Texte ändern.

Systemdefinierte Textlisten befinden sich in der Projektnavigation entweder unterhalb eines Geräts oder unter "Gemeinsame Daten".

### Voraussetzung

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.
- Das Projekt enthält mindestens ein Gerät

### Texte in systemdefinierten Textlisten ändern

Um in systemdefinierten Textlisten Texte, die einem Wertebereich zugeordnet sind, zu bearbeiten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Pfeil links neben einem Gerät oder dem Element "Gemeinsame Daten".  
Die unterhalb angeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Textlisten".  
Alle Textlisten, die dem Gerät zugeordnet sind oder gemeinsam verwendet werden, erscheinen tabellarisch aufgelistet im Arbeitsbereich.
3. Selektieren Sie eine Textliste in der Tabelle.  
Im Arbeitsbereich werden die Inhalte der selektierten Textlisten geöffnet. Dort können Sie einem Wertebereich zugeordnete Texte ergänzen oder bearbeiten.
4. Geben Sie zu jedem Wertebereich einen Text in die Spalte "Eintrag" ein.

## 7.9 Memory Cards verwenden

### 7.9.1 Grundlagen zu Memory Cards

#### Einführung

Memory Cards sind steckbare Speicherkarten, die für unterschiedliche Zwecke eingesetzt werden können und in verschiedenen Typen erhältlich sind. Je nach Gerätetyp bzw. Gerätefamilie können Memory Cards z. B. für folgende Zwecke eingesetzt werden:

- Als Ladespeicher einer CPU
- Speichermedium für Projekte, Firmware-Backups oder beliebige andere Dateien
- Speichermedium für die Durchführung einer Firmware-Aktualisierung
- Speichermedium für den PROFINET-Gerätenamen

Informationen zu den technischen Ausprägungen der jeweiligen Memory Cards und zur allgemeinen Handtierung finden Sie in den jeweiligen Dokumentationen zum Gerät.

Informationen zur Handtierung von Memory Cards im TIA-Portal finden Sie in der Online-Hilfe unter dem Stichwort "Memory Card".

**ACHTUNG**

Verwenden Sie Memory Cards nicht für SIMATIC-fremde Zwecke und formatieren Sie sie weder mit Fremdgeräten noch mit Windows-Mitteln. Sie überschreiben dadurch die interne Struktur der Memory Card, die nicht wiederhergestellt werden kann. Die Memory Card wird dadurch für SIMATIC-Geräte unbrauchbar!

**Siehe auch**

- Benutzerdefinierten Kartenleser hinzufügen (Seite 311)
- Auf Memory Cards zugreifen (Seite 312)
- Eigenschaften von Memory Cards anzeigen (Seite 312)

## 7.9.2 Benutzerdefinierten Kartenleser hinzufügen

**Einführung**

Falls Ihr Kartenleser nicht automatisch erkannt wurde, können Sie ihn manuell hinzufügen.

**Voraussetzung**

Die Projektansicht ist geöffnet.

**Vorgehen**

Um einen Kartenleser hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Benutzerdefinierten Card Reader hinzufügen".  
Der Dialog "Benutzer-definierten Card Reader hinzufügen" wird geöffnet.
3. Wählen Sie in der Klappliste den Pfad für den Kartenleser aus.
4. Bestätigen Sie die Eingabe mit "OK".

**Siehe auch**

- Grundlagen zu Memory Cards (Seite 310)
- Auf Memory Cards zugreifen (Seite 312)
- Eigenschaften von Memory Cards anzeigen (Seite 312)

### 7.9.3 Auf Memory Cards zugreifen

#### Voraussetzung

- Eine Memory Card ist im Kartenleser vorhanden.
- Die Projektansicht ist geöffnet.

#### Vorgehen

Um auf Memory Cards zuzugreifen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Wählen Sie im Menü "Projekt" den Befehl "Card Reader/USB-Speicher > Card Reader/USB-Speicher anzeigen".  
In der Projektnavigation wird der Ordner "Card Reader/USB-Speicher" angezeigt.
3. Öffnen Sie den Ordner "Card Reader/USB-Speicher".  
Sie können nun auf die Memory Card zugreifen.

---

#### Hinweis

Wenn auf der Memory Card Daten eines nicht installierten Produktes gespeichert sind, werden die Ordner, die diese Daten enthalten, grau angezeigt. Beim Zugriff auf einen solchen Ordner erhalten Sie eine Fehlermeldung. Installieren Sie bei Bedarf das entsprechende Produkt.

---

#### Siehe auch

Grundlagen zu Memory Cards (Seite 310)

Benutzerdefinierten Kartenleser hinzufügen (Seite 311)

Eigenschaften von Memory Cards anzeigen (Seite 312)

### 7.9.4 Eigenschaften von Memory Cards anzeigen

Sie können sich die Eigenschaften der verwendeten Memory Cards anzeigen lassen. Beachten Sie, dass abhängig vom Gerät unterschiedliche Memory Cards verwendet werden müssen, die unterschiedliche Eigenschaften besitzen.

#### Voraussetzung

- Eine Memory Card ist im Kartenleser vorhanden.
- Die Projektansicht ist geöffnet.

## Vorgehen

Um die Eigenschaften einer Memory Card anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Memory Card, deren Eigenschaften Sie anzeigen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Der Dialog "Memory Card <Name der Memory Card>" wird geöffnet. In diesem Dialog werden die Eigenschaften angezeigt.

## Siehe auch

Grundlagen zu Memory Cards (Seite 310)

Benutzerdefinierten Kartenleser hinzufügen (Seite 311)

Auf Memory Cards zugreifen (Seite 312)

## 7.10 Bibliotheken verwenden

### 7.10.1 Grundlagen zu Bibliotheken

#### Einführung

In den Bibliotheken legen Sie Objekte ab, die Sie wiederverwenden möchten. Für jedes Projekt gibt es eine Projektbibliothek, die an das Projekt gebunden ist. Neben der Projektbibliothek können Sie beliebig viele globale Bibliotheken anlegen, die sich projektübergreifend verwenden lassen. Die Bibliotheken sind untereinander kompatibel, so dass Sie Bibliothekselemente zwischen den Bibliotheken kopieren und verschieben können. Mithilfe von Bibliotheken erstellen Sie z. B. Vorlagen für Bausteine, die Sie zunächst in die Projektbibliothek einfügen und dort weiterentwickeln. Schließlich kopieren Sie die Bausteine aus der Projektbibliothek in eine globale Bibliothek. Die globalen Bibliotheken stellen Sie anderen Mitarbeitern in Ihrem Projekt zur Verfügung. Diese verwenden die Bausteine weiter, passen sie gegebenenfalls an ihre persönlichen Erfordernisse an.

Sowohl die Projektbibliothek als auch globale Bibliotheken unterscheiden zwei unterschiedliche Arten von Objekten:

- **Kopiervorlagen**  
Nahezu jedes Objekt lässt sich als Kopiervorlage speichern und später wieder in das Projekt einfügen. Sie können z. B. ganze Geräte mit ihren Inhalten oder Deckblätter für die Anlagendokumentation als Kopiervorlagen speichern.
- **Typen**  
Als Typen sind Elemente geeignet, die für den Ablauf von Anwenderprogrammen notwendig sind, z. B. Bausteine, PLC-Datentypen, HMI Anwenderdatentypen oder Bildbausteine. Typen sind versionierbar und unterstützen daher eine professionelle Weiterentwicklung. Projekte, in denen die Typen verwendet werden, lassen sich aktualisieren, sobald neue Versionen der Typen vorliegen.

## Projektbibliothek

Jedes Projekt besitzt eine eigene Bibliothek, die Projektbibliothek. Hier legen Sie die Objekte ab, die Sie innerhalb des Projekts wiederverwenden möchten. Die Projektbibliothek wird immer zusammen mit dem aktuellen Projekt geöffnet, gespeichert und geschlossen.

## Globale Bibliotheken

Zusätzlich zur Projektbibliothek legen Sie globale Bibliotheken an, wenn Sie diese projektübergreifend verwenden möchten. Globale Bibliotheken sind unabhängig von einem bestimmten Projekt und können daher an andere Anwender weitergegeben werden. Ein gemeinsamer Zugriff auf globale Bibliotheken, z. B. auf einem Netzlaufwerk, ist ebenfalls möglich, wenn alle Anwender die globale Bibliothek schreibgeschützt öffnen.

Zu eigenen Software-Produkten liefert Siemens globale Bibliotheken mit. Diese enthalten fertige Funktionen und Funktionsbausteine, die Sie innerhalb Ihres Projekts verwenden können. Die mitgelieferten Bibliotheken lassen sich nicht ändern.

Globale Bibliotheken aus dem TIA-Portal V11.x lassen sich weiterverwenden. Sie müssen diese jedoch zuvor zur Verwendung in der aktuellen TIA-Portal-Version hochrüsten.

## Vergleichen von Bibliotheksobjekten

Sie haben die Möglichkeit, Bausteine und PLC-Datentypen mit den Objekten eines Geräts zu vergleichen. Sie können dadurch z. B. herausfinden, ob bestimmte Bausteine oder PLC-Datentypen in einem Projekt verwendet wurden und ob diese verändert wurden.

## Siehe auch

Übersicht über die Task Card "Bibliotheken" (Seite 315)

Übersicht über die Bibliotheksansicht (Seite 318)

Übersicht über die Bibliotheksverwaltung (Seite 321)

Grundlagen zu Kopiervorlagen (Seite 334)

Grundlagen zu Typen (Seite 337)



## 7.10.2 Task Card "Bibliotheken" verwenden

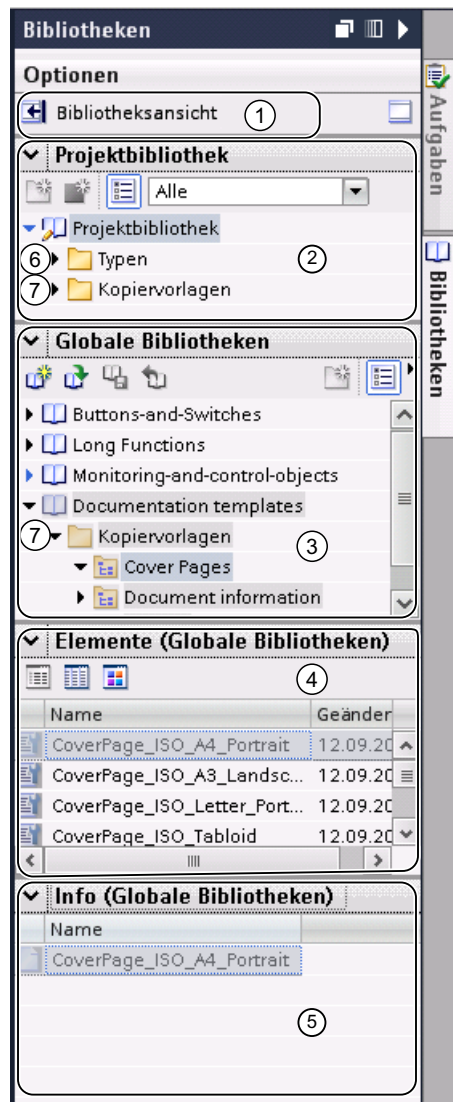
### 7.10.2.1 Übersicht über die Task Card "Bibliotheken"

#### Funktion der Task Card "Bibliotheken"

Die Task Card "Bibliotheken" ermöglicht Ihnen ein effizientes Arbeiten mit der Projektbibliothek und den globalen Bibliotheken.

#### Aufbau der Task Card "Bibliotheken"

Die Task Card "Bibliotheken" besteht aus folgenden Komponenten:



- ① Schaltfläche "Bibliotheksansicht"
- ② Palette "Projektbibliothek"
- ③ Palette "Globale Bibliotheken"
- ④ Palette "Elemente"
- ⑤ Palette "Info"
- ⑥ Ordner "Typen"
- ⑦ Ordner "Kopiervorlagen"

### Schaltfläche "Bibliotheksansicht"

Mit der Schaltfläche "Bibliotheksansicht" wechseln Sie zur Bibliotheksansicht. Die Task Card "Bibliotheken" und die Projektnavigation werden dabei geschlossen.

Siehe auch: Bibliotheksansicht verwenden (Seite 318)

### Palette "Projektbibliothek"

In der Palette "Projektbibliothek" legen Sie Objekte ab, die Sie innerhalb des Projekts mehrfach verwenden möchten.

### Palette "Globale Bibliotheken"

In der Palette "Globale Bibliotheken" legen Sie Objekte ab, die Sie projektübergreifend wiederverwenden möchten.

In der Palette "Globale Bibliotheken" werden auch Bibliotheken aufgeführt, die mit Ihren gekauften Produkten mitgeliefert werden. Diese Bibliotheken stellen Ihnen z. B. fertige Funktionen und Funktionsbausteine zur Verfügung. Die mitgelieferten globalen Bibliotheken sind nicht änderbar.

### Palette "Elemente"

In dieser Palette lassen Sie sich die Inhalte von Ordnern in der Bibliothek anzeigen. Die Palette "Elemente" wird standardmäßig nicht angezeigt. Wenn Sie die Palette "Elemente" anzeigen möchten, muss diese zunächst aktiviert werden. In der Palette "Elemente" gibt es drei Darstellungsformen:

- Detailmodus  
Im Detailmodus werden die Eigenschaften von Ordnern, Kopiervorlagen und Typen in Tabellenform angezeigt.
- Listenmodus  
Im Listenmodus werden die Inhalte von Ordnern aufgelistet.
- Übersichtsmodus  
Im Übersichtsmodus werden die Inhalte von Ordnern mit großen Symbolen angezeigt.

Siehe auch: Elementansicht verwenden (Seite 317)

### **Palette "Info"**

In der Palette "Info" lassen Sie sich die Inhalte der Bibliothekselemente anzeigen. Außerdem werden die einzelnen Versionen von Typen sowie das letzte Änderungsdatum der Version angezeigt.

### **Ordner "Typen"**

In den Verzeichnissen "Typen" verwalten Sie Typen und Typ-Versionen von Objekten, die Sie als Instanzen im Projekt verwenden.

Siehe auch: Verwendung von Typen (Seite 337)

### **Ordner "Kopiervorlagen"**

In den Verzeichnissen "Kopiervorlagen" verwalten Sie Kopiervorlagen von Objekten, die Sie als Kopien im Projekt einfügen können.

Siehe auch: Verwendung von Kopiervorlagen (Seite 334)

### **Siehe auch**

Grundlagen zu Bibliotheken (Seite 313)

Bibliothekselemente vergleichen (Seite 363)

## **7.10.2.2 Elementansicht verwenden**

### **Einführung**

Beim ersten Öffnen der Task Card "Bibliotheken" werden die Paletten "Projektbibliothek" und "Globale Bibliotheken" geöffnet und die Palette "Info" geschlossen angezeigt. Die Palette "Elemente" blenden Sie bei Bedarf ein.

Die Elementansicht zeigt die Elemente der selektierten Bibliothek. In der Elementansicht sind drei Darstellungsformen möglich:

- **Details**  
Im Detailmodus werden die Eigenschaften von Ordnern, Kopiervorlagen und Typen in Tabellenform angezeigt.
- **Liste**  
Im Listenmodus werden die Inhalte von Ordnern aufgelistet.
- **Übersicht**  
Im Übersichtsmodus werden die Inhalte von Ordnern mit großen Symbolen angezeigt.

Die Palette "Info" zeigt den Inhalt des selektierten Bibliothekselements. Wenn Sie beispielsweise einen Typen in der Elementansicht selektieren, dann werden in der Palette "Info" die Typ-Versionen angezeigt.

### Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet.

### Vorgehen

Um die Elementansicht zu verwenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Palette "Projektbibliothek" oder in der Palette "Globale Bibliotheken" auf "Elementansicht öffnen oder schließen".
2. Um die Darstellungsform von der Detailansicht auf die Listenform oder Übersicht zu ändern, klicken Sie in der Funktionsleiste auf das entsprechende Symbol.

### Siehe auch

Grundlagen zu Bibliotheken (Seite 313)

Übersicht über die Task Card "Bibliotheken" (Seite 315)

Globalen Bibliotheken verwenden (Seite 323)

Bibliothekselemente vergleichen (Seite 363)

## 7.10.3 Bibliotheksansicht verwenden

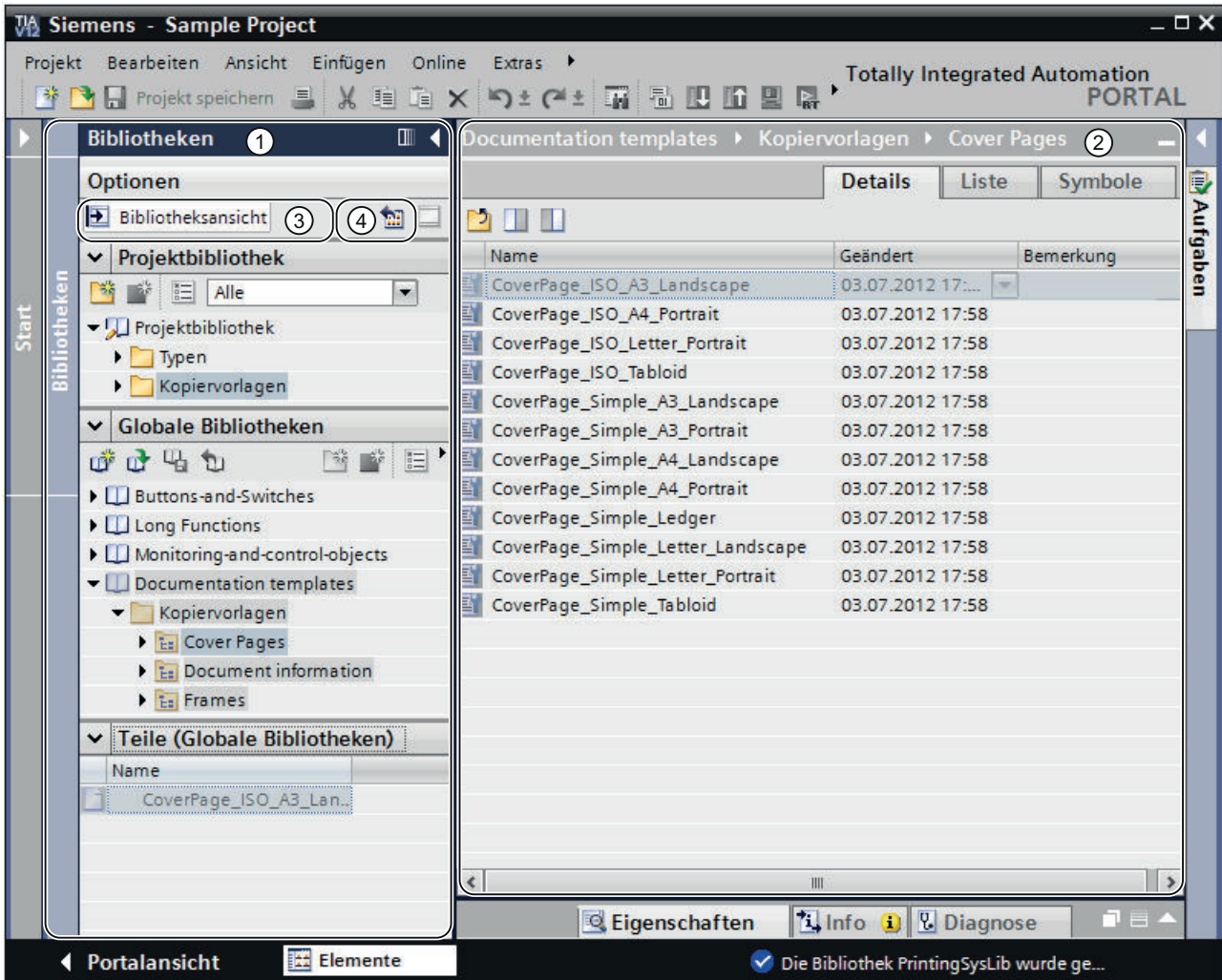
### 7.10.3.1 Übersicht über die Bibliotheksansicht

#### Funktion der Bibliotheksansicht

Die Bibliotheksansicht vereinigt die Funktionalitäten der Task Card "Bibliotheken" und dem Übersichtsfenster. In der Bibliotheksansicht zeigen Sie die Elemente einer Bibliothek in unterschiedlichen Ansichten an und sehen z. B. in der Detailansicht weitere Eigenschaften der einzelnen Elemente. Außerdem bearbeiten und versionieren Sie Typen in der Bibliotheksansicht.

## Aufbau der Bibliotheksansicht

Das folgende Bild zeigt die Komponenten der Bibliotheksansicht:



- ① Bibliotheksnavigation
- ② Bibliotheksübersicht
- ③ Schaltfläche "Bibliotheksansicht"
- ④ Schaltfläche "Bibliotheksübersicht öffnen oder schließen"

## Bibliotheksnavigation

Die Bibliotheksnavigation gleicht bis auf kleinere Abweichungen der Task Card "Bibliotheken". Im Gegensatz zur Task Card gibt es keine Palette "Elemente", da die Elemente in der Bibliotheksübersicht angezeigt werden. Zudem können Sie in der Bibliotheksnavigation die Bibliotheksansicht schließen oder die Bibliotheksübersicht öffnen oder schließen.

Siehe auch: Task Card "Bibliotheken" (Seite 315)

## Bibliotheksübersicht

Die Bibliotheksübersicht entspricht dem Übersichtsfenster und zeigt die Elemente des aktuell in der Bibliotheksnavigation selektierten Objekts an. Sie können die Elemente in drei unterschiedlichen Ansichten anzeigen lassen. Außerdem können Sie in der Bibliotheksübersicht folgende Aktionen durchführen:

- Elemente umbenennen
- Elemente löschen
- Elemente kopieren
- Elemente verschieben
- Instanzen von Typen bearbeiten
- Typen versionieren
- Nur WinCC: Bildbausteine und HMI Anwenderdatentypen bearbeiten

Siehe auch: Übersichtsfenster (Seite 199)

## Siehe auch

Grundlagen zu Kopiervorlagen (Seite 334)

Grundlagen zu Typen (Seite 337)

Bibliotheksansicht öffnen und schließen (Seite 320)

Grundlagen zu Bibliotheken (Seite 313)

Bibliothekselemente vergleichen (Seite 363)

### 7.10.3.2 Bibliotheksansicht öffnen und schließen

Die Bibliotheksansicht wird in einigen Fällen automatisch geöffnet, z. B. wenn Sie die Testinstanz eines Typen bearbeiten oder wenn Sie Bildbausteine und HMI Anwenderdatentypen bearbeiten. Sie können die Bibliotheksansicht jedoch auch manuell öffnen.

## Bibliotheksansicht öffnen

Um die Bibliotheksansicht manuell zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Bibliotheken".
2. Klicken Sie in der Task Card "Bibliotheken" auf die Schaltfläche "Bibliotheksansicht öffnen". Die Bibliotheksnavigation wird geöffnet. Die Task Card "Bibliotheken" und die Projektnavigation werden geschlossen.
3. Wenn die Bibliotheksübersicht nicht angezeigt wird, klicken Sie in der Bibliotheksnavigation auf die Schaltfläche "Bibliotheksübersicht öffnen/schließen". Die Bibliotheksübersicht wird geöffnet.

## **Bibliotheksansicht verlassen**

Um die Bibliotheksansicht zu verlassen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Bibliotheksnavigation auf die Schaltfläche "Bibliotheksansicht schließen". Die Bibliotheksnavigation wird geschlossen. Die Task Card "Bibliotheken" und die Projektnavigation werden geöffnet.

## **Siehe auch**

Übersicht über die Bibliotheksansicht (Seite 318)

Grundlagen zu Bibliotheken (Seite 313)

Task Card "Bibliotheken" verwenden (Seite 315)

Globalen Bibliotheken verwenden (Seite 323)

Bibliothekselemente vergleichen (Seite 363)

## **7.10.4 Bibliotheksverwaltung verwenden**

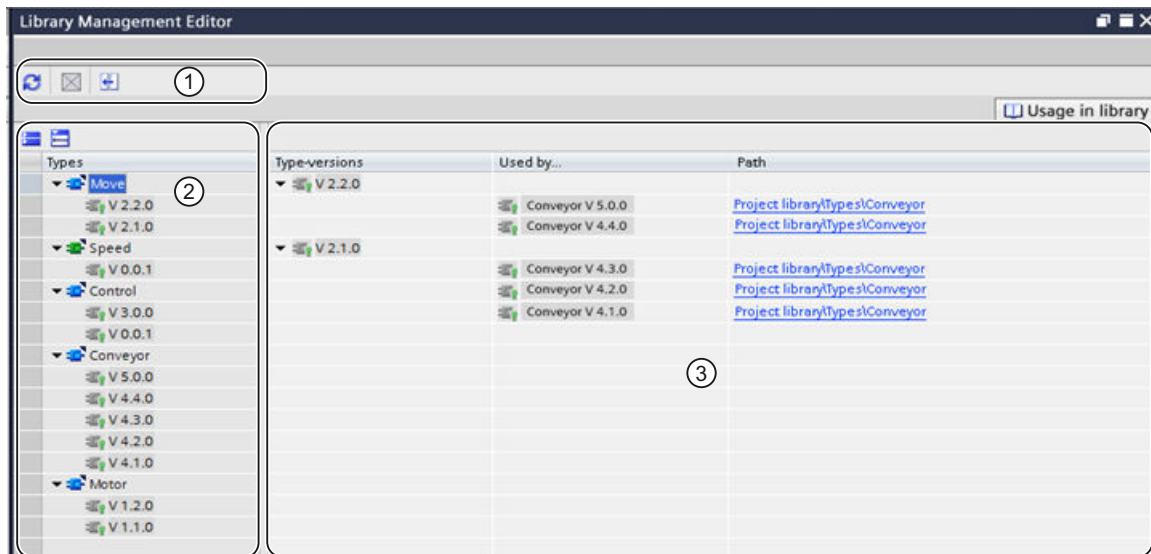
### **7.10.4.1 Übersicht über die Bibliotheksverwaltung**

#### **Funktion der Bibliotheksverwaltung**

Die Bibliotheksverwaltung bietet Ihnen eine Übersicht über die Abhängigkeiten eines Typen zu anderen Bibliothekselementen. Wird ein Typ in anderen Typen oder Kopiervorlagen referenziert, dann werden die Zusammenhänge in der Bibliotheksverwaltung angezeigt. Typen mit Abhängigkeiten zu anderen Bibliothekselementen unterliegen einigen Funktionseinschränkungen. Sie lassen sich beispielsweise nicht löschen, so lange noch Abhängigkeiten bestehen. Auf diese Weise wird verhindert, dass andere Bibliothekselemente unbrauchbar werden.

## Aufbau der Bibliotheksverwaltung

Das folgende Bild zeigt die Komponenten der Bibliotheksverwaltung:



① Funktionsleiste der Bibliotheksverwaltung

② Bereich "Typen"

③ Bereich "Typverwendung"

## Funktionsleiste der Bibliotheksverwaltung

Über die Funktionsleiste der Bibliotheksverwaltung können Sie folgende Aktionen durchführen:

- Ansicht aktualisieren  
Falls das Projekt geändert wurde, können Sie die Ansicht der Bibliotheksverwaltung aktualisieren.
- Bibliothek aufräumen  
Sie können die Projektbibliothek und globale Bibliotheken aufräumen. Mit dem Aufräumen einer Bibliothek werden alle Typen und Typ-Versionen gelöscht, die mit keiner Instanz im Projekt verküpft sind.
- Projekt harmonisieren  
Durch das Harmonisieren eines Projekts werden die Namen und die Pfadstrukturen von Typverwendungen im Projekt an die entsprechenden Namen und Pfadstrukturen der Typen innerhalb einer Bibliothek angepasst.

## Bereich "Typen"

Im Bereich "Typen" werden die Inhalte des Ordners angezeigt, den Sie in der Bibliotheksansicht selektiert haben. Über die Schaltflächen in der Funktionsleiste des Bereichs "Typen" können Sie alle Typen auf- oder zuklappen.



## **Bereich "Typverwendung"**

Im Bereich "Typverwendung" sehen Sie eine Übersicht über die Verwendungsstellen des selektierten Typen.

### **Siehe auch**

Bibliotheksverwaltung öffnen (Seite 323)  
Grundlagen zu Kopiervorlagen (Seite 334)  
Grundlagen zu Typen (Seite 337)  
Grundlagen zu Bibliotheken (Seite 313)

## **7.10.4.2 Bibliotheksverwaltung öffnen**

### **Vorgehen**

Um die Bibliotheksverwaltung zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksansicht.
2. Selektieren Sie einen Typen oder einen beliebigen Ordner, der Typen enthält.
3. Wählen Sie im Kontextmenü den Befehl "Bibliotheksverwaltung".

### **Ergebnis**

Die Bibliotheksverwaltung wird geöffnet und die Typen werden mit ihren Versionen angezeigt.

### **Siehe auch**

Übersicht über die Bibliotheksverwaltung (Seite 321)

## **7.10.5 Globalen Bibliotheken verwenden**

### **7.10.5.1 Globale Bibliothek erstellen**

#### **Voraussetzung**

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

## Vorgehen

Um eine neue globale Bibliothek zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Palette "Globale Bibliotheken" auf das Symbol "Neue globale Bibliothek erstellen" oder wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Neue Bibliothek erstellen".  
Der Dialog "Neue globale Bibliothek erstellen" wird geöffnet.
2. Geben Sie den Namen und den Speicherort für die neue globale Bibliothek an.
3. Bestätigen Sie Ihre Eingaben mit "Erstellen".

## Ergebnis

Die neue globale Bibliothek wird erzeugt und in die Palette "Globale Bibliotheken" eingefügt. Am Speicherort der globalen Bibliothek wird im Dateisystem ein Ordner mit dem Namen der globalen Bibliothek angelegt. Die eigentliche Bibliotheksdatei erhält die Dateinamenerweiterung ".al12".

## Siehe auch

- Grundlagen zu Bibliotheken (Seite 313)
- Globale Bibliothek öffnen (Seite 324)
- Eigenschaften von globalen Bibliotheken anzeigen (Seite 326)
- Globale Bibliothek speichern (Seite 328)
- Globale Bibliothek schließen (Seite 329)
- Globale Bibliothek löschen (Seite 330)

### 7.10.5.2 Globale Bibliothek öffnen

Globale Bibliotheken lassen sich zentral weiter entwickeln und projektübergreifend verwenden. Mehrere Personen können eine globale Bibliothek gleichzeitig von einem zentralen Ablageort aus öffnen, sofern alle Anwender die globale Bibliothek schreibgeschützt öffnen.

## Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

## Vorgehen

Um eine globale Bibliothek zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Palette "Globale Bibliotheken" auf das Symbol "Globale Bibliothek öffnen" oder wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Bibliothek öffnen".  
Der Dialog "Globale Bibliothek öffnen" wird geöffnet.
2. Wählen Sie die globale Bibliothek, die Sie öffnen möchten. Sie erkennen die Bibliotheksdatei an der Dateinamenserweiterung ".al12" oder ".al11" (globale Bibliotheken aus dem TIA-Portal V11.x).
3. Für die Bibliothek ist der Schreibschutz aktiviert. Wenn Sie die globale Bibliothek ändern möchten, deaktivieren Sie die Option "Schreibgeschützt öffnen".
4. Klicken Sie auf "Öffnen".  
Die ausgewählte globale Bibliothek wird geöffnet und in die Palette "Globale Bibliotheken" eingefügt.

---

### Hinweis

#### Verwendung von globalen Bibliotheken aus früheren Versionen des TIA-Portals

Wenn Sie globale Bibliotheken aus früheren Versionen des TIA-Portals weiterverwenden möchten, müssen Sie diese zunächst auf die aktuelle TIA-Portal-Version hochrüsten. Eine entsprechende Aufforderung erhalten Sie beim Öffnen der globalen Bibliothek.

---

## Siehe auch

Bibliotheken aus älteren Versionen hochrüsten (Seite 325)

Globale Bibliotheken dearchivieren (Seite 332)

Grundlagen zu Bibliotheken (Seite 313)

Globale Bibliothek erstellen (Seite 323)

Eigenschaften von globalen Bibliotheken anzeigen (Seite 326)

Globale Bibliothek speichern (Seite 328)

Globale Bibliothek schließen (Seite 329)

Globale Bibliothek löschen (Seite 330)

### 7.10.5.3 Bibliotheken aus älteren Versionen hochrüsten

Wenn Sie eine globale Bibliothek, die mit einer früheren Version des TIA-Portals erstellt wurde, weiterverwenden möchten, dann müssen Sie die Bibliothek zunächst hochrüsten. Damit wird sichergestellt, dass alle verwendeten Elemente mit der aktuellen Version des TIA-Portals kompatibel sind. Die ursprüngliche Bibliothek bleibt unverändert erhalten und es wird eine Kopie erstellt, die zur Verwendung in der aktuellen Version des TIA-Portals geeignet ist. Für Bibliotheken aus dem TIA-Portal V12 erhalten Sie beim Öffnen automatisch eine Aufforderung, die Bibliothek hochzurüsten. Beachten Sie, dass globale Bibliotheken, die einmal mit der aktuellen Version des TIA-Portals gespeichert wurden, nicht rückwärts kompatibel sind.

Eine Ausnahme bilden Projekte, die Sie mit dem TIA-Portal V11.x erstellt haben und die noch nicht auf die aktuelle Version des TIA-Portals hochgerüstet wurden. In einem solchen Projekt lassen sich globale Bibliotheken aus dem TIA-Portal V11.x ohne Einschränkung weiterbenutzen und im Format der früheren Version speichern.

### Voraussetzung

- Sie haben eine globale Bibliothek geladen, die mit dem TIA-Portal V11.x erstellt wurde.
- Die Bibliothek ist nicht schreibgeschützt. Gründe für einen Schreibschutz auf der Bibliothek können z. B. ein schreibgeschützter Ablageort oder ein gleichzeitiger Zugriff auf die Bibliothek durch eine andere Installation des TIA-Portals sein.

### Vorgehen

Um globale Bibliotheken für die Verwendung in der aktuellen TIA-Portal-Version hochzurüsten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie hochrüsten möchten.
2. Wählen Sie im Kontextmenü den Befehl "Bibliothek hochrüsten". Der Dialog "Hochrüsten" wird angezeigt.
3. Bestätigen Sie mit "Ja".

### Ergebnis

Die Bibliothek wird hochgerüstet und als Kopie gespeichert. Die alte Bibliothek wird geschlossen und die neue Bibliothek wird geladen.

### Siehe auch

Projekte hochrüsten (Seite 235)

Globale Bibliothek öffnen (Seite 324)

#### 7.10.5.4 Eigenschaften von globalen Bibliotheken anzeigen

Globale Bibliotheken besitzen Eigenschaften zur näheren Beschreibung der jeweiligen Bibliothek. Zu den Eigenschaften gehören die folgenden Punkte:

- Allgemeine Informationen zur Bibliothek  
Hier erhalten Sie beispielsweise Informationen zum Erstellzeitpunkt, zum Autor, den Speicherpfad, der Dateigröße, Hinweise zum Copyright, usw. Viele der Attribute sind änderbar.
- Bibliothekshistorie  
Die Bibliothekshistorie enthält eine Übersicht über die durchgeführten Migrationen. Hier können Sie auch die Protokolldatei zu den Migrationen aufrufen. Außerdem enthält die Bibliothekshistorie Informationen zu Updates der globalen Bibliothek.

- Support Packages in der Bibliothek  
Sie können sich eine Übersicht über zusätzliche Software anzeigen lassen, die notwendig ist, um alle Geräte des Projekts bearbeiten zu können.
- Softwareprodukte in der Bibliothek  
Sie können eine Übersicht aller installierten und für das Projekt notwendigen Softwareprodukte anzeigen lassen.

### Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

### Vorgehen

Um die Eigenschaften einer globalen Bibliothek anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, deren Eigenschaften Sie anzeigen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Der Dialog mit den Eigenschaften der globalen Bibliothek wird geöffnet.
3. Wählen Sie in der Bereichsnavigation die gewünschten Eigenschaften aus, die angezeigt werden sollen.

### Siehe auch

Globale Bibliothek öffnen (Seite 324)

Grundlagen zu Bibliotheken (Seite 313)

Globale Bibliothek erstellen (Seite 323)

Globale Bibliothek speichern (Seite 328)

Globale Bibliothek schließen (Seite 329)

Globale Bibliothek löschen (Seite 330)

#### 7.10.5.5 Protokolle zu globalen Bibliotheken anzeigen

Bei der Aktualisierung von globalen Bibliotheken werden Protokolle angelegt, in denen alle durchgeführten Änderungen an der globalen Bibliothek aufgelistet sind. Die Protokolle werden zusammen mit der globalen Bibliothek gespeichert und sind immer verfügbar, nachdem Sie die globale Bibliothek geöffnet haben.

## Vorgehen

Um die Protokolle einer globalen Bibliotheken aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die globale Bibliothek in der Task-Card "Bibliotheken" oder in der Bibliotheksansicht.
2. Öffnen Sie den unterlagerten Ordner "Gemeinsame Daten > Protokolle".
3. Doppelklicken Sie auf das gewünschte Protokoll.  
Das Protokoll wird im Arbeitsbereich geöffnet.

## Siehe auch

Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren (Seite 360)

### 7.10.5.6 Globale Bibliothek speichern

Nachdem Sie eine globale Bibliothek geändert haben, speichern Sie diese ab. Über "Bibliothek speichern unter" können Sie eine globale Bibliothek unter einem anderen Namen speichern.

---

#### Hinweis

#### Rückwärtskompatibilität zu älteren Versionen des TIA-Portals

Beachten Sie, dass sich globale Bibliotheken nicht mehr in älteren Versionen des TIA-Portals öffnen lassen, nachdem Sie einmal in der aktuellen Version gespeichert wurden.

---

## Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

## Änderungen speichern

Um eine globale Bibliothek zu speichern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie speichern möchten.
2. Wählen Sie im Kontextmenü den Befehl "Bibliothek speichern".

## Globale Bibliothek unter anderem Namen speichern

Um eine globale Bibliothek unter einem anderen Namen zu speichern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie unter einem anderen Namen speichern möchten.
2. Wählen Sie im Kontextmenü den Befehl "Bibliothek speichern unter".  
Der Dialog "Globale Bibliothek speichern unter" wird geöffnet.

3. Wählen Sie den Speicherort und geben Sie den Dateinamen ein.
4. Bestätigen Sie Ihre Eingaben mit "Speichern".  
Die Bibliothek wird an dem angegebenen Ort unter dem neuen Namen gespeichert. Die ursprüngliche Bibliothek bleibt bestehen.

## Siehe auch

- Arbeiten mit Archiven globaler Bibliotheken (Seite 331)
- Globale Bibliotheken archivieren (Seite 331)
- Grundlagen zu Bibliotheken (Seite 313)
- Globale Bibliothek erstellen (Seite 323)
- Globale Bibliothek öffnen (Seite 324)
- Eigenschaften von globalen Bibliotheken anzeigen (Seite 326)
- Globale Bibliothek schließen (Seite 329)
- Globale Bibliothek löschen (Seite 330)

### 7.10.5.7 Globale Bibliothek schließen

Globale Bibliotheken sind von Projekten unabhängig. Das bedeutet auch, dass globale Bibliotheken nicht zusammen mit dem Projekt geschlossen werden. Sie müssen globale Bibliotheken daher explizit schließen.

## Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

## Vorgehen

Um eine geöffnete globale Bibliothek zu schließen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie schließen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Bibliothek schließen".  
Wenn Sie Änderungen an der globalen Bibliothek vorgenommen haben, wählen Sie, ob Sie die Änderungen speichern möchten oder nicht.  
Die globale Bibliothek wird geschlossen.

## Siehe auch

- Globale Bibliothek erstellen (Seite 323)
- Globale Bibliothek öffnen (Seite 324)
- Eigenschaften von globalen Bibliotheken anzeigen (Seite 326)
- Globale Bibliothek speichern (Seite 328)
- Grundlagen zu Bibliotheken (Seite 313)
- Globale Bibliothek löschen (Seite 330)

### 7.10.5.8 Globale Bibliothek löschen

Wenn Sie eine globale Bibliothek nicht mehr benötigen, können Sie diese löschen. Von Siemens mitgelieferte Bibliotheken können nicht gelöscht werden.

## Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

## Vorgehen

Um eine globale Bibliothek zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek, die Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Bestätigen Sie die Sicherheitsabfrage mit "Ja".

## Ergebnis

Die globale Bibliothek wird aus der Palette "Globale Bibliotheken" entfernt. Das gesamte Verzeichnis für die globale Bibliothek wird aus dem Dateisystem gelöscht.

## Siehe auch

- Grundlagen zu Bibliotheken (Seite 313)
- Globale Bibliothek erstellen (Seite 323)
- Globale Bibliothek öffnen (Seite 324)
- Eigenschaften von globalen Bibliotheken anzeigen (Seite 326)
- Globale Bibliothek speichern (Seite 328)
- Globale Bibliothek schließen (Seite 329)



### 7.10.5.9 Globale Bibliotheken archivieren und deaktivieren

#### Arbeiten mit Archiven globaler Bibliotheken

Wenn Sie globale Bibliotheken beispielsweise auf einer externen Festplatte sichern oder per E-Mail versenden möchten, reduzieren Sie den Speicherplatzbedarf der Bibliothek mit der Archivieren-Funktion.

#### Möglichkeiten zur Reduktion der Projektgröße

Es gibt zwei Möglichkeiten, den Speicherplatzbedarf von globalen Bibliotheken zu reduzieren:

- **Komprimierte Archive globaler Bibliotheken**  
Archive globaler Bibliotheken sind komprimierte Dateien, die jeweils eine ganze globale Bibliothek inklusive der gesamten Ordnerstruktur der Bibliothek enthalten. Bevor das Verzeichnis mit der globalen Bibliothek zur Archivdatei gepackt wird, werden sämtliche Dateien auf ihre wesentlichen Bestandteile reduziert, um den Speicherplatzbedarf zusätzlich zu verringern. Komprimierte Archive globaler Bibliotheken eignen sich daher gut zum Versand per E-Mail.  
Komprimierte Archive globaler Bibliotheken haben die Dateierweiterung ".zal12".  
Um ein komprimiertes Archiv einer globalen Bibliothek zu öffnen, müssen Sie das Archiv dearchivieren. Dabei wird die Archivdatei an einem von Ihnen gewählten Ort mit der gesamten Ordnerstruktur und sämtlichen Dateien entpackt.
- **Minimiertes globale Bibliotheken**  
Sie können auf die zusätzliche Komprimierung in einer Archivdatei verzichten und stattdessen eine Kopie des Verzeichnisses der globalen Bibliothek erstellen. Die enthaltenen Dateien werden auf die wesentlichen Bestandteile reduziert. Somit wird der Speicherplatzbedarf minimiert. Der volle Funktionsumfang der globalen Bibliothek bleibt erhalten und die globale Bibliothek lässt sich gewohnt laden.  
Minimierte globale Bibliotheken sind besonders gut zur Archivierung geeignet, z. B. auf einem externen Datenträger.

#### Siehe auch

Globale Bibliotheken archivieren (Seite 331)

Globale Bibliotheken dearchivieren (Seite 332)

#### Globale Bibliotheken archivieren

Der Speicherplatzbedarf einer globalen Bibliothek lässt sich reduzieren, indem Sie die globale Bibliothek zu einer komprimierten Datei packen. Alternativ reduzieren Sie den Speicherplatzbedarf, indem Sie die globale Bibliothek mit auf das Wesentliche minimiertem Umfang abspeichern. Beides erreichen Sie mit der Archivieren-Funktion für globale Bibliotheken.

#### Voraussetzung

Die globale Bibliothek muss geladen sein.

## Vorgehen

Um eine globale Bibliothek zu archivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die globale Bibliothek, die Sie archivieren möchten.
2. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek und wählen Sie im Kontextmenü den Befehl "Archivieren".  
Der Dialog "Globale Bibliothek archivieren unter..." wird geöffnet.
3. Wählen Sie das Verzeichnis, an dem Sie die Archivdatei oder das neue Verzeichnis der globalen Bibliothek speichern möchten.  
Das Verzeichnis darf nicht in einem Projektverzeichnis eines Projekts oder innerhalb des Verzeichnisses einer globalen Bibliothek liegen.
4. Wählen Sie aus der Klappliste "Dateityp" den Dateityp aus:
  - Globale Bibliotheken Archiv, wenn Sie eine komprimierte Datei des Projekts erstellen möchten.
  - Globale Bibliothek minimiert, wenn Sie lediglich eine Kopie des Projektverzeichnisses mit minimiertem Speicherplatzbedarf erstellen möchten.
5. Geben Sie einen Dateinamen in das Feld "Dateiname" ein, wenn Sie eine Archivdatei erstellen. Wenn Sie eine minimierte globale Bibliothek erstellen, geben Sie stattdessen den Namen des neu zu erstellenden Bibliotheksverzeichnisses in das Feld "Dateiname" ein.
6. Klicken Sie auf "Speichern".

## Ergebnis

Wenn Sie ein Archiv einer globalen Bibliothek erstellt haben, wird eine komprimierte Datei mit der Endung ".zal12" generiert. Die Datei enthält das komplette Verzeichnis der globalen Bibliothek. Die einzelnen Dateien der globalen Bibliothek wurden zusätzlich auf die wesentlichen Bestandteile reduziert, um Speicherplatz zu sparen.

Wenn Sie die globale Bibliothek minimiert haben, wird lediglich eine Kopie des ursprünglichen Verzeichnisses der globalen Bibliothek am gewünschten Ort erstellt. Die darin enthaltenen Dateien wurden auf die wesentlichen Bestandteile reduziert, um Speicherplatz zu sparen.

## Siehe auch

Arbeiten mit Archiven globaler Bibliotheken (Seite 331)

Globale Bibliotheken dearchivieren (Seite 332)

## Globale Bibliotheken dearchivieren

Bevor Sie eine archivierte globale Bibliothek verwenden können, dearchivieren Sie diese. Dabei wird die globale Bibliothek entpackt und anschließend im TIA-Portal geöffnet.

## Vorgehen

Um das Archiv einer globalen Bibliothek zu entpacken, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Globale Bibliotheken > Bibliothek dearchivieren". Der Dialog "Archivierte globale Bibliothek dearchivieren" wird geöffnet.
2. Wählen Sie die Archivdatei aus.
3. Aktivieren Sie das Optionskästchen "Schreibgeschützt öffnen", wenn Sie die globale Bibliothek schreibgeschützt laden möchten.
4. Klicken Sie auf Öffnen.
5. Der Dialog "Ordner suchen" wird geöffnet.
6. Wählen Sie das Zielverzeichnis aus, an dem die archivierte globale Bibliothek entpackt werden soll.
7. Klicken Sie auf "OK".

## Ergebnis

Die globale Bibliothek wird im gewählten Verzeichnis entpackt und sofort geöffnet.

## Siehe auch

Arbeiten mit Archiven globaler Bibliotheken (Seite 331)

Globale Bibliotheken archivieren (Seite 331)

Globale Bibliothek öffnen (Seite 324)

### 7.10.6 Ordner in einer Bibliothek erstellen

Innerhalb der Bibliotheken werden die Bibliothekselemente, abhängig von ihrer Art, in den Ordnern "Typen" und "Kopiervorlagen" gespeichert. Um Kopiervorlagen und Typen weiter zu strukturieren, legen Sie weitere Ordner unterhalb von "Typen" und "Kopiervorlagen" an.

## Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.
- Wenn Sie neue Ordner innerhalb einer globalen Bibliothek erstellen möchten, dann muss die globale Bibliothek mit Schreibrechten geöffnet sein.

## Vorgehen

Um einen neuen Ordner zu erzeugen, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb der Bibliothek mit der rechten Maustaste auf einen beliebigen Ordner.
2. Wählen Sie im Kontextmenü den Befehl "Ordner hinzufügen".  
Ein neuer Ordner wird erstellt.
3. Geben Sie einen Namen für den neuen Ordner ein.

## Siehe auch

Arbeiten mit Typen in der Projektbibliothek (Seite 342)

Kopiervorlagen filtern (Seite 336)

## 7.10.7 Kopiervorlagen einsetzen

### 7.10.7.1 Grundlagen zu Kopiervorlagen

Sie verwenden Kopiervorlagen, um häufig benötigte Elemente standardisiert zu erzeugen. Auf Basis einer Kopiervorlage können Sie bei Bedarf beliebig viele Elemente erstellen und in das Projekt einfügen. Die Elemente erben die Eigenschaften der Kopiervorlage.

Kopiervorlagen legen Sie entweder in der Projektbibliothek oder in einer globalen Bibliothek ab. Kopiervorlagen in der Projektbibliothek können Sie nur innerhalb des Projekts verwenden. Wenn Sie die Kopiervorlage in einer globalen Bibliothek erstellen, lässt sie sich in unterschiedlichen Projekten verwenden.

In der Bibliothek lassen sich z. B. die folgenden Elemente als Kopiervorlagen anlegen:

- Geräte mit ihrer Gerätekonfiguration
- Variablentabellen
- Anweisungsprofile
- Beobachtungstabellen
- Elemente aus den Dokumentationseinstellungen, z. B. Deckblätter und Rahmen

In vielen Fällen enthalten die Objekte, die Sie als Kopiervorlage hinzufügen, weitere Elemente. Eine CPU kann beispielsweise Bausteine enthalten. Wenn die enthaltenen Elemente Verwendungen einer Typ-Version sind, dann werden die verwendeten Versionen der Typen automatisch in der Bibliothek angelegt. Die enthaltenen Elemente sind anschließend als Instanz verwendet und mit dem Typ verbunden.

## Siehe auch

- Kopiervorlagen hinzufügen (Seite 335)
- Kopiervorlagen verwenden (Seite 337)
- Grundlagen zu Typen (Seite 337)
- Kopiervorlagen filtern (Seite 336)

### 7.10.7.2 Kopiervorlagen hinzufügen

Objekte, die Sie mehrfach verwenden möchten, speichern Sie als Kopiervorlage in die Projektbibliothek oder in eine globale Bibliothek. Sie können beim Erstellen von Kopiervorlagen zwischen den folgenden Möglichkeiten wählen:

- Sie selektieren ein oder mehrere Elemente und erzeugen daraus eine Kopiervorlage
- Sie selektieren mehrere Elemente und erzeugen daraus je eine Kopiervorlage pro Element

#### Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet.
- Wenn Sie ein Gerät als Kopiervorlage hinzufügen, dann erfüllt das Gerät die folgenden Anforderungen:
  - Das Gerät ist übersetzt und in konsistentem Zustand.
  - Das Gerät enthält keine Testinstanz eines Typen.
- Wenn Sie die Kopiervorlage zu einer globalen Bibliothek hinzufügen, ist die globale Bibliothek mit Schreibrechten geöffnet.

#### Aus einem oder mehreren Elementen eine Kopiervorlage erzeugen

Um aus einem oder mehreren Elementen eine Kopiervorlage zu erzeugen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliothek in der Task Card "Bibliotheken".
2. Ziehen Sie ein oder mehrere Elemente per Drag & Drop in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen".  
Alternativ: Kopieren Sie die Elemente in die Zwischenablage und fügen Sie diese an der gewünschten Stelle ein.  
Das Element wird als eine einzige Kopiervorlage in die Bibliothek eingefügt.

#### Aus mehreren Elementen je eine Kopiervorlage pro Element erzeugen

Um aus mehreren Elementen je eine Kopiervorlage pro Element zu erzeugen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliothek in der Task Card "Bibliotheken".
2. Kopieren Sie die Elemente, die Sie als Kopiervorlagen anlegen möchten, in die Zwischenablage.

3. Klicken Sie in der Bibliothek mit der rechten Maustaste auf den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner.
4. Wählen Sie im Kontextmenü den Befehl "Als getrennte Kopiervorlagen einfügen". Die Elemente werden als getrennte Kopiervorlagen in die Bibliothek eingefügt. Von eventuell enthaltenen Objekten wird automatisch jeweils ein Typ erzeugt.

### Siehe auch

- Grundlagen zu Kopiervorlagen (Seite 334)
- Kopiervorlagen verwenden (Seite 337)
- Grundlagen zu Bibliotheken (Seite 313)
- Typen zur Projektbibliothek hinzufügen (Seite 342)

### 7.10.7.3 Kopiervorlagen filtern

Um bei einer großen Anzahl an Kopiervorlagen die Übersichtlichkeit zu erhöhen, können Sie die Anzeige nach der Art der Kopiervorlage filtern.

### Voraussetzung

Die Task Card "Bibliotheken" ist eingeblendet oder die Bibliotheksansicht ist geöffnet.

### Vorgehen

Um die Ansicht zu filtern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Kopiervorlagen" in der Projektbibliothek oder einer globalen Bibliothek.
2. Wählen Sie in der Klappliste in der Funktionsleiste die Art der Objekte, die Sie unter "Kopiervorlagen" anzeigen möchten.

### Ergebnis

Es wird nur die gewählte Art von Kopiervorlagen angezeigt. Um zu einer ungefilterten Ansicht zurückzukehren, setzen Sie den Filter auf "Alle".

### Siehe auch

- Grundlagen zu Bibliotheken (Seite 313)
- Ordner in einer Bibliothek erstellen (Seite 333)
- Grundlagen zu Kopiervorlagen (Seite 334)
- Kopiervorlagen verwenden (Seite 337)
- Elementeansicht verwenden (Seite 317)

#### **7.10.7.4 Kopiervorlagen verwenden**

Kopiervorlagen befinden sich entweder in der Projektbibliothek oder in einer globalen Bibliothek.

##### **Voraussetzung**

Die Task Card "Bibliotheken" ist eingeblendet.

##### **Vorgehen**

Um eine Kopiervorlage in das Projekt einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in einer Bibliothek den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen".
2. Ziehen Sie eine oder mehrere Kopiervorlage per Drag & Drop an die gewünschte Verwendungsstelle.  
Von der Kopiervorlage wird eine Kopie eingefügt.

Oder:

1. Öffnen Sie die Elementansicht.
2. Ziehen Sie die Kopiervorlage per Drag & Drop aus der Palette "Elemente" an die gewünschte Verwendungsstelle.  
Von der Kopiervorlage wird eine Kopie eingefügt.

##### **Siehe auch**

Grundlagen zu Kopiervorlagen (Seite 334)  
Kopiervorlagen hinzufügen (Seite 335)  
Kopiervorlagen filtern (Seite 336)  
Grundlagen zu Bibliotheken (Seite 313)  
Elementansicht verwenden (Seite 317)

#### **7.10.8 Typen und ihre Versionen einsetzen**

##### **7.10.8.1 Grundlagen zu Typen**

##### **Verwendung von Typen**

Typen sind Elemente, die für den Ablauf von Anwenderprogrammen notwendig sind. Typen lassen sich versionieren und zentral weiterentwickeln.

Die folgenden Elemente lassen sich als Typ in der Projektbibliothek oder der globalen Bibliothek ablegen:

- Funktionen (FCs)
- Funktionsbausteine (FBs)
- PLC-Datentypen
- Anwenderdatentypen
- Bildbausteine
- Bilder
- Anwenderdefinierte Funktionen

Aus den Versionen von Typen lassen sich beliebig viele Instanzen im Projekt ableiten. Die Instanzen sind anschließend mit der Version des Typen verbunden. Wenn Sie Typen aus einer globalen Bibliothek verwenden, dann wird der Typ zusätzlich in der Projektbibliothek angelegt. Ist der Typ bereits in der Projektbibliothek vorhanden, werden bei Bedarf eventuell fehlende Typ-Versionen ergänzt. Die Instanz ist anschließend nur mit der jeweiligen Typ-Version in der Projektbibliothek verbunden.

Typen und ihre Instanzen werden mit einem schwarzen Dreieck gekennzeichnet. Das folgende Bild zeigt eine Instanz, gekennzeichnet mit schwarzem Dreieck, und einen gewöhnlichen Programmbaustein:



## Grundlagen zur Versionierung von Typen

Mithilfe der Versionierung von Typen haben Sie die Möglichkeit, Typen zentral zu entwickeln und dann die neuste Version an die einzelnen Projekte als Aktualisierung auszurollen. Fehlerkorrekturen und Funktionserweiterungen lassen sich auf diese Weise einfach in bestehende Projekte integrieren. Wenn Sie eine neue Fassung einer globalen Bibliothek erstellt haben, aktualisieren Sie bestehende Projekte in einem automatischen Prozess. Dadurch werden Fehler minimiert und der Arbeitsaufwand zur Pflege großer Automatisierungslösungen mit vielen einzelnen Projekten sinkt.

Die Versionierung macht den Entwicklungsprozess einzelner Typen nachvollziehbar. Bevor Sie eine Version freigeben, können Sie in einer Testumgebung ausprobieren, ob sich die Änderungen an einem Typen reibungslos in ein bestehendes Projekt integrieren. Erst wenn Sie sichergestellt haben, dass alles fehlerfrei funktioniert, geben Sie eine Version für den Produktiveinsatz frei. Sie können jederzeit die Historie einzelner Instanzen im Projekt sehen und ermitteln, aus welcher Version die Instanz abgeleitet wird.

Das TIA-Portal prüft außerdem automatisch, ob es zusammenhängende Objekte mit einzelnen Versionen eines Typen gibt. Zusammenhängende Objekte können beispielsweise in einem Baustein referenzierte PLC-Datentypen oder andere Bausteine sein. Schon beim Erstellen eines Typen oder dem Kopieren zwischen Bibliotheken werden alle zusammenhängenden Objekte berücksichtigt. Damit keine Inkonsistenzen im Projekt entstehen, werden Versionen von Typen vor der Freigabe auf ihre Konsistenz geprüft.



## Versionen von Typen

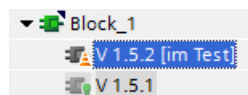
Jedem Typen sind Versionen zugeordnet. Die Versionsnummer wird sowohl in der Task Card "Bibliotheken" als auch in der Bibliotheksansicht neben dem jeweiligen Typen angezeigt. Außerdem wird sie in der Projektnavigation neben den Instanzen von Typen angezeigt. So können Sie jederzeit sehen, in welcher Version eine Instanz im Projekt verwendet wird.

Die Versionsnummer besteht aus drei Stellen. Die ersten beiden Stellen können Sie frei bestimmen. Die dritte Stelle ist die Build-Nummer. Sie wird automatisch um eins erhöht, sobald Sie eine der Version zugehörige Instanz bearbeiten. Wenn Sie die Version (erste Stelle der Versionsnummer) freigeben, wird die Build-Nummer auf Null zurückgesetzt.

Die Versionen von Typen können drei Zustände besitzen:

- In Bearbeitung (Bildbausteine und HMI Anwenderdatentypen)
- Im Test (Alle Arten von Typen außer Bildbausteine und HMI Anwenderdatentypen)
- Freigegeben

Das folgende Bild zeigt einen Typ mit zwei Versionen. Eine Version befindet sich im Zustand "im Test" und eine Version ist freigegeben:



## Siehe auch

Zustand von Typ-Versionen (Seite 339)

Grundlagen zu Kopiervorlagen (Seite 334)

Typen zur Projektbibliothek hinzufügen (Seite 342)

Typen verwenden (Seite 343)

### 7.10.8.2 Zustand von Typ-Versionen

Die Versionen von Typen können sich in drei verschiedenen Zuständen befinden. Die Zustände sind an der Instanz oder in der Bibliothek zu erkennen.

#### Zustand "In Bearbeitung"

Nur Versionen von Bildbausteine und HMI Anwenderdatentypen haben den Zustand "In Bearbeitung". Wenn sich eine Version in Bearbeitung befindet, steht "in Bearbeitung" neben der Version in der Bibliothek.

Wenn Sie einen neuen Typen oder eine neue Version eines freigegebenen Typen erstellen, wird der Typ in den Zustand "in Bearbeitung" gesetzt.

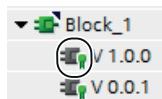
Typen mit dem Zustand "in Bearbeitung" lassen sich in der Bibliotheksansicht bearbeiten. Ein Bezug zu einer Instanz im Projekt muss nicht vorhanden sein. Die Kompatibilität des Typen wird bei der Freigabe mit einer Konsistenzprüfung geprüft.

## Zustand "Im Test"

Im Zustand "im Test" können sich alle Typen außer Bildbausteine und HMI Anwenderdatentypen befinden. Wenn sich eine Version im Test befindet, steht "im Test" neben der Instanz und in der Bibliothek. Eine Version, die sich im Test befindet, ist mit einer Testinstanz im Projekt verknüpft. Sie können so die Auswirkungen Ihrer Änderungen in einer Testumgebung inklusive aller Online-Funktionen ausprobieren, bevor Sie einen Typen zur Verwendung im laufenden Betrieb freigeben.

## Zustand "Freigegeben"

Der Zustand "Freigegeben" ist für alle Typen verfügbar, unabhängig von der Verwendungsstelle. Wenn eine Version freigegeben ist, dann wird das Symbol der Version in der Bibliothek mit einem Siegel gekennzeichnet:



Freigegebene Versionen lassen sich in ihrer Instanz schreibgeschützt öffnen. Wenn Sie eine freigegebene Version bearbeiten möchten, müssen Sie zunächst eine neue Version "in Bearbeitung" oder "im Test" erzeugen.

## Siehe auch

- Grundlagen zu Typen (Seite 337)
- Typen verwenden (Seite 343)
- Testversion eines Typen erstellen (Seite 344)
- Testversion eines Typen bearbeiten (Seite 346)
- Bearbeitungsversion eines Typen erstellen (Seite 346)
- Konsistenzprüfung für eine Typ-Version durchführen (Seite 347)
- Version eines Typen verwerfen (Seite 348)
- Version eines Typen freigeben (Seite 349)
- Gemeinsame Version zuweisen (Seite 351)
- Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 352)
- Verbindung zwischen Instanz und Typ aufheben (Seite 354)

### 7.10.8.3 Freigegebene Typ-Version anzeigen

Wenn Sie eine freigegebene Version anzeigen, aber nicht bearbeiten möchten, öffnen Sie die Instanz schreibgeschützt. Alle Typen außer Bildbausteine und HMI Anwenderdatentypen lassen sich direkt an der Instanz öffnen. Bildbausteine und HMI Anwenderdatentypen lassen sich ausschließlich in der Task Card "Bibliotheken" oder in der Bibliotheksansicht öffnen.

## Voraussetzung

Wenn es sich nicht um einen Bildbaustein oder einen HMI Anwenderdatentyp handelt, besitzt die freigegebene Version eine Instanz im Projekt.

## Typ-Version an einer Instanz öffnen

Um eine freigegebene Version eines Typen ausgehend von einer Instanz schreibgeschützt zu öffnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die freigegebene Version an der Instanz in der Projektnavigation.
2. Klicken Sie mit der rechten Maustaste auf die Instanz und wählen Sie im Kontextmenü den Befehl "Öffnen".  
Die Instanz wird schreibgeschützt geöffnet.

## Typ-Version in der Task Card "Bibliotheken" oder Bibliotheksansicht öffnen

Um eine freigegebene Version eines Typen in der Task Card "Bibliotheken" oder der Bibliotheksansicht zu öffnen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Version.
2. Klicken Sie mit der rechten Maustaste auf die Version und wählen Sie im Kontextmenü den Befehl "Öffnen".  
Wenn es sich um einen Bildbaustein oder HMI Anwenderdatentyp handelt, wird dieser direkt in der Bibliotheksansicht geöffnet. Lassen Sie in diesem Fall die weiteren Schritte aus. Wenn es sich um einen andern Typen handelt, wird der Dialog "Typ öffnen" geöffnet.
3. Selektieren Sie in der Liste der Instanzen die Instanz mit der Version, die Sie anzeigen möchten.
4. Bestätigen Sie mit "OK".  
Die Instanz wird schreibgeschützt geöffnet.

### 7.10.8.4 Eigenschaften eines Typen oder einer Version anzeigen

In den Eigenschaften eines Typen oder seiner Versionen können Sie folgende Eigenschaften sehen und gegebenenfalls ändern:

- Name des Typen
- Versionsnummer (nur bei einer Version sichtbar)
- Erstellungszeitpunkt der Version (nur bei einer Version sichtbar)
- Datum und Uhrzeit der letzten Änderung an der Version (nur bei einer Version sichtbar)
- Autor, der den Typen oder die Version erstellt hat
- Kommentar zu dem Typen oder der Version des Typen  
Der Kommentar ist änderbar.

- Ursprüngliche Bibliothek (nur bei einer Version sichtbar)  
Es wird das Projekt und die Bibliothek angezeigt, aus der die aktuelle Version des Typen generiert wurde. Diese Information ist beispielsweise wichtig, um das Original des Typen wiederzufinden, nachdem er aus einer anderen Bibliothek kopiert wurde.
- ID des Typen oder der Version des Typen  
Mit dieser ID ist der Typ oder die Version des Typen eindeutig identifizierbar, auch wenn es z. B. innerhalb der Projektbibliothek oder der globalen Bibliothek Typen oder Versionen mit einer identischen Benennung gibt. Die ID ist nicht änderbar und wird automatisch vergeben.

## Vorgehen

Um die Eigenschaften eines Typen oder einer Version anzuzeigen und einen Kommentar einzugeben, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Typen oder die Version eines Typen in der Task Card "Bibliotheken" oder in der Bibliotheksansicht.
2. Klicken Sie mit der rechten Maustaste auf den Typen oder auf eine seiner Versionen und wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Der Dialog "Eigenschaften" wird geöffnet.
3. Geben Sie bei Bedarf einen Kommentar zu dem Typen in das Feld "Kommentar" ein oder ändern Sie einen bereits vorhandenen Kommentar.

### 7.10.8.5 Arbeiten mit Typen in der Projektbibliothek

#### Typen zur Projektbibliothek hinzufügen

In der Projektbibliothek lassen sich von etlichen Elementen Typen zur Wiederverwendung im Projekt anlegen. Folgende Elemente können Sie z. B. als Typ anlegen:

- Programmbausteine
- Bildbausteine
- PLC-Datentypen
- HMI Anwenderdatentyp

Wenn Sie ein Element als Typ zur Projektbibliothek hinzufügen, welches Abhängigkeiten zu anderen Elementen besitzt, dann werden die abhängigen Elemente ebenfalls automatisch als Typ angelegt.

Nach dem Hinzufügen eines Typen zur Projektbibliothek ist der Typ mit dem hinzugefügten Element aus dem Projekt verbunden.

#### Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet.
- Die Elemente, die Sie als Typ hinzufügen möchten, sind übersetzt.
- Die Elemente befinden sich in konsistentem Zustand.

## Vorgehen

Um der Projektbibliothek ein bestehendes Element als Typ hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektbibliothek in der Task Card "Bibliotheken".
2. Ziehen Sie ein oder mehrere Elemente per Drag & Drop in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen".  
Alternativ: Kopieren Sie die Elemente in der Projektnavigation in die Zwischenablage und fügen Sie die Elemente in den gewünschten Ordner der Projektbibliothek ein.  
Der Dialog "Typ erzeugen" wird geöffnet.
3. Geben Sie die Eigenschaften des neuen Typen ein:
  - Geben Sie einen Namen für den neuen Typ in das Feld "Name des Typen" ein.
  - Geben Sie in das Feld "Version" eine Versionsnummer für den neuen Typen ein.
  - Geben Sie in das Feld "Bearbeiter" den Namen des für den Typen verantwortlichen Bearbeiters ein.
  - Geben Sie in das Feld "Kommentar" einen Kommentar zu dem Typen ein.
4. Bestätigen Sie mit "OK".  
Der neue Typ wird mit einer freigegebenen Version erzeugt. Die Version ist mit dem hinzugefügten Element verbunden.

## Siehe auch

- Grundlagen zu Typen (Seite 337)
- Grundlagen zu Bibliotheken (Seite 313)
- Kopiervorlagen hinzufügen (Seite 335)

## Typen verwenden

Um die Version eines Typen zu verwenden, erzeugen Sie eine Instanz der Version an einer geeigneten Stelle in der Projektnavigation. Bildbausteine und HMI Anwenderdatentypen können Sie auch direkt im Editor platzieren. Die Instanz ist anschließend mit der Version des jeweiligen Typen verknüpft. Wenn der Typ abhängige Elemente, wie z. B. in einem Baustein referenzierte PLC-Datentypen enthält, dann werden diese ebenfalls als Instanz an geeigneter Stelle angelegt. Einem Gerät können Sie einen Typen in einer bestimmten Version nur einmal zuweisen.

## Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet.
- Die Version ist freigegeben.
- Im Projekt befindet sich bereits ein Gerät, welches die Art des gewünschten Typen unterstützt
- Dem Gerät ist noch keine weitere Instanz desselben Typen zugeordnet.

## Vorgehen

Um von der Version eines Typen eine Instanz zu erzeugen, gehen Sie folgendermaßen vor:

1. Navigieren Sie in der Projektbibliothek zu der Version des Typen, die Sie verwenden möchten.
2. Ziehen Sie die gewünschte Version des Typen per Drag & Drop aus der Projektbibliothek an die Verwendungsstelle.  
Alternativ: Um automatisch die neueste Version zu verwenden, ziehen Sie den Typen selbst per Drag & Drop aus der Projektbibliothek an die Verwendungsstelle.

## Ergebnis

Von dem Typen und seinen abhängigen Elemente wird eine Instanz erzeugt und an der Verwendungsstelle eingefügt. Die Instanzen sind an die jeweilige Version des Typen in der Projektbibliothek gebunden.

## Siehe auch

- Grundlagen zu Typen (Seite 337)
- Zustand von Typ-Versionen (Seite 339)
- Grundlagen zu Bibliotheken (Seite 313)
- Kopiervorlagen verwenden (Seite 337)

## Testversion eines Typen erstellen

Bevor Sie einen Typen für den Produktiveinsatz freigeben, testen Sie den Typen innerhalb eines Projekts und an der Automatisierungsanlage. Der Test erfolgt in einer bestimmten Testumgebung. Diese Testumgebung kann beispielsweise eine CPU sein.

Für den Test erstellen Sie eine Version im Zustand "im Test". Das Erstellen einer Version im Zustand "im Test" ist für alle Arten von Typen geeignet, außer für Bildbausteine und HMI Anwenderdatentypen. Von Bildbausteinen und HMI Anwenderdatentypen lassen sich hingegen Versionen im Zustand "in Bearbeitung" erstellen.

Es gibt zwei Möglichkeiten, eine Testversion eines Typen zu erstellen und die Testumgebung festzulegen:

- In der Task Card "Bibliotheken" oder in der Bibliotheksansicht  
Die neue Version im Zustand "im Test" erzeugen Sie in der Task Card "Bibliotheken" oder in der Bibliotheksansicht. Die neue Version können Sie entweder direkt aus dem Typen oder aus einer bestimmten Version des Typen erzeugen.
- An einer Instanz in der Projektnavigation  
Die Testversion können Sie auch direkt an der Instanz in der Projektnavigation erstellen. Da die Instanz immer in einer bestimmten Version im Projekt verwendet wird, wird aus der an der Instanz verwendeten Version eine neue Version des Typen erzeugt.

Sie können auch von mehreren Typen gleichzeitig Testversionen erstellen.

Für eine Version im Zustand "im Test" gelten die folgenden Regeln:

- Es ist immer nur möglich, eine einzige Version je Typ in den Zustand "im Test" zu versetzen.
- Eine Version im Test darf nur mit einer einzigen Instanz im Projekt verknüpft sein. Daher ist es nicht möglich eine Instanz in die Zwischenablage zu kopieren, sie zu duplizieren oder einen weiteren Typen aus der Instanz zu erstellen, solange diese sich im Zustand "im Test" befindet.

### Voraussetzung

- Es existiert innerhalb des Projekts mindestens eine Instanz des Typen in einer beliebigen Version.
- Wenn Sie die neue Version aus einer bestimmten Version des Typen erstellen möchten, muss die Instanz im Projekt in dieser Version verwendet sein.

### Vorgehen

Um eine neue Testversion eines Typen oder der Version eines Typen zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Typen, eine Version des Typen oder die Instanz.  
Wenn Sie die Testversion direkt an der Instanz erstellen, können Sie über Multiselektion mehrere Elemente oder Ordner selektieren. Da über die selektierte Instanz die Testumgebung bereits definiert ist, können Sie die Schritte 3 und 4 auslassen.
2. Klicken Sie mit der rechten Maustaste auf das selektierte Element und wählen Sie im Kontextmenü den Befehl "Typ bearbeiten".  
Wenn Sie das Bearbeiten in der Task Card "Bibliotheken" oder in der Bibliotheksansicht gestartet haben, wird der Dialog "Typ bearbeiten" geöffnet. Wenn Sie das Bearbeiten an der Instanz in der Projektnavigation begonnen haben, wird die Testinstanz sofort zum Bearbeiten in der Bibliotheksansicht geöffnet.
3. Wählen Sie aus der Liste eine Instanz des Typen im Projekt aus.  
Wenn Sie das Bearbeiten am Typen selbst gestartet haben, dann gilt:
  - Die Stelle, an der die Instanz verwendet wird (z. B. die CPU), dient für das anschließende Bearbeiten des Typen als Testumgebung.
  - Mit der Auswahl der Testinstanz legen Sie auch die Version fest, die bearbeitet wird.Für das Bearbeiten einer bestimmten Version gilt:  
Wenn Ihr Ausgangspunkt eine bestimmte Version ist, können Sie in der Liste nur Instanzen wählen, die in derselben Version verwendet werden.
4. Bestätigen Sie mit "OK".

### Ergebnis

Eine neue Version des Typen wird erstellt. Die neue Version befindet sich im Zustand "im Test" und wird an der Benutzeroberfläche entsprechend gekennzeichnet.

## Siehe auch

- Grundlagen zu Typen (Seite 337)
- Zustand von Typ-Versionen (Seite 339)
- Grundlagen zu Bibliotheken (Seite 313)

## Testversion eines Typen bearbeiten

Wenn Sie eine Version im Zustand "im Test" weiterbearbeiten, wird keine neue Version erstellt. Sie können das Bearbeiten der Testversion an der Instanz in der Projektnavigation, in der Task Card "Bibliotheken" oder in der Bibliotheksansicht starten.

---

### Hinweis

#### Löschen und Umbenennen von Schnittstellen-Parametern

Sie können neue Parameter hinzufügen. Wenn Sie jedoch bereits existierende Parameter umbenennen oder löschen, hat dies den Verlust der Parameterversorgung beim Bausteinaufruf zur Folge.

---

## Vorgehen

Um die Testversion eines Typen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Testversion oder die Instanz.
2. Wählen Sie im Kontextmenü den Befehl "Typ bearbeiten".  
Die Testinstanz wird in der Bibliotheksansicht geöffnet und kann bearbeitet werden.

## Siehe auch

- Grundlagen zu Typen (Seite 337)
- Zustand von Typ-Versionen (Seite 339)
- Grundlagen zu Bibliotheken (Seite 313)
- Konsistenzprüfung für eine Typ-Version durchführen (Seite 347)
- Version eines Typen verwerfen (Seite 348)
- Version eines Typen freigeben (Seite 349)

## Bearbeitungsversion eines Typen erstellen

Wenn Sie einen Typen mit Bildbausteinen oder HMI Anwenderdatentypen bearbeiten möchten, erstellen Sie eine neue Version des Typen im Zustand "in Bearbeitung". Das Bearbeiten der neuen Version erfolgt in der Bibliotheksansicht. Um die Kompatibilität der Änderungen zu prüfen, wird für den Typen vor der Freigabe automatisch eine Konsistenzprüfung durchgeführt.



### Voraussetzung

Die Projektbibliothek ist in der Task Card "Bibliotheken" oder der Bibliotheksansicht geöffnet.

### Vorgehen

Um eine neue Version eines Typen in Bearbeitung zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Typen oder die Version des Typen.
2. Wählen Sie im Kontextmenü den Befehl "Typ bearbeiten".  
Es wird eine neue Version "in Bearbeitung" erstellt und zum Bearbeiten in der Bibliotheksansicht geöffnet.

### Siehe auch

- Grundlagen zu Typen (Seite 337)
- Zustand von Typ-Versionen (Seite 339)
- Grundlagen zu Bibliotheken (Seite 313)

### Konsistenzprüfung für eine Typ-Version durchführen

Bei der Bearbeitung einer Typ-Version kann diese versehentlich in einen inkonsistenten Zustand geraten. Um Fehler rechtzeitig im Entwicklungsprozess zu bemerken, können Sie regelmäßig eine Konsistenzprüfung durchführen. Die Konsistenzprüfung erfolgt jedoch immer automatisch, sobald Sie eine Version freigeben.

Im Folgenden erfahren Sie, wie Sie die Konsistenzprüfung für die Version eines Typen manuell starten.

### Voraussetzung

- Die Projektbibliothek ist in der Task Card "Bibliotheken" oder in der Bibliotheksansicht geöffnet.
- Die Version befindet sich im Zustand "in Bearbeitung" oder "im Test".

### Vorgehen

Um eine Konsistenzprüfung für die Version eines Typen durchzuführen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Version, die Sie auf ihre Konsistenz prüfen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Konsistenzprüfung durchführen".  
Die Konsistenzprüfung wird durchgeführt. Sie erhalten eine Meldung mit dem Ergebnis der Konsistenzprüfung.

## Siehe auch

- Version eines Typen freigeben (Seite 349)
- Version eines Typen verwerfen (Seite 348)
- Testversion eines Typen bearbeiten (Seite 346)
- Grundlagen zu Typen (Seite 337)
- Zustand von Typ-Versionen (Seite 339)
- Grundlagen zu Bibliotheken (Seite 313)

## Version eines Typen verwerfen

Versionen eines Typen im Zustand "im Test" oder "in Bearbeitung" verwerfen Sie, wenn Sie die Version nicht mehr benötigen. Alle Verwendungen der gelöschten Version werden auf den letzten freigegebenen Stand zurückgesetzt. Sie können auch für mehrere Typen gleichzeitig die Versionen verwerfen.

## Voraussetzung

Die Version, die Sie verwerfen möchten, befindet sich im Zustand "im Test" oder "in Bearbeitung".

Sie befinden sich in der Bibliotheksansicht oder die Task Card "Bibliotheken" ist geöffnet.

## Version eines einzelnen Typs verwerfen

Um die Version eines einzelnen Typen zu verwerfen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Version.
2. Wählen Sie im Kontextmenü den Befehl "Änderungen verwerfen und Version löschen". Die Version wird gelöscht.

Alternativ:

1. Klicken Sie, während Sie eine Version zur Bearbeitung geöffnet haben, in der Funktionsleiste auf die Schaltfläche "Version verwerfen". Die Version wird gelöscht.

## Versionen mehrerer Typen verwerfen

Um die Version mehrerer Typen zu verwerfen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksansicht, Bibliotheksverwaltung oder die Task Card "Bibliotheken".
2. Klicken Sie mit der rechten Maustaste auf einen beliebigen Ordner.
3. Wählen Sie im Kontextmenü den Befehl "Änderungen verwerfen und Version löschen". Die Versionen werden gelöscht.

## Siehe auch

- Grundlagen zu Typen (Seite 337)
- Zustand von Typ-Versionen (Seite 339)
- Grundlagen zu Bibliotheken (Seite 313)
- Konsistenzprüfung für eine Typ-Version durchführen (Seite 347)

## Version eines Typen freigeben

Wenn Sie mit dem Bearbeiten der Version eines Typen fertig sind, geben Sie die Version für den Produktiveinsatz frei. Bei der Freigabe weisen Sie eine Versionsnummer zu. Sie können auch mehrere Typ-Versionen auf einmal freigeben.

## Voraussetzung

- Die Version, die Sie freigeben möchten, befindet sich im Zustand "im Test" oder "in Bearbeitung".
- Die Version befindet sich in konsistentem Zustand.  
Es wird eine Konsistenzprüfung durchgeführt, sobald Sie die Freigabe starten. Wenn bei der Konsistenzprüfung freigabeverhindernde Fehler auftreten, erhalten Sie einen Hinweis, wie Sie die Fehler beheben können.

## Version eines einzelnen Typs freigeben

Um eine Version eines Typen freizugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Version des Typen.
2. Wählen Sie im Kontextmenü den Befehl "Version freigeben".  
Der Dialog "Version freigeben" wird geöffnet.
3. Ändern Sie bei Bedarf die Eigenschaften der Version:
  - Geben Sie in das Feld "Name" einen Namen für den Typen ein.
  - Legen Sie im Feld "Version" eine Haupt- und eine Zwischenversionsnummer für die freizugebende Version fest.
  - Geben Sie in das Feld "Autor" den Bearbeiter der freizugebenden Version ein.
  - Geben Sie in das Feld "Kommentar" einen Kommentar zu der freizugebenden Version ein.

4. Wählen Sie die Optionen für die Freigabe:
  - Aktivieren Sie das Optionskästchen "Alle Instanzen des Typen aktualisieren", um alle im Projekt verwendeten Instanzen des Typen mit derselben Ausgangsversion auf die neueste Version zu aktualisieren. Instanzen innerhalb von Kopiervorlagen werden nicht geändert.
  - Aktivieren Sie das Optionskästchen "Nicht verwendete Versionen des Typen aus der Bibliothek löschen", um alle Versionen desselben Typen aus der Bibliothek zu löschen, wenn diese keiner Instanz im Projekt zugewiesen sind und keine anderweitigen Abhängigkeiten bestehen.
5. Bestätigen Sie mit "OK".

Alternativ:

1. Klicken Sie, während Sie eine Version zur Bearbeitung geöffnet haben, in der Funktionsleiste auf das Symbol "Version freigeben".
2. Fahren Sie mit den Schritten 3 bis 5 der voranstehenden Beschreibung fort.

### Versionen mehrerer Typen freigeben

Um für mehrere Typen gleichzeitig Versionen freizugeben, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksansicht, die Bibliotheksverwaltung oder die Task Card "Bibliotheken".
2. Selektieren Sie einen beliebigen Ordner.
3. Wählen Sie im Kontextmenü den Befehl "Version freigeben".  
Der Dialog "Version freigeben" wird geöffnet.
4. Ändern Sie bei Bedarf die Eigenschaften der Version:
  - Geben Sie in das Feld "Autor" den Bearbeiter der freizugebenden Version ein.
  - Geben Sie in das Feld "Kommentar" einen Kommentar zu der freizugebenden Version ein.

Die Inhalte der Felder "Name" und "Version" können Sie nicht ändern.

5. Wählen Sie die Optionen für die Freigabe:
  - Aktivieren Sie das Optionskästchen "Alle Instanzen des Typen aktualisieren", um alle im Projekt verwendeten Instanzen des Typen mit derselben Ausgangsversion auf die neueste Version zu aktualisieren. Instanzen innerhalb von Kopiervorlagen werden nicht geändert.
  - Aktivieren Sie das Optionskästchen "Nicht verwendete Versionen des Typen aus der Bibliothek löschen", um alle Versionen desselben Typen aus der Bibliothek zu löschen, wenn diese keiner Instanz im Projekt zugewiesen sind und keine anderweitigen Abhängigkeiten bestehen.
6. Bestätigen Sie mit "OK".

## Ergebnis

Die Eigenschaften werden für den Typen selbst, die freizugebende Version und für alle zukünftig folgenden Versionen übernommen. Bereits freigegebene Versionen bleiben von den Änderungen unberührt.

Gegebenenfalls werden alle Instanzen mit derselben Ausgangsversion auf die neueste Version aktualisiert und die nicht verwendeten Versionen des Typen gelöscht.

## Siehe auch

Grundlagen zu Typen (Seite 337)

Zustand von Typ-Versionen (Seite 339)

Grundlagen zu Bibliotheken (Seite 313)

Konsistenzprüfung für eine Typ-Version durchführen (Seite 347)

Gemeinsame Version zuweisen (Seite 351)

Typen zu einer globalen Bibliothek hinzufügen (Seite 354)

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 352)

## Gemeinsame Version zuweisen

Wenn Sie mit dem Bearbeiten aller Typen in der Projektbibliothek fertig sind, erstellen Sie eine neue Version aller Typen mit einer gemeinsamen Versionsnummer.

## Voraussetzung

- Die Projektbibliothek ist in der Task Card "Bibliotheken" oder der Bibliotheksansicht geöffnet.
- In der Projektbibliothek darf keine Version eines Typen im Zustand "Im Test" oder "In Bearbeitung" sein.

## Vorgehen

Um allen freigegebenen Typen innerhalb der Projektbibliothek dieselbe Hauptversion zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Projektbibliothek.
2. Wählen Sie im Kontextmenü den Befehl "Gemeinsame Version zuweisen". Der Dialog "Gemeinsame Version zuweisen" wird geöffnet.
3. Ändern Sie bei Bedarf die Eigenschaften der Version:
  - Legen Sie im Feld "Version" die neue Versionsnummer fest. Die Versionsnummer muss höher sein als die höchste Versionsnummer aller Typen innerhalb der Projektbibliothek.
  - Geben Sie in das Feld "Autor" den Verantwortlichen für die freizugebende Version ein.
  - Geben Sie in das Feld "Kommentar" einen Kommentar zu der freizugebenden Version ein.

4. Wählen Sie die Optionen für die Freigabe:
  - Aktivieren Sie das Optionskästchen "Alle Instanzen aktualisieren", um alle Instanzen im Projekt auf die neue Version zu aktualisieren. In Kopiervorlagen verwendete Typen werden nicht aktualisiert.
  - Aktivieren Sie das Optionskästchen "Nicht verwendete Versionen aus der Bibliothek löschen", um alle älteren Versionen von Typen aus der Bibliothek zu löschen, wenn diese keiner Instanz im Projekt zugewiesen sind und keine Abhängigkeit zu anderen Typen aufweisen.
5. Bestätigen Sie mit "OK".

## Ergebnis

Die Projektbibliothek wird wie folgt geändert:

- Es wird eine neue Version aller Typen innerhalb der Projektbibliothek mit der festgelegten Versionsnummer erstellt.
- Die Eigenschaften werden für alle Typen innerhalb der Projektbibliothek, die neue Version und für alle zukünftig folgenden Versionen übernommen. Niedrigere Versionen bleiben von den Änderungen unberührt. Wenn Sie keine Änderungen an den Eigenschaften durchführen, werden die Eigenschaften der letzten, freigegebenen Version jedes Typen übernommen.
- Gegebenenfalls werden alle Instanzen auf die neueste Version aktualisiert und die nicht verwendeten Versionen aller Typen aus der Projektbibliothek gelöscht.

## Siehe auch

Grundlagen zu Typen (Seite 337)

Zustand von Typ-Versionen (Seite 339)

Grundlagen zu Bibliotheken (Seite 313)

Version eines Typen freigeben (Seite 349)

Typen zu einer globalen Bibliothek hinzufügen (Seite 354)

## Projekt auf die neuesten Typ-Versionen aktualisieren

Nachdem Sie mehrere Typen in der Projektbibliothek aktualisiert haben, aktualisieren Sie alle Instanzen im Projekt auf die neueste Version der Typen aus der Projektbibliothek. Wenn Sie die Änderungen nicht auf das gesamte Projekt anwenden möchten, beschränken Sie die Aktualisierung auf einzelne Geräte im Projekt.

Die folgenden Elemente sind jeweils als Quelle für die Aktualisierung wählbar:

- Die gesamte Projektbibliothek
- Einzelne Ordner innerhalb der Projektbibliothek
- Einzelne Typen  
Die Auswahl mehrerer Typen ist zulässig.

## Voraussetzung

Sie befinden sich in der Task Card "Bibliotheken" oder in der Bibliotheksansicht.

## Vorgehen

Um Instanzen in einem Projekt mit den Inhalten aus der Projektbibliothek zu aktualisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gesamte Projektbibliothek oder einzelne Elemente daraus.
2. Klicken Sie mit der rechten Maustaste auf die gewünschten Elemente und wählen Sie im Kontextmenü den Befehl "Aktualisieren > Projekt".  
Der Dialog "Projekt aktualisieren" wird geöffnet.
3. Wählen Sie entweder das gesamte Projekt oder einzelne Geräte für die Aktualisierung aus.
4. Wählen Sie Optionen für den Aktualisierungsvorgang:
  - Das Optionskästchen "Alle Instanzen der gewählten Typen aktualisieren" ist bei diesem Vorgang immer aktiv.
  - Aktivieren Sie das Optionskästchen "Alle nicht verwendeten Versionen aus der Projektbibliothek löschen", um alle älteren Versionen der aktualisierten Typen aus der Projektbibliothek zu löschen.
5. Bestätigen Sie mit "OK".  
Die Aktualisierung wird durchgeführt.

## Ergebnis

Am Projekt wurden folgende Änderungen durchgeführt:

- Gegebenenfalls wurden alle älteren Versionen aus der Projektbibliothek gelöscht.
- Alle Instanzen innerhalb der gewählten Geräte wurden auf die neueste Version des verknüpften Typen aktualisiert.
- In der Projektnavigation unter "Gemeinsame Daten" finden Sie ein Protokoll des Aktualisierungsvorgangs.

## Siehe auch

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 357)

Protokolle verwenden (Seite 231)

Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren (Seite 360)

Grundlagen zu Typen (Seite 337)

Grundlagen zu Bibliotheken (Seite 313)

Zustand von Typ-Versionen (Seite 339)

## Verbindung zwischen Instanz und Typ aufheben

Instanzen von Typen sind immer an die Version des korrespondierenden Typen gebunden. Sie lassen sich nicht wie ein gewöhnliches Objekt bearbeiten. Wenn Sie die Instanz bearbeiten, wird automatisch eine neue Version des Typen in der Bibliothek erstellt und die Änderungen haben somit Auswirkungen auf das gesamte Projekt.

Wenn Sie die Verbindung der Instanz zu ihrem Typen trennen, bearbeiten Sie anschließend das Objekt in der Projektnavigation wie ein gewöhnliches Objekt.

---

### Hinweis

#### Verbindung zum Typen nicht wiederherstellbar

Wenn Sie die Verbindung zwischen Instanz und Typ getrennt haben, ist diese nicht wiederherstellbar.

---

## Voraussetzung

Die Instanz darf sich nicht im Zustand "im Test" befinden.

## Vorgehen

Um die Verbindung zwischen einer Instanz und der Version eines Typen aufzuheben, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Instanz in der Projektnavigation.
2. Klicken Sie mit der rechten Maustaste auf die Instanz und wählen Sie im Kontextmenü den Befehl "Verbindung zum Typ aufheben".
3. Die Verbindung zur Version des korrespondierenden Typen wird aufgehoben.

## Siehe auch

Grundlagen zu Typen (Seite 337)

Grundlagen zu Bibliotheken (Seite 313)

Zustand von Typ-Versionen (Seite 339)

### 7.10.8.6 Arbeiten mit Typen in globalen Bibliotheken

#### Typen zu einer globalen Bibliothek hinzufügen

Globale Bibliotheken dienen als zentrale Ressource für die Arbeit an mehreren Projekten. Ein direktes Bearbeiten von Typen ist daher den Typen in der Projektbibliothek vorbehalten. Wenn Sie an Typen arbeiten möchten, verwenden Sie daher die Projektbibliothek. Wenn Sie mit dem Bearbeiten eines Typen in der Projektbibliothek fertig sind, können Sie den Typen in eine globale Bibliothek hinzufügen. Das Hinzufügen eines Typen aus der Projektbibliothek entspricht einem gewöhnlichen Kopiervorgang aus der Projektbibliothek.



## Voraussetzung

- Die Task Card "Bibliotheken" oder die Bibliotheksansicht ist geöffnet.
- Die globale Bibliothek, der Sie einen Typen hinzufügen möchten, ist mit Schreibrechten geöffnet.

## Vorgehen

Um einer globalen Bibliothek ein bestehendes Element als Typ hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Task Card "Bibliotheken" oder in der Bibliotheksansicht den gewünschten Ordner in der globalen Bibliothek.
2. Ziehen Sie einen Typen aus der Projektbibliothek per Drag & Drop in den Ordner "Typen" oder einen beliebigen Unterordner der globalen Bibliothek.  
Der neue Typ wird erzeugt.

Alternativ:

1. Kopieren Sie den gewünschten Typen aus der Projektbibliothek in die Zwischenablage.
2. Öffnen Sie in der Task Card "Bibliotheken" in der Palette "Globale Bibliothek" die globale Bibliothek, der Sie einen Typ hinzufügen möchten.
3. Klicken Sie mit der rechten Maustaste auf den Ordner "Typen" oder einen beliebigen Unterordner von "Typen".
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".  
Der neue Typ wird erzeugt.

## Ergebnis

Der Typ wird in die globale Bibliothek eingefügt. Abhängige Typen, wie z. B. Typen von HMI Anwenderdatentypen oder Variablen, werden ebenfalls in die globale Bibliothek kopiert, sofern sie dort noch nicht vorhanden sind. Somit wird sichergestellt, dass alle für das Erzeugen einer Instanz notwendigen Elemente in der globalen Bibliothek vorhanden sind.

Wenn der Typ, den Sie der globalen Bibliothek hinzufügen möchten, dort bereits vorhanden ist, dann entspricht das beschriebene Vorgehen einer Aktualisierung der globalen Bibliothek. In diesem Fall werden die neusten, freigegebenen Versionen des Typen in der globalen Bibliothek ergänzt.

## Siehe auch

Grundlagen zu Typen (Seite 337)

Version eines Typen freigeben (Seite 349)

Gemeinsame Version zuweisen (Seite 351)

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 357)

Grundlagen zu Bibliotheken (Seite 313)

## Typen verwenden

Um die Version eines Typen aus der globalen Bibliothek zu verwenden, erzeugen Sie eine Instanz der Version an einer geeigneten Stelle in der Projektnavigation. Bildbausteine und HMI Anwenderdatentypen können Sie auch direkt im Editor platzieren.

Instanzen von Typen aus einer globalen Bibliothek sind im Projekt nicht mit dem Typen in der globalen Bibliothek verbunden. Stattdessen wird beim Einfügen einer Instanz in das Projekt eine Kopie des Typen und seiner abhängigen Elemente in der Projektbibliothek erzeugt. Abhängige Elemente können beispielsweise PLC-Datentypen sein, die in einem Baustein referenziert werden. Die Kopie des Typen und der abhängigen Elemente in der Projektbibliothek enthält jeweils die Version, die Sie mit der Instanz verknüpfen. Wenn der Typ oder ein abhängiges Element bereits in der Projektbibliothek vorhanden sind, dann wird gegebenenfalls nur die fehlende Version bei dem Typen in der Projektbibliothek ergänzt.

Die Instanz ist mit der Kopie des Typen in der Projektbibliothek verbunden. Einem Gerät können Sie, unabhängig von der Version, einen Typen nur einmal zuweisen.

## Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet.
- Im Projekt befindet sich bereits ein Gerät, welches die Art des gewünschten Typen unterstützt.
- Dem Gerät ist noch keine weitere Instanz desselben Typen zugeordnet.

## Vorgehen

Um von der Version eines Typen eine Instanz zu erzeugen und zu verwenden, gehen Sie folgendermaßen vor:

1. Navigieren Sie in der globalen Bibliothek zu der Version des Typen, die Sie verwenden möchten.
2. Ziehen Sie die gewünschte Version des Typen per Drag & Drop an die Verwendungsstelle. Alternativ: Um automatisch die neueste Version zu verwenden, ziehen Sie den Typen selbst per Drag & Drop an die Verwendungsstelle.

## Ergebnis

Fehlende Elemente werden in der Projektbibliothek ergänzt. Von dem Typen und seiner abhängigen Elemente wird eine Instanz erzeugt und an der Verwendungsstelle eingefügt. Die Instanzen sind an die jeweilige Version des Typen in der Projektbibliothek gebunden.

## Siehe auch

Grundlagen zu Typen (Seite 337)

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 357)

Grundlagen zu Bibliotheken (Seite 313)

Elementeansicht verwenden (Seite 317)

## Projekt auf die neuesten Typ-Versionen aktualisieren

Vor allem in großen Betrieben mit vielen Automatisierungsprojekten werden globale Bibliotheken häufig zentral bearbeitet. Nach der Fertigstellung einer neueren Version werden die aktualisierten globalen Bibliotheken den einzelnen Projekten zur Verfügung gestellt. Wenn Sie eine neuere Version einer globalen Bibliothek bekommen haben, ersetzen Sie veraltetete Instanzen in Ihrem Projekt durch die neueste Version. Wenn Sie die Änderungen nicht auf das gesamte Projekt anwenden möchten, beschränken Sie die Aktualisierung auf einzelne Geräte im Projekt.

Während der Aktualisierung des Projekts oder einzelner Geräte wird auch die Projektbibliothek mit den neuen Versionen der Typen in der globalen Bibliothek aktualisiert.

Die folgenden Elemente sind als Quelle für die Aktualisierung wählbar:

- Eine globale Bibliothek
- Einzelne Ordner innerhalb einer globalen Bibliothek
- Einzelne Typen  
Die Auswahl mehrerer Typen ist zulässig.

## Voraussetzung

- Sie befinden sich in der Task Card "Bibliotheken" oder in der Bibliotheksansicht.
- Die aktualisierte globale Bibliothek ist geöffnet.

## Vorgehen

Um Instanzen in einem Projekt mit den Inhalten aus einer globalen Bibliothek zu aktualisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die aktualisierte globale Bibliothek oder einzelne Elemente daraus.
2. Klicken Sie mit der rechten Maustaste auf die globale Bibliothek oder die gewünschten Elemente und wählen Sie im Kontextmenü den Befehl "Aktualisieren > Projekt".  
Der Dialog "Projekt aktualisieren" wird geöffnet.
3. Wählen Sie entweder das gesamte Projekt oder einzelne Geräte für die Aktualisierung aus.
4. Wählen Sie Optionen für den Aktualisierungsvorgang:
  - Das Optionskästchen "Alle Instanzen der gewählten Typen aktualisieren" ist bei diesem Vorgang immer aktiv.
  - Aktivieren Sie das Optionskästchen "Alle nicht verwendeten Versionen aus der Projektbibliothek löschen", um alle älteren Versionen der aktualisierten Typen aus der Projektbibliothek zu löschen.
5. Bestätigen Sie mit "OK".  
Die Aktualisierung wird durchgeführt.

## Ergebnis

Am Projekt wurden folgende Änderungen durchgeführt:

- Die neueste Version der gewählten Typen befindet sich in der Projektbibliothek. Gegebenenfalls wurden alle älteren Versionen gelöscht.
- Alle Instanzen innerhalb der gewählten Geräte wurden auf die neueste Version des verküpften Typen aktualisiert.
- In der Projektnavigation unter "Gemeinsame Daten" finden Sie ein Protokoll des Aktualisierungsvorgangs.

## Siehe auch

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 352)

Protokolle verwenden (Seite 231)

Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren (Seite 360)

Grundlagen zu Typen (Seite 337)

Grundlagen zu Bibliotheken (Seite 313)

Typen zu einer globalen Bibliothek hinzufügen (Seite 354)

### 7.10.9 Bibliothekselemente bearbeiten

Innerhalb der Task Card "Bibliotheken" oder der Bibliotheksansicht lassen sich Typen, Kopiervorlagen und Ordner auf gewohnte Art und Weise ausschneiden, kopieren, einfügen, verschieben, umbenennen oder löschen. Globale Bibliotheken müssen für jeden der genannten Vorgänge mit Schreibrechten geöffnet sein.

#### Kopieren von Typen

Wenn Sie einen Typen in die Zwischenablage kopieren, dann gelten folgende Regeln:

- Typen werden immer mit allen zugehörigen Versionen in die Zwischenablage kopiert. Es werden jedoch nur Versionen kopiert, die bereits freigegeben sind.
- Typen werden immer mit allen abhängigen Elementen in die Zwischenablage kopiert.
- Kopiervorlagen werden immer zusammen mit allen darin verwendeten Typen und allen Versionen der Typen kopiert.

#### Ausschneiden von Elementen

Ausgeschnittene Bibliothekselemente können Sie nur in die gleiche Bibliothek einfügen. Dabei können Sie Kopiervorlagen nur in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen" und Typen nur in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen" einfügen.

## **Einfügen von Typen**

Das Einfügen von Typen in eine andere Bibliothek entspricht einer Aktualisierung der Zielbibliothek.

Wenn Sie einen Typen in eine andere Bibliothek eingefügt haben, gelten folgende Regeln:

- Ein Typ wird immer mit all seinen Versionen eingefügt.
- Wenn der Typ bereits in der Zielbibliothek existiert, werden alle Versionen, die neuer als die bereits vorhandenen Versionen sind, bei dem entsprechenden Typen in der Zielbibliothek ergänzt.
- Wenn eine Version im freigegebenen Zustand bereits in der Zielbibliothek existiert, wird dieselbe Version nicht noch einmal eingefügt.
- Wenn dieselbe Version im Test oder in Bearbeitung bereits in der Zielbibliothek existiert, wird diese durch die freigegebene Version ersetzt.
- Wenn ein Typ andere Typen benötigt, dann werden diese an entsprechender Stelle ebenfalls eingefügt.

## **Einfügen von Kopiervorlagen**

Beim Einfügen von Kopiervorlagen werden alle darin verwendeten Typen und ihre Versionen ebenfalls eingefügt. Wenn der entsprechende Typ bereits in der Bibliothek vorhanden ist, werden nur die fehlenden Versionen ergänzt. Wenn ein Typ noch nicht vorhanden ist, wird er mit allen seinen Versionen auf oberster Ebene in die Bibliothek eingefügt.

## **Elemente verschieben**

Wenn Sie ein Element aus einer Bibliothek in eine andere Bibliothek verschieben, wird das Element kopiert und nicht verschoben. Es gelten dann dieselben Regeln, wie unter "Einfügen von Typen" und "Einfügen von Kopiervorlagen" genannt sind.

## **Löschen von Typen und Typ-Versionen**

Beim Löschen von Typen oder Typ-Versionen ist Folgendes zu beachten:

- Ein Typ oder eine Typ-Version lässt sich nur löschen, wenn keine Abhängigkeiten zu anderen Typen bestehen.
- Wenn Sie einen Typen löschen, dann werden alle Versionen des Typen gelöscht.
- Wenn Sie alle Versionen eines Typen löschen, dann wird der Typ ebenfalls gelöscht.
- Wenn Sie eine Version löschen, die Instanzen im Projekt besitzt, dann werden die Instanzen ebenfalls aus dem Projekt gelöscht.

## **Löschen von Instanzen**

Wenn Sie eine Instanz löschen, die Abhängigkeiten zu anderen Instanzen besitzt, dann wird diese bei der nächsten Übersetzung wiederhergestellt. Die Instanz wird wieder mit der ursprünglichen Typ-Version verknüpft. Somit wird die Konsistenz des Projekts wieder hergestellt.

## Siehe auch

Grundlagen zu Bibliotheken (Seite 313)

Verbindung zwischen Instanz und Typ aufheben (Seite 354)

Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren (Seite 360)

### 7.10.10 Bibliothek mit den Inhalten einer anderen Bibliothek aktualisieren

Eine bereits vorhandene Bibliothek lässt sich mit den Inhalten aus einer anderen Bibliothek aktualisieren. Folgende Möglichkeiten zur Aktualisierung von Bibliotheken bestehen:

- Aktualisieren einer globalen Bibliothek mit Typen aus einer anderen globalen Bibliothek oder der Projektbibliothek
- Aktualisieren der Projektbibliothek mit Typen aus einer globalen Bibliothek

Die folgenden Elemente sind jeweils als Quelle für die Aktualisierung wählbar:

- Eine gesamte Bibliothek
- Einzelne Ordner innerhalb einer Bibliothek
- Einzelne Typen  
Die Auswahl mehrerer Typen ist zulässig.

Bei der Aktualisierung werden bereits in der Zielbibliothek vorhandene Typen um die neuesten Versionen ergänzt. Typen, die noch nicht in der Zielbibliothek vorhanden sind, werden mit allen ihren Versionen in die Zielbibliothek kopiert.

## Voraussetzung

Wenn Sie eine globale Bibliothek aktualisieren möchten, dann muss diese mit Schreibrechten geöffnet sein.

## Vorgehen

Um eine Bibliothek mit Inhalten einer anderen Bibliothek zu aktualisieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine Bibliothek oder einzelne Elemente daraus als Quelle für die Aktualisierung.
2. Klicken Sie mit der rechten Maustaste auf die Quelle und wählen Sie im Kontextmenü den Befehl "Aktualisieren > Bibliothek".  
Der Dialog "Bibliothek aktualisieren" wird geöffnet.
3. Wählen Sie aus, welche Art von Bibliothek Sie aktualisieren möchten:
  - Wählen Sie "Projektbibliothek aktualisieren", um die Projektbibliothek mit Typen aus einer globalen Bibliothek zu aktualisieren.
  - Wählen Sie "Globale Bibliothek aktualisieren", wenn Sie eine globale Bibliothek aktualisieren möchten.

4. Optional: Wählen Sie aus der Klappliste die globale Bibliothek, die Sie aktualisieren möchten, wenn Sie eine globale Bibliothek aktualisieren möchten.
5. Wählen Sie die Optionen für die Aktualisierung:
  - Die Option "Alle Instanzen aktualisieren" ist bei diesem Vorgang immer deaktiviert.
  - Aktivieren Sie das Optionskästchen "Nicht verwendeten Versionen aus der Projektbibliothek löschen", wenn Sie alle Versionen von Typen aus der Projektbibliothek löschen möchten, die mit keiner Instanz im Projekt verbunden sind und die keine Abhängigkeiten zu anderen Typen besitzen. Die Option ist bei der Aktualisierung einer globalen Bibliothek nicht wählbar, da Typen einer globalen Bibliothek niemals eine Verwendungsstelle im Projekt besitzen.
6. Bestätigen Sie mit "OK".  
Die Aktualisierung wird durchgeführt.

## Ergebnis

An der Zielbibliothek wurden folgende Änderungen durchgeführt:

- Typen, die in der Zielbibliothek noch nicht vorhanden sind, wurden mit allen ihren Versionen ergänzt. Typen, die bereits in der Zielbibliothek vorhanden sind, wurden um die neueren Versionen ergänzt. Wenn in der Zielbibliothek bereits eine neuere Version eines Typen vorhanden war, wurde die neueste Version aus der Quellbibliothek trotzdem kopiert und automatisch eine neuere Versionsnummer vergeben.
- Gegebenenfalls wurden alle Versionen von Typen aus der Projektbibliothek gelöscht, wenn diese in keiner Instanz im Projekt verwendet sind.
- Für den Aktualisierungsvorgang wurde ein Protokoll angelegt, das alle durchgeführten Änderungen an der Zielbibliothek enthält.  
Wenn Sie die Projektbibliothek aktualisiert haben, finden Sie das Protokoll in der Projektnavigation unter "Gemeinsame Daten > Protokolle".  
Wenn Sie eine globale Bibliothek aktualisiert haben, finden Sie das Protokoll, der globalen Bibliothek unterlagert, im Ordner "Gemeinsame Daten > Protokolle".

## Siehe auch

Protokolle verwenden (Seite 231)

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 352)

Projekt auf die neuesten Typ-Versionen aktualisieren (Seite 357)

Protokolle zu globalen Bibliotheken anzeigen (Seite 327)

Grundlagen zu Bibliotheken (Seite 313)

### 7.10.11 Namen und Pfadstruktur harmonisieren

Sie haben die Möglichkeit, das Projekt mit einer Bibliothek zu harmonisieren. Dadurch können Sie folgende Punkte korrigieren lassen:

- **Namen der Instanzen:**  
Während der Entwicklungsphase einer Bibliothek können Instanzen entstehen, deren Namen durch eine automatische Korrektur um "\_1", „\_2“ etc. erweitert wurden. Die Erweiterung erfolgt, um doppelte Namen im Projekt zu vermeiden. Beim Harmonisieren erhalten die Instanzen wieder die Namen ihrer zugehörigen Typen.
- **Pfadstruktur:**  
Durch parallele Entwicklung oder Kopieren von abhängigen Instanzen kann die ursprüngliche Pfadstruktur verloren gehen. Dies beeinträchtigt die Übersichtlichkeit des Projekts. Beim Harmonisieren wird die Pfadstruktur innerhalb des Projekts an die Pfadstruktur der Bibliothek angepasst.

#### Vorgehen

Um die Namen und die Pfadstruktur zu harmonisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksverwaltung.
2. Klicken Sie in der Funktionsleiste auf "Projekt harmonisieren".  
Der Dialog "Projekt harmonisieren" wird geöffnet.
3. Wählen Sie das Gerät, mit dem Sie die Bibliothek harmonisieren möchten.
4. Aktivieren Sie das Optionskästchen "Pfade zwischen Projekt und Bibliothek harmonisieren", wenn Sie die Pfadstruktur wiederherstellen möchten.
5. Aktivieren Sie das Optionskästchen "Namen zwischen Projekt und Bibliothek harmonisieren", wenn Sie die Namen korrigieren lassen möchten.
6. Bestätigen Sie Ihre Eingaben mit "OK".  
Abhängig von Ihren Einstellungen werden die Namen und die Pfadstruktur im Projekt mit der Bibliothek harmonisiert.

#### Siehe auch

Grundlagen zu Bibliotheken (Seite 313)

Übersicht über die Bibliotheksansicht (Seite 318)

Übersicht über die Bibliotheksverwaltung (Seite 321)

### 7.10.12 Bibliothek aufräumen

Sie können die Projektbibliothek oder globale Bibliotheken aufräumen, um Typen oder Versionen zu entfernen, wenn diese mit keiner Instanz im Projekt verbunden sind. Dies erhöht die Übersichtlichkeit innerhalb der Bibliotheken und verringert die Größe der Bibliothek.



### **Projektbibliothek aufräumen**

Um die Projektbibliothek aufzuräumen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksverwaltung.
2. Klicken Sie in der Funktionsleiste auf "Bibliothek aufräumen".  
Der Dialog "Projektbibliothek aufräumen" wird geöffnet.
3. Wählen Sie aus, ob Sie nur nicht verwendete Typ-Versionen oder vollständige Typen aus der Bibliothek entfernen möchten.
4. Bestätigen Sie Ihre Eingabe mit "OK".  
Abhängig von Ihrer Auswahl werden entweder nicht verwendete Typ-Versionen oder Typen aus der Projektbibliothek entfernt.

### **Globale Bibliothek aufräumen**

Um eine globale Bibliothek aufzuräumen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bibliotheksverwaltung.
2. Klicken Sie in der Funktionsleiste auf "Bibliothek aufräumen".  
Der Dialog "Globale Bibliothek aufräumen" wird geöffnet.
3. Klicken Sie auf "Fortsetzen".  
Nicht verwendete Typ-Versionen werden gelöscht. Die neueste Version eines Typen bleibt immer erhalten.

### **Siehe auch**

- Grundlagen zu Bibliotheken (Seite 313)
- Übersicht über die Bibliotheksansicht (Seite 318)
- Übersicht über die Bibliotheksverwaltung (Seite 321)

## **7.10.13 Bibliothekselemente vergleichen**

### **Einführung**

Sie können Geräte aus Bibliotheken sowohl mit Geräten aus dem aktuellen Projekt als auch aus der gleichen oder aus anderen Bibliotheken oder aus Referenzprojekten vergleichen. Beachten Sie jedoch, dass Referenzprojekte schreibgeschützt sind. Sie haben auch die Möglichkeit, Instanzen in einem Gerät mit ihrer Typversion in einer Bibliothek zu vergleichen. Beim Vergleich mit Typen stehen Ihnen nicht alle Aktionen zur Verfügung. So können Sie z. B. eine Instanz einer neueren Version nicht mit einer älteren Typversion aus der Bibliothek überschreiben.

Sie können beim Vergleichen von Bibliothekselementen jederzeit zwischen automatischem und manuellem Vergleich wechseln.

## Vorgehen

Um Bibliothekselemente mit den Gerätedaten eines Projekts zu vergleichen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Gerät, dessen Daten Sie mit einem Bibliothekselement vergleichen möchten und das einen Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Öffnen Sie die Task Card "Bibliotheken".
4. Selektieren Sie das Bibliothekselement, das Sie mit den Gerätedaten vergleichen möchten.
5. Ziehen Sie das Bibliothekselement in die rechte Drop-Fläche des Vergleichseditors.  
Den Status der Objekte können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Wenn Sie ein Objekt selektieren, werden die Eigenschaften des Objekts und des entsprechenden Objekts des zugeordneten Geräts übersichtlich im Eigenschaftsvergleich angezeigt.  
Sie können jederzeit aus dem aktuellen Projekt, einer Bibliothek oder aus einem Referenzprojekt andere Geräte in die Drop-Flächen ziehen und somit einen neuen Vergleich starten. Dabei ist es unerheblich, welches Gerät Sie in welche Drop-Fläche ziehen.

## Siehe auch

- Vergleichseditor verwenden (Seite 264)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Bibliotheksansicht verwenden (Seite 318)
- Grundlagen zu Bibliotheken (Seite 313)
- Übersicht über die Task Card "Bibliotheken" (Seite 315)
- Übersicht über die Bibliotheksansicht (Seite 318)
- Elementeansicht verwenden (Seite 317)
- Globalen Bibliotheken verwenden (Seite 323)

## 7.11 Querverweise nutzen

### 7.11.1 Querverweise nutzen

#### Einführung zu Querverweisen

Die Querverweisliste bietet einen Überblick über die Verwendung von Objekten innerhalb des Projekts. Sie können ablesen, welche Objekte in gegenseitiger Abhängigkeit stehen und wo sich die einzelnen Objekte befinden. Querverweise sind somit Teil der Projektdokumentation.

Zusätzlich können Sie direkt zur Verwendungsstelle eines Objekts springen.

Welche Objekte in der Querverweisliste angezeigt und lokalisiert werden können, ist abhängig von den installierten Produkten.

## 7.12 Geräte simulieren

### 7.12.1 Simulation von Geräten

#### Einführung

Mit dem TIA-Portal können Sie die Hardware und Software des Projekts in einer simulierten Umgebung ausführen und testen. Die Simulation wird direkt auf dem PG/PC ausgeführt. Daher ist keine zusätzliche Hardware nötig.

Die Simulationssoftware bietet eine grafische Bedienoberfläche zum Beobachten und Ändern der Projektierung. Sie unterscheidet sich je nach aktuell selektiertem Gerät.

#### Einbindung in das TIA-Portal

Die Simulationssoftware ist vollständig in das TIA-Portal integriert, wird aber nur von bestimmten Geräten unterstützt. Daher ist die Schaltfläche zum Aufruf der Simulationssoftware nur bedienbar, wenn das selektierte Gerät die Simulation unterstützt.

Die Simulationssoftware zu manchen Geräten benötigt eine eigene virtuelle Schnittstelle, um mit den simulierten Geräten zu kommunizieren. Die virtuelle Schnittstelle finden Sie in der Projektnavigation unter dem Eintrag "Online-Zugänge" neben den physikalischen Schnittstellen des PG/PC.

Weiterführende Hilfe zur Simulationssoftware bekommen Sie nach dem Öffnen der Software über einen gesonderten Link.

#### Siehe auch

Simulation starten (Seite 365)

### 7.12.2 Simulation starten

Einige Geräte lassen sich mit einer zusätzlichen Software simulieren. Sie müssen daher nicht tatsächlich über die Geräte verfügen, können aber trotzdem Ihr Projekt umfangreich testen.

## **Vorgehen**

Um die Simulations-Software zu starten, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Gerät, das Sie simulieren möchten, z. B. in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "Simulation > Starten".  
Die Simulations-Software wird aufgerufen.

## **Siehe auch**

Simulation von Geräten (Seite 365)

# Geräte und Netze bearbeiten

## 8.1 Geräte und Netze konfigurieren

### 8.1.1 Hardware- und Netzwerkeditor

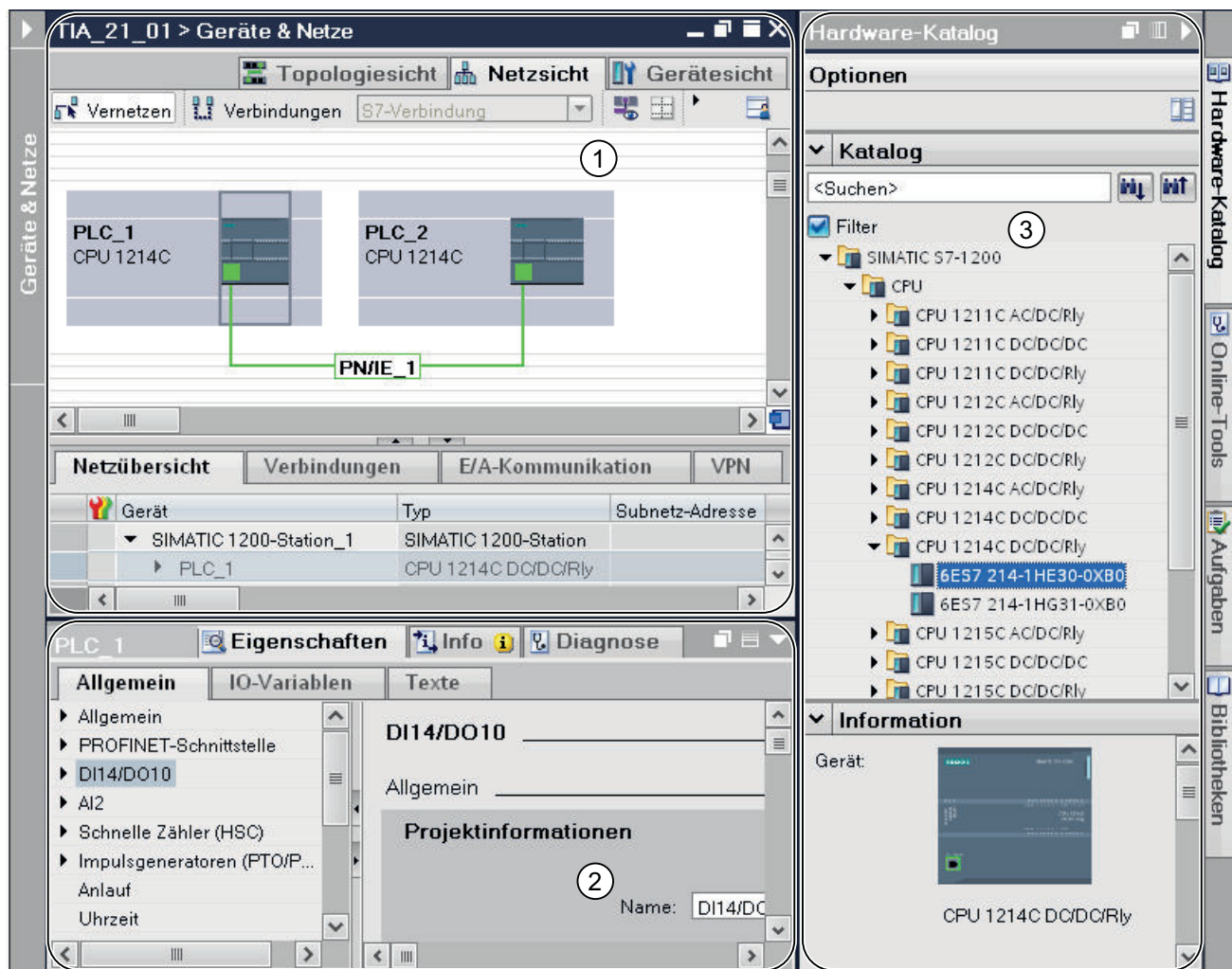
#### 8.1.1.1 Übersicht über den Hardware- und Netzwerkeditor

#### **Funktion des Hardware- und Netzwerkeditors**

Wenn Sie in der Projektnavigation auf den Eintrag "Geräte & Netze" doppelklicken, öffnet sich der Hardware- und Netzwerkeditor. Der Hardware- und Netzwerkeditor ist die integrierte Entwicklungsumgebung für das Konfigurieren, Parametrieren und Vernetzen von Geräten und Baugruppen. Er bietet maximale Unterstützung für die Durchführung des Automatisierungsprojekts.

#### **Aufbau des Hardware- und Netzwerkeditors**

Der Hardware- und Netzwerkeditor besteht aus den folgenden Komponenten:



- ① Gerätesicht (Seite 371) , Netzsicht (Seite 369) , Topologiesicht (Seite 374)
- ② Inspektorfenster (Seite 379)
- ③ Hardware-Katalog (Seite 381)

Der Hardware- und Netzwerkeditor bietet Ihnen drei unterschiedliche Sichten auf Ihr Projekt. Zwischen diesen drei Sichten können Sie jederzeit umschalten, je nachdem, ob Sie einzelne Geräte und Baugruppen, ganze Netzwerke und Gerätekonfigurationen oder den topologischen Aufbau ihres Projektes erstellen und bearbeiten möchten.

Im Inspektorfenster finden Sie Informationen über das aktuell markierte Objekt. Dort können Sie auch die Einstellungen für das markierte Objekt ändern.

Aus dem Hardware-Katalog ziehen Sie sich die für Ihr Automatisierungssystem benötigten Geräte und Baugruppen in die Geräte-, Netz- oder Topologiesicht.

### 8.1.1.2 Netzsicht

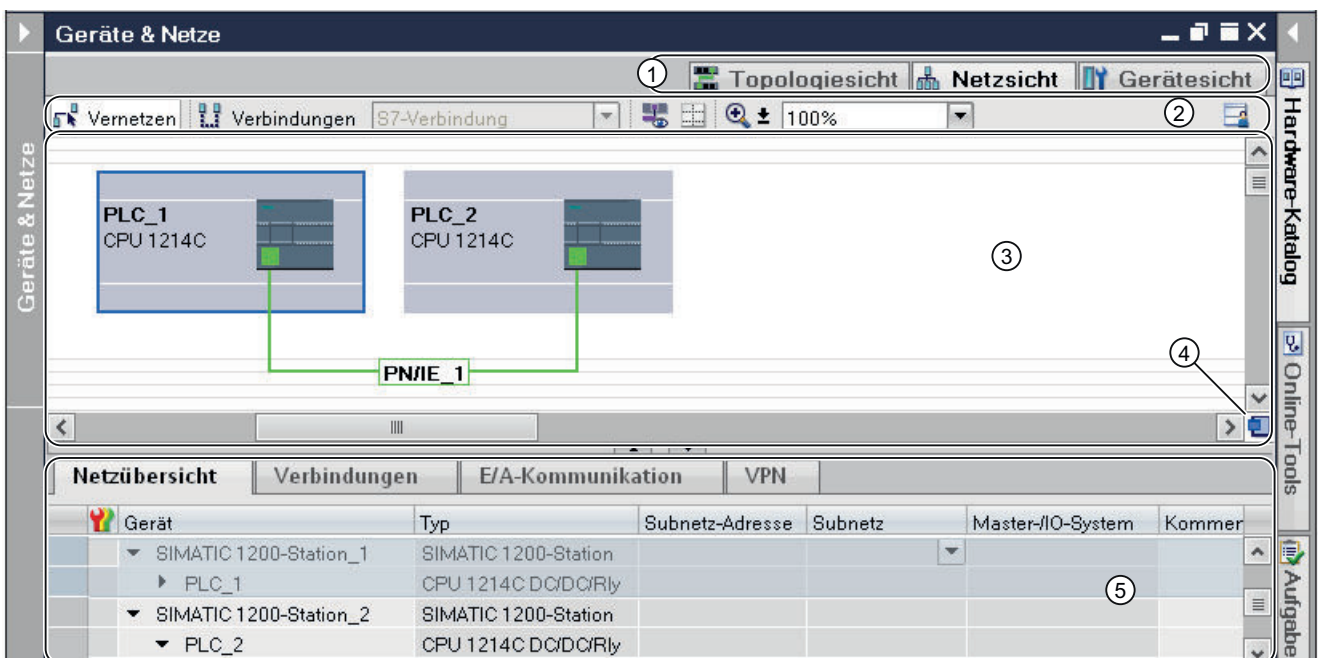
#### Einführung

Die Netzsicht ist einer von drei Arbeitsbereichen des Hardware- und Netzwerkeditors. Hier führen Sie folgende Aufgaben durch:

- Geräte konfigurieren und parametrieren
- Geräte miteinander vernetzen

#### Aufbau

Das folgende Bild zeigt die beiden Bestandteile der Netzsicht:










- ① Umschalter Netzsicht/Gerätesicht/Topologiesicht
- ② Funktionsleiste der Netzsicht
- ③ Grafischer Bereich der Netzsicht
- ④ Übersichtsnavigation
- ⑤ Tabellarischer Bereich der Netzsicht

Die Teilung zwischen grafischem und tabellarischem Bereich der Netzsicht können Sie mit der Maus verändern. Klicken Sie dazu mit der Maus auf den oberen Rand des tabellarischen Bereichs und ziehen Sie den tabellarischen Bereich bei gedrückter Maustaste größer oder kleiner. Über den Speedy Splitter (die beiden kleinen Pfeiltasten) können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

## Funktionsleiste

Über die Funktionsleiste stehen Ihnen die folgenden Funktionen zur Verfügung:

| Symbol  | Bedeutung   |
|---|---|
|  | Modus zum Vernetzen der Geräte.   |
|  | Modus zum Erstellen von Verbindungen.<br>Über die nebenstehende Klappliste können Sie den Verbindungstyp einstellen.  |
|  | Modus zum Erstellen von Relationen.   |
|  | Anzeige der Schnittstellenadressen.   |
|  | Einstellen der Zoomstufe.<br>Über die nebenstehende Klappliste können Sie die Zoomstufe auswählen oder direkt eintragen. Über das Zoomsymbol können Sie die Ansicht auch schrittweise vergrößern oder verkleinern oder einen Rahmen um einen zu vergrößernden Bereich ziehen. |
|  | Zeige Seitenumbrüche<br>Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.  |
|  | Anordnung merken<br>Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.  |

## Grafischer Bereich

Im grafischen Bereich der Netzsicht werden Ihnen netzrelevante Geräte, Netze, Verbindungen und Relationen angezeigt. Hier fügen Sie Geräte aus dem Hardware-Katalog ein, verbinden diese über ihre Schnittstellen miteinander und bearbeiten die Kommunikationseinstellungen.

## Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

## Tabellarischer Bereich

Der tabellarische Bereich der Netzsicht beinhaltet verschiedene Tabellen zu den vorhandenen Geräten, Verbindungen und Kommunikationseinstellungen:

- Netzübersicht
- Verbindungen
- E/A-Kommunikation

Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen.



## Siehe auch

- Gerät zur Hardware-Konfiguration hinzufügen (Seite 399)
- Aufbau der Bedienoberfläche (Seite 176)
- Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 966)
- Geräte in der Netzsicht vernetzen (Seite 414)
- Tabellarische Netzübersicht (Seite 417)

### 8.1.1.3 Gerätesicht

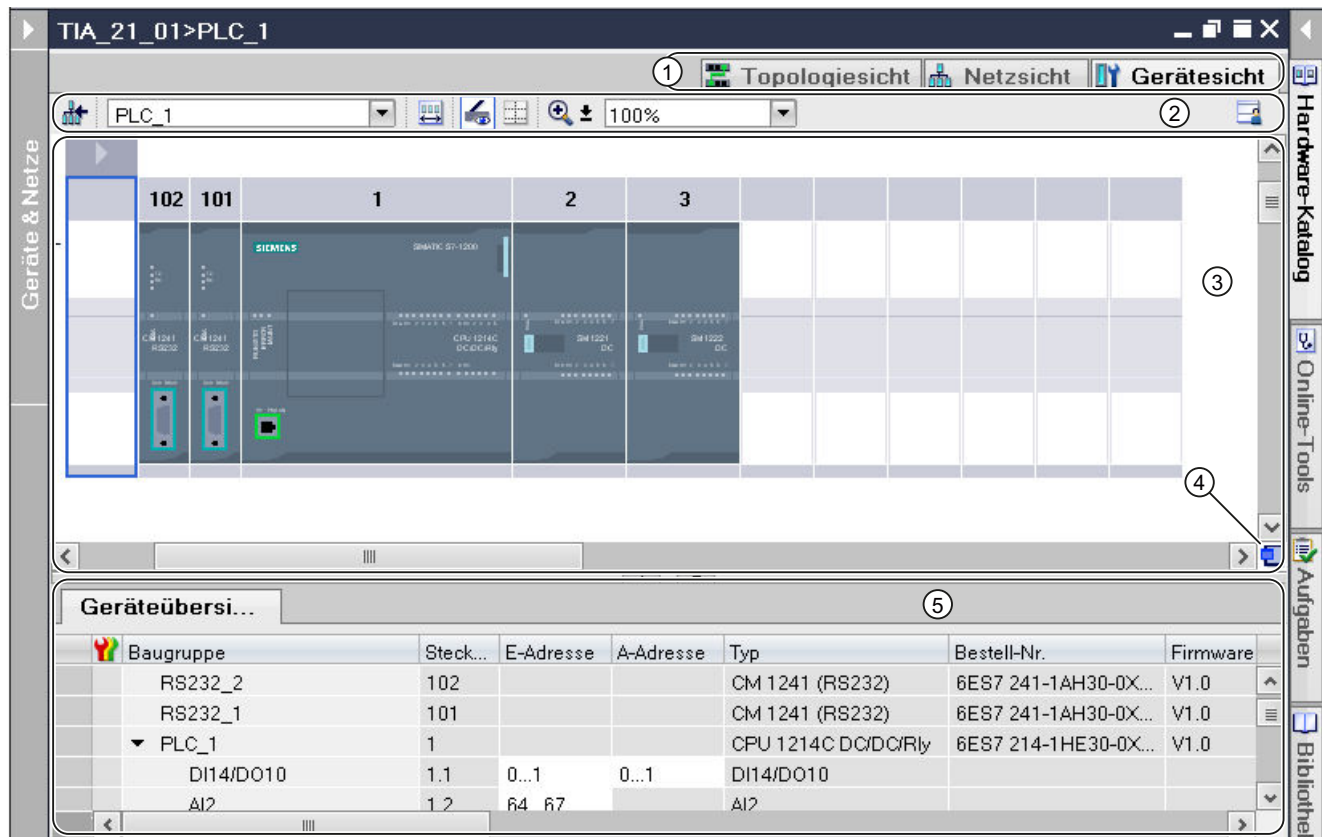
#### Einführung

Die Gerätesicht ist einer von drei Arbeitsbereichen des Hardware- und Netzwerkeditors. Hier führen Sie folgende Aufgaben durch:

- Geräte konfigurieren und parametrieren
- Baugruppen konfigurieren und parametrieren

**Aufbau**

Das folgende Bild zeigt die beiden Bestandteile der Gerätesicht:









- ① Umschalter Netzsicht/Gerätesicht/Topologiesicht
- ② Funktionsleiste der Gerätesicht
- ③ Grafischer Bereich der Gerätesicht
- ④ Übersichtsnavigation
- ⑤ Tabellarischer Bereich der Gerätesicht

Die Teilung zwischen grafischem und tabellarischem Bereich der Gerätesicht können Sie mit der Maus verändern. Klicken Sie dazu mit der Maus auf den oberen Rand des tabellarischen Bereichs und ziehen Sie den tabellarischen Bereich bei gedrückter Maustaste größer oder kleiner. Über den Speedy Splitter (die beiden kleinen Pfeiltasten) können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

## Funktionsleiste

Über die Funktionsleiste stehen Ihnen die folgenden Funktionen zur Verfügung:

| Symbol  | Bedeutung  |
|---|--|
|  | Schaltet in die Netzsicht um.<br>Hinweis: Über die nebenstehende Klappliste kann die Gerätesicht zwischen den vorhandenen Geräten umgeschaltet werden.   |
|  | Anzeige der Ablage nicht gesteckter Baugruppen.  |
|  | Anzeige der Baugruppenbeschriftung.  |
|  | Einstellen der Zoomstufe.<br>Sie können die Zoomstufe in der nebenstehenden Klappliste auswählen oder direkt eintragen. Über das Zoomsymbol können Sie die Ansicht auch schrittweise vergrößern oder verkleinern oder einen Rahmen um einen zu vergrößernden Bereich ziehen.<br>Ab einer Zoomstufe von 200 % können Sie bei Signalmodulen die Adressbezeichnungen der E/A-Kanäle erkennen. |
|  | Zeige Seitenumbrüche<br>Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.   |
|  | Anordnung merken<br>Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.   |

## Grafischer Bereich

Im grafischen Bereich der Gerätesicht werden Ihnen Hardware-Komponenten und ggf. die zugehörigen Baugruppen angezeigt, die über einen oder mehrere Baugruppenträger einander zugeordnet sind. Bei Geräten mit Baugruppenträgern können Sie hier weitere Hardware-Objekte aus dem Hardware-Katalog in die Steckplätze der Baugruppenträger stecken.

## Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

## Tabellarischer Bereich

Der tabellarische Bereich der Gerätesicht gibt Ihnen eine Übersicht über die verwendete Hardware mit den wichtigsten technischen und organisatorischen Daten.

Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen.

## Siehe auch

Arbeiten mit Baugruppenträgern (Seite 392)

Netzsicht (Seite 369)

Ablage nicht gesteckter Baugruppen (Seite 396)

Baugruppe in Baugruppenträger stecken (Seite 402)

Objekte in der Gerätesicht (Seite 394)

Aufbau der Bedienoberfläche (Seite 176)

Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 966)

### 8.1.1.4 Topologiesicht

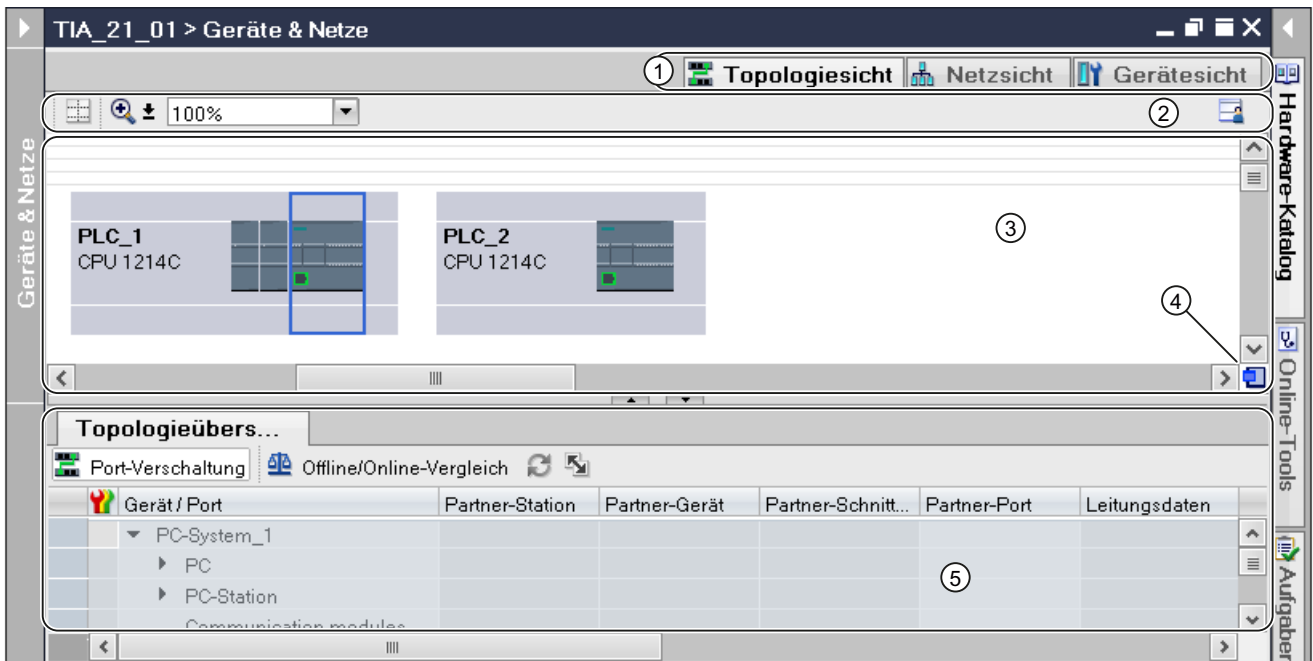
#### Einführung

Die Topologiesicht ist einer von drei Arbeitsbereichen des Hardware- und Netzwerkeditors. Hier führen Sie folgende Aufgaben durch:

- Ethernet-Topologie anzeigen
- Ethernet-Topologie projektieren
- Unterschiede zwischen Soll- und Ist-Topologie ermitteln und minimieren

## Aufbau

Das folgende Bild gibt Ihnen einen Überblick über die Topologiesicht.






- ① Umschalter Gerätesicht/Netzsicht/Topologiesicht
- ② Funktionsleiste der Topologiesicht
- ③ Grafischer Bereich der Topologiesicht
- ④ Übersichtsnavigation
- ⑤ Tabellarischer Bereich der Topologiesicht

Die Teilung zwischen grafischem und tabellarischem Bereich der Topologiesicht können Sie mit der Maus verändern. Klicken Sie dazu mit der Maus auf den oberen Rand des tabellarischen Bereichs und ziehen Sie den tabellarischen Bereich bei gedrückter Maustaste größer oder kleiner. Über den Speedy Splitter (die beiden kleinen Pfeiltasten) können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

### Funktionsleiste

Über die Funktionsleiste stehen Ihnen die folgenden Funktionen zur Verfügung:

| Symbol  | Bedeutung   |
|---|---|
|  | Einstellen der Zoom-Stufe.<br>Über die nebenstehende Klappliste können Sie die Zoom-Stufe auswählen oder direkt eintragen. Über das Zoom-Symbol können Sie die Ansicht auch stufenweise vergrößern oder verkleinern oder einen Rahmen um einen zu vergrößernden Bereich ziehen. |
|  | Zeige Seitenumbrüche<br>Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.  |
|  | Anordnung merken<br>Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.  |

### Grafischer Bereich

Im grafischen Bereich der Topologiesicht werden Ihnen Ethernet-Baugruppen mit den zugehörigen Ports und Portverschaltungen angezeigt. Hier können Sie weitere Hardware-Objekte mit Ethernet-Schnittstellen hinzufügen. Siehe: Gerät zur Hardware-Konfiguration hinzufügen (Seite 399)

### Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

### Tabellarischer Bereich

Hier werden Ihnen die Ethernet- bzw. PROFINET-Baugruppen, deren Ports und Portverschaltungen in tabellarischer Form angezeigt. Diese Tabelle entspricht der Netzübersichtstabelle in der Netzsicht.

Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen.

### Siehe auch

Aufbau der Bedienoberfläche (Seite 176)

Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 966)

### 8.1.1.5 Drucken von Hardware- und Netzkonfigurationen

#### Ausdruck von Hardware- und Netzwerkkonfigurationen

Im Rahmen der Projektdokumentation können Sie die folgenden Elemente der Hardware- und Netzsicht ausdrucken:

- Grafische Netzsicht
- Netzübersichtstabelle
- Grafische Gerätesicht
- Die Geräteübersichtstabelle
- Die Parameter des aktuell im Editor selektierten Objekts

#### Ausdruck der Inhalte eines Editors

Wenn Sie den Druck innerhalb eines geöffneten Editor starten und keine Baugruppe ist selektiert, werden immer die Inhalte des Editors gedruckt. Dazu gehört die grafische Darstellung des Editors wie auch die dem Editor zugehörige Tabelle. Sie können den Druckumfang jedoch anpassen. Sie können einstellen, ob nur die grafische Sicht, nur die Tabelle oder beides zusammen gedruckt werden sollen. Lesen Sie dazu das Kapitel "Optionen für den Druck ändern (Seite 379)".

Ist die Grafik größer als das gewählte Seitenlayout, wird der Ausdruck auf der nächsten Seite fortgesetzt. So gehen keine Inhalte verloren. Alternativ können Sie die Zoomstufe der grafischen Darstellung ändern, um den Ausdruck auf eine Seite anzupassen. Der Ausdruck erfolgt immer in der aktuell gewählten Zoomstufe.

Zur Kontrolle, ob alle Inhalte auf eine Seite passen, können Sie entweder die Druckvorschau nutzen oder Sie aktivieren die Seitenumbruchvorschau. Bei aktivierter Seitenumbruchvorschau werden innerhalb des grafischen Editors gestrichelte Linien an der Stelle angezeigt, wo später ein Seitenumbruch erfolgt.

#### Druck von sehr großen Tabellen

Ist eine Tabelle größer als die Druckfläche und kann somit nicht vollständig gedruckt werden, so werden die Inhalte der Tabelle nicht in tabellarischer Form gedruckt, sondern als Paare zwischen Wert und Schlüssel.

Beispiel:

| Objektname | Eigenschaft 1 | Eigenschaft 2 |
|------------|---------------|---------------|
| Objekt A   | Wert A1       | Wert A2       |
| Objekt B   | Wert B1       | Wert B2       |

In diesem Fall sieht der Ausdruck wie folgt aus:

##### **Objekt A**

Eigenschaft 1: Wert A1

Eigenschaft 2: Wert A2

### Objekt B

Eigenschaft 1: Wert B1

Eigenschaft 2: Wert B2

Sie können Sie diese Druckform auch voreinstellen, so dass Tabellen immer als Paar zwischen Schlüssel und Wert gedruckt werden. Lesen Sie dazu das Kapitel "Einstellungen für den Druck ändern (Seite 278)".

### Ausdruck von Baugruppenparametern

Parameter von selektierten Baugruppen werden zusammen mit den aktuell eingestellten Werten in Textform ausgedruckt. Alle Parameter von zugehörigen Baugruppen werden ebenfalls mit ausgedruckt. Haben Sie beispielsweise eine CPU selektiert, werden die Parameter eines eventuell gesteckten Signal Boards mit gedruckt.

Sie können bestimmen, in welchem Umfang die Parameter von Baugruppen gedruckt werden. Im Dialog "Drucken" können Sie wählen, ob Sie alle Eigenschaften und Parameter einer Baugruppe drucken möchten, oder ob Sie einen Kompakt-Ausdruck wünschen. Wählen Sie die kompakte Form, werden nur die Einträge gedruckt, die im Bereich "Allgemein" in den Eigenschaften der Baugruppe eingetragen sind. Davon ausgenommen sind Kommentare zur Baugruppe sowie der Autor und die Baugruppenbeschreibung. Im Kompakt-Modus werden also beispielsweise die folgenden Parameter einer Baugruppe gedruckt:

- Angaben zur Baugruppe  
Name, Baugruppensteckplatz, Kurzbeschreibung, Bestellnummer, Firmware-Version
- Name der PROFINET-Schnittstelle
- Angaben zum Subnetz  
Name des Subnetzes, ID des S7-Subnetzes

### Siehe auch

Optionen für den Druck ändern (Seite 379)

Dokumentationsfunktion (Seite 276)

Druckvorschau erstellen (Seite 290)

Projektdateien drucken (Seite 293)

Seitenumbruchvorschau für den Druck aktivieren (Seite 378)

#### 8.1.1.6 Seitenumbruchvorschau für den Druck aktivieren

In den grafischen Editoren können Sie eine Seitenumbruchvorschau für den Druck aktivieren. Ist die Option aktiviert, werden an den Stellen innerhalb des grafischen Editors gestrichelte Linien angezeigt, an denen beim späteren Ausdruck der Seitenumbruch erfolgt.



## Vorgehen

Um die Seitenumbruchvorschau zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie den grafischen Bereich der entsprechenden Ansicht aus.
2. Klicken Sie auf das Symbol "Zeige Seitenumbrüche" in der Funktionsleiste des grafischen Editors.  
Innerhalb des grafischen Editors werden gestrichelte Linien an den Stellen angezeigt, wo später ein Seitenumbruch erfolgt.
3. Um das Rahmenlayout zu ändern, wählen Sie im Menü "Projekt" den Befehl "Drucken".
4. Um die Seitenumbruchvorschau zu deaktivieren, klicken Sie erneut auf das Symbol "Zeige Seitenumbrüche" in der Funktionsleiste des grafischen Editors,.

### 8.1.1.7 Optionen für den Druck ändern

#### Umfang des Ausdrucks ändern

Beim Drucken aus einem Editor heraus, können Sie festlegen, ob sowohl Grafiken als auch Tabellen gedruckt werden sollen, oder ob Sie nur eines von beiden ausdrucken möchten. Standardmäßig wird beides ausgedruckt.

## Vorgehen

Um den Umfang des Ausdrucks zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
2. Öffnen Sie in der Bereichsnavigation die Parametergruppe "Druckeinstellungen" unter "Allgemein".
3. Scrollen Sie bis zur Gruppe "Hardware-Konfiguration".
4. Aktivieren oder deaktivieren Sie das Optionskästchen "Aktive grafische Sicht", je nachdem ob Sie die Grafiken der Netz- und Gerätesicht mit ausdrucken möchten.
5. Aktivieren oder deaktivieren Sie das Optionskästchen "Aktive Tabelle", je nachdem ob Sie die einem Editor zugehörige Tabelle mit ausdrucken möchten.

## Siehe auch

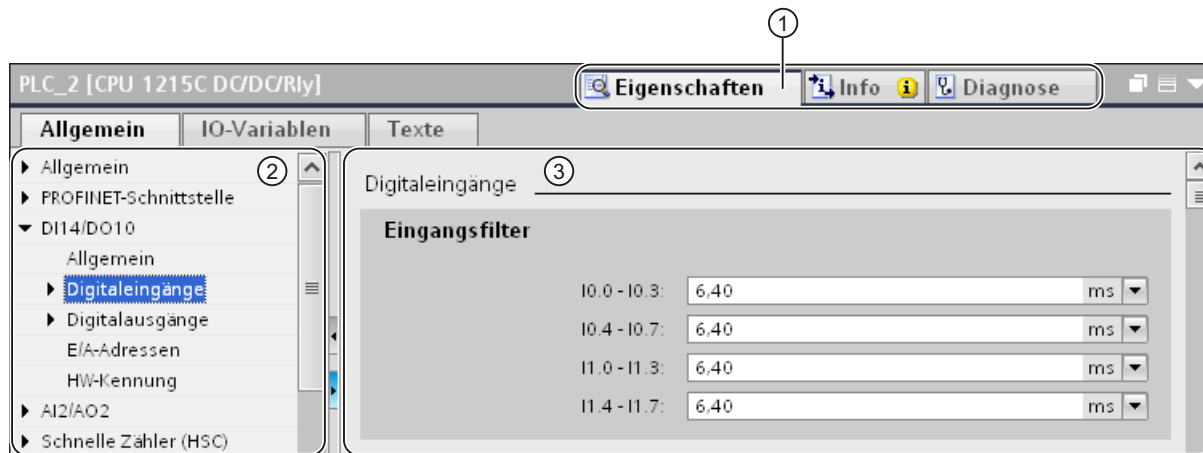
Drucken von Hardware- und Netzkonfigurationen (Seite 377)

### 8.1.1.8 Inspektorfenster

Die angezeigten Eigenschaften und Parameter des ausgewählten Objekts können innerhalb des Inspektorfensters bearbeitet werden.

## Aufbau

Das Inspektorfenster besteht aus folgenden Komponenten:



- ① Umschaltung zwischen verschiedenen Informations- und Arbeitsbereichen
- ② Navigation zwischen verschiedenen Informationen und Parametern
- ③ Anzeige der ausgewählten Informationen und Parameter

## Funktion

Die Informationen und Parameter im Inspektorfenster sind in verschiedene Informationstypen aufgeteilt:

- Eigenschaften
- Info
- Diagnose

Für die Anzeige der entsprechenden Informationen und Parameter klicken Sie auf den gewünschten Bereich. Für das Konfigurieren eines Automatisierungssystems ist der Bereich "Eigenschaften" der wichtigste Bereich. Dieser Bereich wird standardmäßig angezeigt und enthält verschiedene Register:

- Allgemein
- IO-Variablen
- Text

Im linken Teil des Inspektorfensters befindet sich die Bereichsnavigation. Informationen und Parameter sind dort in Gruppen angeordnet. Wenn Sie auf das Pfeilsymbol links neben dem Gruppennamen klicken, können Sie die Gruppe bei vorhandenen Untergruppen aufklappen. Wenn Sie eine Gruppe oder eine Untergruppe auswählen, werden die entsprechenden Informationen und Parameter im rechten Teil des Inspektorfensters angezeigt und können dort auch bearbeitet werden.

**Siehe auch**

Eigenschaften und Parameter bearbeiten (Seite 408)

Übersicht über den Hardware- und Netzwerkkeditor (Seite 367)

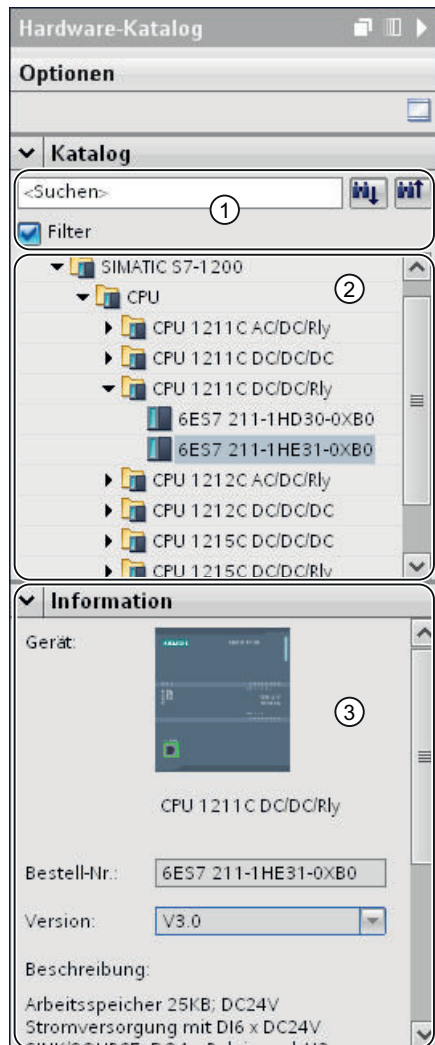
Einzelnen Objekten zugehörige Texte übersetzen (Seite 244)

**8.1.1.9 Hardware-Katalog**

Die Task Card "Hardware-Katalog" ermöglicht Ihnen einen einfachen Zugriff auf verschiedenste Hardware-Komponenten.

### Aufbau

Die Task Card "Hardware-Katalog" besteht aus den folgenden Paletten:



- ① Palette "Katalog", Such- und Filterfunktion
- ② Palette "Katalog", Komponentenauswahl
- ③ Palette "Information"

### Such- und Filterfunktion

Die Such- und Filterfunktionen der Palette "Katalog" erlauben Ihnen eine komfortable Suche nach bestimmten Hardware-Komponenten. Über die Filterfunktion können Sie die Anzeige der HW-Komponenten nach bestimmten Kriterien beschränken. Beispielsweise können Sie die Anzeige auf Objekte beschränken, die Sie auch im aktuellen Kontext platzieren können oder die bestimmte Funktionen beinhalten.

Verwendbare Objekte im aktuellen Kontext sind z. B. vernetzbare Objekte in der Netzsicht oder nur zum Gerät kompatible Baugruppen in der Gerätesicht.

## Komponentenauswahl

Die Komponentenauswahl der Palette "Katalog" enthält die installierten Hardware-Komponenten in einer Baumstruktur. Aus dem Katalog können Sie die gewünschten Geräte oder Baugruppen in den grafischen Arbeitsbereich der Geräte- oder Netzsicht ziehen.

Hardware-Komponenten, die zwar installiert sind, aber für die keine Lizenz vorhanden ist, werden grau angezeigt. Nicht-lizenzierte HW-Komponenten können Sie nicht verwenden.

Hardware-Komponenten, die thematisch zu verschiedenen Komponentengruppen gehören, sind zum Teil als verlinktes Objekt ausgeführt. Wenn Sie solch eine verlinkte Hardware-Komponente anklicken, öffnet sich die Katalogstruktur, in der Sie die entsprechende Hardware-Komponente finden.

## Information

In der Palette "Information" sehen Sie Detail-Informationen des im Katalog selektierten Objekts:

- Schematische Darstellung
- Name
- Bestellnummer
- Versionsnummer
- Beschreibung

## Siehe auch

Hardware-Katalog durchsuchen (Seite 391)

Übersicht über den Hardware- und Netzwerkeditor (Seite 367)

Informationen zu Hardware-Komponenten (Seite 383)

### 8.1.1.10 Informationen zu Hardware-Komponenten

Im Hardware-Katalog haben Sie die Möglichkeit, sich Informationen von ausgewählten Hardware-Komponenten in der Palette "Information" anzeigen zu lassen. Zusätzlich können Sie sich weitere Informationen zu den ausgewählten Hardware-Komponenten über das Kontextmenü anzeigen lassen.

## Zugriff auf weitere Informationen

Wenn Sie im Hardware-Katalog ein Hardware-Objekt selektieren und das Kontextmenü öffnen, stehen Ihnen neben der "Kopieren"-Funktion drei Optionen für den Zugriff auf Informationen zu Service & Support zu Verfügung:

- Informationen zum Produktsupport
- FAQs
- Handbücher

Die gewünschten Informationen werden Ihnen im Arbeitsbereich des Hardware- und Netzwerkeditors angezeigt.

---

**Hinweis**

Den Zugriff auf Service & Support können Sie nur verwenden, wenn Sie mit dem Internet verbunden sind und die Funktion aktiviert ist. Standardmäßig ist die Funktion deaktiviert. Um die Funktion zu aktivieren, beachten Sie das Kapitel "Produktsupport aktivieren (Seite 384)".

---

### Informationen zum Produktsupport

Hier erhalten Sie Zugang zu allgemeinen Informationen zu Hard- und Software-Komponenten. Die Bestellnummer des selektierten Hardware-Objektes ist in der Suchmaske voreingestellt. Sie können aber auch nach anderen Hard- und Software-Komponenten suchen.

### FAQs

Hier erhalten Sie Zugang zu den "Frequently Asked Questions" (FAQ). Sie können sich verschiedene Beiträge zu Hard- und Software-Fragen ansehen. Über eine detaillierte Suchmaske können Sie das gewünschte Themengebiet herausfiltern.

### Handbücher

Hier erhalten Sie Zugang zu den Handbüchern der verschiedenen Hardware-Komponenten. Dies ist vor allem hilfreich, wenn die vorzunehmende Konfiguration, Adressierung oder Parametrierung genauere Kenntnisse über die verwendete Hardware voraussetzt.

### Siehe auch

Hardware-Katalog (Seite 381)

Produktsupport aktivieren (Seite 384)

## 8.1.1.11 Produktsupport aktivieren

### Service & Support-Funktion aktivieren

Sie können sich zu jedem Gerät des Hardware-Katalogs weitere Informationen anzeigen lassen, die im Service & Support-Bereich der Siemens-Internetseite hinterlegt sind. Standardmäßig ist die Funktion deaktiviert. Im Folgenden erfahren Sie, wie Sie die Funktion aktivieren.

### Voraussetzung

Die Software muss Zugriff auf das Internet haben.

## Vorgehen

Um die Service & Support-Funktion zu aktivieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".
2. Öffnen Sie in der Bereichsnavigation die Gruppe "Hardware-Konfiguration".
3. Aktivieren Sie das Optionskästchen "Über Internet".

## Ergebnis

Innerhalb des Hardware-Katalogs erhalten Sie nun im Kontextmenü einer Baugruppe Zugriff auf den Produktsupport, die FAQs und die Handbücher.

## Siehe auch

Informationen zu Hardware-Komponenten (Seite 383)

### 8.1.1.12 Tastaturbedienung: Navigation im Editor

Sie können zwischen den Komponenten des Hardware- und Netzwerkeditor und dessen Objekten in der Netz- und Gerätesicht über Tastenkombinationen navigieren.

## Zwischen Elementen und Funktionen navigieren

| Funktion   | Tastenkombination  |
|--|--|
| Wechsel in die nächsttiefere Auswahlebene<br>Mit <Return> können Sie zum Beispiel von einem selektierten Baugruppenträger in die tiefere Auswahleben der dort aufgeschnappten Geräte und Module wechseln. Ist ein Gerät selektiert, können Sie mit <Return> in die tiefere Auswahlebene der auf dem Gerät angezeigten Schnittstellen wechseln. | <Return>   |
| Wechsel in die nächsthöhere Auswahlebene<br>Mit <Esc> können Sie zum Beispiel von einer selektierten Schnittstelle in die höhere Auswahleben der Geräte und Module wechseln. Ist ein Gerät selektiert, können Sie mit <Esc> in die höhere Auswahlebene der Baugruppenträger wechseln.  | <Esc>  |
| Navigation zwischen Objekten in der aktuellen Auswahlebene<br>Mit den Pfeiltasten können Sie in einer aktuellen Auswahlebene zwischen den Objekten navigieren. Zum Wechseln der Auswahlebene verwenden Sie die Tasten <Return> oder <Esc>.   | <Pfeil-nach-oben><br><Pfeil-nach-unten><br><Pfeil-nach-rechts><br><Pfeil-nach-links> |
| Schaltet auf die Gerätesicht um  | <Strg+Shift+D>   |
| Schaltet auf die Netzsicht um  | <Strg+Shift+N>   |
| Schaltet auf die Topologiesicht um   | <Strg+Shift+T>   |

8.1 Geräte und Netze konfigurieren

| Funktion   | Tastenkombination              |
|--|--------------------------------|
| Umschalten zwischen Editorelementen<br>Mit der Taste <Tab> schalten Sie von einem Editorelement zu dem nächsten Element. Mit <Shift+Tab> schalten Sie zum vorigen Element um. Sie können so zum Beispiel zwischen grafischer Ansicht, Speedy Splitter, tabellarischer Ansicht oder unterlagerten Registerkarten umschalten.          | <Tab><br><Shift+Tab>           |
| Umschalten zwischen den Registerkarten<br>Mit den Tasten <Strg+Tab> schalten Sie von einer Registerkarte zu der nächsten rechten Registerkarte um. Mit <Strg+Shift+Tab> schalten Sie zur nächsten linken Registerkarte um. Sie können mit mit den Tasten zum Beispiel zwischen Gerätesicht, Netzsicht und Topologiesicht umschalten. | <Strg+Tab><br><Strg+Shift+Tab> |

Elemente und Funktionen öffnen

| Funktion   | Tastenkombination |
|--|-------------------|
| Online- und Diagnosesicht öffnen<br>Bei selektiertem Gerät öffnet sich für das selektierte Gerät mit <Strg+D> die Online- und Diagnosesicht.       | <Strg+D>          |
| Dialog zum Laden in Gerät öffnen<br>Bei selektiertem Gerät öffnet sich für das selektierte Gerät mit <Strg+L> der Dialog für das erweiterte Laden. | <Strg+L>          |
| Neues Gerät hinzufügen<br>Mit <Strg+N> öffnet sich der Dialog für das Hinzufügen eines neuen Gerätes.  | <Strg+N>          |
| Öffnet die Task Card "Hardware-Katalog"  | <Strg+Shift+C>    |
| Öffnet die Task Card "Online-Tools"  | <Strg+Shift+O>    |

Siehe auch

Tastaturbedienung im TIA-Portal (Seite 207)

8.1.1.13 Tastaturbedienung: Bearbeiten von Objekten

Sie können im Hardware- und Netzwerkeditor einige Funktionen der Netz- und Gerätesicht direkt über Kombinationen von Tastatur und Maus durchführen. Die Tastaturbedienung in Tabellen (Seite 207) entspricht dem üblichen Verhalten. An dieser Stelle finden Sie die Tastaturbedienung für den grafischen Arbeitsbereich der Netz- und Gerätesicht.

Allgemeine Tastaturbedienung

| Funktion  | Tastenkombination                      |
|---|--|
| Ansicht auf gezogenen Rahmen vergrößern<br>Ziehen Sie einen Rahmen in der grafischen Sicht, um die Größe der Ansicht entsprechend zu verändern. | <Strg+Leertaste> + gedrückte Maustaste |
| Ansicht verschieben<br>Bewegen Sie den Mauszeiger, um die Ansicht zu verschieben.   | <Leertaste> + gedrückte Maustaste      |



| <b>Funktion</b>  | <b>Tastenkombination</b> |
|--|--------------------------|
| Aktuellen Vorgang abbrechen  | <Esc>                    |
| Verbinder lösen<br>Mit <Esc> oder Doppelklick verlassen Sie den Verbindungsmodus beim Ziehen einer Verbindung. | <Esc> oder Doppelklick   |
| Zoom in grafischer Sicht<br>Die Vergrößerung oder Verkleinerung ist von der Drehrichtung abhängig.             | <Strg> + Mausrad drehen  |

## Ausgewählte Objekte

| <b>Funktion</b>   | <b>Tastenkombination</b>     |
|---|------------------------------|
| Objekt selektieren  | Mausklick                    |
| Objekt ausschneiden<br>Das selektierte Objekt wird in die Zwischenablage kopiert und aus der grafischen Sicht gelöscht.   | <Strg+X>                     |
| Objekt kopieren<br>Das selektierte Objekt wird in die Zwischenablage kopiert.   | <Strg+C>                     |
| Objekt einfügen<br>Das Objekt aus der Zwischenablage wird in die Auswahl eingefügt.   | <Strg+V>                     |
| Selektiertes Objekt löschen   | <Entf>                       |
| Mehrere Objekte selektieren 1<br>Sie können mehrere Objekte einzeln durch Mausclick den selektierten Objekten hinzufügen. Alternativ können Sie mit <Shift > + gedrückter Maustaste einen Rahmen um die zu selektierenden Objekte ziehen.   | <Shift > + Mausclick         |
| Mehrere Objekte selektieren 2<br>Sie können mehrere Objekte einzeln durch Mausclick den selektierten Objekten hinzufügen. Alternativ können Sie mit <Shift > + gedrückter Maustaste einen Rahmen um die zu selektierenden Objekte ziehen. Bei gehaltener Taste <Strg> können Sie selektierte Objekte mit Mausclick deselektieren. | <Strg> + Mausclick           |
| Auswahl verschieben<br>Sie können mit gedrückter Maustaste Geräte oder Module auf erlaubte Steckplätze eines Baugruppenträgers ziehen.  | gedrückte Maustaste          |
| Auswahl kopieren<br>Sie können mit <Strg> und gedrückter Maustaste Geräte oder Module auf erlaubte Steckplätze eines Baugruppenträgers ziehen. Die Geräte oder Module werden dabei kopiert.   | <Strg> + gedrückte Maustaste |

## 8.1.2 Geräte konfigurieren

### 8.1.2.1 Grundlagen

#### Einführung zum Konfigurieren der Hardware

Um ein Automatisierungssystem aufzubauen, müssen Sie die einzelnen Komponenten der Hardware konfigurieren, parametrieren und miteinander verbinden. Die dafür notwendigen Arbeiten verrichten Sie in der Geräte- und Netzsicht.

#### Konfigurieren

Unter "Konfigurieren" versteht man das Anordnen, Einstellen und Vernetzen von Geräten und Baugruppen innerhalb der Geräte- oder Netzsicht. Baugruppenträger werden symbolisch repräsentiert. Wie "reale" Baugruppenträger lassen sie das Stecken einer festgelegten Anzahl von Baugruppen zu.

Jeder Baugruppe wird automatisch eine Adresse zugewiesen. Die Adressen können nachträglich geändert werden.

Beim Anlauf des Automatisierungssystems vergleicht die CPU die softwareseitig erstellte Sollkonfiguration mit der tatsächlichen Istkonfiguration der Anlage. Eventuelle Fehler können somit sofort erkannt und gemeldet werden.

#### Parametrieren

Unter "Parametrieren" versteht man das Einstellen der Eigenschaften der verwendeten Komponenten. Dabei werden Hardware-Komponenten und Einstellungen für den Datenaustausch parametrieren:

- Eigenschaften parametrierbarer Baugruppen
- Einstellungen für den Datenaustausch zwischen Komponenten

Die Parameter werden in die CPU geladen und im Anlauf von der CPU an die entsprechenden Baugruppen übertragen. Baugruppen lassen sich sehr einfach ersetzen, da die erstellten Parameter beim Anlauf automatisch in die neue Baugruppe geladen werden.

#### Anpassen der Hardware an Projekterfordernisse

Sie müssen Hardware konfigurieren, wenn Sie ein Automatisierungsprojekt aufbauen, erweitern oder verändern möchten. Dazu fügen Sie Ihrem Aufbau Hardware-Komponenten hinzu, verbinden diese mit bestehenden Komponenten und passen die Eigenschaften der Hardware den Aufgaben an.

Die Eigenschaften der Automatisierungssysteme und der Baugruppen sind so voreingestellt, dass Sie in vielen Fällen nichts mehr parametrieren müssen. In den folgenden Fällen ist das Parametrieren dennoch erforderlich:

- Sie möchten voreingestellte Parameter einer Baugruppe verändern.
- Sie möchten spezielle Funktionen verwenden.
- Sie möchten Kommunikationsverbindungen projektieren.

## Siehe auch

Eigenschaften der Baugruppen ändern (Seite 854)

## Übernehmen bestehender Konfigurationen

### Bestehende Projekte öffnen

Beim Öffnen von bereits bestehenden Projekten wird automatisch geprüft, ob für alle im Projekt verwendeten Baugruppen die entsprechende Software installiert ist. Versuchen Sie ein Projekt mit Baugruppen zu öffnen, die vom aktuellen Installationsumfang des TIA-Portals nicht unterstützt werden, so erscheint bereits beim Öffnen des Projekts ein Hinweis auf die fehlenden Software-Komponenten. Sind die Software-Komponenten nicht zwingend für das Öffnen des Projekts erforderlich, lässt sich das Projekt dennoch öffnen.

### Verhalten bei fehlenden Software-Komponenten

Projekte, die vom aktuellen Installationsumfang nicht unterstützte Baugruppen enthalten, verhalten sich wie folgt:

- Anzeige der Baugruppen an der Benutzeroberfläche
  - Die nicht unterstützten Baugruppen werden in der Projektnavigation mit all ihren unterlagerten Objekten angezeigt, die Baugruppen selbst können jedoch nicht in Editoren oder im Inspektorfenster bearbeitet werden. Falls möglich, wird eine Ersatzbaugruppe verwendet, die am besten zur ursprünglichen Baugruppe passt. Ersatzbaugruppen sind mit einem Ausrufezeichen-Symbol gekennzeichnet.
  - Die Anzeige von Eigenschaften in Tabellen ist eingeschränkt. Dies betrifft insbesondere die Darstellung von Netzparametern, wie beispielsweise der IP-Adresse.
- Funktionale Einschränkungen
  - Nicht unterstützte Baugruppen können nicht gedruckt und nicht übersetzt werden.
  - Es ist nicht möglich eine Online-Verbindung zu den Baugruppen aufzubauen. Somit ist ein Download ebenfalls nicht möglich.
  - Um den Gerätetyp zu ändern, müssen Sie das Gerät zunächst löschen und neu einfügen. Die Funktion "Gerätetyp ändern" wird nicht unterstützt.
  - Das Kopieren und Einfügen von unterlagerten Objekten, wie z. B. Bausteinen, ist möglich, das Kopieren und Einfügen des Geräts selbst ist nicht möglich.
  - Das Ändern der Netzwerkkonfiguration innerhalb der Netzsicht ist mit Ersatzbaugruppen nicht möglich.
  - Querverweise können angezeigt werden. Die Querverweise geben jedoch nur den zuletzt im Projekt gespeicherten Zustand wieder, da ein Online-Abgleich mit der Original-Baugruppe nicht durchgeführt werden kann.

## Siehe auch

Projekte öffnen (Seite 233)

## Allgemeine Steckplatzregeln

### Einführung

Für jedes Automatisierungssystem und jede Baugruppe gelten spezifische Steckplatzregeln.

Wenn Sie in der Gerätesicht eine Baugruppe im Hardware-Katalog selektieren, werden im Baugruppenträger alle möglichen Steckplätze für die selektierte Baugruppe markiert. Sie können Baugruppen nur auf markierte Steckplätze ziehen.

Wenn Sie eine Baugruppe einfügen, verschieben oder tauschen, werden ebenfalls die Steckplatzregeln angewendet.

### Konsistenz

Einige Steckplatzregeln sind von der Konfiguration der Umgebung abhängig. Dies bedeutet, dass Sie manchmal Baugruppen in den Baugruppenträger stecken können, obwohl dies aktuell zu Inkonsistenzen führen würde. Wenn Sie die Konfiguration ändern, z. B. hinsichtlich der Auswahl anderer Baugruppen oder der Parametereinstellungen an den Baugruppen, können Sie die Konsistenz der Konfiguration wieder herstellen.

In Fällen, in denen das Stecken einer Baugruppe zu einer korrigierbaren Inkonsistenz führt, wird das Stecken zugelassen. Beim Übersetzen der Konfiguration wird ein Konsistenzcheck durchgeführt. Inkonsistenzen werden als Meldung im Inspektorfenster unter "Info" angezeigt. Auf Grund des Ergebnisses des Konsistenzchecks können Sie Ihre Konfiguration überarbeiten und die Konsistenz wieder herstellen.

## Regeln für die Anordnung von Baugruppen

Ganz allgemein gelten für Baugruppen in Baugruppenträgern folgende Regeln:

- In einen Baugruppenträger können Sie nur Baugruppen stecken.
- In eine Baugruppe können Sie nur Schnittstellenmodule stecken.
- Sie können nur Baugruppen der gleichen Produkt- oder Systemfamilie in einem Baugruppenträger verwenden.

Darüber hinaus gibt es für einige Baugruppen weitere spezielle Regeln:

- Steckbarkeit nur auf bestimmte Steckplätze
- Steckbarkeit in Abhängigkeit von anderen Baugruppen, CPUs oder Einstellungen
- Beschränkung der Anzahl der Verwendungen in einem Baugruppenträger

## Hardware-Katalog durchsuchen

### Einführung

Um für eine Hardware-Konfiguration die gewünschten Hardware-Komponenten auszuwählen, verwenden Sie die Task Card "Hardware-Katalog". Verwenden Sie den Hardware-Katalog für die Auswahl der vernetzbaren Hardware-Komponenten in der Netz- und Topologiesicht und für die Auswahl der gewünschten Baugruppen in der Gerätesicht.

### Kontextfilter

Über die Option "Filter" des Hardware-Katalogs können Sie die Anzahl der angezeigten und durch die Suche auffindbaren HW-Komponenten einschränken.

Wenn Sie den Filter aktivieren, werden nur die Komponenten im Hardware-Katalog angezeigt, die aktuell ausgewählt werden können. Wenn Sie den Filter nicht aktivieren, wird der gesamte Hardware-Katalog angezeigt.



Wenn Sie zwischen den verschiedenen Sichten wechseln, wird die Ansicht der gefilterten Objekte an den aktuellen Kontext angepasst.

### Suchmöglichkeiten

Über die Suchfunktion suchen Sie gezielt nach Einträgen im Hardware-Katalog. Beachten Sie bei der Eingabe von Suchbegriffen folgende Regeln:

- Zwischen Groß- und Kleinschreibung wird nicht unterschieden.
- Bindestriche und Leerzeichen werden bei der Suche ignoriert.
- Teile eines gesuchten Begriffs werden bei der Suche berücksichtigt.
- Mehrere Suchbegriffe müssen mit einem Leerzeichen getrennt werden

Sie starten die Suche von einem aktuell markierten Objekt im Hardware-Katalog mit der Suchrichtung nach oben oder nach unten.

| Symbol  | Bedeutung      |
|---|----------------|
|  | Suche abwärts  |
|  | Suche aufwärts |

## Hardware-Katalog durchsuchen

Wenn Sie den Hardware-Katalog durchsuchen möchten, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der Maus in das Eingabefeld der Suchfunktion
2. Geben Sie einen Suchbegriff ein. Folgende Elemente werden durchsucht:
  - Name des Geräts oder der Baugruppe
  - Bestellnummer (MLFB)
  - Beschreibung in Palette "Information"
3. Klicken Sie auf eine der Schaltflächen "Suche abwärts" bzw. "Suche aufwärts".

---

### Hinweis

Für die richtige Suchrichtung beachten Sie, welche Stelle Sie im Hardware-Katalog markiert haben. Um alles zu durchsuchen, klicken Sie auf das oberste Objekt des Hardware-Katalogs und starten Sie die Suche nach Eingabe des Suchbegriffs mit "Suche abwärts".

Die erste gefundene Übereinstimmung mit dem Suchbegriff wird als Ergebnis angezeigt. Für weitere Suchergebnisse klicken Sie weitere Male auf die Schaltfläche "Suche abwärts" bzw. "Suche aufwärts".

Beachten Sie den Kontextfilter des HW-Kataloges. Ist dieser eingeschaltet, beschränkt sich die Suche im HW-Katalog nur auf die angezeigten steckbaren HW-Komponenten.

---

## Siehe auch

Hardware-Katalog (Seite 381)

Informationen zu Hardware-Komponenten (Seite 383)

## Arbeiten mit Baugruppenträgern

### Einführung

Um Baugruppen einem Gerät zuzuweisen, benötigen Sie einen Baugruppenträger, z. B. eine Profilschiene. Auf dem Baugruppenträger befestigen Sie die Baugruppen und verbinden diese damit über den Rückwandbus mit der CPU, einer Stromversorgung oder anderen Baugruppen.


### Anlegen eines Baugruppenträgers

Wenn Sie ein Gerät in die Netzsicht einfügen, wird automatisch eine Station und ein zum ausgewählten Gerät passender Baugruppenträger angelegt. In der Gerätesicht wird der Baugruppenträger mit den verfügbaren Steckplätzen angezeigt. Die Anzahl der verfügbaren Steckplätze richtet sich auch wieder nach dem Typ des verwendeten Geräts.

## Aufbau eines Baugruppenträgers

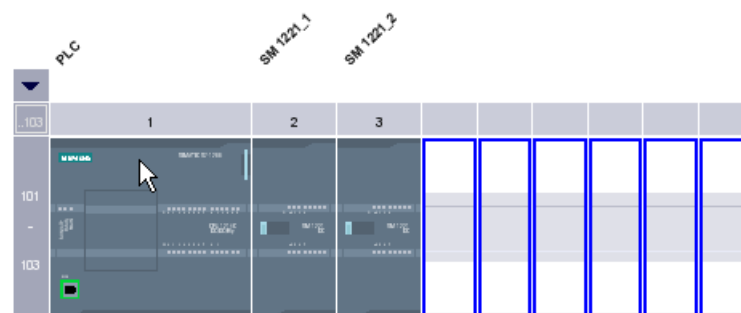
Ein Baugruppenträger beinhaltet immer das Gerät, das in der Netzsicht eingefügt wurde. Das Gerät ist fest einem Steckplatz zugeordnet, wobei der Steckplatz von der Art des Geräts abhängt. Rechts und ggf. links von dem Gerät befinden sich weitere Steckplätze, oberhalb der Steckplätze befinden sich bei gesteckten Baugruppen die Steckplatznummern.

Über den gesteckten Geräten und Baugruppen wird eine entsprechende Kurzbezeichnung angezeigt. Diese Kurzbezeichnung können Sie über die Menüleiste unter "Ansicht" mit dem Befehl "Baugruppenbeschriftungen anzeigen" oder mit dem entsprechenden Symbol in der Funktionsleiste der Gerätesicht (Seite 371) ein- und ausschalten.

| Symbol  | Bedeutung                         |
|---|-----------------------------------|
|  | Baugruppenbeschriftungen anzeigen |

Wenn Baugruppen im Hardware-Katalog selektiert werden, werden alle für diese Baugruppe erlaubten Steckplätze markiert. Sie können dann sofort erkennen, an welchen Steckplatz die selektierte Baugruppe gesteckt werden kann.

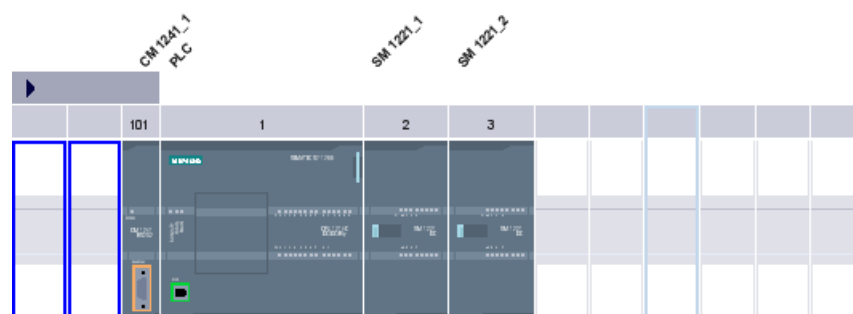
In dem folgenden Bild wurde bei einem teilweise bestückten Baugruppenträger einer S7-1200 ein Signalmodul im Hardware-Katalog selektiert:



Da die Steckplätze 101-103 für Kommunikationsmodule reserviert sind, werden nur die anderen freien Steckplätze als mögliche Steckplätze angezeigt.

Die vordere Steckplatzgruppe können Sie über ein Pfeilsymbol oberhalb des erweiterbaren Steckplatzes auf- und zuklappen. Bei zugeklappter Steckplatzgruppe wird die erste und letzte Steckplatznummer der Gruppe angezeigt.

Das folgende Bild zeigt die aufgeklappte Steckplatzgruppe:



Steckplatzgruppen mit bereits gesteckten Baugruppen können nicht zugeklappt werden.

## Mehrfachselektion von Baugruppen und Steckplätzen

Für die Selektion mehrerer Baugruppen oder Steckplätze stehen Ihnen mehrere Möglichkeiten zur Verfügung:

- Selektieren mehrerer Baugruppen oder Steckplätze bei gleichzeitigem Drücken von <Shift> oder <Strg>.
- Mausklick außerhalb des Baugruppenträgers und bei gedrückter Maustaste einen Rahmen um die zu selektierenden Baugruppen oder Steckplätze ziehen.

## Objekte in der Gerätesicht

Im oberen Teil der Gerätesicht sehen Sie eine grafische Darstellung des Baugruppenträgers und der darauf gesteckten Geräte.

Im unteren Teil der Gerätesicht sehen Sie die Geräteübersicht. Die Geräteübersicht ist eine Tabelle, die Ihnen die wichtigsten Informationen der im Baugruppenträger eingefügten Baugruppen zeigt.

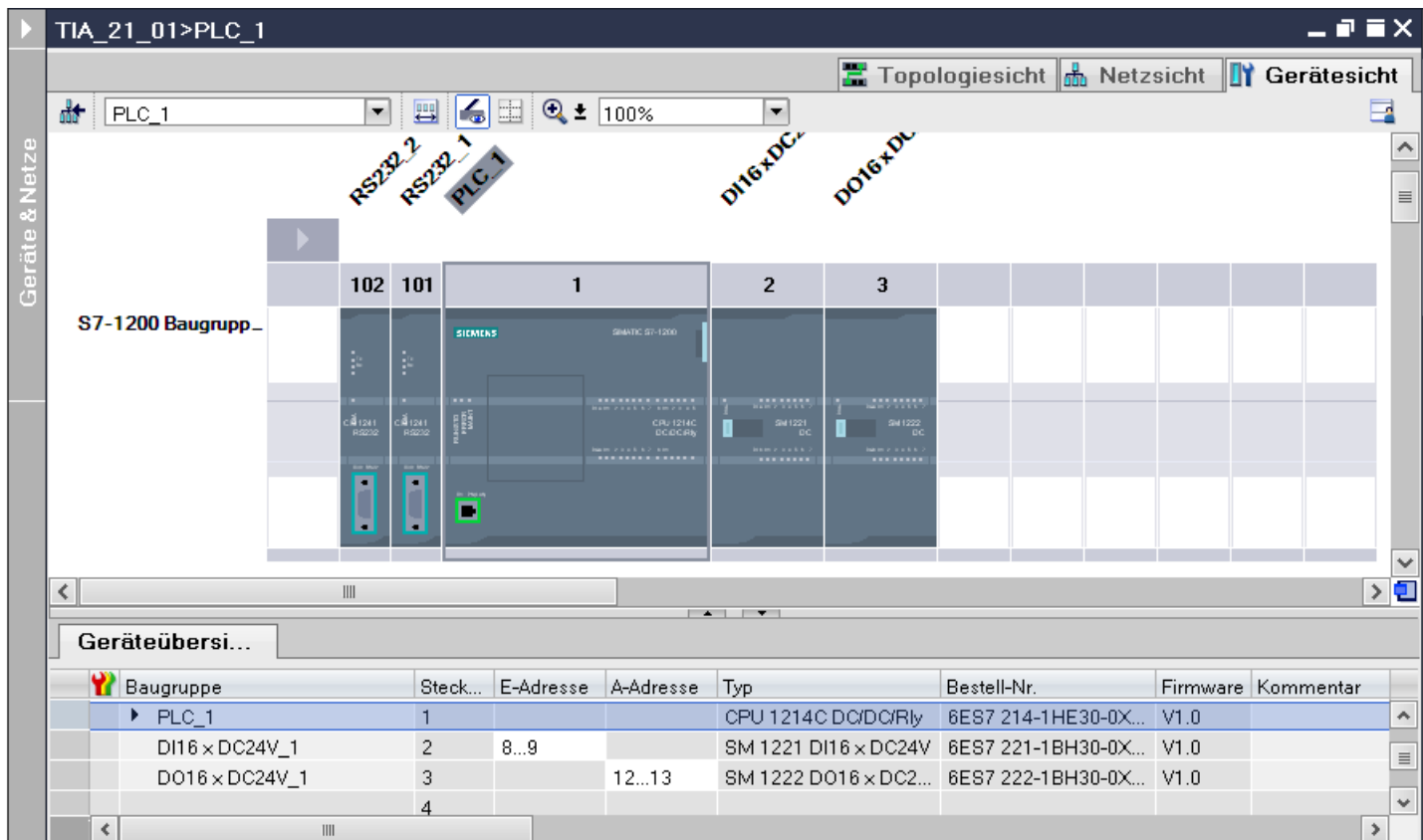
## Aufbau und Inhalt der Gerätesicht

In der grafischen Gerätesicht wird Ihnen die Offline-Konfiguration der Geräte auf dem Baugruppenträger angezeigt. Die Darstellung entspricht symbolisch dem Aufbau auf dem realen Baugruppenträger.

In der Geräteübersicht wird die Belegung des Baugruppenträgers in tabellarischer Form gezeigt. Eine Tabellenzeile beinhaltet jeweils die Informationen für die Belegung eines Steckplatzes.

Das folgende Bild zeigt die Gerätesicht mit der Konfiguration einer SIMATIC S7-1200 CPU.





Im oberen Teil sehen Sie die grafische Sicht mit der Belegung des Baugruppenträgers durch verschiedene Baugruppen auf den Steckplätzen 1 bis 3, sowie 101. Im unteren Teil sehen Sie in der Geräteübersicht die tabellarische Darstellung des Baugruppenträgers.

Jede Zeile in der Geräteübersicht präsentiert einen Steckplatz. Für jeden Steckplatz werden in den verschiedenen Spalten die wichtigsten Informationen angezeigt:

| Spalte      | Bedeutung                                  |
|-------------|--|
| Baugruppe   | Bezeichnung der Baugruppe, frei editierbar |
| Steckplatz  | Nummer des Steckplatzes                    |
| E-Adresse   | Eingangsadressbereich, editierbar          |
| A-Adresse   | Ausgangsadressbereich, editierbar          |
| Typ         | Katalogbezeichnung der Baugruppe           |
| Bestell-Nr. | Bestellnummer der Baugruppe                |
| Firmware    | Firmwareversion der Baugruppe              |
| Kommentar   | Optionaler Kommentar                       |

## Siehe auch

Gerätesicht (Seite 371)

### Ablage nicht gesteckter Baugruppen

In manchen Fällen werden Baugruppen einer Hardware-Konfiguration temporär keinem Steckplatz zugewiesen. Solche nicht gesteckten Baugruppen werden in die Ablage nicht gesteckter Baugruppen geschoben, einem speziellen Bereich in der Gerätesicht.

### Aufnahme von Baugruppen in die Ablage

In die Ablage nicht gesteckter Baugruppen werden automatisch die Baugruppen geschoben, die z. B. über eine Kopieraktion einem Gerät zugeordnet werden sollten, der entsprechende Baugruppenträger jedoch keinen freien kompatiblen Steckplatz aufweist.

Unter folgenden Bedingungen werden Baugruppen automatisch in die Ablage nicht gesteckter Baugruppen aufgenommen:

- Eine Baugruppe wird in der Netzsicht auf ein Gerät gezogen, auf dem Baugruppenträger ist aber kein kompatibler Steckplatz frei.
- Eine Baugruppe wird in der Gerätesicht aus dem Baugruppenträger, dem Hardware-Katalog oder der Projektnavigation direkt in die Ablage geschoben oder kopiert.

CPs und FMs, die eine Netzressource belegen, können in die Ablage nicht gesteckter Baugruppen geschoben werden, verlieren aber ihre bereits zugewiesenen Netzressourcen.

Sie können Baugruppen auch selber beispielsweise über Drag & Drop in die Ablage nicht gesteckter Baugruppen einfügen. Dazu muss die Ablage geöffnet sein.

### Verwendung der Ablage nicht gesteckter Baugruppen

Sie öffnen die Ablage nicht gesteckter Baugruppen mit der entsprechenden Schaltfläche.

Die Ablage nicht gesteckter Baugruppen finden Sie in der Gerätesicht.



Sie öffnen die Ablage nicht gesteckter Baugruppen über das entsprechende Symbol in der Funktionsleiste der Gerätesicht (Seite 371).

| Symbol | Bedeutung                                 |
|--------|---|
|        | Ablage nicht gesteckter Baugruppen öffnen |

---

### **Hinweis**

Um Steckplätze frei zu machen, schieben Sie Baugruppen aus Ihrer Konfiguration in die Ablage und stecken die gewünschten Baugruppen aus der Ablage in die nun freien Steckplätze.

Bereits parametrisierte Baugruppen nehmen Sie auf diese Weise temporär aus der Konfiguration, ohne diese zu löschen.

---

## **Behandlung von Baugruppen in der Ablage**

Für Baugruppen in der Ablage gelten folgende Regeln:

- Die Baugruppen finden Sie in der Projektnavigation unter dem entsprechenden Gerät im Ordner "Lokale Baugruppen".
- Die Baugruppen behalten alle bisher vorgenommenen Einstellungen und Parameter.
- Die Baugruppen werden beim Laden in das Zielsystem nicht berücksichtigt, es findet also kein Konsistenzcheck für Baugruppen in der Ablage nicht gesteckter Baugruppen statt.
- Die Baugruppen können über das Kontextmenü z. B. kopiert, ausgeschnitten oder gelöscht werden.

### **8.1.2.2 Einzelne Geräte konfigurieren**

#### **CPU auswählen**

##### **Einführung**

In der Netzsicht wählen Sie aus dem Hardware-Katalog eine CPU aus und legen diese zusammen mit einem Baugruppenträger an. Auf dieses Gerät ziehen Sie aus dem Hardware-Katalog die gewünschten Baugruppen, die automatisch auf dem Baugruppenträger angeordnet werden.

##### **Auswahl der Komponente im Hardware-Katalog**

Im Hardware-Katalog wird jede Hardware-Komponente als Ordner angezeigt. Wenn Sie diesen Ordner öffnen, sehen Sie die unterschiedlichen Versionen der ausgewählten Hardware-Komponente mit ihren jeweiligen Bestellnummern.

Es wird Ihnen hier beispielhaft gezeigt, wie Sie in der Netzsicht eine CPU mit einem Baugruppenträger anlegen.

##### **Voraussetzung**

- Der Hardware-Katalog ist geöffnet.
- Sie befinden sich in der Netzsicht.

## Vorgehen

Um eine CPU aus dem Hardware-Katalog auszuwählen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog auf den Ordner mit den gewünschten CPUs.
2. Öffnen Sie den Ordner mit dem gesuchten CPU-Typ. Sie sehen alle Bestellnummern des angewählten CPU-Typs.
3. Klicken Sie auf eine CPU-Bestellnummer, um sich in der Palette "Information" über die gewählte CPU zu informieren.



4. Legen Sie die CPU und einen Baugruppenträger an. Sie haben dazu folgende Möglichkeiten:
  - Ziehen Sie mit Drag & Drop die CPU aus dem Hardware-Katalog in die Netzsicht.
  - Kopieren Sie die CPU mit Copy & Paste in die Netzsicht.
  - Doppelklicken Sie auf den CPU-Eintrag im Hardware-Katalog.

## Siehe auch

- Hardware-Katalog durchsuchen (Seite 391)
- Gerät zur Hardware-Konfiguration hinzufügen (Seite 399)
- Baugruppe in Baugruppenträger stecken (Seite 402)
- Arbeiten mit Baugruppenträgern (Seite 392)
- Nicht spezifizierte CPU anlegen (Seite 400)
- Informationen zu Hardware-Komponenten (Seite 383)

## Gerät zur Hardware-Konfiguration hinzufügen

### Einführung

Um ein vernetzbares Gerät der Hardware-Konfiguration hinzuzufügen, gibt es in der Netzsicht und der Topologiesicht folgende Möglichkeiten:

- Befehl "Neues Gerät hinzufügen" in der Projektnavigation
- Doppelklick auf Gerät im Hardware-Katalog
- Drag & Drop aus dem Hardware-Katalog in die Netzsicht bzw. in die Topologiesicht:
  - Texteintrag aus der Palette "Katalog"
  - Vorschaugrafik aus der Palette "Information"
- Befehl "Einfügen > Gerät" aus der Menüleiste der Netzsicht bzw. der Topologiesicht
- Kontextmenü eines Gerätes im Hardware-Katalog für "Kopieren" und "Einfügen"

Zusammen mit dem neuen Gerät wird ein passender Baugruppenträger angelegt. Das ausgewählte Gerät wird auf den ersten zulässigen Steckplatz des Baugruppenträgers gesteckt.

Unabhängig von dem gewählten Weg ist das hinzugefügte Gerät in der Projektnavigation und in der Netzsicht bzw. in der Topologiesicht des Hardware- und Netzwerkeditors sichtbar.

### Gerät über die Projektnavigation hinzufügen

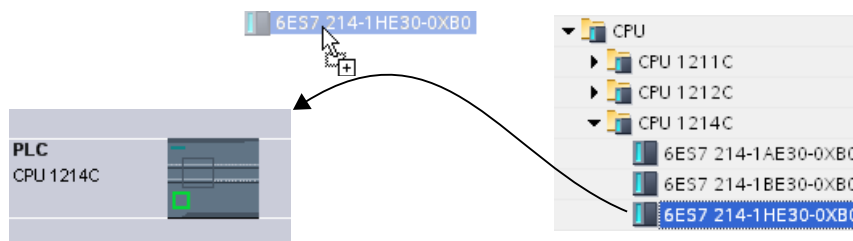
Um über die Projektnavigation ein Gerät der Hardware-Konfiguration hinzuzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Befehl "Neues Gerät hinzufügen".  
Das Dialogfeld "Neues Gerät hinzufügen" wird geöffnet.
2. Lassen Sie sich das gewünschte Gerät in der Baumstruktur anzeigen:
  - Navigieren Sie in der Baumstruktur zu dem gewünschten Gerät.
  - Geben Sie im Eingabefeld einen Gerätenamen an.
3. Wählen Sie aus der Baumstruktur das gewünschte Gerät aus.  
Auf der rechten Seite des Dialogfelds werden Ihnen weitere Informationen zum aktuell selektierten Gerät angezeigt.
4. Stellen Sie ggf. den FW-Stand über die Klappliste des Dialogfeldes ein.
5. Markieren Sie das Optionskästchen "Gerätesicht öffnen", wenn Sie nach dem Hinzufügen des Gerätes sofort in die Gerätesicht wechseln wollen.  
Dort können Sie sofort mit der Gerätekonfiguration und Bestückung des Baugruppenträgers fortfahren.
6. Klicken Sie auf "OK", um das ausgewählte Gerät hinzuzufügen.  
Das Dialogfeld wird geschlossen.

### Gerät aus dem Hardware-Katalog hinzufügen

Um über den Hardware-Katalog ein Gerät der Hardware-Konfiguration hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Netzsicht bzw. die Topologiesicht.
2. Öffnen Sie den Hardware-Katalog.
3. Navigieren Sie im Hardware-Katalog zum gewünschten Gerät.
4. Selektieren Sie das ausgewählte Gerät mit einem Mausklick.
5. Stellen Sie ggf. den FW-Stand über die Klappliste des Hardware-Katalogs ein.
6. Ziehen Sie das Gerät mit Drag & Drop in die Netzsicht bzw. in die Topologiesicht.



Sie haben nun das Gerät in der Netzsicht bzw. in der Topologiesicht platziert. Das angezeigte Rechteck (auch "Station") symbolisiert das gesteckte Gerät mit seinem Baugruppenträger und evtl. unterlagerten Baugruppen. Mit einem Doppelklick auf das Gerät oder die Station öffnen Sie die Gerätesicht und sehen dort den neuen Baugruppenträger mit dem gesteckten Gerät. In den nächsten Schritten können Sie das Gerät in der Gerätesicht konfigurieren und den Baugruppenträger mit Baugruppen bestücken.

### Siehe auch

Netzsicht (Seite 369)

Nicht spezifizierte CPU anlegen (Seite 400)

Informationen zu Hardware-Komponenten (Seite 383)

Topologiesicht (Seite 374)

### Nicht spezifizierte CPU anlegen

#### Einführung

Wenn Sie noch keine CPU ausgewählt haben aber bereits mit der Programmierung beginnen oder ein bestehendes Programm verwenden möchten, haben Sie die Möglichkeit, eine nicht spezifizierte CPU zu verwenden. Sie können auch bei unspezifizierten CPUs einige Einstellungen vornehmen. Die Einstellmöglichkeiten beschränken sich auf Parameter, die allen CPUs derselben CPU-Familie gemeinsam sind.

### Anlegen einer nicht spezifizierten CPU in der Portalsicht

Um eine nicht spezifizierte CPU in der Portalsicht zu erzeugen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf eine der folgenden Möglichkeiten:
  - "Geräte & Netze > Neues Gerät hinzufügen"
  - "PLC-Programmierung" > Schaltfläche "Gerät"
2. Wählen Sie für eine Gerätefamilie eine nicht spezifizierte CPU aus der Baumstruktur des Dialogs "Neues Gerät hinzufügen".
3. Klicken Sie auf "Hinzufügen".

Eine nicht spezifizierte CPU wird angelegt und die Gerätesicht für diese CPU wird geöffnet.

### Weitere Möglichkeiten für das Anlegen von nicht spezifizierten CPUs

In der Projektansicht können Sie nicht spezifizierte CPUs wie spezifizierte CPUs anlegen:

- In der Projektnavigation über die Schaltfläche "Neues Gerät hinzufügen"
- In der Task-Card "Hardware-Katalog"

Sie können auf diesen Wegen auch mehrere nicht spezifizierte CPUs anlegen.

### Nicht spezifizierte CPUs spezifizieren

Es gibt zwei Möglichkeiten, wie Sie nicht spezifizierte CPUs spezifizieren können:

- Weisen Sie einer nicht spezifizierten CPU über den Baugruppentausch (Seite 408) eine vorhandene CPU per Drag & Drop aus dem Hardware-Katalog zu.
- Wählen Sie bei markierter nicht spezifizierter CPU den Menübefehl "Online > Hardware-Erkennung" und weisen Sie eine online ermittelte CPU zu. Vergeben Sie zu diesem Zweck eine IP-Adresse über die Schaltfläche "Adresse für PG/PC hinzufügen".

---

#### **Hinweis**

Wenn Sie nach der Durchführung der Hardware-Erkennung online gehen möchten, müssen Sie zuerst die erkannte Konfiguration in ihr Projekt laden, sonst kann es zu einem Fehler wegen inkonsistenter Konfigurationen kommen. Bei der CPU im Projekt und der tatsächlich vorhandenen CPU sind nach der Hardware-Erkennung zwar die Bestellnummern identisch, nicht aber deren Parameter. Die Parameter der CPU im Projekt haben die voreingestellten Werte, die Parameter der tatsächlich vorhandenen CPU die von Ihnen eingestellten Werte.

---

### Siehe auch

CPU auswählen (Seite 397)

Gerät zur Hardware-Konfiguration hinzufügen (Seite 399)

## Baugruppe in Baugruppenträger stecken

### Einführung

Nachdem Sie in der Netzsicht Geräte aus dem Hardware-Katalog Ihrer Konfiguration hinzugefügt haben, können Sie die Geräte mit Baugruppen bestücken. Um in der Gerätesicht eine Baugruppe in einen Baugruppenträger zu stecken, gibt es folgende Möglichkeiten:

- Bei freiem gültigen Steckplatz Doppelklick auf eine Baugruppe im Hardware-Katalog.
- Drag & Drop aus dem Hardware-Katalog auf einen freien gültigen Steckplatz in den grafischen oder tabellarischen Bereich:
  - Texteintrag aus der Palette "Katalog"
  - Vorschaugrafik aus der Palette "Information"
- "Kopieren" im Kontextmenü einer Baugruppe im Hardware-Katalog und Kontextmenü "Einfügen" auf einem freien gültigen Steckplatz im grafischen oder tabellarischen Bereich.

Um aus der Netzsicht in die Gerätesicht zu gelangen, doppelklicken Sie in der Netzsicht auf ein Gerät oder die Station oder verwenden Sie das Register Gerätesicht. In der Gerätesicht finden Sie eine Darstellung des ausgewählten Geräts innerhalb eines Baugruppenträgers. Die grafische Darstellung des Baugruppenträgers in der Software entspricht dabei dem realen Aufbau, d. h. es sind genauso viele Steckplätze sichtbar wie in dem realen Aufbau.

---

#### Hinweis

Sie können eine Baugruppe auch in der Netzsicht auf einen Baugruppenträger ziehen. Dazu muss die Filterfunktion des Hardware-Katalogs deaktiviert sein. Die Baugruppe wird automatisch auf einen freien und erlaubten Steckplatz gesteckt. Wenn kein Steckplatz vorhanden ist, wird die Baugruppe in die Ablage nicht gesteckter Baugruppen (Seite 396) geschoben.

---

### Bestückung eines Baugruppenträgers

Ordnen Sie Baugruppen auf einem Baugruppenträger gemäß entsprechender Steckplatzregeln an.

Nach dem Stecken einer Baugruppe in einen Baugruppenträger mit einer bereits gesteckten CPU werden die Adressbereiche automatisch überprüft, damit es nicht zu einer Doppelbelegung von Adressen kommt. So besitzt jede Baugruppe nach dem Stecken einen gültigen Adressbereich. DP-Slaves und IO-Devices müssen hierfür mit einer CPU über das entsprechende DP-Master- oder IO-System vernetzt sein.

### Voraussetzung

- Sie befinden sich in der Gerätesicht.
- Der Hardware-Katalog ist geöffnet.



## Baugruppe aus dem Hardware-Katalog hinzufügen

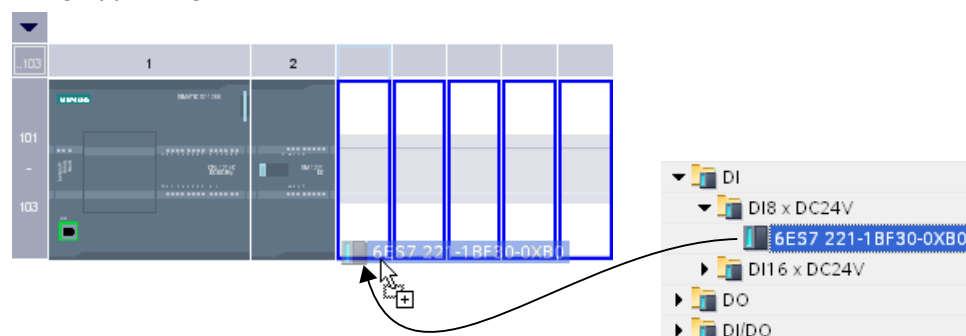
Das Stecken einer Baugruppe aus dem Hardware-Katalog in einen Baugruppenträger wird Ihnen am Beispiel eines digitalen Signalmoduls gezeigt. Gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog zu der gewünschten Baugruppe.

### Hinweis

Wenn Sie die Filterfunktion des Hardware-Katalogs aktivieren, werden Ihnen nur Baugruppen angezeigt, die zu dem ausgewählten Gerätetyp kompatibel sind.

2. Selektieren Sie die ausgewählte Baugruppe.
3. Stellen Sie ggf. den FW-Stand über die Klappliste des Hardware-Katalogs ein.
4. Ziehen Sie das Signalmodul mit Drag & Drop auf einen freien Steckplatz des Baugruppenträgers.

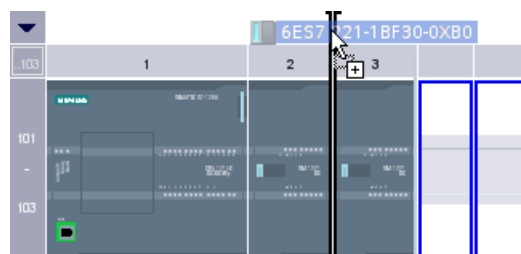


Sie haben nun das digitale Signalmodul in einen Steckplatz im Baugruppenträger gesteckt. Verfahren Sie mit weiteren Baugruppen genauso.

Über den eingefügten Baugruppen wird der Name der Baugruppe angezeigt. In der Menüleiste können Sie über "Ansicht > Baugruppenbeschriftungen anzeigen" die Beschriftung der Baugruppe ein- und ausschalten.

## Baugruppe einfügen

Sie können Baugruppen mit Drag & Drop auch zwischen bereits gesteckte Baugruppen einfügen. Ziehen Sie dazu eine Baugruppe mit gedrückter Maustaste oberhalb zwischen zwei bereits gesteckte Baugruppen.



Es erscheint eine Einfügemarke. Wenn Sie den Mauszeiger loslassen, werden alle rechts der Einfügemarke gesteckten Baugruppen um einen Steckplatz nach rechts verschoben. Eine

eventuell überzählige Baugruppe wird in die Ablage nicht gesteckter Baugruppen verschoben. Auf den entstehenden freien Steckplatz wird die neue Baugruppe gesteckt.

### Siehe auch

Gerätesicht (Seite 371)

Ablage nicht gesteckter Baugruppen (Seite 396)

Informationen zu Hardware-Komponenten (Seite 383)

Allgemeine Steckplatzregeln (Seite 390)

### Hardware-Komponente löschen

Sie können Hardware-Komponenten über unterschiedliche Wege löschen. Gelöschte Hardware-Komponenten werden aus dem System entfernt und belegte Adressen stehen wieder zur Verfügung.

### Regeln

- CPUs oder Baugruppen aus dem Baugruppenträger und aus der Ablage nicht gesteckter Baugruppen können gelöscht werden.
- Wird ein Baugruppenträger in der Gerätesicht gelöscht, werden die gesteckten Hardware-Komponenten in die Ablage nicht gesteckter Baugruppen verschoben.

### Vorgehen

Um eine Hardware-Komponente zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Hardware-Komponenten, die Sie löschen möchten.
  - Netzsicht: Selektieren Sie in der grafischen Sicht oder in der Netzübersicht Geräte oder netzrelevante Hardware-Komponenten.
  - Gerätesicht: Selektieren Sie in der grafischen Sicht oder in der Geräteübersicht Baugruppenträger oder Baugruppen in Baugruppenträgern oder in der Ablage nicht gesteckter Baugruppen.
  - Topologiesicht: Selektieren Sie in der grafischen Sicht oder in der Topologieübersicht Geräte oder Hardware-Komponenten mit Ethernet-Schnittstellen..
  - Projektnavigation: Selektieren Sie Geräte oder einzelne Hardware-Komponenten aus der Baumstruktur.
2. Wählen Sie im Kontextmenü "Löschen" oder drücken Sie <Entf>. Wenn der Menüpunkt "Löschen" gegraut dargestellt wird, enthält Ihre Auswahl mindestens eine Komponente, die nicht gelöscht werden kann.

Die selektierten Hardware-Komponenten werden gelöscht.

---

#### **Hinweis**

Durch das Löschen von Hardware-Komponenten kann das Projekt Inkonsistenzen aufweisen, z. B. Verletzung von Steckplatzregeln. Auf die Inkonsistenzen wird bei dem Konsistenzcheck hingewiesen. Beseitigen Sie die Inkonsistenzen durch entsprechende Handlungen, z. B. stellen Sie die Einhaltung der Steckplatzregeln wieder her.

---

#### **Siehe auch**

Tastaturbedienung: Bearbeiten von Objekten (Seite 386)

#### **Hardware-Komponente kopieren**

Sie können Hardware-Komponenten in der Geräte- oder Netzsicht kopieren. Kopierte Hardware-Komponenten werden in einer Zwischenablage abgelegt und können an einer anderen Stelle aus dieser heraus eingefügt werden. Kopierte Stationen werden als neue Stationen in der Netzsicht eingefügt, kopierte Geräte und Baugruppen können in der Netz- und Gerätesicht in bestehende Baugruppenträger eingefügt werden.

#### **Regeln**

- Es können sowohl einzelne Objekte als auch mehrere Objekte gleichzeitig kopiert werden.
- Gesteckte Baugruppen aus dem Baugruppenträger und aus der Ablage nicht gesteckter Baugruppen können kopiert werden.
- Geräte und Baugruppen können unter Berücksichtigung der Steckplatzregeln nur auf freie und gültige Steckplätze kopiert werden.
- Baugruppenträger mit gesteckter CPU können nicht einzeln kopiert werden, sondern nur komplett mit allen gesteckten Hardware-Komponenten.

#### **Vorgehen**

Um eine Hardware-Komponente zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Hardware-Komponente, die Sie kopieren möchten.
  - Gerätesicht: Selektieren Sie die Baugruppe in einem Baugruppenträger oder in der Ablage nicht gesteckter Baugruppen.
  - Netzsicht: Selektieren Sie die Station oder die in der Netzsicht relevante Hardware-Komponente.
  - Projektnavigation: Selektieren Sie die Station oder Baugruppe.
2. Wählen Sie im Kontextmenü "Kopieren" oder drücken Sie <Strg+C>. Wenn der Menüpunkt "Kopieren" gegraut dargestellt wird, enthält Ihre Auswahl mindestens eine Komponente, die nicht kopiert werden kann.

3. Selektieren Sie den Ort, in den der Inhalt aus der Zwischenablage eingefügt werden soll.
  - Gerätesicht: Selektieren Sie einen freien Steckplatz des Baugruppenträgers oder die Ablage nicht gesteckter Baugruppen.
  - Netzsicht: Selektieren Sie eine Station, um Geräte oder Baugruppen einzufügen oder bewegen Sie den Mauszeiger zu einem freien Ort in der Netzsicht, um eine kopierte Station oder in der Netzsicht relevante Hardware-Komponente einzufügen.
4. Wählen Sie im Kontextmenü "Einfügen" oder drücken Sie <Strg+V>. Wenn der Menüpunkt "Einfügen" graut dargestellt wird, ist die Zwischenablage leer oder enthält mindestens eine Komponente, die an dieser Stelle nicht eingefügt werden kann.

Das selektierte Objekt wird am gewählten Platz eingefügt.

Wenn Sie zum Einfügen einer Baugruppe eine Station in der Netzsicht selektiert haben, wird die Baugruppe auf den ersten freien und gültigen Steckplatz gesteckt. Steht kein freier und gültiger Steckplatz zur Verfügung, wird das Objekt in die Ablage nicht gesteckter Baugruppen eingefügt.

---

#### Hinweis

Sie können eine Baugruppe auch von einem Gerät in ein anderes kopieren:

Kopieren Sie dazu eine Baugruppe im Hardware- und Netzwerkeditor, selektieren Sie dann in der Netzsicht oder der Klappliste der Gerätesicht ein anderes Gerät und fügen Sie die Baugruppe ein.

In der Gerätesicht können Sie das kopierte Objekt direkt auf einen Steckplatz bzw. in die Ablage nicht gesteckter Baugruppen einfügen. Wenn Sie das kopierte Objekt in der Netzsicht einem Gerät oder einer Station hinzufügen, wird das kopierte Objekt in den ersten freien Steckplatz gesteckt.

Ist kein Steckplatz für das Objekt verfügbar, wird das Objekt automatisch in die Ablage nicht gesteckter Baugruppen (Seite 396) eingefügt.

---

#### Hinweis

Sie können eine selektierte Hardware-Komponente auch direkt mit <Strg> und Drag & Drop kopieren.

---

#### Siehe auch

Tastaturbedienung: Bearbeiten von Objekten (Seite 386)

#### Hardware-Komponente verschieben

Sie können Hardware-Komponenten in der Geräte- oder Netzsicht verschieben.

## Regeln

- Geräte, Baugruppen und Module aus dem Baugruppenträger und aus der Ablage nicht gesteckter Baugruppen können unter Berücksichtigung der Steckplatzregeln verschoben werden.
- CPs können in der Netzsicht verschoben werden. Der CP wird im Zielgerät in einen freien und gültigen Steckplatz gesteckt. Ist kein freier Steckplatz vorhanden, wird der einzufügende CP in die Ablage nicht gesteckter Baugruppen verschoben.
- CPUs und Slave-Kopfbaugruppen können in der Netzsicht zwischen den Geräten verschoben werden, je nach CPU-Typ auch innerhalb des Baugruppenträgers.

---

### Hinweis

Verschobene CPs werden von ihrem Netz getrennt, behalten aber die Netzparameter und Adresse. Wenn der CP wieder an das Netz angeschlossen wird, aber dessen Adresse bereits anderweitig vergeben wurde, können Sie dem CP über einen Dialog eine neue eindeutige Adresse zuweisen.

---

## Vorgehen

Um eine Hardware-Komponente zu verschieben, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Hardware-Komponente, die Sie verschieben möchten.
  - Gerätesicht: Selektieren Sie die Baugruppe in einem Baugruppenträger oder in der Ablage nicht gesteckter Baugruppen.
  - Netzsicht: Selektieren Sie die netzsichtrelevante Hardware-Komponente.
2. Wählen Sie im Kontextmenü "Ausschneiden" oder drücken Sie <Strg+X>. Wenn der Menüpunkt "Ausschneiden" gegraut dargestellt wird, enthält Ihre Auswahl mindestens eine Komponente, die nicht ausgeschnitten werden kann.
3. Selektieren Sie den Ort, an den das ausgeschnittene Objekt verschoben werden soll.
  - Gerätesicht: Selektieren Sie einen freien Steckplatz des Baugruppenträgers oder die Ablage nicht gesteckter Baugruppen.
  - Netzsicht: Selektieren Sie eine Station, um Geräte oder Baugruppen einzufügen.
4. Wählen Sie im Kontextmenü "Einfügen" oder drücken Sie <Strg+V>. Wenn der Menüpunkt "Einfügen" gegraut dargestellt wird, ist die Zwischenablage leer oder enthält mindestens eine Komponente, die an dieser Stelle nicht eingefügt werden kann.

Die selektierte Hardware-Komponente wird an den Zielort verschoben. Falls es sich bei der verschobenen Hardware-Komponente um ein vernetztes Objekt handelt, wird es vom Netz entkoppelt.

---

### Hinweis

Sie können eine selektierte Hardware-Komponente auch direkt mit Drag & Drop verschieben.

---

## Siehe auch

Tastaturbedienung: Bearbeiten von Objekten (Seite 386)

## Hardware-Komponente tauschen

Sie können Hardware-Komponenten gegeneinander tauschen. Dadurch können Sie beispielsweise nicht spezifizierte CPUs (Seite 400) durch verfügbare CPUs aus dem Hardware-Katalog ersetzen.

## Regeln

Es können nur Hardware-Komponenten gegeneinander getauscht werden, die den Baugruppentausch unterstützen und die miteinander kompatibel sind.

## Vorgehen

Um Baugruppen gegeneinander zu tauschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Baugruppe, die Sie austauschen möchten.
2. Öffnen Sie das Kontextmenü:
  - Wenn der Eintrag "Gerät tauschen" aktiv ist, ist ein Baugruppentausch möglich.
  - Wenn der Eintrag "Gerät tauschen" deaktiviert ist, ist ein Baugruppentausch nicht möglich.
3. Klicken Sie im Kontextmenü auf den Eintrag "Gerät tauschen". Das Dialogfeld "Gerät tauschen" wird geöffnet.
4. Selektieren Sie in der Baumstruktur unter "Neues Gerät" die Baugruppe, durch die Sie Ihre aktuelle Baugruppe ersetzen möchten.
5. Klicken Sie auf "OK".

Die vorhandene Baugruppe wird durch die neue Baugruppe ersetzt.

Alternativ können Sie auch direkt eine Baugruppe aus dem Hardware-Katalog mit Drag & Drop auf die zu tauschende Baugruppe ziehen. Wenn die beiden Baugruppen gegeneinander ausgetauscht werden können, wird dies durch ein Symbol am Mauszeiger dargestellt.

## Eigenschaften und Parameter bearbeiten

Nachdem Sie Hardware-Komponenten auf Ihren Baugruppenträger gesteckt haben, können Sie in der Netz- oder Gerätesicht deren voreingestellte Eigenschaften bearbeiten, z. B. die Parameter oder Adressen.

## Voraussetzung

Sie befinden sich in der Gerätesicht.

---

### Hinweis

Sie können auch in der Netzsicht Eigenschaften und Parameter bearbeiten. In der grafischen Netzsicht haben Sie Zugriff auf die netzrelevanten Hardware-Komponenten und die Station. Auf Baugruppen und Hardware-Komponenten, die nicht in der grafischen Netzsicht angezeigt werden, können Sie über die tabellarische Netzsicht zugreifen.

---

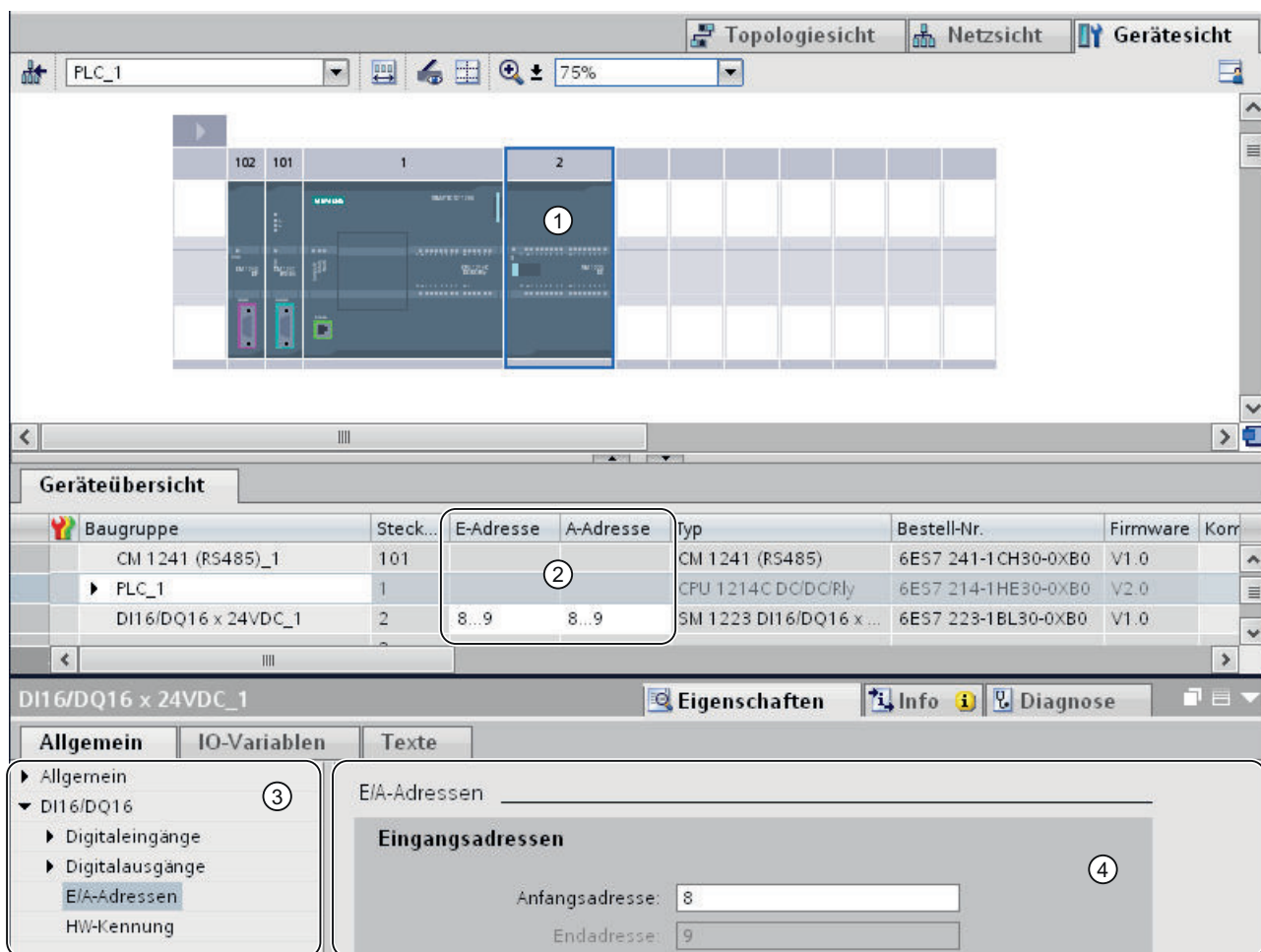
## Vorgehen

Um die Eigenschaften und Parameter der Hardware-Komponenten zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der grafischen Darstellung die CPU, die Baugruppe, den Baugruppenträger oder die Schnittstelle, die Sie bearbeiten möchten.
2. Bearbeiten Sie die Einstellungen für das gewählte Objekt:
  - In der tabellarischen Darstellung können Sie z. B. Adressen und Namen ändern.
  - Im Inspektorfenster stehen Ihnen unter "Eigenschaften" verschiedenste Einstellmöglichkeiten zur Verfügung.

Beachten Sie, dass Baugruppen nur dann vollständig parametrierbar sind, wenn sie einer CPU zugeordnet sind. Daher müssen PROFIBUS- bzw. PROFINET-Interfacemodule zunächst mit der CPU bzw. einem zentral steckenden Kommunikationsmodul vernetzt werden, so dass sie ein Mastersystem bzw. IO-System bilden. Erst dann sind z. B. die Adressen der dezentral gesteckten Komponenten editierbar.

### Beispiel für das Ändern von Einstellungen



- ① Selektion einer Baugruppe
- ② Bearbeitungsmöglichkeit für Adressen in der Geräteübersicht
- ③ Auswahlmöglichkeiten im Inspektorfenster
- ④ Bearbeitungsmöglichkeit für Adressen im Inspektorfenster

### Siehe auch

Inspektorfenster (Seite 379)



## Ein- und Ausgangsadressen in der Adressübersicht

### Einführung

Die aktuell verwendeten Ein- und Ausgangsadressen können in der Adressübersicht tabellarisch angezeigt werden. Die Adressübersicht finden Sie im Inspektorfenster unter "Eigenschaften" der CPU.

### Aufbau der Adressübersicht

Über verschiedene Optionskästchen können Sie einstellen, welche Objekte in der Adressübersicht angezeigt werden sollen:

- Eingänge: Anzeige der Eingangsadressen
- Ausgänge: Anzeige der Ausgangsadressen
- Adresslücken: Anzeige freier Adressräume
- Steckplatz: Anzeige der Steckplatznummer

Typischerweise werden in der Adressübersicht die folgenden Informationen angezeigt:

| Tabellenkopf     | Bedeutung  |
|------------------|--|
| Typ              | Zeigt an, ob es sich um einen Eingangs- oder einen Ausgangsadressbereich handelt.  |
| Adr. von         | Anfangsadresse des Adressbereiches.  |
| Adr. bis         | Endadresse des Adressbereiches.  |
| Baugruppe        | Baugruppe, von der der Adressbereich genutzt wird.   |
| TPA              | Nr. des Teilprozessabbildes. Zeigt an, ob der zyklische OB im Teilprozessabbild liegt.   |
| DP               | Nummer des Mastersystems. So lässt sich ermitteln, welche Slaves einem Master zugeordnet sind. Der Wert in Klammern gibt die PROFIBUS-Adresse der Hardwarekomponente an. |
| PN               | Nummer des IO-Systems. Der Wert in Klammern steht für die Gerätenummer der Hardwarekomponente.   |
| Baugruppenträger | Nummer des Baugruppenträgers, auf dem die Hardwarekomponente gesteckt ist.   |
| Steckplatz       | Nummer des Steckplatzes auf dem Baugruppenträger, in dem die Hardwarekomponente gesteckt ist.  |

### Siehe auch

Ein- und Ausgangsadressen festlegen (Seite 629)

## Modulversion aktualisieren

### Begriffsklärung

Im Folgenden werden die Begriffe "Modulversion" und "Firmware-Version" genauer erklärt.

- Modulversion: diejenige Version der Projektierungs-Software, aus der die Modulbeschreibung stammt.  
Bsp.: V11.0.0.0
- Firmware-Version: der Ausgabestand der Firmware des offline parametrisierten Moduls  
Bsp.: V2.0

### Voraussetzung

- Sie haben eine Gerätekonfiguration erstellt.
- Sie haben zu einem späteren Zeitpunkt beispielsweise ein Update oder ein Optionspaket installiert. Durch diese Installation wurde im Hardware-Katalog bei mindestens einem Modultyp die Modulversion aktualisiert, wobei die neue Version inkompatibel zur bisherigen ist.
- Sie haben in Ihrer Gerätekonfiguration solche Module verwendet und wollen die geänderten bzw. hinzugekommenen Eigenschaften nutzen.

### Vorgehen

Führen Sie für jeden betroffenen Modultyp die folgenden Schritte durch.

1. Selektieren Sie in der Gerätesicht ein betroffenes Modul.
2. Gehen Sie im Inspektorfenster zu Eigenschaften > Allgemein > Kataloginformation. Klicken Sie dort auf die Schaltfläche "Modulversion aktualisieren".
3. Legen Sie in der dann erscheinenden Abfrage fest, ob Sie im aktuellen Projekt nur für das selektierte Modul oder für alle Module dieses Typs die Modulversion aktualisieren wollen.

### Ergebnis

Im aktuellen Projekt werden die ausgewählten Module durch die gleichen Module mit aktualisierter Modulversion ersetzt.

### In welchen Fällen brauchen Sie die Aktualisierung der Modulversion nicht durchzuführen?

Die Aktualisierung der Modulversion ist in den folgenden Fällen überflüssig:

- Sie wollen die geänderten bzw. hinzugekommenen Eigenschaften der Module nicht nutzen.
- Sie öffnen ein bestehendes Projekt mit einem Stand der Projektierungs-Software, der aktueller ist als der Stand, mit dem Sie das Projekt erstellt haben, und das System führt eine automatische Projektkonvertierung z. B. von V11 nach V12 durch. In diesem Fall werden alle veralteten Modulversionen automatisch angepasst.

## 8.1.3 Netze konfigurieren

### 8.1.3.1 Geräte vernetzen

#### Kommunikation und Netze

##### Kommunikation zwischen Geräten

Grundlage für jede Art von Kommunikation ist immer ein zuvor konfiguriertes Netz. Mit der Netzkonfiguration schaffen Sie die notwendigen Voraussetzungen für die Kommunikation:

- Versorgung aller Teilnehmer an einem Netz mit eindeutigen Adressen
- Kommunikation der Teilnehmer mit konsistenten Übertragungseigenschaften

##### Netzprojektierung

Um Netze zu projektieren, sind folgende Schritte notwendig:

- Geräte mit Subnetz verbinden
- Eigenschaften/Parameter für jedes Subnetz festlegen
- Teilnehmereigenschaften für jede vernetzte Baugruppe festlegen
- Konfigurationsdaten in die Geräte laden, um Schnittstellen mit den aus der Netzprojektierung resultierenden Einstellungen zu versorgen
- Netzkonfiguration dokumentieren

Für die Open User Communication wird das Anlegen und Konfigurieren eines Subnetzes von der Verbindungsparametrierung unterstützt.

##### Zusammenhang zwischen Netzkonfiguration und Projekt

Innerhalb eines Projekts werden Subnetze mit ihren Eigenschaften verwaltet. Eigenschaften ergeben sich im Wesentlichen aus einstellbaren Netzparametern und aus der Anzahl und den Kommunikationseigenschaften der angeschlossenen Teilnehmer.

Die zu vernetzenden Teilnehmer müssen sich im selben Projekt befinden.

##### Subnetz-Name und Subnetz-ID

Subnetze werden innerhalb des Projektes durch einen Subnetz-Namen und eine Subnetz-ID eindeutig gekennzeichnet. Die Subnetz-ID ist in allen Komponenten mit vernetzbaren Schnittstellen hinterlegt. Damit können Komponenten auch nach einem Hochladen in ein Projekt eindeutig einem Subnetz zugeordnet werden.

## Möglichkeiten der Vernetzung

Sie können im Projekt Geräte mit kommunikationsfähigen Komponenten anlegen und vernetzen. Für die Vernetzung der Geräte gibt es prinzipiell folgende Möglichkeiten:

- Sie verbinden die Schnittstellen der kommunikationsfähigen Komponenten untereinander. Dadurch wird ein zum Typ der Schnittstelle passendes neues Subnetz angelegt.
- Sie verbinden die Schnittstelle der kommunikationsfähigen Geräte mit einem neuen oder einem bereits angelegten Subnetz.
- Sie erstellen eine Verbindung der Open User Communication. Über die Verbindungsparametrierung der Open User Communication wird automatisch ein Subnetz zwischen den Kommunikationspartnern angelegt.
- Sie projektieren Verbindungen über die grafische Verbindungsprojektierung; fehlende Vernetzungen werden hierbei erkannt und werden automatisch oder dialoggeführt angelegt.

Aufgrund unterschiedlicher Aufgaben der Geräte oder aufgrund der Ausdehnung der Anlage kann es erforderlich sein, mehrere Subnetze zu betreiben. Diese Subnetze werden in einem Projekt verwaltet.

## Geräte in der Netzsicht vernetzen

### Möglichkeiten

In der grafischen Netzsicht behalten Sie die Übersicht über die Subnetze der gesamten Anlage im Projekt. Unterstützend können Sie die tabellarische Netzübersicht verwenden.

Um die Schnittstelle einer kommunikationsfähigen Komponente zu vernetzen, haben Sie je nach Ausgangssituation unterschiedliche Möglichkeiten bei der Projektierung. Nachfolgend werden diese Vorgehensweisen beschrieben:

- Einzelnes Subnetz erstellen
- Mehrere Subnetze gleichzeitig erstellen
- Zwei Zielgeräte über ein neues Subnetz verbinden
- Geräte mit bestehendem Subnetz verbinden
- Vorhandenes Subnetz aus Liste auswählen
- Automatische Vernetzung während der Verbindungsprojektierung;  
Siehe auch: Auto-Hotspot

Mögliche Ausgangssituationen sind:

- Es ist noch kein geeignetes Subnetz vorhanden.
- Das Subnetz, mit dem Sie die Komponente verbinden wollen, ist bereits vorhanden.

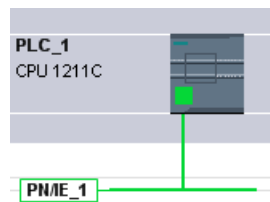
### Vorgehen - Einzelnes Subnetz erstellen

Um ein Subnetz zu erzeugen und es mit einer Schnittstelle zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle einer CPU / eines CP.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Subnetz erstellen".

Die selektierte Schnittstelle wird mit einem neuen Subnetz verbunden. Dabei werden die Adressparameter der Schnittstelle automatisch konsistent eingestellt.

Im folgenden Bild sehen Sie eine Schnittstelle mit ausgehender Verbindungslinie zu einem Subnetz:



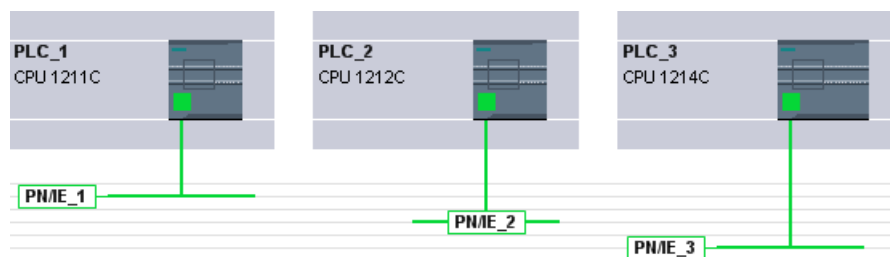
### Vorgehen - Mehrere Subnetze gleichzeitig erstellen

Um mehrere Subnetze gleichzeitig zu erzeugen, gehen Sie folgendermaßen vor:

1. Selektieren Sie mehrere Schnittstellen durch Mausklick und gleichzeitiges Drücken der Taste <Strg>.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Subnetz erstellen".

Jede selektierte Schnittstelle wird mit einem neuen Subnetz verbunden. Dabei werden die Adressparameter der Schnittstelle automatisch konsistent eingestellt.

Im folgenden Bild sehen Sie mehrere angelegte Subnetze durch eine Mehrfachselektion der Schnittstellen:

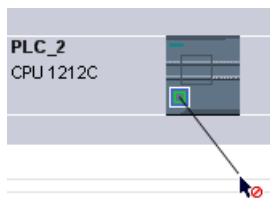


### Vorgehen - Zwei Zielgeräte über ein neues Subnetz verbinden

Um eine Schnittstelle mit einem anderen Gerät über ein noch nicht vorhandenes Subnetz zu verbinden, gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf die zu vernetzende Schnittstelle einer kommunikationsfähigen Komponente.
2. Klicken Sie mit der linken Maustaste und halten Sie die Maustaste gedrückt.

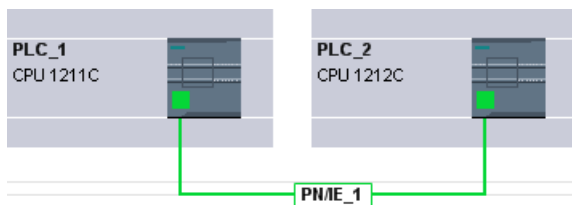
3. Verschieben Sie den Mauscursor.  
Der Mauscursor zeigt jetzt mit dem Vernetzungssymbol den Arbeitsmodus "Vernetzen" an. Gleichzeitig sehen Sie im Mauscursor das Sperrsymbol, das erst über einer gültigen Zielposition verschwindet.



4. Ziehen Sie nun den Mauscursor auf die Schnittstelle des Zielgeräts. Sie können hierbei die linke Maustaste gedrückt halten oder loslassen.
5. Lassen Sie nun die linke Maustaste los oder klicken Sie erneut mit der linken Maustaste.

Ein neues Subnetz wird angelegt. Die Schnittstellen sind jetzt über das neue Subnetz verbunden. Dabei werden die Adressparameter der Schnittstelle automatisch konsistent eingestellt.

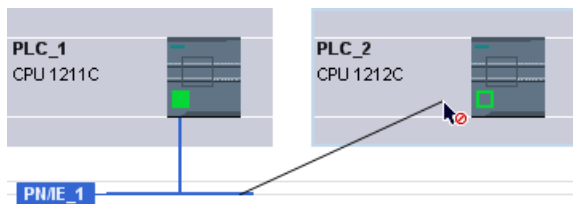
Im folgenden Bild sehen Sie zwei vernetzte Geräte:



### Vorgehen - Geräte mit bestehendem Subnetz verbinden

Um eine Schnittstelle mit einem bereits bestehendem Subnetz zu verbinden, gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf der zu vernetzenden Schnittstelle einer kommunikationsfähigen Komponente oder auf dem vorhandenen Subnetz.
2. Klicken Sie mit der linken Maustaste und halten Sie die Maustaste gedrückt.
3. Verschieben Sie den Mauscursor.  
Der Mauscursor zeigt jetzt mit dem Vernetzungssymbol den Arbeitsmodus "Vernetzen" an. Gleichzeitig sehen Sie im Mauscursor das Sperrsymbol, das erst bei Platzierung auf einer gültigen Zielposition verschwindet.
4. Ziehen Sie nun den Mauscursor auf das vorhandene Subnetz oder auf die zu vernetzende Schnittstelle. Sie können hierbei die linke Maustaste gedrückt halten oder loslassen.



5. Lassen Sie nun die linke Maustaste los oder klicken Sie erneut mit der linken Maustaste.

Ergebnis:

Die Schnittstelle und das ausgewählte Subnetz sind jetzt verbunden. Dabei werden die Adressparameter der Schnittstelle automatisch konsistent eingestellt.

### **Vorgehen - Vorhandenes Subnetz aus Liste auswählen**

Um eine Schnittstelle mit einem bereits angelegten Subnetz zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle einer CPU.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Neuem Subnetz zuweisen". Es erscheint eine Auswahlliste mit den vorhandenen Subnetzen.
3. Wählen Sie aus der Liste ein Subnetz aus.

Die Schnittstelle und das ausgewählte Subnetz sind jetzt verbunden. Dabei werden die Adressparameter der Schnittstelle automatisch konsistent eingestellt.

### **Tabellarische Netzübersicht**

#### **Bedeutung**

Die tabellarische Netzübersicht ergänzt die grafische Netzsicht mit folgenden Funktionen:

- Sie erhalten Detailinformationen über den Aufbau und die Parametrierung der Geräte.
- Sie können über die Spalte "Subnetz" kommunikationsfähige Komponenten mit angelegten Subnetzen verbinden.

#### **Grundfunktionen für Tabellen**

Die Netzübersicht unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spalten der Tabelle ein- und ausblenden  
Hinweis: Die für die Konfiguration maßgeblichen Spalten können nicht ausgeblendet werden.
- Spaltenbreite optimieren
- Tabelle sortieren
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen

### **Geräte in der Gerätesicht vernetzen**

#### **Vernetzung in der Gerätesicht**

In der Gerätesicht können Sie alle Parameter der zu einem Gerät gehörigen Komponenten und der Schnittstellen detailliert prüfen und einstellen. Auch die Zuordnung der Schnittstellen auf die im Projekt angelegten Subnetze ist hier möglich.

### Voraussetzung

- Das Subnetz, mit dem eine Schnittstelle verbunden werden soll, ist bereits angelegt.
- Falls das Subnetz noch nicht angelegt ist, wechseln Sie in die Netzsicht und führen dort die Vernetzung durch.

### Vorgehen - Mit vorhandenem Subnetz verbinden

Um die Schnittstelle mit einem bereits angelegten Subnetz zu verbinden, gehen Sie in der Gerätesicht folgendermaßen vor:

1. Selektieren Sie die komplette kommunikationsfähigen Komponente oder die zu vernetzende Schnittstelle.  
Im Inspektorfenster werden die Eigenschaften der gewählten Schnittstelle oder Komponente angezeigt.
2. Wählen Sie im Inspektorfenster die zur gewählten Schnittstelle gehörende Parametergruppe aus, z. B. bei einer PROFINET-Schnittstelle die Parameter-Gruppe "Ethernet-Adressen".
3. Wählen Sie unter "Schnittstelle vernetzt mit" das zu verbindenden Subnetz aus der Klappliste "Subnetz" aus.

Die Schnittstelle und das ausgewählte Subnetz sind jetzt verbunden. Dabei werden die Adressparameter der Schnittstelle automatisch konsistent eingestellt.

### Vorgehen - Neues Subnetz anlegen

Um ein Subnetz zu erzeugen und es mit der Schnittstelle zu verbinden, gehen Sie in der Gerätesicht folgendermaßen vor:

1. Selektieren Sie die komplette kommunikationsfähigen Komponente oder die zu vernetzende Schnittstelle.  
Im Inspektorfenster werden die Eigenschaften der gewählten Schnittstelle oder Komponente angezeigt.
2. Wählen Sie im Inspektorfenster die zur gewählten Schnittstelle gehörende Parametergruppe aus, z. B. bei einer PROFINET-Schnittstelle die Parameter-Gruppe "Ethernet-Adressen".
3. Klicken Sie unter "Schnittstelle vernetzt mit" auf die Schaltfläche "Neues Subnetz hinzufügen".

Die Schnittstelle wird mit einem neuen Subnetz des geeigneten Subnetztyps verbunden. Dabei werden die Adressparameter der Schnittstelle automatisch konsistent eingestellt.



## Netzparameter und Schnittstellenparameter prüfen oder ändern

### Einführung

Die Kommunikationsfähigkeit zwischen vernetzten Geräten erfordert die Konfiguration folgender Parameter:

- **Netzparameter**  
Netzparameter identifizieren das Netz innerhalb der Anlagen-Projektierung, z. B. über einen Namen.
- **Schnittstellenparameter**  
Schnittstellenparameter legen spezifische Eigenschaften einer kommunikationsfähigen Komponente fest. Eine mit den Netzparametern konsistente Einstellung der Adressen und der Übertragungseigenschaften wird automatisch hergestellt.

---

#### **Hinweis**

Netzparameter und Schnittstellenparameter sind bei der Vernetzung standardmäßig so eingestellt, dass die Kommunikation für viele Anwendungsfälle ohne Parameteränderung möglich ist.

---

### Vorgehen - Netzparameter prüfen oder ändern

Um Netzparameter zu prüfen oder zu ändern, gehen Sie folgendermaßen vor:

1. Gehen Sie in die Netzsicht.
2. Selektieren Sie in der Netzsicht das Subnetz.  
Im Inspektorfenster sind die Netzparameter im Register "Eigenschaften" sichtbar.
3. Prüfen oder modifizieren Sie bei Bedarf die Netzparameter in der jeweiligen Parametergruppe.

### Vorgehen - Schnittstellenparameter prüfen oder ändern

Sie können Schnittstellenparameter in der Netz- und Gerätesicht prüfen und ändern.

Um Schnittstellenparameter zu prüfen oder zu ändern, gehen Sie folgendermaßen vor:

1. Gehen Sie in die Netzsicht oder die Gerätesicht.
2. Selektieren Sie die Schnittstelle.  
Im Inspektorfenster sind die Schnittstellenparameter im Register "Eigenschaften" sichtbar.
3. Prüfen oder modifizieren Sie bei Bedarf die Schnittstellenparameter in der jeweiligen Parametergruppe.

## Vernetzungen ändern

### Einführung

Sie können den Netzanschluss einer Schnittstelle aufheben oder einem anderen Subnetz des selben Subnetztyps zuweisen.

### Auswirkungen

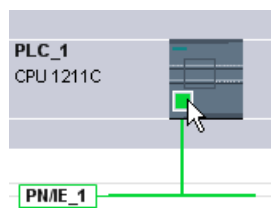
Je nach Ausführung sind zu unterscheiden:

- Netzanschluss einer Schnittstelle aufheben  
Die konfigurierten Parameter der Schnittstelle bleiben unverändert.
- Netzanschluss einem anderen Subnetz zuweisen  
Wenn die Adressen im zugewiesenen Subnetz nicht eindeutig, d. h. bereits vorhanden sind, werden sie automatisch eindeutig gemacht.

### Vorgehen - Netzanschluss aufheben

Um den Netzanschluss einer Schnittstelle aufzuheben, gehen Sie folgendermaßen vor:

1. Selektieren Sie die vernetzte Schnittstelle.



2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Vom Subnetz trennen"

Der Netzanschluss wird gelöscht, Adressen der Schnittstelle werden jedoch nicht verändert.

Projektierte Verbindungen bleiben erhalten, sind in der Verbindungstabelle jedoch wegen der fehlenden Vernetzung rot markiert. Spezifizierte Verbindungen bleiben spezifiziert.

### Siehe auch

Geräte in der Netzsicht vernetzen (Seite 414)

## Subnetze kopieren, ausschneiden oder löschen

### Einführung

Subnetze können Sie als einzelne Objekte oder im Verbund mit vernetzten Geräten oder zusammen mit weiteren Netzen kopieren.

Damit können Sie z. B. komplexe Konfigurationen, die innerhalb des Projektes in mehrfacher Ausprägung angeordnet werden sollen, ohne zusätzlichen Aufwand anlegen.

## Auswirkungen auf das kopierte Subnetz

Eigenschaften, die innerhalb eines Projektes eindeutig vergeben sein müssen, werden beim Kopieren den kopierten Objekten entsprechend neu zugewiesen.

Für Subnetze bedeutet dies: Die Subnetz-ID und der Subnetz-Name werden dem kopierten Subnetz neu zugewiesen.

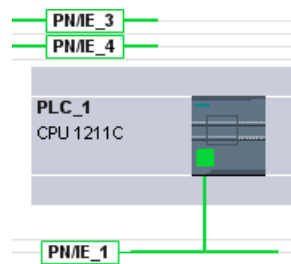
Die konfigurierten Eigenschaften werden in das kopierte Subnetz übernommen.

## Vorgehen - Subnetz kopieren

Um ein oder mehrere Subnetze zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein oder mehrere Subnetze.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Die kopierten Subnetze werden im oberen Bereich der Netzsicht als "verwaiste" Subnetze angezeigt.



## Vorgehen - Subnetz mit angeschlossenen Geräten kopieren

Um ein oder mehrere Subnetze mit vernetzten Geräten zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein oder mehrere Subnetze mit den angeschlossenen Geräten - z. B. mit der Lasso-Funktion.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Es werden vollständige Kopien der Subnetze und der angeschlossenen Geräte angelegt.

Projektierte Verbindungen werden übernommen und bleiben innerhalb der kopierten Geräte erhalten. Verbindungen zu nicht mitkopierten Geräten werden aufgebrochen und sind damit unspezifiziert.

## Netzkonfiguration MPI

### MPI-Adressvergabe

Für Geräte mit einer MPI-Schnittstelle gilt, dass sämtliche Teilnehmer eines Subnetzes eine unterschiedliche Teilnehmeradresse besitzen müssen.

CPUs mit MPI-Adresse werden mit der Default-MPI-Adresse 2 ausgeliefert. Da Sie diese Adresse nur einmal im MPI-Subnetz benutzen dürfen, müssen Sie die voreingestellte Adresse in allen anderen CPUs ändern.

Für Geräte mit den Bestell.-Nr. 6ES7 3xx-xxxxx-0AB0 gilt:

Bei der Planung der MPI-Adressen für mehrere CPUs müssen Sie "MPI-Adresslücken" für FMs und CPs mit eigenen MPI-Adressen vorsehen, um Doppelbelegung von Adressen zu vermeiden.

Erst wenn sämtliche Baugruppen eines Subnetzes unterschiedliche Adressen besitzen und Ihr tatsächlicher Aufbau mit der erstellten Netzkonfiguration übereinstimmt, sollten Sie die Einstellungen über das Netz laden.

### Regeln für die MPI-Adressvergabe

- Vergeben Sie die MPI-Adressen in aufsteigender Reihenfolge.
- Reservieren Sie die MPI-Adresse 0 für ein PG.
- Bis zu 126 (adressierbare) Teilnehmer können Sie in einem MPI-Subnetz miteinander verbinden; bis zu 8 Teilnehmer bei einer Übertragungsgeschwindigkeit von 19,2 kbit/s.
- Alle MPI-Adressen eines MPI-Subnetzes müssen unterschiedlich sein.

Weitere Regeln zum Aufbau eines Netzes finden Sie in den Handbüchern zum Aufbauen von Automatisierungssystemen.

## Netzkonfiguration PROFIBUS

### PROFIBUS-Adressen

### Regeln für die Netzkonfiguration

Sämtliche Teilnehmer eines Subnetzes müssen eine unterschiedliche PROFIBUS-Adresse besitzen.

Erst wenn sämtliche Baugruppen eines Subnetzes unterschiedliche Adressen besitzen und Ihr tatsächlicher Aufbau mit der erstellten Netzkonfiguration übereinstimmt, sollten Sie die Einstellungen über das Netz laden.

Am PROFIBUS-Subnetz können Teilnehmer angeschlossen sein, die über projektierte Verbindungen kommunizieren oder die zu einem PROFIBUS DP-Mastersystem gehören.

Zur Konfiguration eines DP-Mastersystems finden Sie weitere Informationen in den nachfolgenden Kapiteln.

## Voraussetzung

Die CPU 121xC ist ab der Firmware-Version 2.0 PROFIBUS-fähig.

## Regeln für die PROFIBUS-Adressvergabe

- Vergeben Sie die PROFIBUS-Adressen in aufsteigender Reihenfolge.
- Reservieren Sie die PROFIBUS-Adresse "0" für ein Programmiergerät.
- Vergeben Sie für jeden Teilnehmer am PROFIBUS-Netz bzw. für jeden DP-Master und jeden DP-Slave im PROFIBUS-Netz eine eindeutige PROFIBUS-Adresse im Bereich 0 bis 126.
- Es gibt Baugruppen, bei denen die kleinste einstellbare Adresse größer als 1 sein muss.
- Alle PROFIBUS-Adressen eines PROFIBUS-Subnetzes müssen unterschiedlich sein.

Weitere Regeln zum Aufbau eines Netzes finden Sie in den Handbüchern zum Aufbauen von Automatisierungssystemen, z. B. SIMATIC S7-1200.

---

### Hinweis

#### **PROFIBUS-Adresse "0"**

Reservieren Sie die PROFIBUS-Adresse "0" für ein Programmiergerät, das Sie später für Servicezwecke kurzzeitig an das PROFIBUS-Netz anschließen.

---

## Siehe auch

Wissenswertes zu PROFIBUS-Busparametern (Seite 423)

## Wissenswertes zu PROFIBUS-Busparametern

### Parameter aufeinander abgestimmt

Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Parameter für das Busprofil aufeinander abgestimmt sind. Sie sollten die voreingestellten Werte deshalb nur dann ändern, wenn Sie mit der Parametrierung des Busprofils für PROFIBUS vertraut sind.

---

### Hinweis

Die Busparameter sind in Abhängigkeit vom Busprofil einstellbar oder nicht einstellbar. Wenn die Busparameter nicht einstellbar sind, dann sind sie grau dargestellt. Es werden immer die Offline-Werte der Busparameter angezeigt, auch wenn Sie online mit dem Zielsystem verbunden sind.

---

Die angezeigten Parameter gelten für das gesamte PROFIBUS-Subnetz und sind im Folgenden kurz erklärt.

### Zyklisches Verteilen der Busparameter einschalten

Wenn bei ausgewähltem PROFIBUS-Subnetz im Inspektorfenster unter "Busparameter" das Optionskästchen "Zyklisches Verteilen der Busparameter einschalten" aktiviert ist, dann werden die Busparameter im Betrieb zyklisch von den Baugruppen gesendet, die diese Funktion unterstützen. Auf diese Weise kann z. B. ein PG ohne Probleme im laufenden Betrieb an den PROFIBUS angeschlossen werden.

Sie sollten diese Funktion deaktivieren:

- Bei heterogenem PROFIBUS-Subnetz (genauer: bei angeschlossenen Fremdgeräten, deren Protokoll den DSAP 63 für Multicast benutzt)
- Bei Äquidistanz-Betrieb (Buszyklus minimieren!)

### Busparameter für das Busprofil von PROFIBUS-Subnetzen

| Busparameter        | Einstellbar? | Grenzwerte  |
|---------------------|--------------|---|
| Tslot_Init          | Ja           | Max. Tsdr + 15 <= Tslot_init <= 16.383 t_bit  |
| Max. Tsdr           | Ja           | 35 + 2*Tset + Tqui <= Max. Tsdr <= 1.023 t_bit  |
| Min. Tsdr           | Ja           | 11 t_bit <= Min. Tsdr <= MIN(255 t_bit, ...<br>... Max. Tsdr - 1, 34 + 2*Tset + Tqui) |
| Tset                | Ja           | 1 t_bit <= Tset <= 494 t_bit  |
| Tqui                | Ja           | 0 t_bit <= Tqui <= MIN(31 t_bit, Min. Tsdr - 1)                                       |
| GAP-Faktor          | Ja           | 1 <= GAP-Faktor <= 100  |
| Retry Limit         | Ja           | 1 <= Retry Limit <= 15  |
| Tslot               | Nein         | ---   |
| Tid2                | Nein         | Tid2 = Max. Tsdr  |
| Trdy                | Nein         | Trdy = Min. Tsdr  |
| Tid1                | Nein         | Tid1 = 35 + 2*Tset + Tqui   |
| Ttr                 | Ja           | 256 t_bit <= Ttr <= 16.777.960 t_bit  |
| Ttr typisch         | Nein         | Diese Zeit dient nur der Information und wird nicht an die Teilnehmer übertragen.     |
| Ansprechüberwachung |              | 10 ms <= Ansprechüberwachung (watchdog) <= 650 s                                      |

Wenn Sie ein benutzerdefiniertes Busprofil erstellen wollen, dann empfehlen wir folgende Einstellungen:

- minimale Target-Rotation-Time (Ttr) = 5000mal HSA (höchste PROFIBUS-Adresse)
- minimale Ansprechüberwachung (Watchdog) = 6250mal HSA

### Neu berechnen

Über die Schaltfläche "Neu berechnen" können Sie die Parameter neu berechnen lassen.

### Siehe auch

PROFIBUS-Adressen (Seite 422)

Beschreibung der Busparameter (Seite 425)

## Beschreibung der Busparameter

### Detaillierte Beschreibung der PROFIBUS-Busparameter

| <b>Busparameter</b> | <b>Bedeutung</b>   |
|---------------------|--|
| Tslot_Init          | Die Warte-auf-Empfang-Zeit legt fest, wie lange der Sender maximal wartet, um vom angesprochenen Partner eine Antwort zu erhalten. Wird in der Parametergruppe "Leitungskonfiguration" der Einfluss der Leitungs-Komponenten auf die Telegrammlaufzeiten konfiguriert, so müssen diese Anteile zusätzlich berücksichtigt werden. Der Anteil wird auf die vorgegebene Tslot_Init aufaddiert und die Summe als Tslot verwendet.              |
| Max. Tsdr           | Die maximale Protokoll-Bearbeitungszeit legt fest, nach welcher Zeit der antwortende Teilnehmer spätestens geantwortet haben muss.   |
| Min. Tsdr           | Die minimale Protokoll-Bearbeitungszeit legt fest, nach welcher Zeit der antwortende Teilnehmer frühestens antworten darf.   |
| Tset                | Die Auslösezeit ist die Zeit, die zwischen dem Empfang eines Datentelegramms und der Reaktion darauf im Teilnehmer verstreichen darf.  |
| Tqui                | Die Modulator-Ausklingzeit ist die Zeit, die ein sendender Teilnehmer nach Telegrammende für das Umstellen von Senden auf Empfangen benötigt.  |
| GAP-Faktor          | Der GAP-Aktualisierungsfaktor legt fest, nach wie vielen Token-Umläufen ein neu hinzugekommener, aktiver Teilnehmer in den logischen Tokenring aufgenommen werden kann.  |
| Retry Limit         | Mit dem Parameter wird festgelegt, wie viele Versuche (Telegrammwiederholungen) maximal unternommen werden, um einen Teilnehmer zu erreichen.  |
| Tslot               | Die Warte-auf-Empfang-Zeit (slot-time) legt fest, wie lange der Sender maximal wartet, um vom angesprochenen Partner eine Antwort zu erhalten.<br>Wird im Register "Leitungen" der Einfluss der Busphysik-Komponenten auf die Telegrammlaufzeiten eingetragen, so müssen diese Anteile zusätzlich berücksichtigt werden. Der Anteil wird auf die vorgegebene Tslot_Init aufaddiert und die Summe als Tslot verwendet.                      |
| Tid2                | Die Ruhezeit 2 legt fest, nach welcher Zeit ein sendender Teilnehmer nach der Versendung eines nicht quittierten Telegramms frühestens das nächste Telegramm versenden darf.   |
| Trdy                | Die Bereitschaftszeit gibt an, nach welcher Zeit ein sendender Teilnehmer frühestens ein Antworttelegramm empfangen kann.  |
| Tid1                | Die Ruhezeit 1 legt fest, nach welcher Zeit ein sendender Teilnehmer nach dem Empfang einer Antwort frühestens das nächste Telegramm versenden darf.   |
| Ttr                 | Die Soll-Token-Umlaufzeit ist die maximal zur Verfügung gestellte Zeit für einen Token-Umlauf. In dieser Zeit erhalten alle aktiven Teilnehmer (DP-Master etc.) einmal das Senderecht (Token). Die Differenz zwischen der Soll-Token-Umlaufzeit und der tatsächlichen Token-Haltezeit eines Teilnehmers bestimmt, wie viel Zeit den anderen aktiven Teilnehmern (PG, weitere DP-Mastern etc.) für das Senden von Telegrammen übrig bleibt. |

| Busparameter        | Bedeutung   |
|---------------------|---|
| Ttr typisch         | Die typische Datenzykluszeit ist die durchschnittliche Reaktionszeit am Bus, wenn alle projektierten Slaves mit dem DP-Master Daten austauschen. Keiner der Slaves meldet Diagnose und es erfolgt kein zusätzlicher Telegrammverkehr mit PGs oder anderen aktiven Teilnehmern etc. am Bus.  |
| Ansprechüberwachung | Die Ansprechüberwachungszeit wird nur für PROFIBUS-DP-Bussysteme benötigt. Sie legt fest, nach welcher Zeit ein DP-Slave spätestens von seinem DP-Master mit einem neuen Datentelegramm angesprochen werden muss. Geschieht dies nicht, so geht der DP-Slave davon aus, dass der DP-Master ausgefallen ist und setzt seine Ausgänge in einen sicheren Zustand zurück. |

**Siehe auch**

Wissenswertes zu PROFIBUS-Busparametern (Seite 423)

**Busprofile bei PROFIBUS**

**Einführung**

Je nach angeschlossenen Gerätetypen und verwendeten Protokollen am PROFIBUS stehen unterschiedliche Profile zur Verfügung. Die Profile unterscheiden sich hinsichtlich der Einstellmöglichkeiten und hinsichtlich der Berechnung der Busparameter. Im Folgenden werden die Profile erläutert.

**Teilnehmer mit unterschiedlichen Profilen am selben PROFIBUS-Subnetz**

Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Busparameter aller Teilnehmer dieselben Werte haben. Wenn z. B. an einem Subnetz sowohl DP- als auch FMS-Dienste genutzt werden, dann müssen immer die "langsameren" Busparametersätze für alle Teilnehmer eingestellt werden, also auch für DP-Teilnehmer das Profil "Universell (DP/FMS)"!

**Profile und Übertragungsgeschwindigkeiten**

| Profile             | Unterstützte Übertragungsgeschwindigkeiten in kbit/s |
|---------------------|--|
| DP                  | 9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000  |
| Standard            | 9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000  |
| Universell (DP-FMS) | 9,6 19,2 93,75 187,5 500 1500                        |
| Benutzerdefiniert   | 9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000  |



## Bedeutung der Profile

| Profil              | Bedeutung   |
|---------------------|---|
| DP                  | <p>Wählen Sie das Busprofil "DP", wenn nur Geräte an das PROFIBUS-Subnetz angeschlossen sind, welche die Anforderungen der Norm EN 50170 Volume 2/3, Part 8-2 PROFIBUS erfüllen. Die Einstellung der Busparameter ist auf diese Geräte optimiert.</p> <p>Hierzu zählen Geräte mit DP-Master- und DP-Slave-Schnittstellen der SIMATIC S7 sowie Dezentrale Peripheriegeräte anderer Hersteller.</p>   |
| Standard            | <p>Gegenüber dem Profil "DP" bietet das Profil "Standard" zusätzlich die Möglichkeit, Teilnehmer eines anderen Projektes oder Teilnehmer, die hier nicht projektiert wurden, bei der Berechnung der Busparameter zu berücksichtigen. Die Busparameter werden dann nach einem einfachen, nicht optimierten Algorithmus berechnet.</p>  |
| Universell (DP/FMS) | <p>Wählen Sie das Busprofil "Universell (DP/FMS)", wenn einzelne Teilnehmer am PROFIBUS-Subnetz den Dienst PROFIBUS-FMS nutzen.</p> <p>Hierzu gehören z. B. folgende Geräte:</p> <ul style="list-style-type: none"> <li>• CP 343-5 (SIMATIC S7)</li> <li>• PROFIBUS-FMS-Geräte anderer Hersteller</li> </ul> <p>Auch hier besteht wie im Profil "Standard" die Möglichkeit, zusätzliche Teilnehmer bei der Berechnung der Busparameter zu berücksichtigen.</p>  |
| Benutzerdefiniert   | <p>Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Parameter für das Busprofil aufeinander abgestimmt sind. Wählen Sie das Busprofil "Benutzerdefiniert", wenn zum Betrieb eines PROFIBUS-Gerätes keines der übrigen Profile "passt" und Sie die Busparameter Ihrem speziellen Aufbau anpassen müssen. Informationen hierzu sind der Dokumentation des PROFIBUS-Gerätes zu entnehmen.</p> <p>Sie sollten die voreingestellten Werte nur dann ändern, wenn Sie mit der Parametrierung des Busprofils für PROFIBUS vertraut sind.</p> <p>Auch mit diesem Busprofil sind nicht alle theoretisch einstellbaren Kombinationen einbaubar. Die PROFIBUS-Norm schreibt einige Parametergrenzwerte in Abhängigkeit von anderen Parametern fest vor. So ist es z. B. nicht zulässig, dass ein Responder bereits antworten darf (Min T<sub>sdr</sub>), bevor der Initiator das Telegramm auch empfangen kann (Tr<sub>dy</sub>). Diese Normvorgaben werden auch im Profil "Benutzerdefiniert" überprüft.</p> <p>Tipp: Als benutzerdefiniert werden automatisch immer die Busparameter eingestellt, die zuletzt am PROFIBUS-Subnetz gültig waren. Wenn beispielsweise das Busprofil "DP" für das Subnetz gültig war, dann werden die Busparameter für "DP" im Busprofil "Benutzerdefiniert" eingestellt. Ausgehend davon können Sie die Parameter modifizieren.</p> <p>Die Neuberechnung der Überwachungszeiten wird nicht automatisch durchgeführt, um die Gleichheit der eingestellten Werte z. B. zu Projektierungen anderer Projektierungswerkzeuge nicht ohne ihr Wissen zu gefährden.</p> <p>Die Überwachungszeiten T<sub>tr</sub> und Ansprechüberwachung können Sie sich auf Basis der von Ihnen eingestellten Parameter berechnen lassen: Klicken Sie hierzu auf die Schaltfläche "Neu berechnen".</p> |

---

**Hinweis**

Bei allen PROFIBUS-Profilen ist sowohl Mono-Master-Betrieb als auch Multi-Master-Betrieb möglich.

---

## Wissenswertes zur PROFIBUS-Leitungskonfiguration

### Leitungskonfiguration und Busparameter

Angaben zur Leitungskonfiguration können bei der Berechnung der Busparameter berücksichtigt werden. Hierzu müssen Sie das Optionskästchen "Leitungskonfiguration berücksichtigen" in den Eigenschaften des PROFIBUS-Subnetzes aktivieren.

Die weiteren Angaben hängen dann vom Typ der verwendeten Leitung ab; einstellbar sind:

- Kupferleitung
- Lichtwellenleiter / optischer Ring

### PROFIBUS-Leitungskonfiguration, optischer Ring

Die Berechnung ist abhängig von den eingesetzten OLM-Typen. Die Auswahl erfolgt durch Aktivieren der Optionskästchen (Mehrfach-Aktivierung ist möglich, mindestens ein OLM-Typ muss ausgewählt werden):

- OLM/P12
- OLM/G12
- OLM/G12-EEC
- OLM/G12-1300

Folgende Anpassungen der Busparameter werden durchgeführt:

- Projektierung eines nicht vorhandenen Teilnehmers

---

**Hinweis**

**Die folgenden Einschränkungen gelten bei optischen Ringen auch für passive Teilnehmer (z. B. DP-Slaves):**

Es dürfen maximal HSA-1 Teilnehmer am PROFIBUS Netz angeschlossen sein. Bei einer HSA von 126 dürfen die Adressen 126 und 125 nicht benutzt werden. Damit sind maximal 125 Teilnehmer am Bus möglich (Nr. 0 bis 124).

Wenn HSA kleiner bzw. gleich 125 ist, dann dürfen die Adressen HSA und größer nicht benutzt werden. Die Adresse HSA-1 darf dagegen benutzt werden.

---

- Erhöhung des Retry-Werts auf 3
- Einstellung der für den Ring-Betrieb erforderlichen Mindest-Slotzeit

---

**Hinweis**

Für OLM/P12 sind kurze, für OLM/G12 und OLM/G12-EEC mittlere sowie für OLM/G12-1300 hohe Slotzeit-Werte erforderlich. Hieraus ergeben sich hohe Performance für kleine Netzausdehnung bzw. mittlere bis niedrige Performance bei mittlerer bis großer Netzausdehnung.

---

## PROFIBUS-Kommunikationslast

### Kommunikationslast - zusätzliche Netzteilnehmer berücksichtigen

Die Busparameter sind vom Kommunikationsaufkommen der aktiven Netzteilnehmer abhängig. Hierbei gibt es Unterschiede zwischen der zyklischen Kommunikation (DP) und der verbindungsorientierten, azyklischen Kommunikation (S7-Kommunikation, Send/Receive (FDL), FMS). Im Gegensatz zu DP ist die Anzahl und Größe der Kommunikationsaufträge (Kommunikationslast) vom Anwenderprogramm abhängig. Daher ist die Kommunikationslast nicht immer automatisch ermittelbar.

Zur Berechnung der Buszeiten können Sie eine Netzkonfiguration in der Parametergruppe "zusätzliche Netzteilnehmer" festlegen, die von der projektierten Netzkonfiguration abweicht.

### Profil berücksichtigen

Die Netzkonfiguration ist festlegbar für die Profile "Standard", "Universell (DP/FMS)" und "Benutzerdefiniert". Beim Profil "DP" sind die Parameter in der Parametergruppe "zusätzliche Netzteilnehmer" nicht eingebbar.

## Quantifizierung der Kommunikationslast

Zur Berücksichtigung der Kommunikationslast sind folgende Einstellungen möglich:

- Angaben zur Anzahl nicht projektierter Netzteilnehmer;
- Angaben zu der aus den Anwenderprogrammen für FDL- oder S7-Kommunikation resultierenden Kommunikationslast. Hierzu können Sie aus folgenden Stufen wählen:
  - Niedrig  
Typisch für DP, keine größere Datenkommunikation außer DP.
  - Mittel  
Typisch für Mischbetrieb von DP und anderen Kommunikationsdiensten (z. B. S7-Kommunikation), wenn DP hohe Zeitanforderungen hat und bei mittlerem azyklischen Kommunikationsaufkommen.
  - Hoch  
Für Mischbetrieb von DP und anderen Kommunikationsdiensten (z. B. S7-Kommunikation), wenn DP geringe Zeitanforderungen hat und bei hohem azyklischen Kommunikationsaufkommen.

## Netzkonfiguration Industrial Ethernet

### Regeln für die Netzkonfiguration

Die Ethernet-Schnittstellen der Geräte besitzen als IP-Adresse eine Default-Adresse, die von Ihnen geändert werden kann.

### IP-Adresse

Die IP-Parameter sind sichtbar, wenn die kommunikationsfähige Baugruppe das TCP/IP-Protokoll unterstützt. Dies ist in der Regel bei allen Ethernet-Baugruppen der Fall.

Die IP-Adresse besteht aus 4 Dezimalzahlen mit dem Wertebereich 0 bis 255. Die Dezimalzahlen sind durch einen Punkt voneinander getrennt.

Beispiel: 140.80.0.2

Die IP-Adresse setzt sich zusammen aus:

- Adresse des (Sub-) Netzes
- Adresse des Teilnehmers (im allgemeinen auch Host oder Netzknoten genannt)

### Subnetzmaske

Die Subnetzmaske trennt diese beiden Adressen. Sie bestimmt, welcher Teil der IP-Adresse das Netz adressiert und welcher Teil der IP-Adresse den Teilnehmer adressiert.

Die gesetzten Bits der Subnetzmaske bestimmen den Netzteil der IP-Adresse.

Beispiel:

Subnetzmaske: 255.255.0.0 = 11111111.11111111.00000000.00000000

Im Beispiel für die oben genannte IP-Adresse hat die hier gezeigte Subnetzmaske folgende Bedeutung:

Die ersten 2 Bytes der IP-Adresse bestimmen das Subnetz - also 140.80. Die letzten beiden Bytes adressieren den Teilnehmer - also 0.2.

Allgemein gilt:

- Die Netzadresse ergibt sich aus der UND-Verknüpfung von IP-Adresse und Subnetzmaske.
- Die Teilnehmeradresse ergibt sich aus der UND-NICHT-Verknüpfung von IP-Adresse und Subnetzmaske.

### Zusammenhang IP-Adresse und Default-Subnetzmaske

Es gibt eine Vereinbarung hinsichtlich der Zuordnung von IP-Adressbereichen und so genannten "Default-Subnetzmasken". Die erste Dezimalzahl der IP-Adresse (von links) bestimmt den Aufbau der Default-Subnetzmaske hinsichtlich der Anzahl der Werte "1" (binär) wie folgt:

| IP-Adresse (dez.) | IP-Adresse (bin.)    | Adressklasse | Default-Subnetzmaske |
|-------------------|----------------------|--------------|----------------------|
| 0 bis 126         | 0xxxxxxx.xxxxxxxx... | A            | 255.0.0.0            |
| 128 bis 191       | 10xxxxxx.xxxxxxxx... | B            | 255.255.0.0          |
| 192 bis 223       | 110xxxxx.xxxxxxxx... | C            | 255.255.255.0        |

#### Hinweis

##### Wertebereich für erste Dezimalstelle

Für die erste Dezimalzahl der IP-Adresse ist auch ein Wert zwischen 224 und 255 möglich (Adressklasse D etc). Dies ist jedoch nicht empfehlenswert, da für diese Werte keine Adressprüfung erfolgt.

### Weitere Subnetze maskieren

Über die Subnetzmaske können Sie ein Subnetz, das einer der Adressklassen A, B oder C zugeordnet ist, weiter strukturieren und "private" Subnetze bilden, indem Sie weitere niederwertige Stellen der Subnetzmaske auf "1" setzen. Pro jedem auf "1" gesetztem Bit verdoppelt sich die Anzahl der "privaten" Netze und halbiert sich die Anzahl der darin enthaltenen Teilnehmer. Nach außen wirkt das Netzwerk nach wie vor wie ein einzelnes Netzwerk.

Beispiel:

Sie ändern bei einem Subnetz der Adressklasse B (z. B. IP-Adresse 129.80.xxx.xxx) die Default-Subnetzmaske wie folgt:

| Masken               | Dezimal       | Binär                                   |
|----------------------|---------------|---|
| Default-Subnetzmaske | 255.255.0.0   | 11111111.11111111.00000000.<br>00000000 |
| Subnetzmaske         | 255.255.128.0 | 11111111.11111111.10000000.<br>00000000 |

### 8.1 Geräte und Netze konfigurieren

Ergebnis:

Alle Teilnehmer mit Adressen von 129.80.001.xxx bis 129.80.127.xxx befinden sich auf einem Subnetz, alle Teilnehmer mit Adressen von 129.80.128.xxx bis 129.80.255.xxx auf einem anderen Subnetz.

#### Netzübergang (Router)

Die Netzübergänge (Router) haben die Aufgabe, die Subnetze zu verbinden. Wenn ein IP-Datagramm an ein anderes Netzwerk geschickt werden soll, dann muss es zunächst an einen Router vermittelt werden. Damit das möglich ist, müssen Sie für jeden Teilnehmer des Subnetzes die Adresse des Routers eingeben.

Die IP-Adresse eines Teilnehmers im Subnetz und die Adresse des Netzübergangs (Router) dürfen nur an den Stellen unterschiedlich sein, an denen in der Subnetzmaske "0" steht.

#### Netzkonfiguration AS-Interface

Ein AS-Interface besteht aus einem AS-i Master und AS-i Slaves, die durch ein AS-i Subnetz miteinander verbunden sind.

#### Regeln für die AS-i Netzkonfiguration

Sämtliche Teilnehmer eines AS-i Subnetzes müssen eine unterschiedliche AS-i Teilnehmeradresse besitzen.

Erst wenn sämtliche Baugruppen eines Subnetzes unterschiedliche Adressen besitzen und der tatsächliche Aufbau mit der erstellten Netzkonfiguration übereinstimmt, sollten Sie die Einstellungen über das Netz laden.

An einem AS-i Subnetz können ein AS-i Master und bis zu 31 AS-i Slaves betrieben werden.

Zur Konfiguration eines AS-Interface mit AS-i Master und AS-i Slaves finden Sie weitere Informationen im Kapitel AS-Interface sowie in der Gerätedokumentation der AS-i Masterbaugruppen.

### 8.1.3.2 Kommunikation über Verbindungen

#### Arbeiten mit Verbindungen

#### S7-Verbindung

#### Einführung zum Projektieren von Verbindungen

##### Definition

Eine Verbindung definiert eine logische Zuordnung zweier Kommunikationspartner zur Ausführung von Kommunikationsdiensten. Eine Verbindung legt Folgendes fest:

- Beteiligte Kommunikationspartner
- Typ der Verbindung (z. B. S7-Verbindung)
- Spezielle Eigenschaften (z. B. ob eine Verbindung permanent aufgebaut bleibt oder ob sie im Anwenderprogramm dynamisch auf- und abgebaut wird und ob Betriebszustandsmeldungen gesendet werden sollen)
- Verbindungsweg

##### Wissenswertes zur Verbindungsprojektierung

Bei der Verbindungsprojektierung wird bei einer S7-Verbindung ein lokaler Verbindungsname als eindeutige lokale Kennung vergeben.

In der Netzsicht wird neben dem Register "Netzübersicht" ein Register "Verbindungen" angezeigt. In diesem Register befindet sich die Verbindungstabelle. Eine Zeile in dieser Verbindungstabelle repräsentiert eine projektierte Verbindung aus Sicht des lokalen Kommunikationspartners mit ihren Eigenschaften, z. B. zwischen zwei S7-1200 CPUs.

##### Wissenswertes zum Verbrauch von Verbindungsressourcen

##### Einführung

Jede Verbindung benötigt auf den beteiligten Geräten Verbindungsressourcen für den Endpunkt bzw. für den Übergangspunkt. Die Anzahl der Verbindungsressourcen ist gerätespezifisch.

Sind alle Verbindungsressourcen eines Kommunikationspartners belegt, so kann keine neue Verbindung aufgebaut werden. Sie erkennen das daran, dass eine neu angelegte Verbindung in der Verbindungstabelle rot hinterlegt ist. Die Konfiguration ist dann inkonsistent und kann nicht übersetzt werden.

## **S7-Verbindungen**

Bei S7-Verbindungen über die PN-Schnittstelle wird bei der S7-1200 CPU pro S7-Verbindung eine Verbindungsressource für den Endpunkt belegt. Für den Verbindungspartner wird ebenfalls eine Verbindungsressource benötigt.

Sie finden eine Übersicht der verfügbaren und belegten Verbindungsressourcen bei selektierter S7-1200 CPU im Inspektorfenster unter "Eigenschaften > Verbindungsressourcen".



## Ansichten mit Informationen zu den projektierten Verbindungen

Über die nachfolgend beschriebenen Ansichten haben Sie umfassenden Zugriff auf sämtliche Informationen und Funktionen zum Projektieren und Prüfen von Kommunikationsverbindungen.

- Verbindungsanzeige in der Netzsicht
- Verbindungstabelle
- Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

The screenshot displays the WinCC Basic V12.0 SP1 interface for configuring network connections. It is divided into three main sections:

**1. Network Overview (Netzübersicht):** Shows a graphical representation of two PLCs, PLC\_1 and PLC\_2, both with CPU 1215C. They are connected via a connection labeled 'S7\_Verbindung\_1'. The interface includes tabs for 'Topologiesicht', 'Netzsicht', and 'Gerätesicht', and a search bar for 'Hervorgehoben: Verbindung'.

**2. Connection Table (Verbindungen):** A table listing the configured connections. The table has columns for 'Lokaler Verbindungsname', 'Lokaler Endpunkt', 'Lokale ID (Hex)', 'Partner-ID (Hex)', 'Partner', and 'Verbindungstyp'.

| Lokaler Verbindungsname | Lokaler Endpunkt | Lokale ID (Hex) | Partner-ID (Hex) | Partner | Verbindungstyp |
|-------------------------|------------------|-----------------|------------------|---------|----------------|
| S7_Verbindung_1         | PLC_1            | 100             | 100              | PLC_2   | S7-Verbindung  |
| S7_Verbindung_1         | PLC_2            | 100             | 100              | PLC_1   | S7-Verbindung  |

**3. Properties Dialog (Eigenschaften):** The 'S7\_Verbindung\_1 [S7-Verbindung]' dialog is open, showing the 'Allgemein' (General) tab. It details the connection name and the path between the local and partner PLCs.

**Verbindung:** Name: S7\_Verbindung\_1

**Verbindungsweg:**

| Parameter         | Lokal                                    | Partner                                  |
|-------------------|--|--|
| Endpunkt          | PLC_1                                    | PLC_2                                    |
| Schnittstelle     | PLC_1, PROFINET-Schnittstelle_1[X1 : PN( | PLC_2, PROFINET-Schnittstelle_1[X1 : PN( |
| Schnittstellentyp | Ethernet                                 | Ethernet                                 |
| Subnetz           | PN/IE_1                                  | PN/IE_1                                  |
| Adresse           | 192.168.0.1                              | 192.168.0.2                              |

A 'Finde Verbindungsweg' button is located at the bottom right of the dialog.

## Vorteile

Die in diesen Ansichten dargestellten Informationen sind in Bezug auf augenblickliche Benutzeraktionen immer aktuell. Das bedeutet:

- Die Verbindungstabelle zeigt alle angelegten Verbindungen.
- Wenn Sie eine Verbindung in der Verbindungstabelle angewählt haben, gilt:
  - Bei aktivem Verbindungsmodus wird der Verbindungsweg in der Netzsicht grafisch hervorgehoben.
  - Das Register "Eigenschaften" im Inspektorfenster zeigt die Parameter dieser Verbindung.

## Die Verbindungstabelle

Die Verbindungstabelle bietet folgende Funktionen:

- Auflistung aller Verbindungen im Projekt
- Auswahl einer Verbindung und damit verbundene Anzeige der Verbindung in der Netzsicht (bei aktivem Verbindungsmodus)
- Ändern von Verbindungspartnern
- Anzeige von Zustandsinformationen

## Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

Der Eigenschaften-Dialog hat folgende Bedeutung:

- Anzeige für Verbindungsparameter
- Anzeige des Verbindungswegs
- Nachträgliches Spezifizieren von Verbindungen über die Schaltfläche "Finde Verbindungsweg"

## Anlegen einer neuen Verbindung

### Anlegen einer Verbindung - Alternativen

Sie haben folgende Möglichkeiten, eine Verbindung in der Netzsicht anzulegen:

- Grafische Verbindungsprojektierung
- Dialoggeführte Verbindungsprojektierung

Die einzelnen Schritte hierzu finden Sie in den Folgekapiteln.

## Voraussetzung und Ergebnis

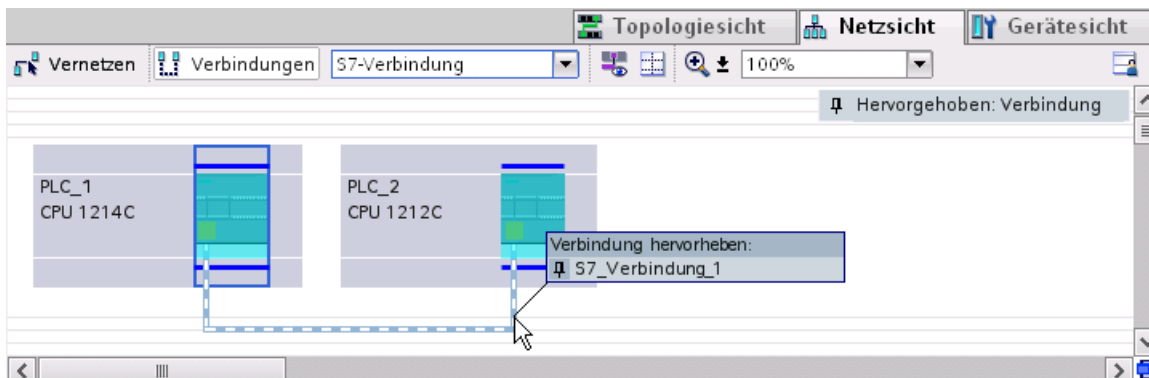
Sie haben in der Netzsicht die Geräte eingefügt, zwischen denen die Verbindungen projektiert werden sollen.

## Verbindung spezifizieren

Wenn beide Partner für den gewählten Verbindungstyp am selben Netz vernetzt sind, legen Sie durch die grafische oder dialoggeführte Auswahl beider Verbindungspartner eine voll-spezifizierte Verbindung an.

Diese wird automatisch in die Verbindungstabelle der S7-1200 CPU eingetragen. Für diese Verbindung wird ein lokaler Verbindungsname vergeben.

Die folgende Darstellung zeigt eine projektierte Verbindung bei vernetztem Gerät:



## Neue Verbindung grafisch anlegen

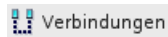
## Verbindungen grafisch projektieren

Bei der grafischen Verbindungsprojektierung wird der Verbindungsweg, sofern Schnittstellen und Ressourcen vorhanden sind, automatisch bestimmt. Wählen Sie in der aktuellen Konfiguration die zu verbindenden Geräte aus.

### Verbindungsweg automatisch ermitteln

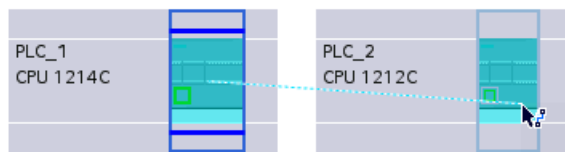
Um eine Verbindung grafisch anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Verbindungen".



Damit ist der Verbindungsmodus aktiviert. Sie können jetzt den gewünschten Verbindungstyp auswählen. Sie erkennen dies an folgendem Verhalten:  
In der Netzsicht sind die für den gewählten Verbindungstyp in Ihrem Projekt verwendbaren Geräte farblich hervorgehoben.

2. Ziehen Sie den Mauscursor bei gedrückter Maustaste von dem Gerät, von dem eine Verbindung ausgehen soll, auf das Gerät, bei dem die Verbindung enden soll.



3. Lassen die auf dem Zielgerät die Maustaste los, um die Verbindung zwischen den beiden Geräten zu erstellen.

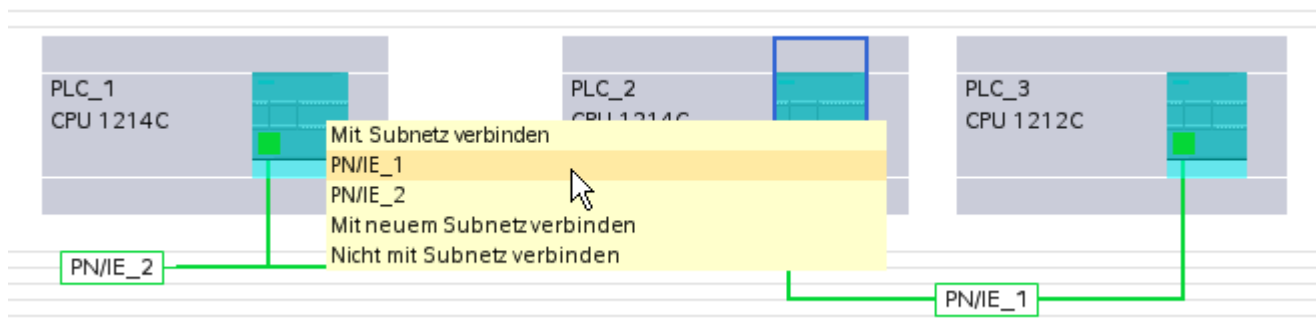
### Ergebnis

- Eine spezifizierte Verbindung ist angelegt.
- Der Verbindungsweg ist hervorgehoben dargestellt.
- Die Verbindung ist in der Verbindungstabelle eingetragen.

### Verbindung bei fehlender oder nicht eindeutiger Netzzuordnung projizieren

Fehlende Vernetzungen werden soweit möglich beim Anlegen einer Verbindung automatisch angelegt. Falls eine eindeutige Netzzuordnung nicht möglich ist, erfolgt eine Abfrage beim Abschluss der Verbindungsprojektierung. Hierbei werden die bestehenden Subnetze zur Auswahl angeboten.

Beispiel in folgendem Bild: Beim Anlegen einer Verbindung zwischen den noch nicht vernetzten Stationen PLC\_1 und PLC\_2 erfolgt eine Abfrage.



## Neue Verbindung dialoggeführt anlegen

## Verbindungen dialoggeführt projektieren

Bestimmen Sie das lokale Gerät und seinen Verbindungspartner.

## Vorgehen

Um eine Verbindung dialoggeführt anzulegen, gehen Sie folgendermaßen vor:

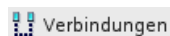
1. Wählen Sie im Kontextmenü eines Verbindungspartners, für den eine Verbindung angelegt werden soll, den Befehl "Neue Verbindung hinzufügen".  
Der Dialog "Neue Verbindung erstellen" wird geöffnet.
2. Wählen Sie den Partner-Endpunkt.  
Im rechten Teil des Dialogs wird passend zum markierten Endpunkt ein möglicher Verbindungsweg angezeigt, soweit vorhanden. Unvollständige Wege wie z. B. bei einer Nicht spezifizierten CPU werden durch ein Ausrufezeichen auf rotem Grund gekennzeichnet.
3. Um die projektierte Verbindung zu übernehmen und weitere Verbindungen zu anderen Endpunkten zu projektieren, klicken Sie auf "Hinzufügen".  
Um die den Dialog zu beenden klicken Sie auf "OK".

## Arbeiten in der Netzsicht

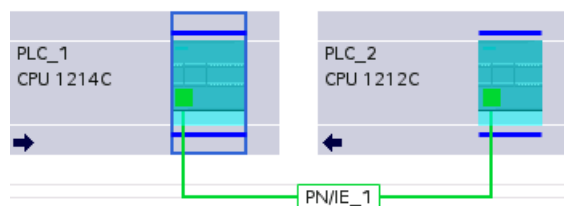
## Verbindungsweg und -partner in der Netzsicht hervorheben

Um die Verbindungspartner für alle oder für bestimmte Verbindungstypen in der Netzsicht anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Verbindungen".



2. Wählen Sie im Kontextmenü der S7-CPU, deren Verbindungspartner Sie in der Netzsicht anzeigen möchten, den Befehl "Verbindungspartner hervorheben".
3. Wählen Sie im Folgemenu "Alle Verbindungspartner".  
Das lokale Gerät und die CPUs der Zielgeräte sind markiert. Der lokale Verbindungspartner zeigt einen Pfeil nach rechts und die entfernten Verbindungspartner zeigen einen Pfeil nach links.
4. Öffnen Sie eine Liste mit Angabe der Zielgeräte, indem Sie den Pfeil des lokalen Gerätes anwählen. Diese Zusatzfunktion ist bei komplexen Netzkonfigurationen nützlich, bei denen nicht alle Geräte sichtbar sind.



---

### Hinweis

Sie können einen der Verbindungspartner, der im aktuellen Anzeigebereich der Netzsicht nicht sichtbar ist, anzeigen. Klicken Sie in der aufgeblendeten Liste auf den Kommunikationspartner. Ergebnis: Die Anzeige wird so verschoben, dass der Verbindungspartner sichtbar wird.

---

## Arbeiten mit der Verbindungstabelle

### Grundfunktionen für Tabellen

Die Verbindungstabelle unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spaltenbreite verändern
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen.

### Spaltenbreite ändern

Um die Breite einer Spalte dem Inhalt anzupassen, so dass alle Texte in den Zeilen lesbar sind, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Verbindungstabelle rechts neben der zu optimierenden Spalte, bis der Mauszeiger die Form von zwei parallelen Linien annimmt (so, als wollten Sie die Breite der Spalte durch Ziehen mit dem Mauszeiger verändern).
2. Doppelklicken Sie auf diese Stelle.

oder


1. Öffnen Sie das Kontextmenü auf der Kopfzeile der Tabelle.
2. Klicken Sie auf
  - "Spaltenbreite optimieren" oder
  - "Breite aller Spalten optimieren".

Bei zu schmal eingestellten Spalten wird der komplette Inhalt einzelner Felder eingeblendet, wenn Sie den Mauszeiger über dem betreffenden Feld kurze Zeit stehen lassen.

### Spalten ein-/ausblenden

Über das Kontextmenü der Kopfzeilen der Verbindungstabelle können Sie die Anzeige der verschiedenen Tabellenspalten steuern. Über den Kontextmenüeintrag "Spalten ein-/ausblenden" sehen Sie eine Übersicht der verfügbaren Spalten. Mit den Optionskästchen steuern Sie die Sichtbarkeit der Spalten.

Wenn Sie die Anordnung, Breite und Sichtbarkeit der Tabellenspalten speichern möchten, klicken Sie in der Netzsicht rechts oben auf die Funktion "Anordnung merken".

| Symbol  | Bedeutung  |
|---|--|
|  | Anordnung merken<br>Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert. |

### Mit Cursortasten in der Verbindungstabelle navigieren

Mit den Cursortasten NACH-OBEN und NACH-UNTEN können Sie eine Verbindung in der Verbindungstabelle auswählen; die ausgewählte Verbindung ist markiert und wird in der Netzsicht hervorgehoben dargestellt.

### Eigenschaften der Verbindung ändern

Sie können einige der in der Verbindungstabelle angezeigten Parameter direkt bearbeiten. Der Name der Verbindung beispielsweise nur in der Verbindungstabelle geändert werden.

### Verbindungspartner ändern

Sie können den Verbindungspartner einer Verbindung wie folgt ändern:

1. Selektieren Sie die Verbindung.
2. Wählen Sie den neuen Verbindungspartner über die aktivierte Klappliste in der Spalte "Partner".

### Verbindungen löschen

Sie können projizierte Verbindungen über die Netzsicht oder die Verbindungstabelle löschen.

In der Netzsicht können Sie jeweils eine hervorgehobene Verbindung löschen. In der Verbindungstabelle können Sie eine oder mehrere Verbindungen löschen.

### Vorgehen

Um eine Verbindung zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie die zu löschende Verbindung aus:
  - In der Netzsicht: Markieren Sie die zu löschende Verbindung.
  - In der Verbindungstabelle: Markieren Sie die Zeilen der zu löschenden Verbindungen (Mehrfachauswahl möglich).
2. Öffnen Sie mit der rechten Maustaste das Kontextmenü.
3. Wählen Sie den Befehl "Löschen".

### Ergebnis

Die markierte Verbindung wird vollständig gelöscht.

## Kopieren von Verbindungen

### Einführung

Verbindungen werden nicht einzeln kopiert, sondern immer im Kontext mit dem Projekt bzw. mit dem Gerät.

Sie können kopieren:

- gesamte Projekte
- eine oder mehrere Geräte innerhalb eines Projekts oder über Projektgrenzen hinweg

### Projekt kopieren

Wenn Sie ein Projekt kopieren, dann werden sämtliche projektierten Verbindungen mitkopiert. Es sind keinerlei Einstellungen für die kopierten Verbindungen notwendig, da die Verbindungen konsistent bleiben.

### Geräte kopieren

Wenn Sie Geräte kopieren, für die Verbindungen projektiert wurden, dann werden die Verbindungen ebenfalls kopiert. Um den Verbindungsweg zu vervollständigen, müssen Sie noch die Vernetzung nachziehen.

Eine S7-1200 CPU mit einer V1.0 Firmware ist lediglich Server für Verbindungen und hat selber keine Verbindungsprojektion. Daher werden beim Kopieren einer S7-1200 CPU mit einer V1.0 Firmware auch keine Verbindungen mitkopiert.

## Inkonsistente Verbindungen - Verbindungen ohne Zuordnung

Bei einer inkonsistenten Verbindung ist die Struktur der Verbindungsdaten zerstört oder die Verbindung ist im Kontext mit dem Projekt nicht funktionsfähig.

Inkonsistente Verbindungen können nicht übersetzt und geladen werden - der Betrieb mit einer solchen Verbindung ist nicht möglich.

In der Verbindungstabelle sind inkonsistente Verbindungen durch die Farbe Rot zu erkennen.

## Mögliche Ursachen für inkonsistente Verbindungen

- Löschen oder Änderungen der Hardware-Konfiguration
- Fehlende Vernetzungen von Schnittstellen im Projekt, die für eine Verbindung notwendig sind.
- Überschreitung von Verbindungsressourcen
- Verbindungen zu einem un spezifizierten Verbindungspartner ohne Angabe der Partneradresse.

Detailinformationen über die Ursache der Inkonsistenz finden Sie im Register "Übersetzen" nach dem Übersetzen (Bearbeiten > Übersetzen).



## Abhilfen

Um einem bestehenden offenen Verbindungsweg einen geschlossenen Verbindungsweg zuzuweisen, ergänzen Sie die Gerätekonfiguration so, dass die für den Verbindungstyp benötigten Schnittstellen bei beiden Partnern vorhanden sind. Im Inspektorfenster unter "Eigenschaften > Allgemein > Schnittstelle" kann mit der Schaltfläche "Finde Verbindungsweg" eine Verbindung zu einem vorhandenem Partner erstellt werden.

## S7-Verbindung Allgemeine Einstellungen

### Allgemeine Verbindungsparameter

In der Parametergruppe "Allgemein" unter den Eigenschaften der Verbindung werden allgemeine Verbindungsparameter angezeigt, die den lokalen Verbindungsendpunkt identifizieren.

Sie haben hier die Möglichkeit, den Verbindungsweg zuzuordnen und die Verbindungspartner vollständig zu spezifizieren.

### Lokale ID

Hier wird die lokale ID der Baugruppe angezeigt, von der aus die Verbindung betrachtet wird (lokaler Partner). Sie können die lokale ID ändern. Das ist dann notwendig, wenn Sie bereits Kommunikations-Funktionsbausteine programmiert haben und Sie die dort angegebene Lokale ID für die Verbindung verwenden wollen.

### Besondere Verbindungseigenschaften

Anzeige der Verbindungseigenschaften (in Abhängigkeit der verwendeten Komponenten änderbar):

- Einseitig  
Einseitig bedeutet, dass der Verbindungspartner Server für diese Verbindung ist und nicht aktiv senden oder empfangen kann.
- Aktiver Verbindungsaufbau  
Bei einer einseitigen Verbindung, z. B. mit einer S7-1200 CPU des Firmwarestands V1.0, kann nur ein Verbindungspartner für den aktiven Verbindungsaufbau zur Verfügung stehen. Bei einer zweiseitigen Verbindung können Sie einstellen, welcher Verbindungspartner den aktiven Part übernehmen soll.
- Betriebszustandsmeldungen senden  
Zeigt an, ob der lokale Partner Betriebszustandsmeldungen an den Verbindungspartner sendet.

### Adressdetails

Anzeige der Adressdetails der S7-Verbindung. Bei einem unspezifizierten Partner können die Werte für Baugruppenträger und Steckplatz geändert werden. Alle anderen Werte werden aus der aktuellen Konfiguration ermittelt und sind nicht änderbar.

## S7-Verbindung Adressdetails

### Bedeutung

In den Adressdetails sehen Sie die Endpunkte der Verbindung und können diese über die Angabe von Baugruppenträger und Steckplatz lokalisieren.

Wenn eine Verbindung aufgebaut wird, dann werden verbindungs-spezifische Ressourcen einer Baugruppe dieser Verbindung fest zugeordnet. Diese Zuordnung setzt voraus, daß die Verbindungsressource adressiert werden kann. Der TSAP (Transport Service Access Point) ist quasi die Adresse der Ressource, die unter anderem mit Hilfe der Verbindungsressource oder bei S7-1200 CPUs ab Firmware V2.0 mit der SIMATIC-ACC (SIMATIC Application Controlled Communication) gebildet wird.

### Aufbau des TSAPs für S7-1200

- Für S7-1200 CPU mit Firmware ab V2.0:  
"SIMATIC-ACC"<nnn><mm>  
nnn = lokale ID  
mm = freier Wert
- Für S7-1200 CPU mit Firmware V1.0:  
<xx>.<y>  
xx = Nummer der Verbindungsressource  
y = Baugruppenträgernummer  
z = Steckplatznummer

### TSAP-Aufbau in Abhängigkeit des Partners

Der Aufbau des TSAPs ist bei S7-1200 CPUs abhängig von der jeweiligen Firmware und von dem fernen Verbindungspartner. Wenn eine S7-1200 CPU mit einer S7-300/400 CPU verbunden wird, verwendet auch eine S7-1200 CPU den TSAP-Aufbau mit Einbeziehung der Verbindungsressource.

Siehe folgende Beispiele für TSAPs verschiedener Verbindungsprojektierungen

- Verbindung zwischen zwei S7-1200 CPUs (beide Firmware V2.0):
  - S7-1200 CPU "A" mit Firmware V2.0 und lokaler ID 100:  
TSAP: SIMATIC-ACC10001
  - S7-1200 CPU "B" mit Firmware V2.0 und lokaler ID 5AE:  
TSAP: SIMATIC-ACC5AE01
- Verbindung zwischen zwei S7-1200 CPUs (Firmware V2.0 und V1.0):
  - S7-1200 CPU mit Firmware V2.0 und lokaler ID 1FF:  
TSAP: SIMATIC-ACC1FF01
  - S7-1200 CPU mit Firmware V1.0 (Baugruppenträger 0, Steckplatz 1, Verbindungsressource 03):  
TSAP: 03.01
- Verbindung zwischen S7-1200 CPU (Firmware V2.0) und S7-300/400 CPU:
  - S7-1200 CPU mit Firmware V2.0 (Baugruppenträger 0, Steckplatz 1, Verbindungsressource 12):  
TSAP: 12.01
  - S7-300/400 CPU (Baugruppenträger 0, Steckplatz 2, Verbindungsressource 11):  
TSAP: 11.02

## S7-Verbindungen über CM/CP

### Einleitung

S7-1200 CPUs mit einem Firmwarestand ab V2.0 unterstützen einseitige und zweiseitige S7-Verbindungen über CM/CP-Schnittstellen. Dadurch wird die Anzahl der für S7-Verbindungen nutzbaren Ethernet-Ports und Netze erhöht. Obwohl die Verbindung dann über den CM/CP geführt wird, ist die zugehörige S7-1200 CPU ein Endpunkt der Verbindung. Der andere Endpunkt kann bei zweiseitigen Verbindungen ein beliebiges anderes Gerät sein. Dieses andere Gerät muss auch S7-Verbindungen unterstützen.

### Datenvolumen und Mengengerüst

Die Anzahl Kommunikationsverbindungen, die über CM/CP unterstützt werden, entnehmen Sie bitte dem Gerätehandbuch, das jedem CM/CP beiliegt. Durch Hinzunahme weiterer CM/CPs kann die Anzahl der Verbindungen pro Gerät weiter erhöht werden.

Sind in einem Gerät mehrere CM/CPs eingebaut, so wird bei Überschreitung dieser Grenze automatisch auf den nächsten CP umgeschaltet. Bei Bedarf ordnen Sie Verbindungen über die Wegewahl gezielt zu.

---

#### Hinweis

Der Datentransfer > 240 Byte wird von den aktuellen CPs unterstützt.

CPs mit älterem Ausgabestand unterstützen den Datentransfer mit einer Datenlänge von bis zu 240 Byte.

Beachten Sie hierzu bitte die Angaben im Gerätehandbuch des Ethernet-CP.

---

### Aufgaben des Ethernet-CM/CP im Online-Betrieb

Der Ethernet-CM/CP übernimmt beim Datentransfer über eine Verbindung folgende Aufgaben:

- Empfangen  
Empfangen von Daten vom Ethernet und Weitergabe an den Anwender-Datenbereich in der CPU.
- Senden  
Übernehmen von Daten aus dem Anwender-Datenbereich der CPU und Senden der Daten über Ethernet.

Der Verbindungsaufbau erfolgt automatisch, sobald der Partner erreichbar ist.

### HMI-Verbindung

#### Einführung zum Projektieren von Verbindungen

#### Definition

Eine Verbindung definiert eine logische Zuordnung zweier Kommunikationspartner zur Ausführung von Kommunikationsdiensten. Eine Verbindung legt Folgendes fest:

- Beteiligte Kommunikationspartner
- Typ der Verbindung (z. B. HMI-Verbindung)
- Spezielle Eigenschaften (z. B. ob eine Verbindung permanent aufgebaut bleibt oder ob sie im Anwenderprogramm dynamisch auf- und abgebaut wird und ob Betriebszustandsmeldungen gesendet werden sollen)
- Verbindungsweg

#### Wissenswertes zur Verbindungsprojektierung

Bei der Verbindungsprojektierung wird bei einer HMI-Verbindung ein lokaler Verbindungsname als eindeutige lokale Kennung vergeben.

In der Netzsicht wird neben dem Register "Netzübersicht" ein Register "Verbindungen" angezeigt. In diesem Register befindet sich die Verbindungstabelle. Eine Zeile in dieser

Verbindungstabelle repräsentiert eine projektierte Verbindung, z. B. zwischen einem HMI-Gerät und einer PLC, mit ihren Eigenschaften.

## Wissenswertes zum Verbrauch von Verbindungsressourcen

### Einführung

Jede Verbindung benötigt auf den beteiligten Geräten Verbindungsressourcen für den Endpunkt bzw. für den Übergangspunkt. Die Anzahl der Verbindungsressourcen ist gerätespezifisch.

Sind alle Verbindungsressourcen eines Kommunikationspartners belegt, so kann keine neue Verbindung aufgebaut werden. Sie erkennen das daran, dass eine neu angelegte Verbindung in der Verbindungstabelle rot hinterlegt ist. Die Konfiguration ist dann inkonsistent und kann nicht übersetzt werden.

### HMI-Verbindungen

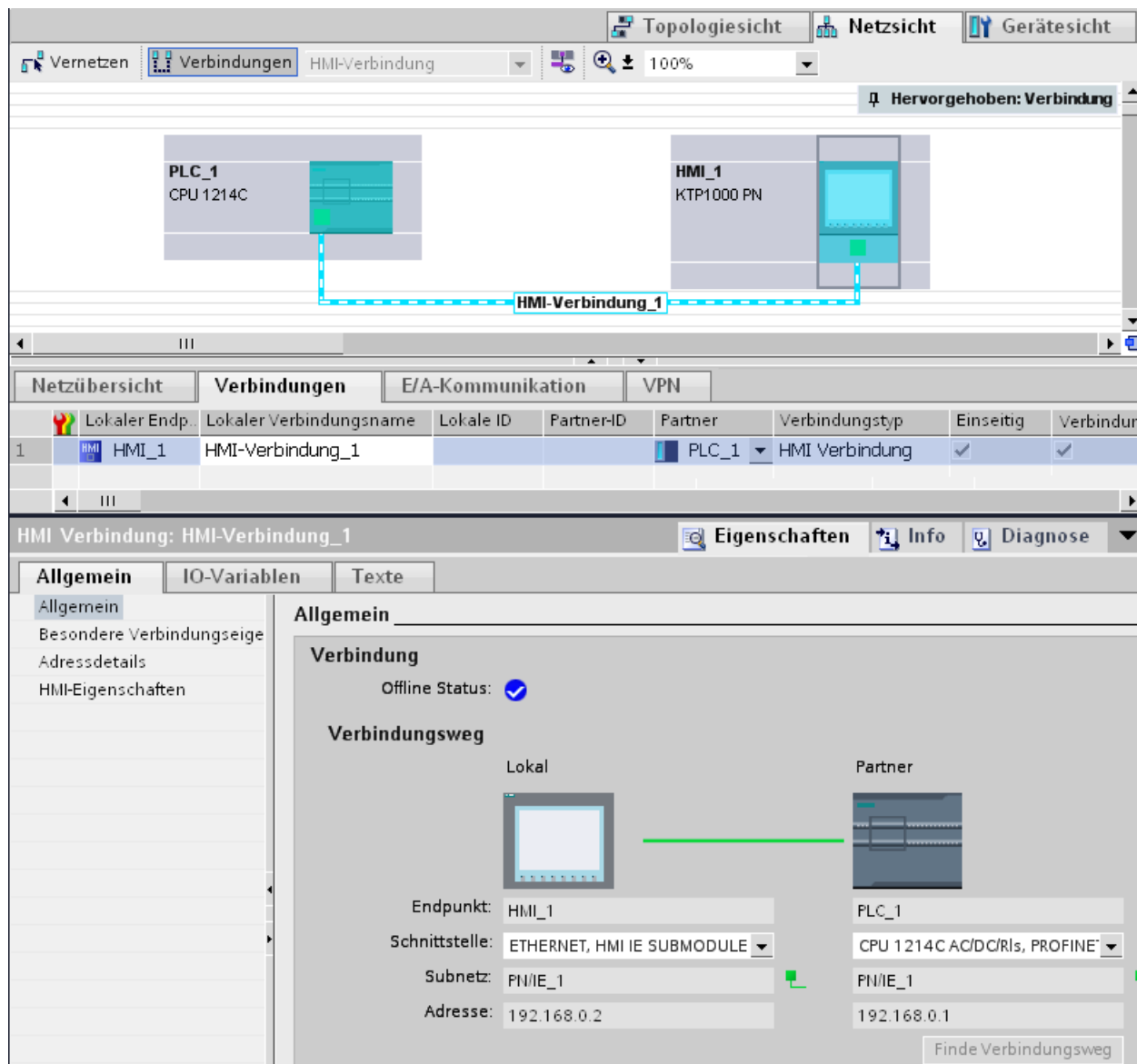
Bei HMI-Verbindungen über die **integrierte** PN-Schnittstelle wird beim HMI-Gerät pro HMI-Verbindung eine Verbindungsressource für den Endpunkt belegt.

Für den Verbindungspartner (PLC) wird ebenfalls eine Verbindungsressource benötigt.

### Ansichten mit Informationen zu den projektierten Verbindungen

Über die nachfolgend beschriebenen Ansichten haben Sie umfassenden Zugriff auf sämtliche Informationen und Funktionen zum Projektieren und Prüfen von Kommunikationsverbindungen.

- Verbindungsanzeige in der Netzsicht
- Verbindungstabelle
- Register "Eigenschaften" zu einer Verbindung im Inspektorfenster



## Vorteile

Die in diesen Ansichten dargestellten Informationen sind in Bezug auf augenblickliche Benutzeraktionen immer aktuell. Das bedeutet:

- Die Verbindungstabelle zeigt alle angelegten Verbindungen.
- Wenn Sie eine Verbindung in der Verbindungstabelle ausgewählt haben, gilt:
  - Sie sehen in der Netzsicht den Verbindungsweg grafisch.
  - Das Register "Eigenschaften" im Inspektorfenster zeigt die Parameter dieser Verbindung.

## Die Verbindungstabelle

Die Verbindungstabelle bietet folgende Funktionen:

- Auflistung aller Verbindungen im Projekt
- Auswahl einer Verbindung und damit verbundene Anzeige der Verbindung in der Netzsicht
- Ändern von Verbindungspartnern
- Anzeige von Zustandsinformationen

## Register "Eigenschaften" zu einer Verbindung im Inspektorfenster

Der Eigenschaften-Dialog hat folgende Bedeutung:

- Anzeige für Verbindungsparameter
- Anzeige des Verbindungswegs
- Nachträgliches Spezifizieren von Verbindungen über die Schaltfläche "Finde Verbindungsweg"

## Anlegen einer neuen Verbindung

### Anlegen einer Verbindung - Alternativen

Sie haben folgende Möglichkeiten, eine Verbindung in der Netzsicht anzulegen:

- Grafische Verbindungsprojektierung
- Dialoggeführte Verbindungsprojektierung

Die einzelnen Schritte hierzu finden Sie in den Folgekapiteln.

## Voraussetzung und Ergebnis

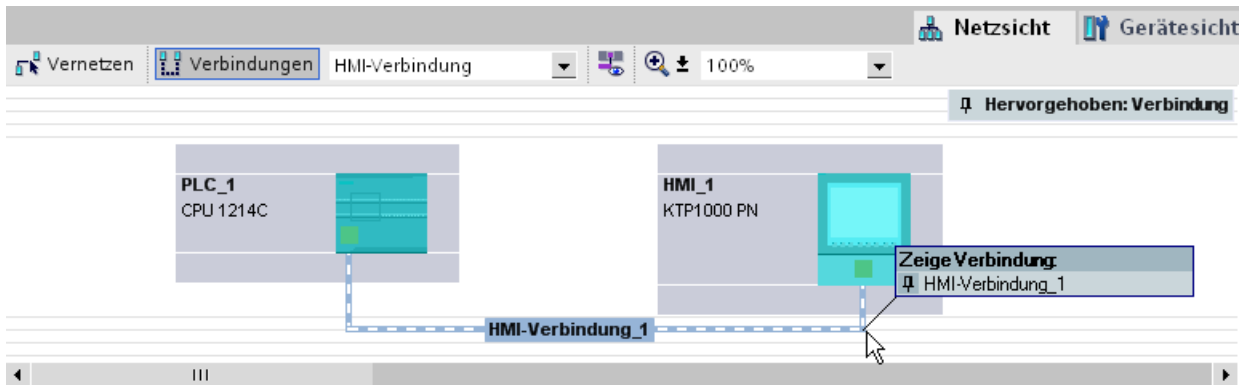
Sie haben in der Netzsicht die Geräte mit CPUs und HMI-Geräten angelegt, zwischen denen die Verbindungen projektiert werden sollen.

### Verbindung spezifizieren

Wenn beide Partner für den gewählten Verbindungstyp am selben Netz vernetzt sind, legen Sie durch die grafische oder dialoggeführte Auswahl beider Verbindungspartner eine voll-spezifizierte Verbindung an.

Diese wird automatisch in die Verbindungstabelle des HMI-Gerätes eingetragen. Für diese Verbindung wird ein lokaler Verbindungsname vergeben.

Die folgende Darstellung zeigt eine projektierte Verbindung bei vernetztem Gerät:



### Neue Verbindung grafisch anlegen

### Verbindungen grafisch projektieren

Bei der grafischen Verbindungsprojektierung werden Sie bei Bedarf vom System aufgefordert, den Verbindungsweg zu bestimmen. Wählen Sie in der aktuellen Konfiguration die zu verbindenden Geräte aus.



## Verbindungsweg automatisch ermitteln

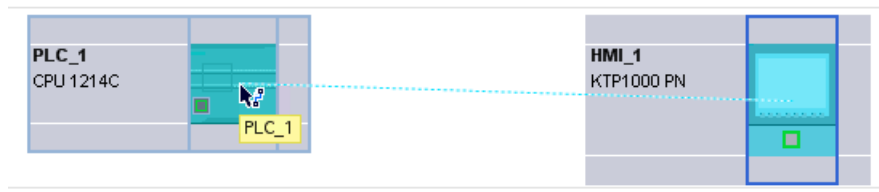
Um eine Verbindung grafisch anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Verbindungen".



Damit ist der Verbindungsmodus für den von Ihnen gewählten Verbindungstyp aktiviert. Sie erkennen dies an folgendem Verhalten: In der Netzsicht sind die für den gewählten Verbindungstyp in Ihrem Projekt verwendbaren Geräte farblich hervorgehoben.

2. Ziehen Sie den Mauscursor bei gedrückter Maustaste von dem Gerät, von dem eine Verbindung ausgehen soll, auf das Gerät, bei dem die Verbindung enden soll.



3. Lassen die auf dem Zielgerät die Maustaste los, um die Verbindung zwischen den beiden Geräten zu erstellen.

## Ergebnis

- Eine spezifizierte Verbindung ist angelegt.
- Der Verbindungsweg ist hervorgehoben dargestellt.
- Die Verbindung ist in der Verbindungstabelle eingetragen.

## Neue Verbindung dialoggeführt anlegen

### Verbindungen dialoggeführt projektieren

Bestimmen Sie das lokale Gerät und seinen Verbindungspartner.

### Vorgehen

Um eine Verbindung dialoggeführt anzulegen, gehen Sie folgendermaßen vor:

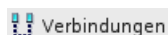
1. Wählen Sie im Kontextmenü eines Verbindungspartners, für den eine Verbindung angelegt werden soll, den Befehl "Neue Verbindung erstellen".  
Der Dialog "Neue Verbindung erstellen" wird geöffnet.
2. Wählen Sie den Partner-Endpunkt.  
Im rechten Teil des Dialogs wird passend zum markierten Endpunkt ein möglicher Verbindungsweg angezeigt, soweit vorhanden. Unvollständige Wege wie z. B. bei einer Nicht spezifizierten CPU werden durch ein Ausrufezeichen auf rotem Grund gekennzeichnet.
3. Um die den Dialog zu beenden klicken Sie auf "OK".  
Um die projektierte Verbindung zu übernehmen und weitere Verbindungen zu anderen Endpunkten zu projektieren, klicken Sie auf "Übernehmen".

### Arbeiten in der Netzsicht

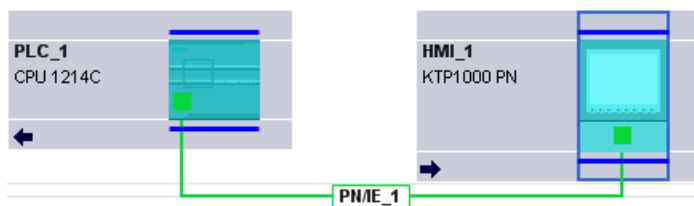
#### Verbindungsweg und -partner in der Netzsicht hervorheben

Um die Verbindungspartner für alle oder für bestimmte Verbindungstypen in der Netzsicht anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Verbindungen".



2. Wählen Sie im Kontextmenü des HMI-Gerätes, deren Verbindungspartner Sie in der Netzsicht anzeigen möchten, den Befehl "Verbindungspartner hervorheben".
3. Wählen Sie im Folgemenu "Alle Verbindungspartner".  
Das lokale Gerät und die CPUs der Zielgeräte sind markiert. Der lokale Verbindungspartner zeigt einen Pfeil nach rechts und die entfernten Verbindungspartner zeigen einen Pfeil nach links.
4. Öffnen Sie eine Liste mit Angabe der Zielgeräte, indem Sie den Pfeil des lokalen Gerätes anwählen. Diese Zusatzfunktion ist bei komplexen Netzkonfigurationen nützlich, bei denen nicht alle Geräte sichtbar sind.



---

### **Hinweis**

Sie können einen der Verbindungspartner, der im aktuellen Anzeigebereich der Netzsicht nicht sichtbar ist, anzeigen. Klicken Sie in der aufgeblendeten Liste auf den Kommunikationspartner. Ergebnis: Die Anzeige wird so verschoben, dass der Verbindungspartner sichtbar wird.

---

### **Siehe auch**

Neue Verbindung grafisch anlegen (Seite 450)

### **Arbeiten mit der Verbindungstabelle**

#### **Grundfunktionen für Tabellen**

Die Verbindungstabelle unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spaltenbreite verändern
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen.

#### **Spaltenbreite ändern**

Um die Breite einer Spalte dem Inhalt anzupassen, so dass alle Texte in den Zeilen lesbar sind, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Mauszeiger in der Kopfzeile der Verbindungstabelle rechts neben der zu optimierenden Spalte, bis der Mauszeiger die Form von zwei parallelen Linien annimmt (so, als wollten Sie die Breite der Spalte durch Ziehen mit dem Mauszeiger verändern).

2. Doppelklicken Sie auf diese Stelle.

oder

1. Öffnen Sie das Kontextmenü auf der Kopfzeile der Tabelle.
2. Klicken Sie auf
  - "Spaltenbreite optimieren" oder
  - "Breite aller Spalten optimieren".

Bei zu schmal eingestellten Spalten wird der komplette Inhalt einzelner Felder eingeblendet, wenn Sie den Mauszeiger über dem betreffenden Feld kurze Zeit stehen lassen.

#### **Spalten ein-/ausblenden**

Über das Kontextmenü der Kopfzeilen der Verbindungstabelle können Sie die Anzeige der verschiedenen Tabellenspalten steuern. Über den Kontextmenüeintrag "Spalten ein-/

ausblenden" sehen Sie eine Übersicht der verfügbaren Spalten. Mit den Optionskästchen steuern Sie die Sichtbarkeit der Spalten.

### Mit Cursortasten in der Verbindungstabelle navigieren

Mit den Cursortasten NACH-OBEN und NACH-UNTEN können Sie eine Verbindung in der Verbindungstabelle auswählen; die ausgewählte Verbindung ist markiert und wird in der Netzsicht hervorgehoben dargestellt.

### Eigenschaften der Verbindung ändern

Sie können die in der Verbindungstabelle angezeigten Parameter teilweise direkt bearbeiten. Um den Verbindungsnamen einer Verbindung zu ändern müssen Sie nicht zum Inspektorfenster navigieren.

### Verbindungspartner ändern

Sie können den Verbindungspartner einer Verbindung wie folgt ändern:

1. Selektieren Sie die Verbindung.
2. Wählen Sie den neuen Verbindungspartner über die aktivierte Klappliste in der Spalte "Partner".

### Verbindungen löschen

Sie können projizierte Verbindungen über die Netzsicht oder die Verbindungstabelle löschen.

In der Netzsicht können Sie jeweils eine hervorgehobene Verbindung löschen. In der Verbindungstabelle können Sie eine oder mehrere Verbindungen löschen.

### Vorgehen

Um eine Verbindung zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie die zu löschende Verbindung aus:
  - In der Netzsicht: Markieren Sie die zu löschende Verbindung.
  - In der Verbindungstabelle: Markieren Sie die Zeilen der zu löschenden Verbindungen (Mehrfachauswahl möglich).
2. Öffnen Sie mit der rechten Maustaste das Kontextmenü.
3. Wählen Sie den Befehl "Löschen".

### Ergebnis

Die markierte Verbindung wird vollständig gelöscht.

## Kopieren von Verbindungen

### Einführung

Verbindungen werden nicht einzeln kopiert, sondern immer im Kontext mit dem Projekt bzw. mit dem Gerät.

Sie können kopieren:

- gesamte Projekte
- eine oder mehrere Geräte innerhalb eines Projekts oder über Projektgrenzen hinweg

### Projekt kopieren

Wenn Sie ein Projekt kopieren, dann werden sämtliche projektierten Verbindungen mitkopiert. Es sind keinerlei Einstellungen für die kopierten Verbindungen notwendig, da die Verbindungen konsistent bleiben.

### Geräte kopieren

Wenn Sie Geräte kopieren, für die Verbindungen projektiert wurden (HMI-Geräte), dann werden die Verbindungen ebenfalls kopiert. Um den Verbindungsweg zu vervollständigen, müssen Sie noch die Vernetzung nachziehen.

Eine S7-1200 CPU mit einer V1.0 Firmware ist lediglich Server für HMI-Verbindungen und hat selber keine Verbindungsprojektion. Daher werden beim Kopieren einer S7-1200 CPU mit einer V1.0 Firmware auch keine Verbindungen mitkopiert.

## Inkonsistente Verbindungen - Verbindungen ohne Zuordnung

Bei einer inkonsistenten Verbindung ist die Struktur der Verbindungsdaten zerstört oder die Verbindung ist im Kontext mit dem Projekt nicht funktionsfähig.

Inkonsistente Verbindungen können nicht übersetzt und geladen werden - der Betrieb mit einer solchen Verbindung ist nicht möglich.

In der Verbindungstabelle sind inkonsistente Verbindungen durch die Farbe Rot zu erkennen.

## Mögliche Ursachen für inkonsistente Verbindungen

- Löschen oder Änderungen der Hardware-Konfiguration
- Fehlende Vernetzungen von Schnittstellen im Projekt, die für eine Verbindung notwendig sind.
- Überschreitung von Verbindungsressourcen
- Fehler bei der Datensicherung durch unzureichenden Speicher
- Verbindungen zu einem un spezifizierten Verbindungspartner ohne Angabe der Partneradresse.

Detailinformationen über die Ursache der Inkonsistenz finden Sie im Register "Übersetzen" nach dem Übersetzen (Bearbeiten > Übersetzen).

## Abhilfen

Wenn die Verbindung durch Öffnen der Verbindungseigenschaften, ändern bzw. rückgängig machen in der Projektierung nicht reparierbar ist, kann ein Löschen und Neuanlegen der Verbindung notwendig sein.

## HMI-Verbindung Allgemeine Einstellungen

### Allgemeine Verbindungsparameter

In der Parametergruppe "Allgemein" unter den Eigenschaften der Verbindung werden allgemeine Verbindungsparameter angezeigt, die den lokalen Verbindungsendpunkt identifizieren.

Sie haben hier außerdem die Möglichkeit, den Verbindungsweg zuzuordnen und die Verbindungspartner vollständig zu spezifizieren.

### Besondere Verbindungseigenschaften

Anzeige der Verbindungseigenschaften (nicht änderbar):

- Aktiver Verbindungsaufbau  
Der Verbindungsaufbau geht immer vom HMI-Gerät aus. Die Option ist standardmäßig angewählt, wenn die Adresse des Partners spezifiziert ist.
- Einseitig  
Einseitig bedeutet, dass der Verbindungspartner Server für diese Verbindung ist und nicht aktiv senden oder empfangen kann.
- Betriebszustandsmeldungen senden  
Nicht relevant für HMI-Geräte.

### Adressdetails

Anzeige der Adressdetails der HMI-Verbindung. Bei einem unspezifizierten Partner können die Werte für Baugruppenträger und Steckplatz geändert werden. Alle anderen Werte werden aus der aktuellen Konfiguration ermittelt und sind nicht änderbar.

### Verschiedenes

Anzeige der Zugangspunkte für die Online-Verbindung zwischen HMI-Gerät und Verbindungspartner.

## Open User Communication anwenden

### Grundlagen der Open User Communication

#### Einführung

Open User Communication ist eine Bezeichnung für ein programmgesteuertes Kommunikationsverfahren zur Kommunikation über die integrierte PN/IE-Schnittstelle der CPUs der S7-1200/1500 und der S7-300/400. Für dieses Kommunikationsverfahren stehen unterschiedliche Verbindungstypen zur Verfügung.

Die Open User Communication zeichnet sich durch eine hohe Flexibilität hinsichtlich der zu übertragenden Datenstrukturen aus und ermöglicht damit den offenen Datenaustausch mit beliebigen Kommunikationsteilnehmern, sofern diese die hier zur Verfügung gestellten Verbindungstypen unterstützen. Da die Kommunikation ausschließlich über Anweisungen im Anwenderprogramm gesteuert wird, ist ein ereignisgesteuertes Auf- und Abbauen von Verbindungen möglich. Verbindungen können außerdem zur Laufzeit durch das Anwenderprogramm modifiziert werden.

Für CPUs mit einer integrierten PN/IE-Schnittstelle stehen Ihnen für die Open User Communication die Verbindungstypen TCP, UDP und ISO-on-TCP zur Verfügung. Die Kommunikationspartner können zwei SIMATIC-PLCs oder auch eine SIMATIC-PLC und ein geeignetes Fremdgerät sein.

#### Anweisungen für die Open User Communication

Zum Erstellen der Verbindungen stehen Ihnen nach dem Öffnen im Programmiereditor in der Task Card "Anweisungen > Kommunikation > Open User Communication" je nach CPU-Typ verschiedene Anweisungen zur Verfügung:

- Kompakte Anweisungen zum Senden oder Empfangen von Daten über mit den integrierten Funktionen zum Auf- oder Abbau der Verbindung (nur S7-1200/1500):
  - TSEND\_C (Verbindungsauf-/abbau, Senden)
  - TRCV\_C (Verbindungsauf-/abbau, Empfangen)
- Einzelanweisungen zum Senden und Empfangen von Daten oder für den Auf- oder Abbau der Verbindungen:
  - TCON (Verbindungsaufbau)
  - TDISCON (Verbindungsabbau)
  - TSEND (TCP oder ISO-on-TCP: Senden)
  - TRCV (TCP oder ISO-on-TCP: Empfangen)
  - TUSEND (UDP: Senden)
  - TURCV (UDP: Empfangen)

#### Verbindungsaufbau

Für die Open User Communication müssen für beide Kommunikationspartner Anweisungen zum Auf- und Abbau der Verbindung vorhanden sein. Ein Kommunikationspartner sendet

seine Daten über TSEND, TUSEND oder TSEND\_C, während der andere Kommunikationspartner die Daten über TRCV, TURCV oder TRCV\_C empfängt.

Einer der Kommunikationspartner beginnt als aktiver Partner mit dem Verbindungsaufbau. Der andere Kommunikationspartner reagiert darauf und startet als passiver Partner seinen Verbindungsaufbau. Wenn beide Kommunikationspartner ihren Verbindungsaufbau angestoßen haben, wird die Kommunikationsverbindung vollständig aufgebaut.

## Verbindungsparametrierung

Sie können den Verbindungsaufbau über einen Verbindungsbeschreibungs-DB mit der Struktur TCON\_Param, TCON\_IP\_v4 oder TCON\_IP\_RFC folgendermaßen parametrieren:

- Manuell erstellen, parametrieren und direkt an die Anweisung schreiben.
- Unterstützt über die Verbindungsparametrierung.

Die Verbindungsparametrierung unterstützt den Aufbau der Verbindung und ist daher gegenüber der anderen Methoden zu empfehlen.

In der Verbindungsparametrierung treffen Sie folgende Festlegungen:

- Verbindungspartner
- Verbindungstyp
- Verbindungs-ID
- Verbindungsbeschreibungs-DB
- Adressdetails in Abhängigkeit vom gewählten Verbindungstyp

Weiterhin legen Sie hier fest, welcher Kommunikationspartner den Verbindungsaufbau aktiviert und welcher auf eine Anforderung des Kommunikationspartners hin einen passiven Verbindungsaufbau durchführt.

## Siehe auch

Arbeitsweise verbindungsorientierter Protokolle (Seite 470)

## Verbindungsparametrierung

### Übersicht über die Verbindungsparametrierung

#### Einführung

Die Verbindungsparametrierung finden Sie im Inspektorfenster des Programmiereditors, wenn Sie mit den Kommunikationsanweisungen TSEND\_C, TRCV\_C oder TCON eine Open User Communication programmieren möchten.

Die Verbindungsparametrierung unterstützt die flexible Funktionalität der Kommunikations-Programmierung: In einem automatisch erzeugten von der Struktur des Typs TCON\_Param, TCON\_IP\_v4 oder TCON\_IP\_RFC abgeleiteten Global-DB werden die bei der Verbindungsparametrierung eingegebenen Parameter gespeichert. In diesem Verbindungsbeschreibungs-DB können Sie die Verbindungsparameter modifizieren.



## **Aufbau der Verbindungsparametrierung**

Die Verbindungsparametrierung besteht aus folgenden Komponenten:

The screenshot displays the SIMATIC Manager interface. The top window shows a ladder logic program for PLC\_1 [CPU 1215C DC/DC/Rly] in the Main [OB1] network. A TCON block (SFB102) is configured with the following parameters:

- EN: Connected to a power rail.
- REQ: Set to False.
- ID: Set to 1.
- CONNECT: Connected to %DB2 "PLC\_1\_Send\_DB".
- ENO: Connected to a power rail.
- DONE, BUSY, ERROR, STATUS: Connected to output lines.

The bottom window shows the configuration dialog for T\_CON [SFB102], specifically the "Verbindungsparameter" (Connection Parameters) tab. The configuration is as follows:

| Parameter                 | Lokal (PLC_1)                    | Partner (PLC_2)         |
|---------------------------|----------------------------------|-------------------------|
| Endpunkt                  | PLC_1                            | PLC_2                   |
| Schnittstelle             | CPU 1215C DC/DC/Rly, PF          | CPU 1215C DC/DC/Rly, PF |
| Subnetz                   | PN/IE_1                          | PN/IE_1                 |
| Adresse                   | 192.168.0.2                      | 192.168.0.1             |
| Verbindungstyp            | TCP                              | TCP                     |
| Verbindungs-ID (dez)      | 1                                | 1                       |
| Verbindungsdaten          | PLC_1_Send_DB                    | PLC_2_Receive_DB        |
| Aktiver Verbindungsaufbau | <input checked="" type="radio"/> | <input type="radio"/>   |

The "Adressdetails" section shows the local port is empty and the partner port is set to 2000.

- ① Kommunikationsanweisung für TCON, TSEND\_C oder TRCV\_C
- ② Register "Konfiguration" im Register "Eigenschaften"
- ③ Bereichsnavigation des Registers "Konfiguration"
- ④ Allgemeine Eigenschaften der Verbindungsparameter
- ⑤ Adressdetails der Verbindungsparameter (bei ausgewählten Verbindungs-DBs)

## Register "Konfiguration"

Tragen Sie die gewünschten Verbindungsparameter im Register "Konfiguration" ein. Sie finden in der Bereichsnavigation des Registers "Konfiguration" die Gruppe "Verbindungsparameter". Diese Gruppe enthält die Verbindungsparametrierung. Hier können Sie systemunterstützt die Parameter für die Verbindungen und die Adressdetails eintragen. Hier beschalten Sie auch die Bausteinparameter CONNECT (TCON, TSEND\_C, TRCV\_C) oder ID (TCON, TSEND, TRCV, TUSEND, TURCV) der gewählten Kommunikationsanweisungen.

Wenn alle erforderlichen Parameter vorhanden sind, wird in der Bereichsnavigation vor die Gruppe "Verbindungsparameter" ein Häkchen gesetzt.

---

### Hinweis

Die Verbindungsparametrierung prüft nicht auf Eindeutigkeit von Verbindungs-IDs und Port-Nummern (TCP, UDP) bzw. TSAPs (ISO-on-TCP, ISO). Sorgen Sie daher bei der Konfiguration der Open User Communication für eine eindeutige Vergabe dieser Parameter innerhalb eines Gerätes.

---

## Siehe auch

- Verbindungsparameter nach TCON\_Param (Seite 472)
- Verbindungsparameter nach TCON\_IP\_v4 (Seite 475)
- Verbindungsparameter nach TCON\_IP\_RFC (Seite 476)

## Beschreibung der Verbindungsparameter

### Überblick

Die folgende Tabelle zeigt die allgemeinen Verbindungsparameter:

| Parameter      | Beschreibung   |
|----------------|--|
| Endpunkt       | <p>Die Namen des lokalen Endpunkts und des Partner-Endpunkts wird angezeigt.</p> <p>Der lokale Endpunkt ist die CPU, für die der TCON, TSEND_C oder TRCV_C programmiert wird. Der lokale Endpunkt ist daher immer bekannt.</p> <p>Der Partner-Endpunkt wird über die Klappliste ausgewählt. In der Klappliste werden alle verfügbaren möglichen Verbindungspartner angezeigt, darunter auch unspezifizierte Verbindungspartner für Geräte, deren Daten innerhalb des Projekts nicht bekannt sind.</p> <p>Bei S7-1500 kann als Partner-Endpunkt Broadcast ausgewählt werden (Nachricht an alle Teilnehmer des Subnetzes). Bei S7-1500 CPs/CMs kann als Partner-Endpunkt auch Multicast ausgewählt werden (Nachricht an eine Gruppe innerhalb des Subnetzes). Der Verbindungstyp wird dabei automatisch auf UDP umgestellt.</p> <p>Solange noch kein Verbindungspartner eingestellt ist, bleiben alle weiteren Parameter in der Maske deaktiviert.</p> |
| Schnittstelle  | <p>Die Schnittstelle des lokalen Endpunkts wird angezeigt. Sind mehrere Schnittstellen verfügbar, zum Beispiel auch über CPs oder CMs, kann die Schnittstelle über die Klappliste ausgewählt werden. Die Anzeige oder Auswahl der Partner-Schnittstelle erfolgt erst nach Auswahl eines spezifizierten Partner-Endpunkts.</p>  |
| Subnetz        | <p>Das Subnetz des lokalen Endpunkts wird angezeigt, wenn dieses vorhanden ist. Die Anzeige des Partner-Subnetzes erfolgt erst nach Auswahl des Partner-Endpunkts.</p> <p>Wenn mindestens einer der beiden Verbindungspartner nicht mit einem Subnetz verbunden ist, werden die beiden Verbindungspartner miteinander vernetzt.</p> <p>Eine Verbindung von Verbindungspartnern an unterschiedlichen Subnetzen ist nur über IP-Routing möglich. Die Routing-Einstellungen können über die betroffenen Schnittstellen-Eigenschaften bearbeitet werden.</p>   |
| Adresse        | <p>Die IP-Adresse des lokalen Endpunkts wird angezeigt. Die Anzeige der IP-Adresse des Partners erfolgt erst nach Auswahl des Partner Endpunkts.</p> <p>Wenn Sie einen unspezifizierten Verbindungspartner ausgewählt haben, ist das Eingabefeld leer und rot hinterlegt. Sie müssen in dem Fall eine gültige IP-Adresse angeben.</p> <p>Broadcast (nur S7-1500): Ist als Partner-Endpunkt "Broadcast" eingestellt, wird für den Verbindungspartner automatisch eine nicht änderbare IP-Adresse mit der Host-Adresse 255 eingetragen. Der Netzwerkanteil entspricht dem des Senders. Beispiel: Lokale IP-Adresse 192.168.0.1, Partner-IP-Adresse 192.168.0.255.</p> <p>Multicast (nur S7-1500 CPs/CMs): Ist als Partner-Endpunkt "Multicast" eingestellt, wird für den Verbindungspartner automatisch die änderbare IP-Adresse 224.0.1.0 eingetragen.</p>  |
| Verbindungstyp | <p>Wählen Sie über die Klappliste "Verbindungstyp" den zu verwendenden Verbindungstyp aus:</p> <ul style="list-style-type: none"> <li>• TCP</li> <li>• ISO-on-TCP</li> <li>• UDP</li> </ul> <p>Bei S7-1500 können Sie an TSEND_C und TRCV_C bei der Konfigurationsart der konfigurierten Verbindungen auch den Verbindungstyp ISO auswählen.</p> <p>Die Verbindungstypen sind nur für Partner verwendbar, die das entsprechende Protokoll auch unterstützen.</p>   |

| Parameter                              | Beschreibung  |
|--|---|
| Konfigurationsart<br>(nur bei S7-1500) | <p>Bei S7-1500 können an TSEND_C und TRCV_C zwei verschiedene Konfigurationsarten eingestellt werden:</p> <ul style="list-style-type: none"> <li>• Programmierte Verbindungen nutzen Programmbausteine für die Verbindungsbeschreibung.</li> <li>• Konfigurierte Verbindungen werden direkt nach Auswahl der Option erstellt, sofern es nicht bereits eine entsprechende Verbindung gibt. Über die konfigurierte Verbindung lässt sich auch der Verbindungstyp ISO auswählen.</li> </ul> <p>Die vorgegebene Konfigurationsart hängt von dem ausgewählten Verbindungstyp ab. Wenn beide Konfigurationsarten möglich sind, ist die programmierte Verbindung voreingestellt. Bei beiden Verbindungspartnern muss dieselbe Konfigurationsart eingestellt sein.</p>  |
| Verbindungs-ID                         | <p>Tragen Sie im Eingabefeld die Verbindungs-ID ein. Sie können die Verbindungs-ID in den Eingabefeldern ändern oder direkt am TCON angeben.</p> <p>Sorgen Sie für eine eindeutige Vergabe der Verbindungs-ID innerhalb des Geräts!</p>   |
| Verbindungsdaten                       | <p>Die Namen der für die Verbindungsbeschreibung nach TCON_Param, TCON_IP_v4 oder TCON_IP_RFC strukturierten Verbindungsbeschreibungs-DB werden in den Klapplisten angezeigt.</p> <p>Nachdem ein Verbindungspartner ausgewählt wurde, ist die Klappliste noch leer. Sie können über die Klappliste einen neuen Datenbaustein generieren oder einen bereits vorhandenen Datenbaustein auswählen. Dieser wird automatisch mit den Werten aus der Verbindungsparametrierung befüllt. Der Name der gewählten Datenbausteine wird automatisch am Bausteinparameter CONNECT der markierten Anweisungen TSEND_C, TRCV_C oder TCON eingetragen.</p> <p>Sie können über die Klappliste auch einen anderen gültigen Datenbaustein referenzieren. Wenn über den Eingangsparameter CONNECT der erweiterten Anweisungen TSEND_C, TRCV_C oder TCON ein DB referenziert wird, der nicht der Struktur eines TCON_Param, TCON_IP_v4 oder TCON_IP_RFC entspricht, wird die Klappliste bei leerem Inhalt rot hinterlegt.</p> |
| Verbindungsname<br>(nur bei S7-1500)   | <p>Wenn bei S7-1500 an TSEND_C und TRCV_C die Konfigurationsart der konfigurierten Verbindungen eingestellt ist, wird der Parameter "Verbindungsdaten" durch den Parameter "Verbindungsname" ersetzt. Als Verbindungsdatum dient hier nun der Name der konfigurierten Verbindung.</p> <p>Nachdem ein Verbindungspartner ausgewählt wurde, ist die Klappliste noch leer. Sie können über die Klappliste eine neue Verbindung generieren oder eine bereits vorhandene Verbindung auswählen. Ein Datenbaustein wird ggf. erzeugt und automatisch mit den Werten aus der Verbindungsparametrierung befüllt. Der Name des Datenbausteins wird automatisch am Bausteinparameter CONNECT der Anweisung TSEND_C oder TRCV_C eingetragen.</p> <p>Sie können über die Klappliste auch eine vorhandene Verbindung referenzieren.</p>   |
| Aktiver Verbindungsaufbau              | <p>Bestimmen Sie über das Optionsfeld "Aktiver Verbindungsaufbau" den aktiven Partner der Open User Communication (nur bei TCP und ISO-on-TCP).</p>   |
| Port<br>(nur bei TCP und UDP)          | <p>Adresskomponente für eine TCP- oder UDP-Verbindung. Der Vorgabewert nach dem Anlegen einer neuen TCP-Verbindung ist 2000.</p> <p>Die Port-Nummern können Sie ändern.</p> <p>Die Port-Nummern müssen im Gerät eindeutig sein!</p>   |
| TSAP<br>(nur bei ISO-on-TCP)           | <p>Adresskomponente für eine ISO-on-TCP-Verbindung. Der Vorgabewert nach dem Anlegen einer neuen ISO-on-TCP-Verbindung ist E0.01.49.53.4F.6F.6E.54.43.50.2D.31 (S7-1200/1500) bzw. E0.02.49.53.4F.6F.6E.54.43.50.2D.31 (S7-300/400).</p> <p>Sie können die TSAP-ID mit Erweiterung oder als ASCII-TSAP eingeben.</p> <p>Die TSAPs müssen im Gerät eindeutig sein!</p>   |

---

**Hinweis**

**UDP-Verbindung für die Einstellung "Broadcast" (S7-300/400/1200)**

Die Parameter der UDP-Verbindung für die Einstellung "Broadcast" am Partner-Endpunkt werden in einem Verbindungsbeschreibungs-DB TCON\_IP\_v4 abgelegt: Bezüglich der UDP-Kommunikation mit TCON und TUSEND/TURCV wird der TCON\_IP\_v4 mit keinen Partner-Parametern befüllt (Wert=0). Die Partneradresse und der Partnerport sind aber für das Senden der Daten notwendig und müssen vom Anwender in den TADDR\_Param eingetragen werden. Der TADDR\_Param für die UDP-Kommunikation wird von dem TUSEND-/TURCV-Bausteinparameter ADDR referenziert. Die Werte für beide Parameter können der Verbindungsparametrierung entnommen werden.

Auch für die anderen Empfänger der UDP-Kommunikation muss die Konfiguration angepasst werden: Um die Broadcast-Telegramme zu empfangen, muss an der Empfängerseite der Partnerport konfiguriert werden. Hierzu ist am ADDR-Baustein der Parameter RemotePort des TADDR\_Param zu befüllen.

---

**Hinweis**

**Kommunikation über TSEND\_C und TRCV\_C (S7-1500)**

Bei der Verwendung von TSEND\_C und TRCV\_C wird für jede Kommunikation ein eigenes Bausteinpaar TSEND\_C und TRCV\_C mit einer konfigurierten Verbindung benötigt. Mehrere Bausteinpaare TSEND\_C und TRCV\_C können nicht gleichzeitig dieselbe konfigurierte Verbindung für ihre Kommunikation benutzen.

Weitere Verbindungen für eine Anweisung TSEND\_C oder TRCV\_C können im Inspektorfenster bei den Verbindungsparametern über die entsprechende Schaltfläche neben den Verbindungsdaten angelegt werden.

Die über TSEND\_C und TRCV\_C konfigurierten Verbindungen werden bei angewähltem Baustein TSEND\_C oder TRCV\_C im Inspektorfenster unter "Eigenschaften > Konfiguration > Übersicht über die konfigurierten Verbindungen" in einer Verbindungstabelle angezeigt.

---

**Siehe auch**

Vergabe von Port-Nummern (Seite 477)

TSAP-Struktur (Seite 480)

Beispiele zur TSAP-Vergabe (Seite 482)

Rücklesbarkeit der Parameter für die Verbindungsbeschreibung (Seite 479)

Verbindungen anlegen und parametrieren (Seite 466)

Verbindungsparameter nach TCON\_Param (Seite 472)

Verbindungsparameter nach TCON\_IP\_v4 (Seite 475)

Verbindungsparameter nach TCON\_IP\_RFC (Seite 476)

## Verbindungsparametrierung starten

Die Verbindungsparametrierung zur Open User Communication wird aktiviert, sobald Sie eine Anweisung zur Kommunikation TCON, TSEND\_C oder TRCV\_C in einem Programmbaustein auswählen.

## Voraussetzung

- Ihr Projekt beinhaltet mindestens eine S7-CPU.
- Der Programmiereditor ist geöffnet.
- Ein Netzwerk ist vorhanden.

## Vorgehen

Um die erweiterten Anweisungen für die Open User Communication einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card, Palette und Ordner "Anweisungen > Kommunikation > Open User Communication".
2. Ziehen Sie eine der folgenden Anweisungen via Drag & Drop in ein Netzwerk:

- TSEND\_C
- TRCV\_C
- TCON

Der Dialog "Aufrufoptionen" wird geöffnet.

3. Bearbeiten Sie im Dialog "Aufrufoptionen" die Eigenschaften des Instanz-DBs. Sie haben dabei folgende Möglichkeiten:
  - Ändern Sie den vorgegebenen Namen.
  - Wählen Sie das Optionsfeld "manuell", um eine eigene Nummer zu vergeben.
  - Für Funktionsbausteine können Sie den DB auch als Multiinstanz ausführen.
4. Klicken Sie auf "OK", um die Eingabe abzuschließen.

## Ergebnis

Für die eingefügte Anweisung TSEND\_C, TRCV\_C oder TCON wird bei einer Einzelinstanz ein entsprechender Instanz-DB angelegt. Im Falle einer Multiinstanz wird der Instanz-DB des Funktionsbausteins verwendet.

Bei markiertem TSEND\_C, TRCV\_C oder TCON sehen Sie im Inspektorfenster unter "Eigenschaften" das Register "Konfiguration". Die Gruppe "Verbindungsparameter" in der Bereichsnavigation enthält die Verbindungsparametrierung, die Sie nun durchführen können.

## Siehe auch

Verbindungen anlegen und parametrieren (Seite 466)

## Verbindungen anlegen und parametrieren

In der Verbindungsparametrierung der Open User Communication können Sie Verbindungen vom Typ TCP, UDP und ISO-on-TCP anlegen und parametrieren.

### Voraussetzung

Es existiert eine CPU mit einer Kommunikationsanweisung TCON, TSEND\_C oder TRCV\_C.

### Vorgehen

Um eine Verbindung der Open User Communication anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Programmiereditor einen Baustein der Open User Communication TCON, TSEND\_C oder TRCV\_C.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften > Konfiguration".
3. Selektieren Sie die Gruppe "Verbindungsparameter". Solange Sie noch keinen Verbindungspartner selektiert haben, ist nur die leere Klappliste für den Partner-Endpunkt aktiv. Alle anderen Eingabemöglichkeiten sind deaktiviert.  
Es werden die bereits bekannten Verbindungsparameter angezeigt:
  - Name des lokalen Endpunkts
  - Schnittstelle des lokalen Endpunkts
  - IP-Adresse des lokalen Endpunkts
4. Wählen Sie in der Klappliste des Partner-Endpunkts einen Verbindungspartner. Als Kommunikationspartner kommt ein unspezifiziertes Gerät oder eine im Projekt vorhandene CPU in Frage. Bestimmte Verbindungsparameter werden danach als Vorgabe automatisch eingetragen.  
Die folgenden Parameter werden eingestellt:
  - Name des Partner-Endpunkts
  - Schnittstelle des Partner-Endpunkts
  - IP-Adresse des Partner-EndpunktsWenn die Verbindungspartner vernetzt sind, wird der Name des Subnetzes angezeigt.
5. Bei S7-1500 wählen Sie in der Klappliste "Konfigurationsart" zwischen der Verwendung von Programmbausteinen oder konfigurierten Verbindungen.



6. Wählen Sie in der Klappliste "Verbindungsdaten" einen vorhandene Verbindungsbeschreibungs-DBs oder bei konfigurierten Verbindungen unter "Verbindungsname" eine vorhandene Verbindung. Sie können auch einen neue Verbindungsbeschreibungs-DBs oder eine neue konfigurierte Verbindung anlegen. Sie können später noch andere Verbindungsbeschreibungs-DBs oder konfigurierte Verbindungen wählen oder die Namen der Verbindungsbeschreibungs-DBs ändern, um neue Datenbausteine zu erstellen:
  - Den ausgewählten Datenbaustein sehen Sie auch an der Beschaltung des Eingangsparameters CONNECT der ausgewählten Anweisung TCON, TSEND\_C oder TRCV\_C.
  - Wenn Sie für den Verbindungspartner bereits einen Verbindungsbeschreibungs-DB über den Parameter CONNECT der Anweisung TCON, TSEND\_C oder TRCV\_C angegeben haben, können Sie entweder diesen DB verwenden oder einen neuen DB anlegen.
  - Wenn Sie den Namen des angezeigten Datenbausteins in der Klappliste bearbeiten, wird automatisch ein neuer Datenbaustein mit dem geänderten Namen, aber derselben Struktur und demselben Inhalt generiert und für die Verbindung verwendet.
  - Geänderte Namen eines Datenbausteins müssen im Kontext des Kommunikationspartners eindeutig sein.
  - Ein Verbindungsbeschreibungs-DB muss je nach CPU-Typ und Verbindung die Struktur TCON\_Param, TCON\_IP\_v4 oder TCON\_IP\_RFC haben.
  - Ein Datenbaustein kann nicht für einen un spezifizierten Partner ausgewählt werden.

Nach Auswahl oder Anlegen des Verbindungsbeschreibungs-DBs oder der konfigurierten Verbindung werden weitere Werte ermittelt und eingetragen.

Für spezifizierte Verbindungspartner gilt:

- Verbindungstyp ISO-on-TCP
- Verbindungs-ID mit dem Vorgabewert 1
- Aktiver Verbindungsaufbau vom lokalen Partner
- TSAP-ID
  - für S7-1200/1500: E0.01.49.53.4F.6F.6E.54.43.50.2D.31
  - für S7-300/400: E0.02.49.53.4F.6F.6E.54.43.50.2D.31

Für un spezifizierte Verbindungspartner gilt:

- Verbindungstyp TCP
- Partnerport 2000

Bei konfigurierter Verbindung mit spezifiziertem Verbindungspartner gilt:

- Verbindungstyp TCP
- Verbindungs-ID mit dem Vorgabewert 257
- Aktiver Verbindungsaufbau vom lokalen Partner
- Partnerport 2000

Bei konfigurierter Verbindung mit un spezifiziertem Verbindungspartner gilt:

- Verbindungstyp TCP

- Lokaler Port 2000

7. Geben Sie ggf. eine Verbindungs-ID für den Verbindungspartner an. Für einen un spezifizierten Partner kann keine Verbindungs-ID vergeben werden.
- 

**Hinweis**

Sie müssen bei einem bekannten Verbindungspartner einen eindeutigen Wert für die Verbindungs-ID eingeben. Die Eindeutigkeit der Verbindungs-ID wird nicht durch die Verbindungsparametrierung geprüft und es wird bei Anlegen einer neuen Verbindung kein Vorgabewert für die Verbindungs-ID eingetragen!

---

8. Wählen Sie den gewünschten Verbindungstyp aus der entsprechenden Klappliste. Die Adressdetails werden abhängig vom Verbindungstyp mit Werten vorbelegt. Sie haben die Wahl zwischen:

- TCP
- ISO-on-TCP
- UDP

Für konfigurierte Verbindungen bei S7-1500 gilt zusätzlich ISO.

9. Sie können die Eingabefelder in den Adressdetails bearbeiten. Je nach eingestelltem Protokoll können Sie die Ports (für TCP und UDP) oder die TSAPs (für ISO-on-TCP und ISO) bearbeiten.
10. Stellen Sie bei TCP, ISO und ISO-on-TCP das Verhalten für den Verbindungsaufbau über die Optionsfelder "Aktiver Verbindungsaufbau" ein. Sie können auswählen, welcher Kommunikationspartner die Verbindung aktiv aufbauen soll.

Geänderte Werte werden von der Verbindungsparametrierung sofort auf Eingabefehler geprüft und in den Datenbaustein für die Verbindungsbeschreibung eingetragen.

---

**Hinweis**

Die Open User Communication zwischen zwei Kommunikationspartnern ist erst dann lauffähig, wenn auch der Programmteil für den Partner-Endpunkt in die Hardware geladen wurde. Achten Sie darauf, dass Sie für eine funktionierende Kommunikation nicht nur die Verbindungsbeschreibung der lokalen CPU in das Gerät laden, sondern auch die der Partner-CPU.

---

**Siehe auch**

- Beschreibung der Verbindungsparameter (Seite 462)
- Verbindungsparametrierung starten (Seite 465)
- TSAP-Struktur (Seite 480)
- Vergabe von Port-Nummern (Seite 477)
- Verbindungsparameter nach TCON\_Param (Seite 472)
- Verbindungsparameter nach TCON\_IP\_v4 (Seite 475)
- Verbindungsparameter nach TCON\_IP\_RFC (Seite 476)

## Verbindungen löschen

### Einführung

Die Daten einer angelegten Verbindung der Open User Communication werden in einer Verbindungsbeschreibungs-DB abgelegt. Sie können die Verbindung löschen, indem Sie den Datenbaustein mit der enthaltenen Verbindungsbeschreibung löschen.

### Voraussetzung

Sie haben eine Verbindung der Open User Communication erstellt.

### Vorgehen

Um eine Verbindung zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Kommunikationspartner der Open User Communication in der Projektnavigation.
2. Öffnen Sie den Ordner "Programmteile > Systemteile > Programmressourcen" unterhalb des selektierten Kommunikationspartners.
3. Wählen Sie im Kontextmenü des Datenbausteines mit der Verbindungsparametrierung den Befehl "Löschen".

---

#### Hinweis

Wenn Sie nicht genau wissen, welchen Baustein Sie löschen müssen, öffnen Sie die erweiterte Anweisung TCON, TSEND\_C oder TRCV\_C. Sie finden den Namen des Datenbausteines als Eingangsparameter CONNECT oder in der Verbindungsparametrierung als Parameter "Verbindungsdaten".

Wenn Sie nur die Instanz-DBs der erweiterten Anweisungen TCON, TSEND\_C oder TRCV\_C löschen, werden die parametrisierten Verbindungen nicht mitgelöscht.

---

#### Hinweis

Wenn die Verbindungs-DB noch von weiteren Bausteinen der erweiterten Anweisungen verwendet wird, müssen die entsprechenden Aufrufe, deren Instanz-DBs und wenn vorhanden, die Kombinationsbausteine TSEND\_C und TRCV\_C ebenfalls aus dem Bausteinordner gelöscht werden, sofern sie nicht anderweitig verwendet werden.

Durch diese Maßnahme wird verhindert, dass das Programm inkonsistent wird.

---

## Ergebnis

Sie haben die Verbindung gelöscht.

---

### Hinweis

Fügen Sie eine erweiterte Anweisung TCON, TSEND\_C oder TRCV\_C erneut ein, um eine existierende Verbindungsbeschreibungen mit der Struktur TCON\_Param, TCON\_IP\_v4 oder TCON\_IP\_RFC über den Parameter "Verbindungsdaten" erneut referenzieren.

---

## Funktionsweise der Protokolle

### Arbeitsweise verbindungsorientierter Protokolle

#### Einführung

Verbindungsorientierte Protokolle bauen vor der Datenübertragung eine logische Verbindung zum Kommunikationspartner auf und bauen diese nach Abschluss der Datenübertragung ggf. wieder ab. Verbindungsorientierte Protokolle werden insbesondere dann eingesetzt, wenn es auf eine zuverlässige Datenübertragung ankommt. Über eine physikalische Leitung können mehrere logische Verbindungen bestehen.

Bei der Open User Communication werden die folgenden Verbindungstypen unterstützt:

- TCP
- ISO-on-TCP
- ISO (nur S7-1500)
- UDP

Für eine Verbindung müssen beide Kommunikationspartner denselben Verbindungstyp unterstützen. Wird beispielsweise eine Verbindung vom Typ ISO-on-TCP von einem Kommunikationspartner nicht unterstützt, verwenden Sie stattdessen vielleicht den Verbindungstyp TCP - sofern dieser auch unterstützt wird.

Für Kommunikationspartner, die nicht im TIA-Portal konfiguriert werden können, beispielsweise bei Fremdgeräten oder PCs, tragen Sie in der Verbindungsparametrierung als Partner-Endpunkt "unspezifiziert" ein. Der benötigte Verbindungstyp für unspezifizierte Geräte ist den jeweiligen Dokumentationen zu entnehmen.

---

### Hinweis

#### Verbindungen mit ISO

Bei S7-1500 CPUs können über die Anweisungen TSEND\_C und TRCV\_C konfigurierte Verbindungen vom Typ ISO erstellt werden. Beachten Sie für weitere Informationen zu diesen Verbindungstypen die allgemeinen Verbindungsbeschreibungen.

---

## **Merkmale von TCP**

TCP ist ein Streaming-Protokoll, bei welchem die Länge des Datenstroms an den Empfänger übertragen wird, damit dieser den Datenstrom als einzelne TCP-Segmente empfangen kann. Bei der Datenübertragung über eine TCP-Verbindung werden also keine Informationen über Anfang und Ende einer Nachricht übertragen. Der Empfänger kann über die empfangenen Segmente des Datenstroms nicht erkennen, wo eine Nachricht im Datenstrom endet und wo die nächste beginnt. Es wird daher empfohlen, der Anzahl der zu empfangenden Bytes (Parameter LEN, Anweisung TRCV/TRCV\_C) denselben Wert zuzuweisen wie der Anzahl der zu sendenden Bytes (Parameter LEN, Anweisung TSEND/TSEND\_C).

Wenn die Länge der gesendeten Daten und die Länge der erwarteten Daten nicht übereinstimmen, wird folgendermaßen verfahren:

- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TRCV/TRCV\_C) größer als Länge der gesendeten Daten (Parameter LEN, Anweisung TSEND/TSEND\_C): TRCV/TRCV\_C kopiert die empfangenen Daten erst dann in den vorgegebenen Empfangsbereich (Parameter DATA), wenn die parametrierte Länge erreicht ist. Wenn die parametrierte Länge erreicht wird, werden bereits Daten eines nachfolgenden Auftrags empfangen. Im Empfangsbereich befinden sich dann Daten aus zwei unterschiedlichen Sendeaufträgen. Wenn Sie die exakte Länge der ersten Nachricht nicht wissen, haben Sie keine Möglichkeit, das Ende der ersten bzw. den Anfang der zweiten Nachricht zu erkennen.
- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TRCV/TRCV\_C) kleiner als Länge der gesendeten Daten (Parameter LEN, Anweisung TSEND/TSEND\_C): TRCV/TRCV\_C kopiert so viele Bytes in den Empfangsbereich (Parameter DATA), wie Sie am Parameter LEN vorgegeben haben. Anschließend setzt er den Zustandsparameter NDR auf TRUE (Auftrag wurde erfolgreich abgeschlossen) und weist RCVD\_LEN (Anzahl der tatsächlich empfangenden Daten) den Wert von LEN zu. Mit jedem weiteren Aufruf erhalten Sie einen weiteren Block der gesendeten Daten.

Über die Protokollvariante des Ad-hoc-Modus kann in den Anweisungen TRCV/TRCV\_C ein Empfangsbereich mit fester Datenlänge festgelegt werden.

## **Merkmale von ISO-on-TCP**

ISO-on-TCP ist ein Message-orientiertes Protokoll, welches auf Empfängerseite das Ende der Nachricht erkennt und die zur Nachricht gehörenden Daten an den Anwender indiziert. Dies ist von der spezifizierten Empfangslänge der Nachricht unabhängig. Bei der Datenübertragung über eine ISO-on-TCP-Verbindung werden also Informationen zur Länge und zum Ende einer Nachricht mitgeliefert.

Wenn die Länge der gesendeten Daten und die Länge der erwarteten Daten nicht übereinstimmen, wird folgendermaßen verfahren:

- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TRCV/TRCV\_C) größer als Länge der gesendeten Daten (Parameter LEN, Anweisung TSEND/TSEND\_C): TRCV/TRCV\_C kopiert die gesendeten Daten vollständig in den Empfangsdatenbereich (Parameter DATA). Anschließend setzt er den Zustandsparameter NDR auf TRUE (Auftrag wurde erfolgreich abgeschlossen) und weist RCVD\_LEN (Anzahl der tatsächlich empfangenden Daten) die Länge der gesendeten Daten zu.
- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TRCV/TRCV\_C) kleiner als Länge der gesendeten Daten (Parameter LEN, Anweisung TSEND/TSEND\_C): TRCV/TRCV\_C kopiert keine Daten in den Empfangsdatenbereich (Parameter DATA), sondern liefert folgende Fehlerinformation: ERROR=1, STATUS=W#16#8088 (Zielpuffer zu klein).

## Merkmale von UDP

UDP ist ein Message-orientiertes Protokoll, welches auf Empfängerseite das Ende der Nachricht erkennt und die zur Nachricht gehörenden Daten an den Anwender indiziert. Dies ist von der spezifizierten Empfangslänge der Nachricht unabhängig. Bei der Datenübertragung über eine UDP-Verbindung werden also Informationen zur Länge und zum Ende einer Nachricht mitgeliefert.

Wenn die Länge der gesendeten Daten und die Länge der erwarteten Daten nicht übereinstimmen, wird folgendermaßen verfahren:

- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TURCV/TRCV\_C) größer als Länge der gesendeten Daten (Parameter LEN, Anweisung TUSEND/TSEND\_C): TURCV/TRCV\_C kopiert die gesendeten Daten vollständig in den Empfangsdatenbereich (Parameter DATA). Anschließend setzt er den Zustandsparameter NDR auf TRUE (Auftrag wurde erfolgreich abgeschlossen) und weist RCVD\_LEN (Anzahl der tatsächlich empfangenden Daten) die Länge der gesendeten Daten zu.
- Länge der zu empfangenden Daten (Parameter LEN, Anweisung TURCV/TRCV\_C) kleiner als Länge der gesendeten Daten (Parameter LEN, Anweisung TUSEND/TSEND\_C): TURCV/TRCV\_C kopiert so viele Daten in den Empfangsdatenbereich (Parameter DATA), wie im Parameter LEN angefordert. Es wird keine weitere Fehlermeldung generiert. Der Anwender muss in diesem Fall nochmals einen T\_URCV aufrufen um die restlichen Bytes zu empfangen.

## Siehe auch

Grundlagen der Open User Communication (Seite 457)

## Verbindungsparameter nach TCON\_Param

### Datenbaustein für Verbindungsbeschreibung

Um die Kommunikationsverbindungen bei TCP, UDP und ISO-on-TCP zu parametrieren, wird für einige CPUs der S7-1200 ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON\_Param verwendet. Die feste Datenstruktur des TCON\_Param enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden. Der Verbindungsbeschreibungs-DB wird automatisch von der Verbindungsparametrierung der Open User Communication bei

Verwendung der Anweisungen TSEND\_C, TRCV\_C oder TCON für eine neue Verbindung angelegt.

Der Verbindungsparameter CONNECT der Instanz-DBs für TSEND\_C, TRCV\_C oder TCON enthält einen Verweis auf den verwendeten Datenbaustein.

### Aufbau der Verbindungsbeschreibung nach TCON\_Param

| Byte    | Parameter         | Datentyp | Startwert | Beschreibung   |
|---------|-------------------|----------|-----------|--|
| 0 ... 1 | block_length      | UINT     | 64        | Länge: 64 Bytes (fest)   |
| 2 ... 3 | id                | CONN_OUC | 1         | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Den Wert dieses Parameters müssen Sie für die Anweisung TSEND_C, TRCV_C oder TCON unter ID angeben.  |
| 4       | connection_type   | USINT    | 17        | Verbindungstyp: <ul style="list-style-type: none"> <li>• 17: TCP</li> <li>• 18: ISO-on-TCP</li> <li>• 19: UDP</li> </ul>   |
| 5       | active_est        | BOOL     | TRUE      | Kennung für die Art des Verbindungsaufbaus. Bei UDP gilt immer FALSE, da über die lokale ID Daten gesendet oder empfangen werden können.<br>Für TCP und ISO-on-TCP gilt: <ul style="list-style-type: none"> <li>• FALSE: passiver Verbindungsaufbau</li> <li>• TRUE: aktiver Verbindungsaufbau</li> </ul>                            |
| 6       | local_device_id   | USINT    | 1         | Kennung für die lokale PN/IE-Schnittstelle.  |
| 7       | local_tsap_id_len | USINT    | 0         | Verwendete Länge des Parameters local_tsap_id in Byte; mögliche Werte: <ul style="list-style-type: none"> <li>• 0 oder 2, falls Verbindungstyp = 17 (TCP)<br/>Für die aktive Seite ist nur der Wert 0 zulässig.</li> <li>• 2 bis 16, falls Verbindungstyp = 18 (ISO-on-TCP)</li> <li>• 2, falls Verbindungstyp = 19 (UDP)</li> </ul> |
| 8       | rem_subnet_id_len | USINT    | 0         | Dieser Parameter wird nicht verwendet.   |
| 9       | rem_staddr_len    | USINT    | 4         | Länge der Adresse des Partner-Endpunkts in Byte: <ul style="list-style-type: none"> <li>• 0: un spezifiziert, d. h. Parameter rem_staddr ist irrelevant.</li> <li>• 4: gültige IP-Adresse im Parameter rem_staddr (nur TCP und ISO-on-TCP)</li> </ul>  |
| 10      | rem_tsap_id_len   | USINT    | 2         | Verwendete Länge des Parameters rem_tsap_id in Byte; mögliche Werte: <ul style="list-style-type: none"> <li>• 0 oder 2, falls Verbindungstyp = 17 (TCP)<br/>Für die passive Seite ist nur der Wert 0 zulässig.</li> <li>• 2 bis 16, falls Verbindungstyp = 18 (ISO-on-TCP)</li> <li>• 0, falls Verbindungstyp = 19 (UDP)</li> </ul>  |
| 11      | next_staddr_len   | USINT    | 0         | Dieser Parameter wird nicht verwendet.   |

| Byte         | Parameter     | Datentyp                    | Startwert | Beschreibung  |
|--------------|---------------|-----------------------------|-----------|---|
| 12 ...<br>27 | local_tsap_id | ARRAY<br>[1..16] of<br>BYTE | -         | <p>Lokale Adresskomponente der Verbindung:</p> <ul style="list-style-type: none"> <li>TCP und UDP: lokale Port-Nr. (mögliche Werte: 1...49151; empfohlene Werte: 2000...5000);<br/>local_tsap_id[1] = High Byte der Port-Nr. in hexadezimaler Darstellung;<br/>local_tsap_id[2] = Low Byte der Port-Nr. in hexadezimaler Darstellung;<br/>local_tsap_id[3-16] = irrelevant</li> <li>ISO-on-TCP: lokale TSAP-ID:<br/>local_tsap_id[1] = B#16#E0;<br/>local_tsap_id[2] = Rack und Steckplatz der lokalen Endpunkts (Bits 0 bis 4: Steckplatznummer, Bits 5 bis 7: Baugruppenträgernummer);<br/>local_tsap_id[3-16] = TSAP-Erweiterung, optional</li> </ul> <p>Hinweis: Stellen Sie sicher, dass jeder Wert von local_tsap_id innerhalb der CPU eindeutig ist.</p> |
| 28 ...<br>33 | rem_subnet_id | ARRAY<br>[1..6] of<br>USINT | -         | Dieser Parameter wird nicht verwendet.  |
| 34 ...<br>39 | rem_staddr    | ARRAY<br>[1..6] of<br>USINT | -         | <p>Nur TCP und ISO-on-TCP: IP-Adresse des Partner-Endpunkts, z. B. für 192.168.002.003:</p> <ul style="list-style-type: none"> <li>rem_staddr[1] = 192</li> <li>rem_staddr[2] = 168</li> <li>rem_staddr[3] = 002</li> <li>rem_staddr[4] = 003</li> <li>rem_staddr[5-6]= irrelevant</li> </ul>   |
| 40 ...<br>55 | rem_tsap_id   | ARRAY<br>[1..16] of<br>BYTE | -         | <p>Partner-Adresskomponente der Verbindung</p> <ul style="list-style-type: none"> <li>TCP: Partnerport-Nr. (mögliche Werte: 1...49151; empfohlene Werte: 2000...5000);<br/>rem_tsap_id[1] = High Byte der Port-Nr. in hexadezimaler Darstellung;<br/>rem_tsap_id[2] = Low Byte der Port-Nr. in hexadezimaler Darstellung;<br/>rem_tsap_id[3-16] = irrelevant</li> <li>ISO-on-TCP: Partner-TSAP-ID:<br/>rem_tsap_id[1] = B#16#E0;<br/>rem_tsap_id[2] = Rack und Steckplatz des Partner-Endpunkts (Bits 0 bis 4: Steckplatznummer, Bits 5 bis 7: Baugruppenträgernummer);<br/>rem_tsap_id[3-16] = TSAP-Erweiterung, optional</li> <li>UDP: Dieser Parameter wird nicht verwendet.</li> </ul>  |
| 56 ...<br>61 | next_staddr   | ARRAY<br>[1..6] of<br>BYTE  | -         | Dieser Parameter wird nicht verwendet.  |
| 62 ...<br>63 | spare         | WORD                        | W#16#0000 | Reserviert.   |



---

**Hinweis****TCON\_Param für S7-1500 CPU**

Der Verbindungsbeschreibungs-DB mit der Struktur nach TCON\_Param wird aus Migrationsgründen auch von CPUs der S7-1500 unterstützt. Es wird jedoch empfohlen, die neuen Strukturen TCON\_IP\_v4 und TCON\_IP\_RFC zu verwenden.

---

**Siehe auch**

- Arbeitsweise verbindungsorientierter Protokolle (Seite 470)
- Beschreibung der Verbindungsparameter (Seite 462)
- Rücklesbarkeit der Parameter für die Verbindungsbeschreibung (Seite 479)
- Übersicht über die Verbindungsparametrierung (Seite 458)
- TSAP-Struktur (Seite 480)
- Vergabe von Port-Nummern (Seite 477)

**Verbindungsparameter nach TCON\_IP\_v4****Datenbaustein für Verbindungsbeschreibung**

Um die Kommunikationsverbindungen bei TCP und UDP zu parametrieren, wird für CPUs der S7-1200 ab V4.0 und S7-1500 ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON\_IP\_v4 verwendet. Die feste Datenstruktur des TCON\_IP\_v4 enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden. Der Verbindungsbeschreibungs-DB wird automatisch von der Verbindungsparametrierung der Open User Communication bei Verwendung der Anweisungen TSEND\_C, TRCV\_C oder TCON für eine neue Verbindung angelegt.

Der Verbindungsparameter CONNECT der Instanz-DBs für TSEND\_C, TRCV\_C oder TCON enthält einen Verweis auf den verwendeten Datenbaustein.

**Aufbau der Verbindungsbeschreibung nach TCON\_IP\_v4**

| Byte    | Parameter    | Datentyp | Startwert | Beschreibung   |
|---------|--------------|----------|-----------|--|
| 0 ... 1 | interface_id | HW_ANY   | 64        | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).  |
| 2 ... 3 | id           | CONN_OUC | 1         | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095).<br>Den Wert dieses Parameters müssen Sie für die Anweisung TSEND_C, TRCV_C oder TCON unter ID angeben. |

| Byte      | Parameter          | Datentyp             | Startwert | Beschreibung  |
|-----------|--------------------|----------------------|-----------|---|
| 4         | connection_type    | BYTE                 | 11        | Verbindungstyp: <ul style="list-style-type: none"> <li>• 11: TCP</li> <li>• 13: UDP</li> </ul>  |
| 5         | active_established | BOOL                 | TRUE      | Kennung für die Art des Verbindungsaufbaus: <ul style="list-style-type: none"> <li>• FALSE: passiver Verbindungsaufbau</li> <li>• TRUE: aktiver Verbindungsaufbau</li> </ul>                      |
| 6 ... 9   | remote_address     | ARRAY [1..4] of BYTE | -         | IP-Adresse des Partner-Endpunkts, z. B. für 192.168.0.1: <ul style="list-style-type: none"> <li>• addr[1] = 192</li> <li>• addr[2] = 168</li> <li>• addr[3] = 0</li> <li>• addr[4] = 1</li> </ul> |
| 10 ... 11 | remote_port        | UINT                 | 2000      | Port-Adresse des entfernten Verbindungspartners (Wertebereich: 1 bis 49151).  |
| 12 ... 13 | local_port         | UINT                 | 2000      | Port-Adresse des lokalen Verbindungspartners (Wertebereich: 1 bis 49151).   |

**Siehe auch**

- Arbeitsweise verbindungsorientierter Protokolle (Seite 470)
- Beschreibung der Verbindungsparameter (Seite 462)
- Rücklesbarkeit der Parameter für die Verbindungsbeschreibung (Seite 479)
- Übersicht über die Verbindungsparametrierung (Seite 458)
- Vergabe von Port-Nummern (Seite 477)

**Verbindungsparameter nach TCON\_IP\_RFC**

**Datenbaustein für Verbindungsbeschreibung**

Um die Kommunikationsverbindungen bei ISO-on-TCP zu parametrieren, wird für CPUs der S7-1200 ab V4.0 und S7-1500 ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON\_IP\_RFC verwendet. Die feste Datenstruktur des TCON\_IP\_RFC enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden. Der Verbindungsbeschreibungs-DB wird automatisch von der Verbindungsparametrierung der Open User Communication bei Verwendung der Anweisungen TSEND\_C, TRCV\_C oder TCON für eine neue Verbindung angelegt.

Der Verbindungsparameter CONNECT der Instanz-DBs für TSEND\_C, TRCV\_C oder TCON enthält einen Verweis auf den verwendeten Datenbaustein.

## Aufbau der Verbindungsbeschreibung nach TCON\_IP\_RFC

| Byte      | Parameter          | Datentyp             | Startwert | Beschreibung  |
|-----------|--------------------|----------------------|-----------|---|
| 0 ... 1   | interface_id       | HW_ANY               | 64        | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).   |
| 2 ... 3   | id                 | CONN_OUC             | 1         | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095).<br>Den Wert dieses Parameters müssen Sie für die Anweisung TSEND_C, TRCV_C oder TCON unter ID angeben.  |
| 4         | connection_type    | BYTE                 | 12        | Verbindungstyp: ISO-on-TCP  |
| 5         | active_established | BOOL                 | TRUE      | Kennung für die Art des Verbindungsaufbaus: <ul style="list-style-type: none"> <li>• FALSE: passiver Verbindungsaufbau</li> <li>• TRUE: aktiver Verbindungsaufbau</li> </ul>                                  |
| 8 ... 11  | remote_address     | ARRAY [1..4] of BYTE | -         | IP-Adresse des Partner-Endpunkts, z. B. für 192.168.0.1: <ul style="list-style-type: none"> <li>• addr[1] = 192</li> <li>• addr[2] = 168</li> <li>• addr[3] = 0</li> <li>• addr[4] = 1</li> </ul>             |
| 12 ... 45 | remote_tselector   | TSelector            | -         | TSelector des entfernten Verbindungspartners: <ul style="list-style-type: none"> <li>• TSelLength = Wertebereich 0 bis 32 als UINT</li> <li>• TSel[1-32] = Wertebereich jeweils 0 bis 255 in Bytes</li> </ul> |
| 46 ... 79 | local_tselector    | TSelector            | -         | TSelector des lokalen Verbindungspartners: <ul style="list-style-type: none"> <li>• TSelLength = Wertebereich 0 bis 32 als UINT</li> <li>• TSel[1-32] = Wertebereich jeweils 0 bis 255 in Bytes</li> </ul>    |

### Siehe auch

Arbeitsweise verbindungsorientierter Protokolle (Seite 470)

Beschreibung der Verbindungsparameter (Seite 462)

Rücklesbarkeit der Parameter für die Verbindungsbeschreibung (Seite 479)

Übersicht über die Verbindungsparametrierung (Seite 458)

TSAP-Struktur (Seite 480)

### Vergabe von Port-Nummern

#### Einführung

Beim Anlegen einer Open User Communication wird automatisch der Wert 2000 als Port-Nummer vergeben.

Für die Port-Nummern sind Werte von 1 bis 49151 erlaubt. Innerhalb dieses Bereichs können Sie Port-Nummern frei vergeben. Da einige Ports jedoch bereits systembedingt verwendet

werden können, wird die Verwendung von Port-Nummern im Bereich von 2000 bis 5000 empfohlen.

**Hinweis**

Port-Nummern müssen eindeutig sein. Bei doppelt vergebenen Port-Nummern wird die Verbindungsprojektierung bzw. ein entsprechender Bausteinanruf mit einem Fehler abgelehnt.

**Übersicht der Port-Nummern**

Die folgende Tabelle zeigt eine Zusammenfassung der Systemreaktionen auf verschiedene Port-Nummern.

| Port-Nr.  | Beschreibung                                      | Systemreaktion   |
|---|---|--|
| 2000 ... 5000   | empfohlener Bereich                               | Keine Warnung, keine Fehlermeldung bei Eingabe<br>Port-Nummer wird zugelassen und übernommen |
| 1 ... 1999, 5001 ... 49151                                  | nutzbar aber ausserhalb des empfohlenen Bereiches | Warnmeldung bei Eingabe<br>Port-Nummer wird zugelassen und übernommen                        |
| 0, 20, 21, 25, 80, 102, 135, 161, 34962 ... 34964           | bedingt verwendbar*                               |  |
| 53, 80, 102, 135, 161, 162, 443, 520, 9001, 34962 ... 34964 | bedingt verwendbar**                              |  |

\* Für bestimmte Funktionen festgelegte Ports:  
 0: ANY - Port-Nummer wird automatisch von S7-1500 CPU als Wert vergeben (>49151)  
 20: FTP-Datenübertragung  
 21: FTP-Kontrolle  
 25: TMAIL\_C (Simple Mail transfer Protokoll)  
 80: Webserver  
 102: ISO-on-TCP (RFC1006)  
 135: DCE Endpoint Mapper für PROFINET  
 161: SNMP (Simple Network Management Protokoll)  
 34962 ... 34964: PROFINET

**Hinweis**

Bei UDP/TCP gibt der Anwender auf dem aktiven Verbindungsendpunkt üblicherweise den Wert 0 für den lokalen Port vor. In diesem Fall wählt das CPU- Betriebssystem den nächsten freien Port oberhalb 49151 aus. Beim passiven Verbindungsendpunkt wird umgekehrt der Partner-Port meist mit 0 vorbelegt. In der Verbindungsparametrierung wird der entsprechende Parameter deaktiviert.

\*\* Diese Ports sind abhängig vom Funktionsumfang der verwendeten CPU gesperrt. Die Belegung dieser Ports geht aus den Dokumentationen der jeweiligen CPUs hervor.

## Siehe auch

- Beschreibung der Verbindungsparameter (Seite 462)
- Verbindungen anlegen und parametrieren (Seite 466)

## Rücklesbarkeit der Parameter für die Verbindungsbeschreibung

### Ändern von Parameterwerten in der Verbindungsbeschreibung

Im Verbindungsbeschreibungs-DB wird von der Verbindungsparametrierung die Verbindungsbeschreibung für genau eine Verbindung der Open User Communication eingetragen.

Sie können die Parameterwerte des Verbindungsbeschreibungs-DB außerhalb der Verbindungsparametrierung im Anwenderprogramm ändern. Verbindungsbeschreibungs-DBs mit nachträglich von Ihnen geänderten Werten können von der Verbindungsparametrierung rückgelesen werden. Im Inspektorfenster werden unter "Eigenschaften > Konfiguration > Verbindungsparameter" nur die im Verbindungsbeschreibungs-DB abgelegten Verbindungsparameter angezeigt.

---

#### Hinweis

Im laufenden Anwenderprogramm dürfen die Werte nur geändert werden, wenn die Anweisungen TCON, TSEND\_C oder TRCV\_C nicht bearbeitet werden bzw. die referenzierte Verbindung nicht aufgebaut ist.

---

Verschachtelt eingetragene Verbindungsbeschreibungen in nur per Offset-Referenzierung auffindbare DB-Typen (z. B. Global-DB) werden von der Verbindungsparametrierung nicht unterstützt.

Die Struktur der Verbindungsbeschreibung kann nicht geändert werden.

## Rücklesbarkeit der einzelnen Verbindungsparameter

Für den Parameter "Adresse" des Kommunikationspartners in einer TCP- oder ISO-on-TCP-Verbindung wird dessen IP-Adresse aus dem Parameter "rem\_staddr" der Verbindungsbeschreibung angezeigt.

Folgende Werte sind außerdem aus der Verbindungsbeschreibung rückladbar:

- Verbindungstyp
- Lokale Verbindungs-ID
- Verbindungsaufbau aktiv/passiv (nicht bei UDP)
- Lokaler TSAP (nur bei ISO-on-TCP)
- Partner-TSAP (nur bei ISO-on-TCP)
- Lokaler Port (nur bei TCP und UDP)
- Partnerport (nur bei TCP)

Die Werte der Parameter der Verbindungs-ID des Kommunikationspartners, der Verbindungsdaten sowie des Verbindungsaufbaus sind nicht Inhalt der Verbindungsbeschreibung im lokalen Verbindungsbeschreibungs-DB. Diese Parameter können somit beim erneuten Öffnen der Verbindungsparametrierung nicht angezeigt werden. Der Verbindungsaufbau des Partners ergibt sich aber aus dem lokalen Verbindungsaufbau und wird daher auch angezeigt.

In der Klappliste "Partner" kann jederzeit ein neuer Kommunikationspartner ausgewählt werden.

Bei Auswahl einer im Projekt erkannten CPU als spezifizierten Kommunikationspartner werden die Eingabemöglichkeiten für die Verbindungs-ID und die Verbindungsdaten wieder eingeblendet.

### Siehe auch

Verbindungsparameter nach TCON\_Param (Seite 472)

Beschreibung der Verbindungsparameter (Seite 462)

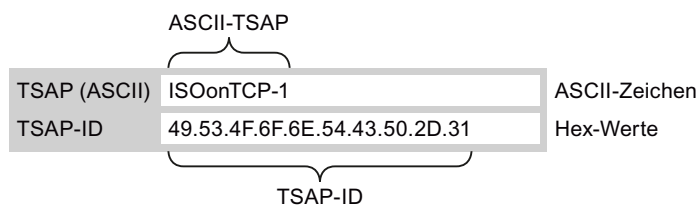
### TSAP-Struktur

#### Einführung

Für eine Verbindung mit dem Verbindungstyp ISO-on-TCP müssen für beide Kommunikationspartner Transportation Service Access Points (TSAPs) vergeben werden. TSAP-IDs werden automatisch nach dem Anlegen einer ISO-on-TCP-Verbindung vergeben. Um die Eindeutigkeit von TSAP-IDs innerhalb eines Geräts zu gewährleisten, können Sie in der Verbindungsparametrierung die vorgegebenen TSAPs ändern.

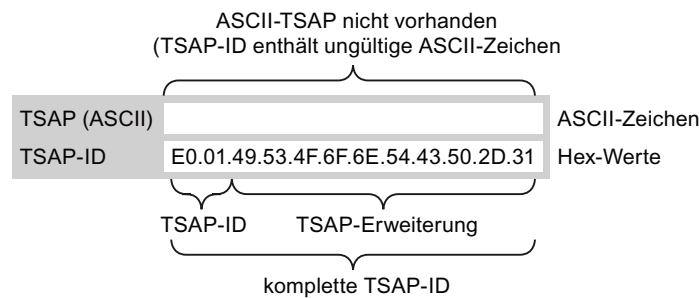
#### Aufbau der TSAPs

Für die freie Vergabe von TSAPs müssen Sie bestimmte Regeln einhalten. Ein TSAP muss eine bestimmte Anzahl an Bytes beinhalten, die in der Verbindungsparametrierung als hexadezimale Werte (TSAP-ID) oder als ASCII-Zeichen (ASCII-TSAP) angezeigt und eingegeben werden können:



Einträge oder Änderungen der TSAP-ID oder der ASCII-TSAP in den entsprechenden Eingabefeldern wirken sich immer auch auf das jeweils andere Anzeigeformat aus.

Wenn ein TSAP keine gültigen ASCII-Zeichen enthält, wird der TSAP nur als TSAP-ID angezeigt, aber nicht als ASCII-TSAP. Dies ist nach dem Anlegen einer Verbindung der Fall. Die ersten beiden Hex-Zeichen als TSAP-ID kennzeichnen die Kommunikationsart und den Baugruppenträger/Steckplatz. Da diese Zeichen für eine CPU keine gültigen ASCII-Zeichen sind, wird in diesem Fall der ASCII-TSAP nicht angezeigt:



Neben den Regeln für Länge und Aufbau der TSAPs müssen Sie auf die Eindeutigkeit der TSAP-ID achten. TSAPs werden nicht automatisch eindeutig vergeben.

### Länge und Inhalt der TSAPs

So ist ein TSAP aufgebaut:

- TSAP-ID mit TSAP-Erweiterung  
Länge = 2 bis 16 Byte  
x\_tsap\_id[0] = 0xE0 (Open User Communication)  
x\_tsap\_id[1] (Bits 0 bis 4) = Steckplatz-Nummer der CPU  
x\_tsap\_id[1] (Bits 5 bis 7) = Baugruppenträger-Nummer der CPU  
x\_tsap\_id[2...15] = beliebige Zeichen (TSAP-Erweiterung, optional)  
(x = loc (lokal) oder x = rem (Partner))
- TSAP-ID als ASCII-TSAP  
Länge = 3 bis 16 Byte  
x\_tsap\_id[0 bis 2] = 3 ASCII-Zeichen (0x20 bis 0x7E)  
x\_tsap\_id[3...15] = beliebige Zeichen (optional)  
(x = loc (lokal) oder x = rem (Partner))

Die folgende Tabelle zeigt den schematischen Aufbau einer TSAP-ID:

| TSAP-ID            | tsap_id_len | tsap_id[0]  | tsap_id[1]                      | tsap_id[2..15]            | tsap_id[3..15]            |
|--------------------|-------------|-------------|---------------------------------|---------------------------|---------------------------|
| ...mit Erweiterung | 2...16 Byte | 0xE0        | 0x01 oder<br>0x02 oder<br>0x00* | Erweiterung<br>(optional) | Erweiterung<br>(optional) |
| ...als ASCII-TSAP  | 3...16 Byte | 0x20...0x7E | 0x20...0x7E                     | 0x20...0x7                | beliebig<br>(optional)    |

\* Eine S7-1200/1500 CPU steckt für gewöhnlich auf Baugruppenträger 0 und Steckplatz 1, eine S7-300/400 CPU auf Baugruppenträger 0 und Steckplatz 2. Aus diesem Grund gilt für die zweite Stelle der TSAP-ID mit Erweiterung der Hex-Wert 01 oder 02. Sollte es sich bei dem Verbindungspartner um eine un spezifizierte CPU handeln, z. B. um ein Fremdgerät, ist für die Steckplatzadresse auch der Hex-Wert 00 zulässig.

#### Hinweis

Bei un spezifizierten Kommunikationspartnern dürfen die lokale TSAP-ID und die Partner-TSAP-ID eine Länge von 0 bis 16 Byte haben, wobei alle Hex-Werte von 00 bis FF erlaubt sind.

### ASCII-Codetabelle für die Eingabe von ASCII-TSAPs

Für die Eingabe eines ASCII-TSAPs in der Verbindungsparametrierung sind lediglich die Hexadezimal-Werte von 20 bis 7E zulässig:

| Code | ..0 | ..1 | ..2 | ..3 | ..4 | ..5 | ..6 | ..7 | ..8 | ..9 | ..A | ..B | ..C | ..D | ..E | ..F |
|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 2..  |     | !   | "   | #   | \$  | %   | &   | '   | (   | )   | *   | +   | ,   | -   | .   | /   |
| 3..  | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | :   | ;   | <   | =   | >   | ?   |
| 4..  | @   | A   | B   | C   | D   | E   | F   | G   | H   | I   | J   | K   | L   | M   | N   | O   |
| 5..  | P   | Q   | R   | S   | T   | U   | V   | W   | X   | Y   | Z   | [   | \   | ]   | ^   | _   |
| 6..  | `   | a   | b   | c   | d   | e   | f   | g   | h   | i   | j   | k   | l   | m   | n   | o   |
| 7..  | p   | q   | r   | s   | t   | u   | v   | w   | x   | y   | z   | {   |     | }   | ~   |     |

#### Siehe auch

- Beispiele zur TSAP-Vergabe (Seite 482)
- Beschreibung der Verbindungsparameter (Seite 462)
- Verbindungen anlegen und parametrieren (Seite 466)

### Beispiele zur TSAP-Vergabe

Die folgenden Beispiele zeigen die Bearbeitung der TSAPs für CPUs der S7-1200/1500 (CPU auf Steckplatz 1) unter verschiedenen Gesichtspunkten:

- Beispiel 1: Anlegen einer neuen Verbindung für eine PLC-PLC-Kommunikation
- Beispiel 2: Eingabe eines lokalen ASCII-TSAPs
- Beispiel 3: Eingabe einer TSAP-Erweiterung in der TSAP-ID
- Beispiel 4: Fehlerhaftes Editieren der TSAP-ID
- Beispiel 5: Eingabe eines ASCII-TSAPs über das Eingabefeld "TSAP-ID"

#### Beispiel 1: Anlegen einer neuen Verbindung für eine PLC-PLC-Kommunikation

Nachdem Sie für die Open User Communication eine neue Verbindung mit zwei PLCs angelegt haben, wird automatisch die TSAP-Erweiterung "ISOonTCP-1" vergeben.

Mit dieser TSAP-Erweiterung ergibt sich die TSAP-ID E0.01.49.53.4F.6F.6E.54.43.50.2D.31, die automatisch in den Verbindungsbeschreibungs-DB und in die Eingabefelder des lokalen und Partner-TSAPs eingetragen wird. Die Eingabefelder der ASCII-TSAPs bleiben leer:

|              | Lokaler TSAP                        | Partner-TSAP                        |
|--------------|-------------------------------------|-------------------------------------|
| TSAP (ASCII) |                                     |                                     |
| TSAP-ID      | E0.01.49.53.4F.6F.6E.54.43.50.2D.31 | E0.01.49.53.4F.6F.6E.54.43.50.2D.31 |

Sie können die Werte in den Eingabefeldern der TSAP-ID und des ASCII-TSAP jederzeit ändern.



Das Eingabefeld der TSAP-ID zeigt den vollständigen TSAP, abgelegt im Datenbaustein der Verbindungsbeschreibung. Die auf 16 Zeichen begrenzte TSAP-ID mit TSAP-Erweiterung wird nicht im Eingabefeld "TSAP (ASCII)" angezeigt, da das Zeichen E0 für den ASCII-TSAP kein gültiges Zeichen darstellt.

Wenn es sich bei der angezeigten TSAP-ID um eine gültige ASCII-TSAP handelt, wird sie im Eingabefeld "TSAP (ASCII)" angezeigt.

Änderungen in den Eingabefeldern für TSAP-ID und ASCII-TSAP beeinflussen sich gegenseitig.

### Beispiel 2: Eingabe eines lokalen ASCII-TSAPs

Wenn Sie eine neue Verbindung angelegt haben und im Eingabefeld "TSAP (ASCII)" einen ASCII-Wert für den lokalen TSAP vergeben, z. B. "ISOonTCP-1", wird die resultierende TSAP-ID automatisch erzeugt.

Wenn Sie das Eingabefeld "TSAP (ASCII)" verlassen, wird automatisch die Einhaltung der ASCII-Zeichenbeschränkung von 3 bis 16 Zeichen geprüft und die resultierende TSAP-ID in das entsprechende Eingabefeld eingetragen:

|              | Lokaler TSAP                  | Partner-TSAP                        |
|--------------|-------------------------------|-------------------------------------|
| TSAP (ASCII) | ISOonTCP-1                    |                                     |
| TSAP-ID      | 49.53.4F.6F.6E.54.43.50.2D.31 | E0.01.49.53.4F.6F.6E.54.43.50.2D.31 |

### Beispiel 3: Eingabe einer TSAP-Erweiterung in der TSAP-ID

Wenn Sie nach dem Anlegen einer Verbindung und der Eingabe eines ASCII-TSAPs (siehe Beispiele 1 und 2) in das Eingabefeld der lokalen TSAP-ID dem TSAP-Wert die Zeichen "E0.01." voranstellen, wird nach Verlassen des Eingabefeldes kein ASCII-TSAP mehr angezeigt:

|              | Lokaler TSAP                        | Partner-TSAP                        |
|--------------|-------------------------------------|-------------------------------------|
| TSAP (ASCII) |                                     |                                     |
| TSAP-ID      | E0.01.49.53.4F.6F.6E.54.43.50.2D.31 | E0.01.49.53.4F.6F.6E.54.43.50.2D.31 |

Nachdem Sie das Eingabefeld der TSAP-ID verlassen haben, wird automatisch geprüft, ob das erste Zeichen der TSAP-ID ein gültiges ASCII-Zeichen ist. Da das Zeichen "E0" in der TSAP-ID nun kein gültiges Zeichen für den ASCII-TSAP ist, wird im Eingabefeld "TSAP (ASCII)" kein ASCII-TSAP mehr angezeigt.

Bei der Verwendung gültiger ASCII-Zeichen erfolgt die Prüfung auf die Einhaltung der Länge von 2 bis 16 Zeichen.

**Beispiel 4: Fehlerhaftes Editieren der TSAP-ID**

Wenn Sie bei einer TSAP-ID beginnend mit "E0.01" den Hex-Wert "E0" entfernen, entspricht die nun mit "01" beginnende TSAP-ID nicht mehr den Regeln und ist ungültig:

|              | Lokaler TSAP                     | Partner-TSAP                        |
|--------------|----------------------------------|-------------------------------------|
| TSAP (ASCII) |                                  |                                     |
| TSAP-ID      | 01.49.53.4F.6F.6E.54.43.50.2D.31 | E0.01.49.53.4F.6F.6E.54.43.50.2D.31 |

Nach Verlassen des Eingabefeldes wird eine Meldung ausgegeben, da es sich bei der TSAP-ID weder um einen gültigen ASCII-TSAP handelt (dieser müsste als ersten Wert einen Hex-Wert im Bereich von 20 bis 7E haben), noch um eine gültige TSAP-ID (diese müsste als ersten Wert die Kennung "E0" besitzen).

**Beispiel 5: Eingabe eines ASCII-TSAPs über das Eingabefeld "TSAP-ID"**

Wenn Sie aus der fehlerhaften TSAP-ID des Beispiels 4 nach dem Wert "E0" auch noch den Wert "01" entfernen, beginnt die TSAP-ID mit dem Hex-Wert 49. Dieser Wert liegt im erlaubten Bereich für ASCII-TSAPs:

|              | Lokaler TSAP                  | Partner-TSAP                        |
|--------------|-------------------------------|-------------------------------------|
| TSAP (ASCII) |                               |                                     |
| TSAP-ID      | 49.53.4F.6F.6E.54.43.50.2D.31 | E0.01.49.53.4F.6F.6E.54.43.50.2D.31 |

Wenn Sie das Eingabefeld verlassen, wird die TSAP-ID als gültiger ASCII-TSAP erkannt und der resultierende ASCII-TSAP "ISOonTCP-1" wird in das Eingabefeld "TSAP (ASCII)" geschrieben.

**Siehe auch**

TSAP-Struktur (Seite 480)

Beschreibung der Verbindungsparameter (Seite 462)

**Kommunikation über PUT- und GET-Anweisungen**

**Grundlagen zur Kommunikation über die PUT/GET-Anweisung**

**Grundlagen zu den PUT/GET-Anweisungen**

Um Daten zwischen zwei CPUs über eine S7-Verbindung auszutauschen, verwenden Sie die Anweisungen PUT und GET.

Mit Hilfe der Anweisung GET können Sie Daten aus einer Partner-CPU auslesen. Mit der Anweisung PUT steuern Sie über das Anwenderprogramm das Schreiben von Variablen beim Kommunikationspartner. Neben den Anweisungen PUT und GET sind keine zusätzlichen Kommunikationsfunktionen für das Lesen und Schreiben von Variablen vorgesehen.

Um Ihnen die Verwendung der beiden Anweisungen zu erleichtern, legen Sie alle notwendigen Parameter für die Verbindung sowie alle Bausteinparameter im Inspektorfenster des Programmiereditors fest.

## Voraussetzung

Um die Anweisungen PUT und GET verwenden zu können, müssen die folgenden Voraussetzungen erfüllt sein:

- Mindestens eine CPU der S7-1200/1500 oder S7-300/400 muss im Projekt angelegt sein. Auf einer S7-1200 CPU muss die Firmware 2.0 oder höher installiert sein. Wenn Sie noch keine zweite CPU im Projekt angelegt haben, können Sie die Verbindung zunächst zu einem un spezifizierten Partner aufbauen.
- Es muss eine S7-Verbindung zwischen beiden CPUs bestehen. Wenn Sie noch keine Verbindung zwischen zwei CPUs hergestellt haben, wird automatisch eine Verbindung während der Projektierung der Anweisungen erstellt.
- Für beide Anweisungen ist ein Instanz-Datenbaustein nötig, in dem alle Daten hinterlegt werden, die von der Anweisung genutzt werden. Der Instanz-Datenbaustein wird automatisch angelegt, sobald Sie die Anweisung PUT oder GET auf ein Netzwerk im Programmiereditor ziehen. Die Instanz-Datenbausteine dürfen für den korrekten Programmablauf nicht verändert werden und sind daher know-how-geschützt. Sie haben nur lesenden Zugriff auf die Instanz-Datenbausteine.

## Siehe auch

Übersicht über die Verbindungsparametrierung (Seite 485)

Auftragsstart parametrieren (Seite 491)

PUT: Schreib- und Sendebereich parametrieren (Seite 491)

GET: Lese- und Speicherbereich parametrieren (Seite 492)

## Verbindungsparametrierung

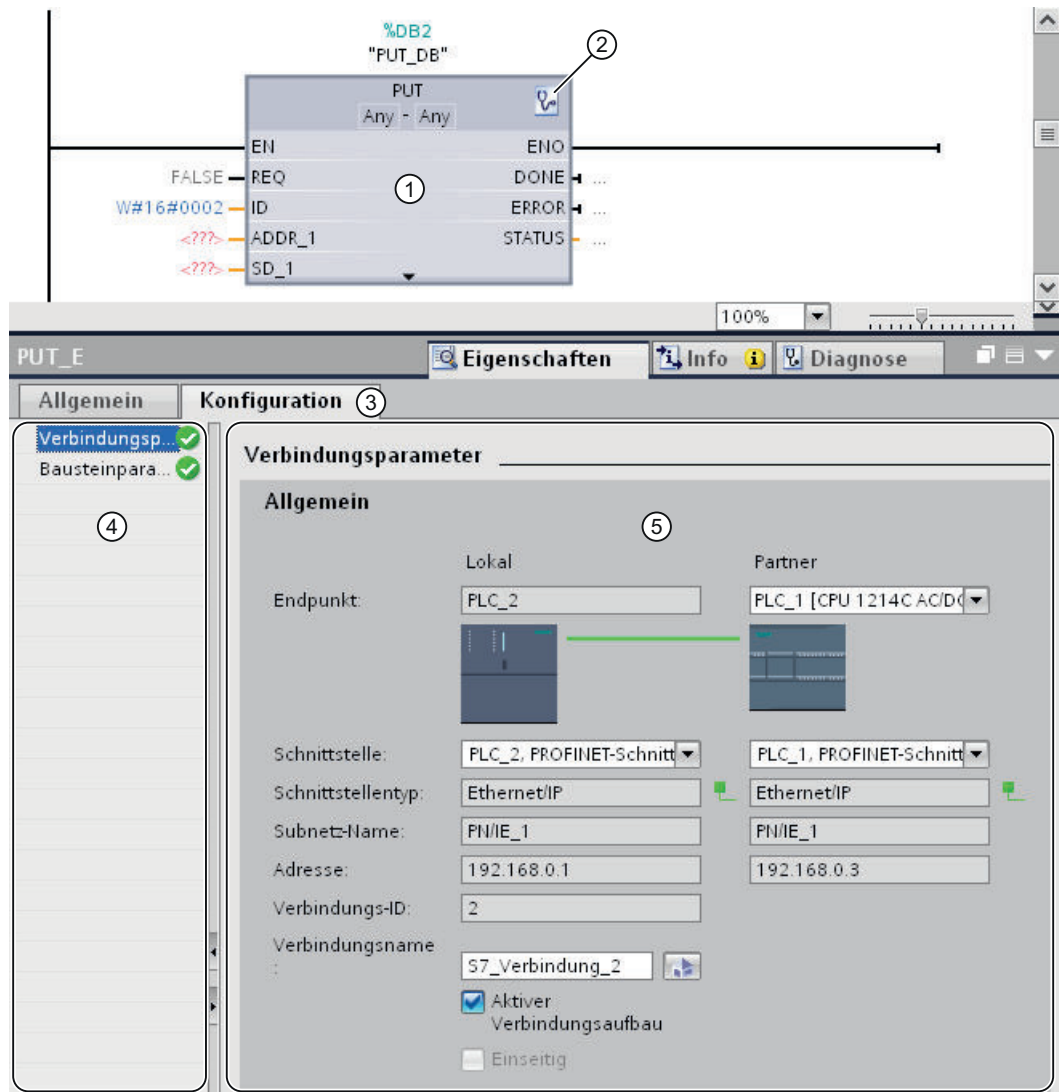
### Übersicht über die Verbindungsparametrierung

### Einführung

Die Verbindungsparametrierung für die Anweisungen PUT und GET erfolgt im Inspektorfenster des Programmiereditors. Alle Parameter werden im zugehörigen Instanz-Datenbaustein gespeichert.

### Aufbau der Verbindungsparametrierung

Die Verbindungsparametrierung besteht aus folgenden Komponenten:



- ① Kommunikationsanweisung für PUT oder GET
- ② Aufruf von Online- und Diagnosefunktionen
- ③ Register "Konfiguration" im Register "Eigenschaften"
- ④ Bereichsnavigation des Registers "Konfiguration"
- ⑤ Allgemeine Eigenschaften der Verbindungsparameter

### Anzeigen von Online- und Diagnosefunktionen

Wenn Sie auf das Symbol zum Start der Online- und Diagnosefunktionen klicken, wird die zugehörige CPU automatisch online verbunden. Die Verbindungstabelle in der Netzansicht wird geöffnet. Außerdem werden im Inspektorfenster das Register "Diagnose" und die Verbindungsinformationen angezeigt.

## Eintragen der Verbindungsparameter

Tragen Sie die gewünschten Verbindungsparameter im Register "Konfiguration" ein. Sie finden in der Bereichsnavigation des Registers "Konfiguration" die Gruppe "Verbindungsparameter". Diese Gruppe enthält die Verbindungsparametrierung. Hier können Sie systemunterstützt die Parameter für die Verbindungen eintragen. Wenn alle erforderlichen Parameter vorhanden sind, wird in der Bereichsnavigation hinter die Gruppe "Verbindungsparameter" ein Häkchen gesetzt.

## Siehe auch

- Auftragsstart parametrieren (Seite 491)
- PUT: Schreib- und Sendebereich parametrieren (Seite 491)
- GET: Lese- und Speicherbereich parametrieren (Seite 492)

## Beschreibung der Verbindungsparameter

### Überblick

Die folgende Tabelle zeigt die allgemeinen Verbindungsparameter:

| Parameter         | Beschreibung  |
|-------------------|---|
| Endpunkt          | Die Namen des lokalen Endpunkts und des Partner Endpunkts wird angezeigt. <ul style="list-style-type: none"> <li>• Lokaler Endpunkt<br/>Der lokale Endpunkt ist die CPU, in der Sie die PUT/GET-Anweisung programmieren.</li> <li>• Partner-Endpunkt<br/>Der Partner-Endpunkt wird über die Klappliste ausgewählt. In der Klappliste werden alle verfügbaren möglichen Verbindungspartner angezeigt, darunter auch un spezifizierte Verbindungspartner für Geräte, deren Daten innerhalb des Projekts nicht bekannt sind. Solange noch kein Verbindungspartner eingestellt ist, bleiben alle weiteren Parameter in der Maske deaktiviert.</li> </ul>  |
| Schnittstelle     | Die Schnittstelle der Partner-CPU wird angezeigt. Die Anzeige der Partner-Schnittstelle erfolgt erst nach Auswahl einer spezifizierten Partner-CPU.   |
| Schnittstellentyp | Die Art der Schnittstelle, über die die Kommunikation abgewickelt wird, wird angezeigt.   |
| Subnetz-Name      | Das Subnetz des lokalen Endpunkts wird angezeigt, wenn dieses vorhanden ist. Die Anzeige des Partner-Subnetzes erfolgt erst nach Auswahl des Partner-Endpunkts.<br>Wenn mindestens einer der beiden Verbindungspartner nicht mit einem Subnetz verbunden ist, werden die beiden Verbindungspartner automatisch miteinander vernetzt. Dabei wird der nicht vernetzte Partner an dasselbe Subnetz angeschlossen, über das der andere Partner bereits mit einem Netz verbunden ist.<br>Eine Verbindung von Verbindungspartnern an unterschiedlichen Subnetzen ist nur über IP- oder S7-Routing möglich. Die IP-Routing-Einstellungen können über die betroffenen Schnittstellen-Eigenschaften bearbeitet werden. |
| Adresse           | Die IP-Adresse des lokalen Endpunkts wird angezeigt. Die Anzeige der IP-Adresse des Partners erfolgt erst nach Auswahl des Partner Endpunkts.<br>Wenn Sie einen un spezifizierten Verbindungspartner ausgewählt haben, ist das Eingabefeld leer und rot hinterlegt. Sie müssen in dem Fall eine gültige IP-Adresse für den Verbindungspartner angeben.  |

| Parameter                 | Beschreibung  |
|---------------------------|---|
| Verbindungs-ID            | Die aktuell eingestellte Verbindungs-ID wird angezeigt. Sie können die Verbindungs-ID in der Verbindungstabelle in der Netzsicht ändern. Außerdem können Sie während der Verbindungsparametrierung direkt auf die Verbindungstabelle zugreifen. Klicken Sie dazu auf das Symbol "Neue Verbindung erstellen".  |
| Verbindungsname           | Der Name der Verbindung, welche beim Einfügen der PUT/GET-Anweisung automatisch angelegt wurde, wird angezeigt. Sie können den Namen der Verbindung ändern, indem Sie einen anderen Namen in das Feld eingeben. Sie können außerdem eine neue Verbindung erstellen oder bereits vorhandene Verbindungen bearbeiten, indem Sie auf das Symbol "Neue Verbindung erstellen" klicken. |
| Aktiver Verbindungsaufbau | Bestimmen Sie über das Optionsfeld "Aktiver Verbindungsaufbau", welcher Partner die Kommunikation startet. Beim Anlegen der Verbindung wird automatisch zunächst der lokale Partner für den Verbindungsaufbau bestimmt. Wenn ein Gerät den aktiven Verbindungsaufbau nicht unterstützt, müssen Sie beim anderen Partner den aktiven Verbindungsaufbau aktivieren.                 |
| Einseitig                 | Ist das Optionskästchen aktiviert, ist der Verbindungspartner Server für diese Verbindung. Er kann nicht aktiv senden oder empfangen. Dies entspricht dem Verhalten der PUT/GET-Anweisungen. Andere Anweisungen sind in diesem Fall nicht möglich. Wenn das Optionskästchen deaktiviert ist, dann ist die Verwendung anderer Anweisungen zur Kommunikation ebenfalls möglich.     |

### Verbindungsparametrierung starten

Die Verbindungsparametrierung für PUT und GET können Sie im Inspektorfenster durchführen, sobald Sie eine PUT/GET-Anweisung in einen Programmbaustein eingefügt und selektiert haben.

### Vorgehen

Um PUT/GET-Anweisungen einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen" den Ordner "Kommunikation > S7-Kommunikation".
2. Ziehen Sie eine PUT/GET-Anweisung per Drag & Drop in ein Netzwerk. Der Dialog "Aufrufoptionen" wird geöffnet.
3. Optional: Bearbeiten Sie im Dialog "Aufrufoptionen" die Eigenschaften des Instanz-DBs. Sie haben dabei folgende Möglichkeiten:
  - Ändern Sie den vorgegebenen Namen.
  - Wählen Sie das Optionsfeld "manuell", um eine eigene Nummer zu vergeben.
4. Klicken Sie auf "OK".

### Ergebnis

Für die eingefügte Anweisung PUT oder GET wird ein entsprechender Instanz-Datenbaustein angelegt. Bei S7-300 CPUs wird zusätzlich ein Funktionsbaustein in den Programmressourcen angelegt.

Bei einer selektierten PUT/GET-Anweisung sehen Sie im Inspektorfenster unter "Eigenschaften" das Register "Konfiguration". Die Gruppe "Verbindungsparameter" in der Bereichsnavigation enthält die Verbindungsparametrierung, die Sie nun durchführen können.

### **Siehe auch**

Verbindungen anlegen und parametrieren (Seite 489)

Verbindungen löschen (Seite 490)

### **Verbindungen anlegen und parametrieren**

In der Verbindungsparametrierung der PUT/GET-Anweisungen können Sie S7-Verbindungen anlegen und parametrieren. Geänderte Werte werden von der Verbindungsparametrierung sofort auf Eingabefehler geprüft.

### **Voraussetzung**

Es existiert eine CPU mit einer Kommunikationsanweisung PUT oder GET.

### **Vorgehen**

Um eine S7-Verbindung über PUT/GET-Anweisungen zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Programmiereditor den Aufruf der Anweisung PUT oder GET.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften > Konfiguration".
3. Selektieren Sie die Gruppe "Verbindungsparameter". Solange Sie noch keinen Verbindungspartner selektiert haben, ist nur die leere Klappliste für den Partner-Endpunkt aktiv. Alle anderen Eingabemöglichkeiten sind deaktiviert.  
Es werden die bereits bekannten Verbindungsparameter angezeigt:
  - Name des lokalen Endpunkts
  - Schnittstelle des lokalen Endpunkts
  - IP-Adresse des lokalen Endpunkts

4. Wählen Sie in der Klappliste des Partner-Endpunkts einen Verbindungspartner. Als Kommunikationspartner kommt ein unspezifiziertes Gerät oder eine im Projekt vorhandene CPU in Frage.  
Die folgenden Parameter werden automatisch eingetragen, sobald Sie den Verbindungspartner gewählt haben:
  - Name des Partner-Endpunkts
  - Schnittstelle des Partner-Endpunkts. Wenn mehrere Schnittstellen zur Verfügung stehen, dann können Sie die Schnittstelle bei Bedarf ändern.
  - Schnittstellentyp des Partner-Endpunkts
  - Subnetz-Name beider Endpunkte
  - IP-Adresse des Partner-Endpunkts
  - Name der Verbindung, die für die Kommunikation genutzt wird. Ist noch keine Verbindung vorhanden, wird automatisch eine erstellt.
5. Ändern Sie bei Bedarf den Verbindungsnamen im Eingabefeld "Verbindungsname" ab. Wenn Sie eine neue Verbindung erstellen, oder eine vorhandene Verbindung bearbeiten möchten, klicken Sie auf das Symbol "Neue Verbindung erstellen".

---

**Hinweis**

Die Anweisungen PUT und GET zwischen zwei Kommunikationspartnern sind erst dann lauffähig, wenn sowohl die Hardware-Konfiguration wie auch der Programmteil für den Partner-Endpunkt in die Hardware geladen wurden. Achten Sie darauf, dass Sie für eine funktionierende Kommunikation nicht nur die Verbindungsbeschreibung der lokalen CPU in das Gerät laden, sondern auch die der Partner-CPU.

---

## Verbindungen löschen

Eine Verbindung, die beim Einfügen einer PUT/GET-Anweisung automatisch erstellt wurde, taucht wie jede gewöhnliche Verbindung in der Verbindungstabelle der Netzsicht auf. Sie lässt sich daher in der Verbindungstabelle löschen.

## Vorgehen

Um eine Verbindung zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Verbindungstabelle in der Netzsicht.
2. Selektieren Sie die Verbindung, die Sie löschen möchten, in der Verbindungstabelle.
3. Klicken Sie mit der rechten Maustaste auf die Verbindung und wählen Sie im Kontextmenü den Befehl "Löschen".

## Ergebnis

Die Verbindung ist gelöscht. Die PUT/GET-Anweisung sowie die zugehörigen Instanzdatenbausteine bleiben erhalten und müssen bei Bedarf manuell gelöscht werden.

Um die PUT/GET-Anweisung weiter zu verwenden, müssen Sie die Verbindungsprojektierung im Inspektorfenster des Programmiereditors erneut durchführen, da beim Löschen der



Verbindung alle Verbindungsparameter ebenfalls gelöscht werden. Legen Sie in diesem Fall einen neuen Kommunikationspartner und eine passende Verbindung fest.

## **Bausteinparametrierung**

### **Auftragsstart parametrieren**

Um eine Kommunikation über die PUT/GET-Anweisung zu beginnen, müssen Sie ein Ereignis festlegen, welches die Anweisung aktiviert. Dieses Ereignis wird als Steuerparameter (REQ) bezeichnet. Die Aktivierung des Kommunikationsauftrags erfolgt, sobald am Steuerparameter REQ eine positive Flanke anliegt.

Beachten Sie, dass der Steuerparameter REQ beim Erstaufwurf mit FALSE belegt ist.

### **Voraussetzung**

- Der Programmiereditor ist geöffnet.
- Sie haben bereits eine PUT/GET-Anweisung eingefügt.
- Es besteht eine Verbindung zwischen zwei Kommunikationspartnern.

### **Vorgehen**

Um den Steuerparameter REQ festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die PUT/GET-Anweisung im Programmiereditor.
2. Öffnen Sie das Register "Konfiguration" im Inspektorfenster.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Bausteinparametrierung".
4. Wählen Sie im Feld "REQ" eine Variable des Datentyps "BOOL" aus, um die Ausführung der Anweisung zu initialisieren. Sie können alternativ auch eine vorhergehende Anweisung im Programmiereditor verschalten.

### **Siehe auch**

PUT: Schreib- und Sendebereich parametrieren (Seite 491)

GET: Lese- und Speicherbereich parametrieren (Seite 492)

### **PUT: Schreib- und Sendebereich parametrieren**

Sie müssen bei der Kommunikation über die PUT-Anweisung festlegen, in welchem Speicherbereich der Partner-CPU die Daten geschrieben werden sollen. Außerdem müssen Sie den Speicherbereich der lokalen CPU festlegen, aus dem die Daten gelesen werden sollen.

### Voraussetzung

- Der Programmiereditor ist geöffnet.
- Sie haben bereits eine PUT-Anweisung eingefügt.
- Es besteht eine Verbindung zwischen zwei Kommunikationspartnern.

### Vorgehen

Um den Lese- und Speicherbereich für die Anweisung festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die PUT-Anweisung im Programmiereditor.
2. Öffnen Sie das Register "Konfiguration" im Inspektorfenster.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Bausteinparametrierung".
4. Wählen Sie im Feld "Ein-Ausgänge > Schreibbereich (ADDR\_1) > Start" einen Zeiger mit dem Datentyp "REMOTE" auf den zu beschreibenden Bereich der Partner-CPU.  
Es ist nur absolute Adressierung zulässig.  
Beispiel: P#DB10.DBX5.0 Byte 10
5. Geben Sie im Feld "Länge" die Länge des Schreibbereichs ein und wählen Sie aus der Klappliste den Datentyp des Speicherbereichs aus.
6. Wählen Sie im Feld "Ein-Ausgänge > Sendebereich (SD\_1) > Start" einen Zeiger auf den Bereich in der lokalen CPU, der die zu sendenden Daten enthält.
7. Geben Sie die Länge des zu lesenden Speicherbereichs im Feld Länge ein und wählen Sie den Datentyp aus der Klappliste.  
Zulässig sind nur die Datentypen BOOL (bei einem Bitfeld muss als Adresse "0" und als Länge ein ganzzahlig Vielfaches von Byte verwendet werden), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER.  
Wenn der VARIANT-Zeiger auf einen DB zugreift, ist der DB immer zu spezifizieren (z. B.: P#DB10.DBX5.0 Byte 10).

### Siehe auch

GET: Lese- und Speicherbereich parametrieren (Seite 492)

### GET: Lese- und Speicherbereich parametrieren

Sie müssen bei der Kommunikation über die GET-Anweisung festlegen, in welchen Speicherbereich der lokalen CPU die Daten geschrieben werden sollen. Außerdem müssen Sie den Lesebereich der Partner-CPU festlegen, aus dem die Daten gelesen werden sollen.

### Voraussetzung

- Der Programmiereditor ist geöffnet.
- Sie haben bereits eine GET-Anweisung eingefügt.
- Es besteht eine Verbindung zwischen zwei Kommunikationspartnern.

## Vorgehen

Um den Lese- und Speicherbereich für die Anweisung festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die GET-Anweisung im Programmiereditor.
2. Öffnen Sie das Register "Konfiguration" im Inspektorfenster.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Bausteinparametrierung".
4. Wählen sie im Feld "Ein-Ausgänge > Lesebereich (ADDR\_1) > Start" einen Zeiger mit dem Datentyp "REMOTE" auf den zu lesenden Bereich der Partner-CPU.  
Es ist nur absolute Adressierung zulässig.  
Beispiel: P#DB10.DBX5.0 Byte 10
5. Geben Sie im Feld "Länge" die Länge des Lesebereichs ein und wählen Sie aus der Klappliste den Datentyp des Speicherbereichs aus.
6. Wählen Sie im Feld "Ein-Ausgänge > Speicherbereich (RD\_1) > Start" einen Zeiger auf den Bereich in der lokalen CPU, in dem die gelesenen Daten abgelegt werden.
7. Geben Sie die Länge des Speicherbereichs im Feld Länge ein und wählen Sie den Datentyp aus der Klappliste.  
Zulässig sind nur die Datentypen BOOL (bei einem Bitfeld muss als Adresse "0" und als Länge ein ganzzahlig Vielfaches von Byte verwendet werden), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER.

## Siehe auch

PUT: Schreib- und Sendebereich parametrieren (Seite 491)

### 8.1.3.3 Topologie anzeigen und projektieren

#### Überblick über die Topologiesicht

#### Funktionen der Topologiesicht

Die Topologiesicht ist einer von drei Arbeitsbereichen des Hardware- und Netzwerkeditors. Hier führen Sie folgende Aufgaben durch:

- Ethernet-Topologie anzeigen
  - Sämtliche PROFINET-Geräte und passiven Ethernet-Komponenten des Projekts samt Ports anzeigen
  - Verschaltungen zwischen den Ports anzeigen
  - Zugehörige logische Netze anzeigen
  - Diagnoseinformationen aller Ports anzeigen
- Ethernet-Topologie projektieren
  - Verschaltungen der Ports anlegen, ändern und löschen
  - Stationen, Geräte, Schnittstellen, Ports umbenennen
  - PROFINET-Geräte und passive Ethernet-Komponenten aus dem Hardware-Katalog zum Projekt hinzufügen
- Unterschiede zwischen Soll- und Ist-Topologie ermitteln und minimieren
  - Offline-/Online-Vergleich von Ethernet-Baugruppen, -Ports und -Portverschaltungen durchführen
  - Online vorliegende Topologieinformationen in das Offline-Projekt übernehmen

#### Unterschiede zwischen Netzsicht und Topologiesicht

- Die Netzsicht zeigt alle logischen Subnetze des Projekts an. Die Topologiesicht zeigt alle Ethernet-Komponenten des Projekts an. Dazu gehören auch passive Komponenten wie Switches und Medienkonverter und Leitungen.

---

#### Hinweis

Darüber hinaus werden auch Stationen mit Nicht-Ethernet-Komponenten angezeigt, wenn sich in der Station mindestens eine Ethernetkomponente befindet.

---

- Die Position eines Geräts in der Netzsicht und seine Position in der Topologiesicht sind unabhängig voneinander, d. h. in der Regel befindet sich ein und dasselbe Gerät in den beiden Sichten jeweils an einer anderen Stelle.
- Wenn Sie den Hardware-Katalog aus der Topologiesicht heraus öffnen, werden Ihnen ausschließlich Geräte mit Ethernet-Schnittstelle angezeigt.

## Aufbau der Topologiesicht

Die Topologiesicht (Seite 374) besteht im Wesentlichen aus einem grafischen Bereich (im Folgenden als grafische Ansicht bezeichnet) und einem tabellarischen Bereich (im Folgenden als tabellarische Ansicht bezeichnet).

## Welche Funktionen gibt es in der grafischen, welche in der tabellarischen Ansicht?

- Ethernet-Topologie anzeigen

| Funktion   | Grafische Ansicht | Tabellarische Ansicht |
|--|-------------------|-----------------------|
| Sämtliche PROFINET-Geräte und passiven Ethernet-Komponenten des Projekts samt Ports anzeigen | ja                | ja                    |
| Verschaltungen zwischen den Ports anzeigen (einschließlich Medientyp)                        | ja                | ja                    |
| Zugehörige logische Netze anzeigen   | nein              | ja                    |
| Eigenschaften der Leitungen zwischen den Ports anzeigen                                      | nein              | ja                    |
| Diagnoseinformationen aller Ports anzeigen   | ja                | ja                    |

- Ethernet-Topologie projektieren

| Funktion   | Grafische Ansicht  | Tabellarische Ansicht  |
|--|--|--|
| Verschaltungen von Ports anlegen, ändern und löschen   | <ul style="list-style-type: none"> <li>• Anlegen: ja</li> <li>• Ändern: nein</li> <li>• Löschen: ja</li> </ul> | <ul style="list-style-type: none"> <li>• Anlegen: ja</li> <li>• Ändern: ja</li> <li>• Löschen: ja</li> </ul> |
| Stationen, Geräte, Schnittstellen, Ports umbenennen  | nein   | ja   |
| PROFINET-Geräte und passive Ethernet-Komponenten aus dem Hardware-Katalog zum Projekt hinzufügen | ja   | nein   |

- Unterschiede zwischen Soll- und Ist-Topologie ermitteln und minimieren

| Funktion  | Grafische Ansicht | Tabellarische Ansicht |
|---|-------------------|-----------------------|
| Offline-/Online-Vergleich von Ethernet-Baugruppen, -Ports und -Portverschaltungen durchführen | nein              | ja                    |
| Online vorliegende Topologieinformationen in das Offline-Projekt übernehmen                   | nein              | ja                    |

## Topologiesicht starten

### Voraussetzung

Die Geräte- oder die Netzsicht innerhalb des Hardware- und Netzwerkeditors ist geöffnet.

## Vorgehen

Um die Topologiesicht Ihres Projekts zu starten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Register "Topologiesicht".

Oder:

1. Öffnen Sie die Netzsicht des Hardware-Editors.
2. Selektieren Sie ein PROFINET-Gerät oder eine PROFINET-Baugruppe.
3. Wählen Sie im Kontextmenü den Befehl "Gehe zur Topologiesicht".

## Ergebnis

Die grafische Ansicht der Topologiesicht wird gestartet. Falls Sie die Topologiesicht mittels Kontextmenü aufgerufen haben, bleibt die selektierte Komponente nach der Umschaltung selektiert.

## Topologie anzeigen

### Grafische Ansicht der projektierten Topologie anzeigen

#### Was wird dargestellt?

Die grafische Ansicht der projektierten Topologie zeigt das Folgende an:

- Projektierte PROFINET-Geräte und passive Ethernet-Komponenten samt Ports
- Projektierte Stationen mit Nicht-Ethernet-Komponenten, wenn sich in der Station mindestens eine Ethernet-Komponente befindet
- Projektierte Verschaltungen zwischen den Ports

#### Art der Darstellung

Die grafische Ansicht der Topologiesicht und der Netzsicht sind sich hinsichtlich der Art der Darstellung sehr ähnlich:

- Komponenten werden gegenüber der Gerätesicht vereinfacht dargestellt.
- Die Verschaltungen zwischen Ports werden als waagrechte und senkrechte Linien dargestellt. Diese sind gestrichelt, wenn es sich um die Verschaltung zwischen einem Werkzeugwechslerport und seinen möglichen Partnerports handelt.

## Tabellarische Ansicht der projektierten Topologie anzeigen

### Was wird dargestellt?

Die tabellarische Ansicht der projektierten Topologie zeigt mit Ausnahme der projektierten logischen PROFINET-Subnetze das Gleiche an wie die grafische Ansicht:

- Sämtliche projektierten PROFINET-Geräte und passiven Ethernet-Komponenten samt Ports
- Sämtliche projektierten Stationen mit Nicht-Ethernet-Komponenten, wenn sich in der Station mindestens eine Ethernet-Komponente befindet
- Projektierte Verschaltungen zwischen den Ports  
Zu einem Port mit der Eigenschaft "wechselnder Partnerport" gehören so viele ausgefüllte Zeilen, wie er potenzielle Partnerports hat, plus eine Leerzeile.

### Art der Darstellung

Wie der Name schon sagt, besteht die tabellarische Ansicht der Topologiesicht aus einer Tabelle, der Topologieübersichtstabelle. Sie ist wie die Netzübersichtstabelle aufgebaut. Sie besteht aus den folgenden Spalten:

- **Gerät/Port**  
Das ist die wichtigste Spalte der Tabelle. Die Einträge dieser Spalte sind hierarchisch aufgebaut, wobei das letzte Element in der Hierarchie die PROFINET-Ports sind. Sie können die hierarchischen Einträge aufklappen und zuklappen. Für eine CPU beispielsweise besteht ein Eintrag aus den folgenden Elementen:
  - Stationsname
  - Gerätename
  - Name der PROFINET-Schnittstelle
  - Name der PortsHinweis: Alle weiteren Spalten enthalten nur in den Zeilen mit den Port-Namen Einträge.
- **Typ** (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)  
Zeigt an, auf welche Art von Station, Gerät oder Schnittstelle sich die Tabellenzeile bezieht oder ob sie zu einem Port gehört.
- **Bestell-Nr.** (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)  
Bestell-Nr. des Geräts
- **Subnetz** (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)  
Projektiertes Subnetz, zu dem die Schnittstelle gehört
- **Master-/IO-System** (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)  
Zeigt an, ob die Schnittstelle zu einem PROFIBUS DP-Mastersystem oder einem PROFINET IO-System gehört.
- **Teilnehmeradresse** (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)  
Projektierte Adresse der Schnittstelle im Subnetz
- **Partner-Station**  
Name der Station, die den Partnerport enthält

- Partner-Gerät  
Name des Geräts, das den Partnerport enthält
- Partner-Schnittstelle  
Schnittstelle, zu der der Partnerport gehört
- Partnerport
- Leitungsdaten  
Enthält die Leitungslänge und die Signallaufzeit der Leitung, die die Ports verbindet

### Grundfunktionen für Tabellen

Die Topologieübersichtstabelle unterstützt die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spalten der Tabelle ein- und ausblenden  
Hinweis: Die für die Konfiguration maßgeblichen Spalten können nicht ausgeblendet werden.
- Spaltenbreite optimieren
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen.

### Diagnosestatus von Ports in der grafischen Ansicht anzeigen

#### Voraussetzung

Die grafische Ansicht der Topologiesicht ist geöffnet.

#### Vorgehen

Um den Diagnosestatus der Ports zu ermitteln, gehen Sie folgendermaßen vor:

1. Gehen Sie mit der gewünschten Komponente bzw. den gewünschten Komponenten online.

#### Ergebnis

Die folgenden Symbole werden angezeigt:

- Bei jedem Gerät wird das zugehörige Diagnosesymbol angezeigt.
- Falls in mindestens einer unterlagerten Komponente ein Fehler vorliegt, wird zusätzlich das Diagnosesymbol "Fehler in unterlagerter Komponente" in die linke untere Ecke des Diagnosesymbols eingeblendet.
- Für jeden Port wird das zugehörige Diagnosesymbol angezeigt.
- Jede Leitung zwischen zwei Ports, die online sind, erhält die ihrem Diagnosezustand entsprechende Farbe.

Die möglichen Diagnosesymbole für Ports und die farbliche Kennzeichnung von Ethernet-Leitungen finden Sie bei der Beschreibung der Hardware-Diagnose. Siehe: Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 966)



## Diagnosestatus von Hardware-Komponenten in der tabellarischen Ansicht anzeigen

### Voraussetzung

Die tabellarische Ansicht der Topologiesicht ist geöffnet.

### Vorgehen

Um den Diagnosestatus von Hardware-Komponenten der Topologieübersichtstabelle zu ermitteln, gehen Sie folgendermaßen vor:

1. Gehen Sie mit den gewünschten Komponenten online.

### Ergebnis

Die folgenden Symbole werden am linken Rand der Topologieübersichtstabelle angezeigt, und zwar in jeder Zeile, die zu der betroffenen Komponente gehört:

- Das zur Hardware-Komponente gehörige Diagnosesymbol wird angezeigt.
- Bei Hardware-Komponenten mit unterlagerten Komponenten wird - falls in mindestens einer unterlagerten Komponente ein Fehler vorliegt - zusätzlich das Diagnosesymbol "Fehler in unterlagerter Komponente" in die linke untere Ecke des Diagnosesymbols der Hardware-Komponente eingeblendet.

Die möglichen Diagnosesymbole für Hardware-Komponenten finden Sie bei der Beschreibung der Hardware-Diagnose. Siehe: Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 966)

---

#### Hinweis

Die Anzeige des Diagnosestatus von Hardware-Komponenten in der Topologieübersichtstabelle und der Netzübersichtstabelle ist gleich.

---

## Offline-/Online-Vergleich durchführen und das Ergebnis anzeigen

### Voraussetzung

Die Topologiesicht ist geöffnet. Eine Online-Verbindung zu einem oder mehreren Geräten kann bestehen, muss aber nicht.

### Vorgehen

Um die Unterschiede zwischen projektierte und tatsächlich vorhandener Topologie zu ermitteln, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Topologieübersicht auf die Schaltfläche "Offline-/Online-Vergleich".

### Ergebnis

Die Spalten "Partner-Station", "Partner-Schnittstelle" und "Leistungsdaten" in der Topologieübersichtstabelle werden entfernt.

Die Tabelle wird an ihrer rechten Seite um zwei Gruppen von Spalten erweitert, die zunächst leer sind:

- Ganz rechts werden Spalten für die online zu ermittelnde Topologie hinzugefügt.
- Zwischen den Spalten für die Offline- und die Online-Topologie werden die Spalten "Status", "Handlung" und "Beschreibung" für das Ergebnis des Offline-/Online-Vergleichs hinzugefügt.



---

#### Hinweis

Voreinstellung ist, dass die Spalte "Beschreibung" nicht angezeigt wird.

---

Die folgenden Schaltflächen in der Funktionsleiste der Tabelle werden aktiv geschaltet:

| Schaltfläche  | Name            | Bedeutung   |
|---|-----------------|---|
|  | Aktualisieren   | Die Ermittlung der online vorhandenen Topologie wird erneut gestartet.  |
|  | Synchronisieren | <ul style="list-style-type: none"><li>• Online ermittelte Portverschaltungen in das Projekt übernehmen (Seite 509)</li><li>• Online ermittelte Geräte in das Projekt übernehmen (Seite 510)</li></ul> |

Nach Ermittlung der tatsächlich vorhandenen Topologie werden die hinzugefügten Spalten befüllt. Darauf wird im Folgenden näher eingegangen.

---

#### Hinweis

Für denjenigen Port, der mit dem nur online verfügbaren PG/PC verbunden ist, wird ein Unterschied zwischen Offline- und Online-Sicht angezeigt. Das liegt daran, dass das PG / der PC offline nicht projiziert werden kann.

---

### Spalten für die online ermittelte Topologie

Folgende Spalten werden angezeigt:






- "Gerät/Port"
- "Typ" (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
- "Bestell-Nr." (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)

- "IP-Adresse" (Voreinstellung ist, dass diese Spalte nicht angezeigt wird.)
- "Partner-Gerät"
- "Partnerport"
- "Leistungsdaten"



### Spalten für das Ergebnis des Offline-/Online-Vergleichs

Folgende Spalten werden angezeigt:

- "Status"  
 Hier wird in Form von Diagnosesymbolen das Ergebnis des Offline-/Online-Vergleichs angezeigt. Folgende Symbole sind möglich:

| Diagnosesymbol  | Bedeutung  |
|---|--|
|    | Unterschiedliche Topologieinformation in mindestens einer unterlagerten Komponente     |
|    | identische Topologieinformation  |
|    | Topologieinformation nur offline vorhanden oder Gerät ist deaktiviert                  |
|   | Topologieinformation nur online vorhanden  |
|  | Unterschiedliche Topologieinformation  |
|   | Wenn ein Gerät keine Topologiefunktionen unterstützt, bleibt die Spalte "Status" leer. |

- "Handlung"  
 Hier werden in Form von Symbolen die möglichen Handlungen angezeigt. Folgende Symbole sind möglich:

| Symbol  | Bedeutung                                     |
|---|---|
|  | keine Handlung möglich                        |
|  | die online ermittelte Verschaltung übernehmen |

- "Beschreibung"  
Diese Spalte beschreibt die gewählte Handlung in Worten.

## Topologie projektieren

## Ports verschalten

## Übersicht

### Ports verschalten in der Topologiesicht

Sie haben in der Topologiesicht die folgenden Möglichkeiten, Ports zu verschalten:

- in der grafischen Ansicht (Seite 503)
- in der grafischen Ansicht eines Werkzeugwechslers (Seite 505)
- in der tabellarischen Ansicht (Seite 504)
- in der tabellarischen Ansicht eines Werkzeugwechslers (Seite 505)
- durch die Übernahme von online ermittelten Portverschaltungen (Seite 509)

### Welche Auswirkungen auf die Netzsicht hat das Verschalten von Ports?

---

#### Hinweis

Sie können in der Netzsicht bei den Eigenschaften eines Subnetzes vorgeben, dass beim Anlegen einer Portverschaltung zwischen zwei nicht vernetzten Geräten dieses Subnetz verwendet wird.

---

Beim Anlegen einer Verschaltung zwischen zwei Ports sind folgende Auswirkungen in der Netzsicht möglich:

- Wenn die zugehörigen Schnittstellen nicht vernetzt sind: Falls Sie ein Subnetz voreingestellt haben, wird dies verwendet. Andernfalls wird ein neues Subnetz angelegt, um die beiden Schnittstellen zu verbinden.
- Wenn genau eine der beiden zugehörigen Schnittstellen vernetzt ist: Die unvernetzte Schnittstelle wird mit demselben Subnetz verbunden wie die bereits vernetzte Schnittstelle.
- In allen anderen Fällen: Die zugehörigen Schnittstellen werden nicht mit einem logischen Subnetz verbunden.

### Siehe auch

Ports verschalten (Seite 810)

## Ports verschalten in der grafischen Ansicht

### Voraussetzung

Sie befinden sich in der grafischen Ansicht der Topologiesicht.

### Vorgehen – Neue Verschaltung zwischen zwei Ports anlegen

Um einen Port eines Geräts mit einem Port eines anderen Geräts zu verschalten, gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf den zu verschaltenden Port.
2. Klicken Sie mit der linken Maustaste und halten Sie diese gedrückt.
3. Verschieben Sie den Mauscursor.  
Der Mauscursor zeigt jetzt mit dem Verschaltungssymbol den Arbeitsmodus "Verschalten" an. Gleichzeitig sehen Sie im Mauscursor das Sperrsymbol, das erst über einer gültigen Zielposition verschwindet.
4. Ziehen Sie nun den Mauscursor auf den Ziel-Port. Sie können hierbei die linke Maustaste gedrückt halten oder loslassen.
5. Lassen Sie nun die linke Maustaste los oder klicken Sie erneut mit der linken Maustaste.

Ergebnis: Eine neue Portverschaltung wird angelegt.

---

#### Hinweis

##### Erzeugen eines Rings bei S7-300-, S7-400- und S7-1500-CPU's

Wenn Sie bei S7-300-, S7-400- oder S7-1500-CPU's mittels Portverschaltungen einen Ring erzeugen, dann wird automatisch eine MRP-Domain erzeugt.

---

### Vorgehen – Eine bestehende Portverschaltung ändern, ohne sie vorher zu löschen

Gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf denjenigen Port einer bestehenden Verschaltung, der einen neuen Partner-Port erhalten soll.
2. Ziehen Sie ihn auf den neuen Partner-Port.

Ergebnis: Die bestehende Portverschaltung wird gelöscht. Die neue Portverschaltung wird angelegt.

Alternative Vorgehensweise:

1. Platzieren Sie den Mauscursor auf einen bisher unverschalteten Port, der mit einem bereits verschalteten Port verbunden werden soll.
2. Ziehen Sie ihn auf den bereits verschalteten Port.

Ergebnis: Die bestehende Portverschaltung wird gelöscht. Die neue Portverschaltung wird angelegt.

### **Vorgehen – Zwei bereits verschaltete Ports miteinander verschalten, ohne die beiden bestehenden Portverschaltungen vorher zu löschen**

Gehen Sie folgendermaßen vor:

1. Platzieren Sie den Mauscursor auf einen bereits verschalteten Port, der einen neuen Partner-Port erhalten soll.
2. Ziehen Sie ihn auf den neuen Partner-Port, der ebenfalls bereits verschaltet ist.

Ergebnis: Die beiden bestehenden Portverschaltungen werden gelöscht. Die neue Portverschaltung wird angelegt.

### **Ports verschalten in der tabellarischen Ansicht**

#### **Welche Aktionen mit Portverschaltungen sind in der tabellarischen Ansicht möglich?**

In der tabellarischen Ansicht sind folgende Aktionen mit Portverschaltungen möglich:

- Neue Portverschaltung anlegen
- Bestehende Portverschaltung ändern
- Bestehende Portverschaltung löschen

#### **Voraussetzung**

in der Topologieübersicht ist die Zeile mit demjenigen Port sichtbar, dessen Verschaltung Sie neu anlegen, ändern oder löschen wollen.

#### **Vorgehen**

Um die Verschaltung eines Port erstmalig anzulegen, zu ändern oder zu löschen, gehen Sie folgendermaßen vor:

1. Bewegen Sie den Mauszeiger in der Zeile des Quell-Ports in die Spalte "Partnerport".
2. Klicken Sie auf die dort befindliche Klappliste.
3. Wählen Sie den gewünschten Partnerport (bei der Neuanlage oder der Änderung einer Portverschaltung) bzw. den Eintrag "unverschaltet" (beim Löschen einer Portverschaltung).

#### **Ergebnis**

Die gewünschte Aktion wird durchgeführt. In der Spalte "Partnerport" wird der neue Partnerport (nach der Neuanlage oder Änderung einer Portverschaltung) bzw. der Schriftzug "Wähle Port" (nach dem Löschen einer Portverschaltung) angezeigt.

## **Einen Port mit mehreren Partnerports verschalten in der grafischen Ansicht**

### **Voraussetzung**

- Sie haben einen Port eines PROFINET-Geräts mit der Eigenschaft "alternative Partner" parametrisiert und seine möglichen Partnerports vorgegeben.
- Die grafische Ansicht der Topologiesicht ist geöffnet.

### **Vorgehen**

1. Verschalten Sie diesen Port (im Folgenden als Quell-Port bezeichnet) mit einem der von Ihnen vorgegebenen Partnerports (im Folgenden als Ziel-Ports bezeichnet).
2. Verschalten Sie den Quell-Port mit einem weiteren Ziel-Port.  
Dies ist auf mehrere Arten möglich:
  - Ziehen Sie den Mauszeiger von einem bereits verschalteten Partnerport zu einem Ziel-Port.
  - Ziehen Sie den Mauszeiger von einer bereits angelegten Verschaltung auf einen Ziel-Port.
  - Ziehen Sie den Mauszeiger von einem Ziel-Port auf einen bereits verschalteten Partnerport.
  - Ziehen sie den Mauszeiger von einem Ziel-Port auf eine bereits angelegte Verschaltung.
3. Wiederholen Sie ggf. den vorhergehenden Schritt einmal oder mehrmals.

### **Ergebnis**

Es wird eine Verschaltung zwischen dem Quell-Port und den wechselnden Partnerports angelegt. Diese wird in Form einer gestrichelten Linie angezeigt.

## **Einen Port mit mehreren Partnerports verschalten in der tabellarischen Ansicht**

### **Welche Aktionen mit Portverschaltungen zu mehreren Partnerports sind in der tabellarischen Ansicht möglich?**

Bei einem Werkzeugwechsler sind in der tabellarischen Ansicht folgende Aktionen mit Portverschaltungen zu mehreren Partnerports möglich:

- Neue Portverschaltung anlegen
- Bestehende Portverschaltung ändern
- Bestehende Portverschaltung löschen

### Voraussetzung

- Sie haben einen Port eines PROFINET-Geräts mit der Eigenschaft "alternative Partner" parametrisiert und seine möglichen Partnerports vorgegeben.
- In der Topologieübersicht ist die Zeile mit demjenigen Port sichtbar, dessen Verschaltung Sie neu anlegen, ändern oder löschen wollen.

### Vorgehen

Um die Verschaltung eines Port zu einem von mehreren Partnerports erstmalig anzulegen, zu ändern oder zu löschen, gehen Sie folgendermaßen vor:

1. Bewegen Sie den Mauszeiger in der Zeile des Quell-Ports in die Spalte "Partnerport".
2. Klicken Sie auf die dort befindliche Klappliste.
3. Wählen Sie den gewünschten Partnerport (bei der Neuanlage oder der Änderung einer Portverschaltung) bzw. den Eintrag "unverschaltet" (beim Löschen einer Portverschaltung).

### Ergebnis

Die gewünschte Aktion wird durchgeführt:

- Bei einer Neuanlage wird eine neue Zeile in die Topologieübersichtstabelle eingefügt. Dort wird in der Spalte "Partnerport" der neue Partnerport angezeigt.
- Bei einer Änderung wird in der Spalte "Partnerport" der neue Partnerport angezeigt.
- Beim Löschen wird die Zeile mit der bisherigen Portverschaltung gelöscht.

---

#### Hinweis

Bei einem Werkzeugwechsler gehören zu einem Port mit Portverschaltungen zu mehreren Partnerports in der Regel mehrere Zeilen. Dabei ist die letzte Zeile stets eine Leerzeile. Die erste Zeile ist editierbar, alle weiteren Zeilen sind nur lesbar.

---

## Stationen, Geräte, Schnittstellen und Ports umbenennen

### Eine Station, ein Gerät, eine Schnittstelle oder einen Port umbenennen

#### Voraussetzung

Die tabellarische Ansicht der projektierten Topologie ist geöffnet.



## Vorgehen

Um eine Station, ein Gerät, eine Schnittstelle oder einen Port umzubenennen, gehen Sie folgendermaßen vor:

1. Klicken Sie zwei Mal in das zugehörige Feld der Topologieübersichtstabelle (Mit dem zweiten Klick gelangen Sie in den Editiermodus.).
2. Geben Sie den neuen Namen ein und drücken Sie die danach ENTER-Taste (beendet den Editiermodus).

## Ergebnis

Das Objekt wird umbenannt.

## Offline-/Online-Vergleich

### Automatische Zuordnung von Geräten durch den Offline-/Online-Vergleich

## Übersicht

Beim Offline-/Online-Vergleich wird die projektierte Topologie mit der tatsächlich vorhandenen Topologie verglichen. Dabei werden online ermittelte Geräte projektierten Geräten automatisch zugeordnet, soweit dies möglich ist.

## Start der Verfügbarkeitsermittlung

Sie starten die Verfügbarkeitsermittlung das erste Mal, indem Sie in der Funktionsleiste der Topologieübersicht auf die Schaltfläche "Offline-/Online-Vergleich" klicken.

Sie starten die Verfügbarkeitsermittlung erneut, indem Sie auf die Schaltfläche "Aktualisieren" klicken.

---

### Hinweis

Die Verfügbarkeitsermittlung kann einige Sekunden dauern. In dieser Zeit sind keine Bedieneingaben möglich.

---


## Automatische Zuordnung

Ein online ermitteltes Gerät wird einem projektierten Gerät automatisch zugeordnet, wenn die beiden Geräte in folgenden Eigenschaften übereinstimmen:

- Gerätename
- Bestellnummer
- Port-Anzahl



Im Folgenden wird beschrieben, welche Fälle dabei auftreten können und welche Maßnahmen Sie ggf. ergreifen können:

- Identische Portverschaltungen  
Das ist der Idealfall. Hier sind keine Maßnahmen erforderlich.


| Spalte "Handlung"   | Bedeutung      |
|---|----------------|
|  | keine Handlung |

- Beim ermittelten und beim projektierten Gerät sind Verschaltungen vorhanden, es gibt jedoch Unterschiede.  
Hier gibt es folgende Handlungsalternativen:

- Wenn die Übernahme der Online-Konfiguration möglich ist



| Spalte "Handlung"   | Bedeutung                                  |
|---|--|
|  | Online-Verschaltung übernehmen (Seite 509) |
|  | keine Handlung                             |

- Wenn die Übernahme der Konfiguration nicht möglich ist


| Spalte "Handlung"   | Bedeutung      |
|---|----------------|
|  | keine Handlung |

- Eine Verschaltung ist nur online vorhanden.  
Hier gibt es folgende Handlungsalternativen:



- Wenn die Übernahme der Online-Konfiguration möglich ist

| Spalte "Handlung"   | Bedeutung                                  |
|---|--|
|  | Online-Verschaltung übernehmen (Seite 509) |
|  | keine Handlung                             |

- Wenn die Übernahme der Konfiguration nicht möglich ist

| Spalte "Handlung"   | Bedeutung      |
|---|----------------|
|  | keine Handlung |

- Eine Verschaltung ist nur in der Projektierung vorhanden.  
Hier gibt es folgende Handlungsalternativen:

| Spalte "Handlung"   | Bedeutung  |
|---|--|
|  | Online-Verschaltung übernehmen (Seite 509) , d. h. die Verschaltung in der Projektierung wird gelöscht |
|  | keine Handlung   |

## Keine automatische Zuordnung

In den folgenden Fällen ist keine automatische Zuordnung möglich:

- Zu einem projektierten Gerät gibt es kein zugehöriges online ermitteltes Gerät (Dann sind die entsprechenden Spalten im Bereich "Online-Topologie" der Topologieübersichtstabelle leer.).  
In diesem Fall sollten Sie Ihre Anlage um das bereits projektierte Gerät ergänzen oder das projektierte Gerät aus der Projektierung löschen.
- Ein online ermitteltes Gerät kann keinem projektierten Gerät zugeordnet werden (Dann sind die entsprechenden Spalten im Bereich "Offline-Topologie" der Topologieübersichtstabelle leer.).  
In diesem Fall können Sie das online ermittelte Gerät in das Projekt übernehmen (Seite 510).

## Online ermittelte Portverschaltungen in das Projekt übernehmen

### Voraussetzung

Sie haben in der Topologiesicht einen Offline-/Online-Vergleich durchgeführt. Dieser hatte als Ergebnis, dass mindestens ein online ermitteltes Gerät einem projektierten Gerät automatisch zugeordnet wurde und es dabei jedoch Unterschiede bei der Verschaltung gibt.

### Vorgehen

Um eine oder mehrere online ermittelte Portverschaltungen manuell in das Projekt zu übernehmen, gehen Sie folgendermaßen vor:

1. Wählen Sie für einen Port eines projektierten Geräts, dem ein online ermitteltes Gerät zugeordnet wurde, in der Spalte "Handlung" den Wert "Übernehmen".
2. Wiederholen Sie diesen Schritt ggf. für weitere Ports desselben projektierten Geräts.
3. Wiederholen Sie die bisherigen Schritte ggf. für weitere projektierte Geräte, denen online ermittelte Geräte zugewiesen wurden, und dabei Unterschiede bei der Verschaltung bestehen.
4. Klicken Sie auf die Schaltfläche "Synchronisieren".

## Ergebnis

Für die zugehörigen Geräte werden die online ermittelten Portverschaltungen und die Leitungsinformationen in das Projekt übernommen. Eine erfolgreiche Übernahme wird durch das Diagnosesymbol "identische Topologieinformation" für jeden Port angezeigt.

---

### Hinweis

Wenn für ein online ermitteltes Gerät andere Portverschaltungen erkannt werden als die im Projekt vorliegenden, führt deren Übernahme in das Projekt dazu, dass die bisher im Projekt vorhandenen Portverschaltungen durch die online erkannten ersetzt werden. Wenn nun für ein online ermitteltes Gerät gar keine Portverschaltungen erkannt werden, führt die Übernahme in das Projekt dazu, dass sämtliche Portverschaltungen dieses Geräts im Projekt gelöscht werden.

---

## Online ermittelte Geräte in das Projekt übernehmen

### Voraussetzung

Sie haben in der Topologiesicht einen Offline-/Online-Vergleich durchgeführt. Dieser hatte als Ergebnis, dass mindestens ein online ermitteltes Gerät keinem projektierten Gerät zugeordnet werden konnte.

### Vorgehen

Um eine oder mehrere online ermittelte Geräte manuell in das Projekt zu übernehmen, gehen Sie folgendermaßen vor:

1. Bewegen Sie bei einem projektierten Gerät ohne Online-Partner den Mauszeiger in die Spalte "Gerät/Port" der Online-Topologie.
2. Wählen Sie in der Klappliste dieses Felds das Gerät, das Sie dem projektierten Gerät zuordnen wollen.
3. Wiederholen Sie die bisherigen Schritte ggf. für weitere projektierte Geräte ohne Online-Partner.

### Ergebnis

Das ausgewählte online ermittelte Gerät wird vom Tabellenende nach oben verschoben. Danach befindet es sich in der Zeile desjenigen projektierten Geräts, dem Sie es soeben zugeordnet haben.

### 8.1.3.4 Industrial Ethernet Security

#### Security projektieren

#### Allgemein

#### Unterstützte Geräte

#### Unterstützte Geräte

Security-Funktionen können für folgende Produkte projektiert werden:

- SCALANCE S:
  - S602 V2/V3
  - S612 V2/V3
  - S613 V2
  - S623 V3

Anmerkung: SCALANCE S V3 Geräte können derzeit lediglich mit denjenigen Funktionen projektiert werden, die von SCALANCE S V2 Geräten unterstützt werden. Für den SCALANCE S623 ist der DMZ-Port derzeit nicht projektierbar.
- SOFTNET Security Client:
  - SOFTNET Security Client V4
- S7-CPs: CP 343-1 GX31 Advanced, CP 443-1 GX30 Advanced, CP 1543-1  
Anmerkung: Die S7-CPs CP 343-1 GX31 Advanced und CP 443-1 GX30 Advanced werden im weiteren Verlauf unter der Benennung "CP x43-1 Advanced" zusammengefasst.
- PC-CP: CP 1628
- SCALANCE M: SCALANCE M875 und MD741-1

#### Allgemeine Benennung "Security-Modul"

Im vorliegenden Abschnitt des Informationssystems werden die folgenden Produkte unter der Benennung "Security-Modul" zusammengefasst: CP 343-1 GX31 Advanced, CP 443-1 GX30 Advanced, CP 1543-1, SCALANCE S602 / SCALANCE S612 / SCALANCE S613 / SCALANCE S623, CP 1628, SCALANCE M875/MD741-1.

#### Struktur dieses Hilfeabschnitts

Themen, die für alle Security-Module relevant sind, finden Sie im Kapitel "Allgemein". Informationen, die nur für bestimmte Modultypen relevant sind, finden Sie in den jeweiligen modulspezifischen Abschnitten.

## Übersicht - Leistungsumfang und Arbeitsweise

### Allgemeine Benennung "STEP 7"

Die Projektierung von Security-Funktionen wird ab STEP 7 V12 unterstützt. Deshalb wird im vorliegenden Abschnitt des Informationssystems die Benennung "STEP 7" stellvertretend für alle Versionsstände von STEP 7 ab V12 verwendet.

### Leistungsumfang

Folgende Security-Funktionen können Sie in STEP 7 nutzen:

- Projektierung der Security-Module
- Erstellen von Konfigurationsdateien für SOFTNET Security Client V4
- Erstellen von Konfigurationsdaten für SCALANCE M875/MD741-1
- Diagnosefunktionen und Statusanzeigen

### Offline- und Online-Ansicht

Die Security-Funktionen werden in zwei Sichtweisen konfiguriert:

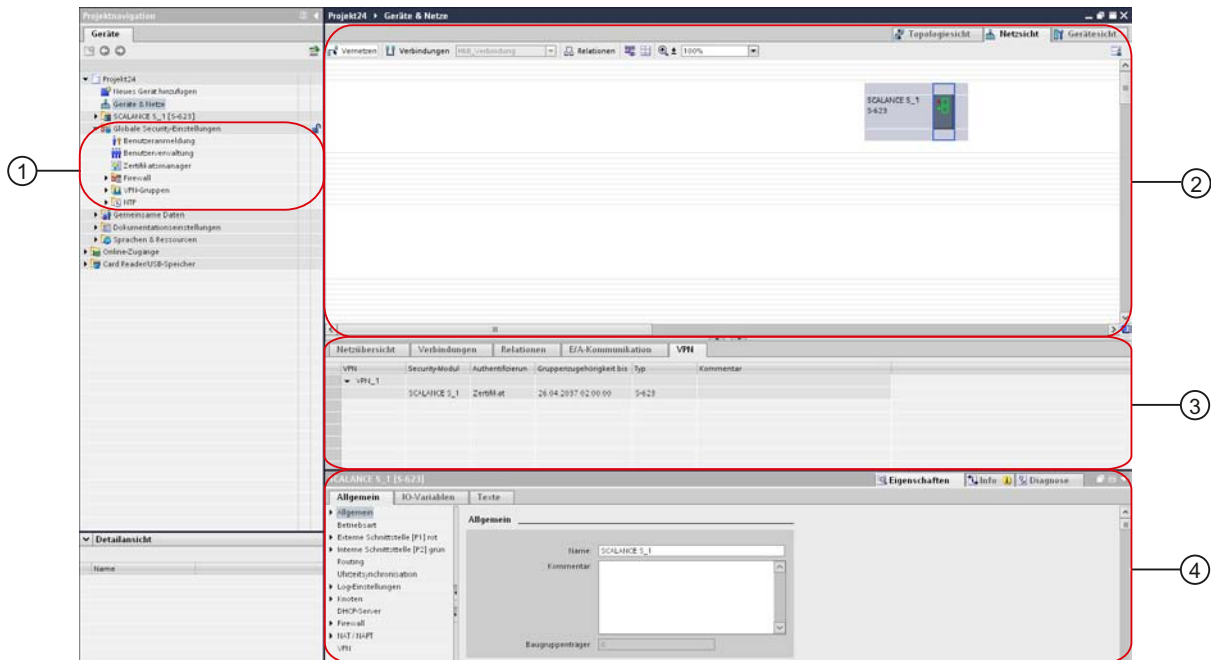
- Offline-Projektierungssicht  
In der Offline-Projektierungssicht erfolgt die Projektierung der Konfigurationsdaten für die Security-Module und den SOFTNET Security Client. Vor dem Ladevorgang muss hierbei keine Verbindung zu dem Security-Modul bestehen.
- Online-Diagnosesicht  
Die Online-Diagnosesicht dient der Diagnose eines Security-Moduls und bietet unter anderem die Möglichkeit, ein Firmware-Update durchzuführen.

### Arbeitsweise - Sicherheit und Konsistenz

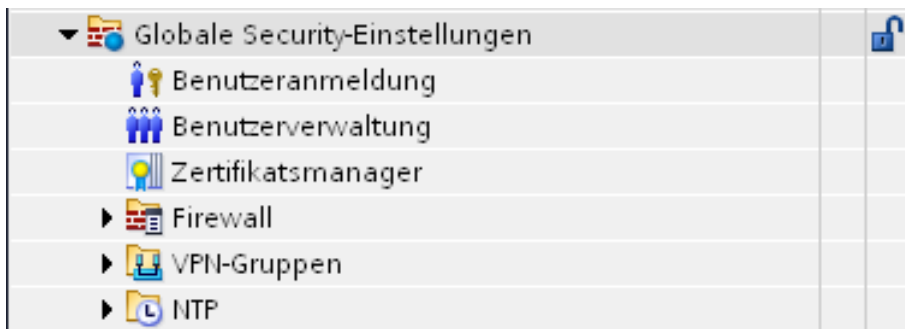
- Zugriff nur für autorisierte Benutzer  
Die Security-Funktionen jedes Projekts sind durch Benutzername- und Passwortvergabe vor unberechtigtem Zugriff geschützt.
- Konsistente Projektdaten  
Schon während der Eingabe in den einzelnen Dialogen erfolgen Konsistenzprüfungen. Zusätzlich werden dialogübergreifende, projektweite Konsistenzprüfungen durchgeführt. Auf die Security-Module können nur konsistente Projektdaten geladen werden.
- Schutz der Projektdaten durch Verschlüsselung  
Die Security-relevanten Projekt- und Konfigurationsdaten sind jeweils durch Verschlüsselung geschützt. Die Datenablage kann je nach Security-Modul im Projekt und/oder auf dem C-PLUG erfolgen.

## Bedienoberfläche - Aufbau und Menübefehle

### Bedienoberfläche für Security-Funktionen in STEP 7



① Globale Security-Einstellungen



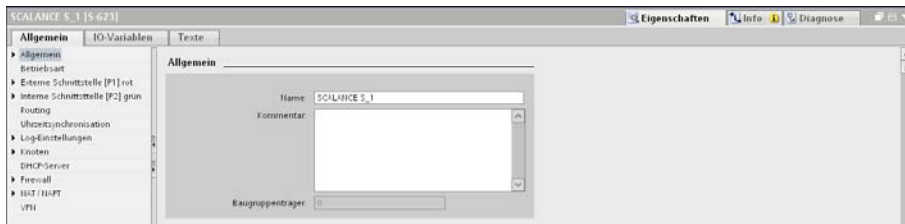
In der Projektnavigation befinden sich die globalen Security-Einstellungen. Diese Security-Einstellungen können modulunabhängig projektiert und im Anschluss ggf. einzelnen Security-Modulen zugewiesen werden. Wenn sich ausschließlich CPs in der Security-Konfiguration befinden, werden die globalen Security-Einstellungen erst dann angezeigt, wenn die Security-Funktionen in den lokalen Security-Einstellungen eines CPs aktiviert wurden. Folgende Hauptordner und Einträge sind in den globalen Security-Einstellungen verfügbar:

- **Benutzeranmeldung**  
Für die Security-Konfiguration innerhalb eines Projekts existiert eine eigene Benutzerverwaltung. Melden Sie sich über den Eintrag "Benutzeranmeldung" an der Security-Konfiguration mit einem bestehenden Benutzer an. Einen neuen Benutzer können Sie beim Erstellen einer Security-Konfiguration und in der Benutzerverwaltung anlegen.
- **Benutzerverwaltung**  
In der Benutzerverwaltung können Sie Benutzer erstellen, Rechte für Rollen definieren und diese Rollen Benutzern zuweisen.
- **Zertifikatsmanager**  
Im Zertifikatsmanager erhalten Sie eine Übersicht über alle im Projekt verwendeten Zertifikate. Sie können z. B. neue Zertifikate importieren sowie bestehende Zertifikate exportieren, ändern oder ersetzen.
- **Firewall**  
Unter dem Eintrag "Firewall" können Sie globale IP- und MAC-Firewall-Regelsätze definieren und Security-Modulen zuweisen. Mit Hilfe von IP- und MAC-Dienst-Definitionen lassen sich die IP- und MAC-Firewall-Regeln kompakt und übersichtlich definieren.
- **VPN-Gruppen**  
In diesem Ordner sind alle erstellten VPN-Gruppen enthalten. Sie können hier neue VPN-Gruppen erstellen und Security-Module diesen VPN-Gruppen zuordnen.
- **NTP**  
Hier können Sie gesicherte NTP-Server anlegen und einem oder mehreren Security-Modulen zuweisen. Damit erreichen Sie, dass die Zeitsynchronisation über den zugewiesenen NTP-Server erfolgt. Nicht gesicherte NTP-Server sind in den lokalen Security-Einstellungen konfigurierbar.





④ Lokale Security-Einstellungen



Lokale Security-Einstellungen werden für ein bestimmtes Security-Modul projiziert. Nachdem ein Security-Modul im Arbeitsbereich selektiert wurde, sind dessen lokale Security-Einstellungen im Inspektorfenster unter "Eigenschaften" > "Allgemein" verfügbar.

**Hinweis für CPs:**

Bevor Security-Einstellungen für CPs projiziert werden können, müssen die lokalen Security-Einstellungen aktiviert werden. Melden Sie sich hierzu ggf. an Ihrem Security-Projekt an und aktivieren Sie anschließend im Inspektorfenster unter "Eigenschaften" > "Allgemein" > "Security" das Kontrollkästchen "Aktiviere Security-Funktionen". Die lokalen Security-Einstellungen werden danach unterhalb des Eintrags "Security" angezeigt. Durch das Aktivieren des Kontrollkästchens werden die folgenden Einstellungen, sofern sie aktiviert wurden, automatisch in die lokalen Security-Einstellungen migriert.

CP x43-1 Advanced:

- SNMP
- FTP-Konfiguration
- Uhrzeitsynchronisation
- Webserver
- Einträge von IP-Zugriffslisten

CP 1543-1:

- SNMP
- FTP-Konfiguration
- Uhrzeitsynchronisation

CP 1628:

- SNMP
- Uhrzeitsynchronisation

Zusätzlich werden für projizierte Verbindungen automatisch Firewall-Regeln angelegt, die den Verbindungsaufbau freigeben. Für unspezifizierte Verbindungen müssen Sie diese Firewall-Regeln im erweiterten Firewall-Modus manuell konfigurieren. Zum Aufzeichnen geblockter Pakete stehen Ihnen entsprechende Log-Einstellungen zur Verfügung.

## Konsistenzprüfung durchführen

### Übersicht

Folgende Konsistenzprüfungen gibt es:

- Lokale Konsistenzprüfungen
- Projektweite Konsistenzprüfungen

In den einzelnen Dialogbeschreibungen dieser Hilfe sind unter dem Stichwort "Konsistenzprüfung" die Regeln aufgeführt, auf die Sie jeweils bei der Eingabe achten müssen.

### Lokale Konsistenzprüfungen

Eine Konsistenzprüfung heißt lokal, wenn sie direkt innerhalb eines Dialogs durchgeführt werden kann. Bei folgenden Aktionen werden lokale Konsistenzprüfungen durchgeführt:

- nach dem Verlassen eines Feldes
- nach dem Verlassen einer Zeile in einer Tabelle
- beim Bestätigen eines Dialogs mit OK.

### Projektweite Konsistenzprüfungen

Projektweite Konsistenzprüfungen geben Aufschluss über die Korrektheit der konfigurierten Projektdaten. Bei folgenden Aktionen wird eine projektweite Konsistenzprüfung durchgeführt:

- beim Übersetzen einer Konfiguration
- beim Laden einer Konfiguration

---

#### Hinweis

Projektierte Daten können Sie nur laden, wenn das Projekt insgesamt konsistent ist.

---

## Zertifikate verwalten

### Übersicht zu Zertifikaten

#### Wie verwalten Sie Zertifikate?

Im Zertifikatsmanager erhalten Sie eine Übersicht aller im Projekt verwendeten Zertifikate, z. B. CA-Zertifikate mit Angaben zu Antragssteller, Aussteller, Gültigkeit, Verwendung und Vorhandensein eines privaten Schlüssels.

Das CA-Zertifikat ist ein durch eine Zertifizierungsstelle, die so genannte "Certificate Authority", ausgestelltes Zertifikat, von dem die Gerätezertifikate abgeleitet werden. Dazu gehören die SSL-Zertifikate, die zur Authentifizierung bei der gesicherten Kommunikation zwischen einem Netzwerkteilnehmer und einem Security-Modul benötigt werden sowie die VPN-

Gruppenzertifikate, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Zertifizierungsstellen können sein:

- STEP 7 selbst. Sind "Antragsteller" und "Aussteller" gleich, dann handelt es sich um ein selbst-signiertes, also durch STEP 7 ausgestelltes Zertifikat.
- Eine übergeordnete Zertifizierungsstelle. Diese projektexternen Fremdzertifikate werden importiert und im Zertifikatsspeicher von STEP 7 abgelegt.

Zertifikate, die von einer der beiden Zertifizierungsstellen angelegt werden, haben immer einen privaten Schlüssel, damit die Gerätezertifikate abgeleitet werden können.

Zusätzlich stehen Ihnen im Zertifikatsmanager die folgenden Funktionen zur Auswahl:

- Ändern von bestehenden Zertifikaten (z. B. Gültigkeitsdauer).
- Importieren von neuen Zertifikaten und Zertifizierungsstellen.
- Importieren von SSL-Zertifikaten (nur S7-CPs), z. B. für FTP-Kommunikation.
- Exportieren der im Projekt verwendeten Zertifikate und Zertifizierungsstellen.
- Erneuern von abgelaufenen Zertifikaten und Zertifizierungsstellen.
- Ersetzen bestehender Zertifizierungsstellen durch andere.
- Hinzufügen von vertrauenswürdigen Zertifikaten und Zertifizierungsstellen.
- Löschen von manuell importierten Zertifikaten.

---

#### **Hinweis**

##### **Laden der Projektierung**

Nach dem Ersetzen oder Erneuern von Zertifikaten muss die Projektierung auf die entsprechenden Security-Module geladen werden.

Nach dem Ersetzen oder Erneuern von CA-Zertifikaten muss die Projektierung auf alle Security-Module geladen werden.

---

---

#### **Hinweis**

##### **Aktuelles Datum und aktuelle Uhrzeit auf den Security-Modulen**

Achten Sie bei der Verwendung von gesicherter Kommunikation (z. B. HTTPS, VPN...) darauf, dass die betroffenen Security-Module über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die verwendeten Zertifikate werden sonst als nicht gültig ausgewertet und die gesicherte Kommunikation funktioniert nicht.

---

### **So erreichen Sie diese Funktion**

Doppelklicken Sie in den globalen Security-Einstellungen auf den Eintrag "Zertifikatsmanager".

In den einzelnen Registern stehen Ihnen die folgenden Befehle über das Kontextmenü zur Verfügung:

| Befehl  | Bedeutung  |
|---|--|
| Importieren / Exportieren                                       | Import / Export von Gerätezertifikaten bzw. CA-Zertifikaten. Die Zertifikate werden auf das Security-Modul übertragen. Folgende Formate sind erlaubt:<br>*.cer (nur Zertifikat)<br>*.crt (nur Zertifikat)<br>*.pem (nur Zertifikat)<br>*.p12 (Zertifikat mit dazugehörigem privaten Schlüssel) |
| Anzeigen  | Öffnet den Zertifikatsdialog von Windows, in dem Sie eine Übersicht aller Zertifikatsdaten angezeigt bekommen.   |
| Erneuern<br>(nur in den Registern "CA" und "Gerätezertifikate") | Öffnet den Dialog "Neues Zertifikat erstellen", in dem Sie bei Bedarf, z. B. bei kompromittierten Zertifikaten, ein Zertifikat importieren oder ein neues Zertifikat von STEP 7 erzeugen lassen.   |
| Ersetzen<br>(nur in Register "CA")                              | Öffnet den Dialog "Zertifizierungsstelle (CA) ändern", in dem Sie eine bestehende Zertifizierungsstelle durch eine neue ersetzen können.   |
| Löschen<br>(nur bei manuell importierten Zertifikaten)          | Löscht ein Zertifikat im Register "Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen".   |

## Zertifizierungsstellen

### Register "CA"

Die hier angezeigten Zertifikate werden durch eine Zertifizierungsstelle angelegt.

- CA-Zertifikate eines Projekts: Beim Erstellen eines neuen Projekts wird für das Projekt ein CA-Zertifikat erzeugt. Von diesem Zertifikat werden die SSL-Zertifikate für die einzelnen Security-Module abgeleitet.
- CA-Gruppenzertifikate: Beim Erstellen einer neuen VPN-Gruppe wird für die Gruppe ein CA-Zertifikat erzeugt.

## Gerätezertifikate

### Register "Gerätezertifikate"

Anzeige der gerätespezifischen Zertifikate, die von STEP 7 für ein Security-Modul erzeugt werden. Dazu gehören:

- SSL-Zertifikate: Für jedes angelegte Security-Modul wird ein SSL-Zertifikat erzeugt, das von dem CA-Zertifikat des Projekts abgeleitet ist. SSL-Zertifikate werden zur Authentifizierung bei der gesicherten Kommunikation zwischen PG/PC und Security-Modul, beim Laden der Konfiguration in ein SCALANCE S-Modul sowie beim Logging herangezogen.
- Gruppenzertifikate: Zusätzlich wird für jedes Security-Modul pro VPN-Gruppe, in der es sich befindet, ein Gruppenzertifikat erzeugt.

## Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen

### Register "Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen"

Anzeige der in STEP 7 importierten Fremdzertifikate. Importiert werden können z. B. Server-Zertifikate von externen FTPS-Servern oder Projektzertifikate von anderen Projekten, die mit STEP 7 erstellt wurden.

Bei CPs wird das importierte Fremdzertifikat auf alle im Projekt verwalteten CPs übertragen, welche das Zertifikat prüfen. Stufen die Security-Module das Zertifikat als vertrauenswürdig ein, kann eine Verbindung beispielsweise zu einem FTPS-Server hergestellt werden. Das importierte Zertifikat wird nicht zusätzlich an einer anderen Stelle in STEP 7 verwendet.

Bei SCALANCE S Modulen werden in diesem Register die Zertifizierungsstellen angezeigt, die für die Verifizierung der Security-Module durch externe Dienste notwendig sind.

### Zertifikate erneuern

#### Bedeutung

In diesem Dialog erneuern Sie CA-Zertifikate und Gerätezertifikate. Sie können bei Bedarf, z. B. bei kompromittierten Zertifikaten, ein Zertifikat importieren oder ein neues Zertifikat von STEP 7 erzeugen lassen.

#### So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf einen Listeneintrag im Zertifikatsmanager.
2. Wählen Sie im Kontextmenü den Befehl "Erneuern".
3. Wählen Sie aus, ob das neue Zertifikat selbst signiert oder durch eine Zertifizierungsstelle signiert werden soll.
4. Wenn das Zertifikat durch eine Zertifizierungsstelle signiert werden soll, wählen Sie über die Schaltfläche "Auswählen" die zu verwendende Zertifizierungsstelle aus. Zur Auswahl stehen dabei nur Zertifizierungsstellen, die im Zertifikatsspeicher des aktuellen Projekts abgelegt sind.

5. Geben Sie je nach Zertifikat die folgenden Werte in das Eingabefeld "Antragsteller" bzw. "Alternativer Antragstellername" ein:

| Zu erneuerndes Zertifikat  | Parameter                      |   |
|--|--------------------------------|---|
|  | Antragsteller                  | Alternativer Antragstellername  |
| CA-Zertifikate des Projekts  | Name des CA-Zertifikats        | -   |
| CA-Gruppenzertifikat   | Name des CA-Gruppenzertifikats | -   |
| SSL-Zertifikat für S7-CP   | Name des Security-Moduls       | IP-Adressen der Gigabit- und PROFINET-Schnittstelle, durch ein Komma getrennt |
| SSL-Zertifikat für PC-CP   | Name des Security-Moduls       | IP-Adresse des Security-Moduls  |
| SSL-Zertifikat für SCALANCE S, SCALANCE M, SOFTNET Security Client | Name des Security-Moduls       | -   |
| Gruppenzertifikat des Security-Moduls                              | Name des Gruppenzertifikats    | Von der CA abgeleitet   |

6. Wählen Sie einen Zeitraum aus, in dem das Zertifikat gültig ist. Standardmäßig wird in das Feld "Gültig von:" die aktuelle Zeit und in das Feld "Gültig bis:" der Wert des aktuellen Zertifikats eingetragen.

## Zertifikate ersetzen

### Bedeutung

In dem Dialog "Zertifizierungsstelle (CA) ändern" ersetzen Sie das bestehende CA-Zertifikat des Projekts oder CA-Gruppenzertifikat durch ein neues.

### So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf einen Listeneintrag im Register "CA".
2. Wählen Sie im Kontextmenü den Befehl "Ersetzen".
3. Der Dialog "Zertifizierungsstelle (CA) ändern" wird geöffnet.

Alle in der Tabelle "Betroffene Zertifikate" aufgelisteten Zertifikate werden neu abgeleitet. Somit kann das CA-Gruppenzertifikat einer bereits projektierten VPN-Gruppe innerhalb des Projekts durch das CA-Gruppenzertifikat eines anderen Projekts ersetzt werden. Die Gruppenzertifikate für die VPN-Gruppenteilnehmer werden also in beiden Projekten von dem gleichen CA-Gruppenzertifikat abgeleitet.

Laden Sie nach Änderungen im Zertifikatsmanager die Projektierung auf alle betroffenen Security-Module.

### Welches Format darf das Zertifikat haben?

Von der importierten Zertifizierungsstelle werden weitere Zertifikate in STEP 7 abgeleitet. Deshalb können Sie nur Zertifikate mit privatem Schlüssel auswählen.

- \*.p12

### Benutzer und Rollen verwalten

#### Regeln für Benutzernamen, Rollen und Passwörter

#### Welche Regeln gelten für Benutzernamen, Rollennamen und Passwörter?

Beachten Sie beim Anlegen oder Ändern eines Benutzers, einer Rolle oder eines Passworts die folgenden Regeln:

Tabelle 8-1 Regeln für die Benutzerverwaltung

|  |   |
|--|---|
| Erlaubte Zeichen                                 | Erlaubt sind folgende Zeichen des US-ASCII-Zeichensatzes:<br>0123456789<br>A...Z a...z<br>!#\$%&()*+,-./:;<=>?@ [\_{}~ ^` |
| Nicht erlaubte Zeichen                           | " ' "   |
| Länge des Benutzernamens                         | 1 ... 32 Zeichen  |
| Länge des Passworts                              | 8 ... 32 Zeichen  |
| Länge des Rollennamens                           | 1 ... 32 Zeichen  |
| Maximale Benutzeranzahl pro Projekt              | 128   |
| Maximale Benutzeranzahl auf einem Security-Modul | 32 + 1 Administrator beim Anlegen des Projekts  |
| Maximale Rollenanzahl pro Projekt                | 124 benutzerdefinierte + 4 systemdefinierte   |
| Maximale Rollenanzahl auf einem Security-Modul   | 33 benutzerdefinierte + 4 systemdefinierte  |

#### Hinweis

##### Benutzernamen und Passwörter

Als wichtige Maßnahme zur Erhöhung der Sicherheit achten Sie stets darauf, dass Benutzernamen und Passwörter mindestens 8 Zeichen lang sind und Sonderzeichen, Groß-/ Kleinschreibung, Zahlen etc. enthalten.



## Benutzer anlegen

### Bedeutung

Die in STEP 7 projektierten Security-Funktionen werden durch eine eigene Benutzerverwaltung vor unbefugtem Zugriff geschützt. Bevor Sie auf die globalen und lokalen Security-Einstellungen von Security-Modulen zugreifen können, müssen Sie sich in den lokalen Security-Einstellungen eines Security-Moduls unter dem Eintrag "Security-Eigenschaften" bei der Security-Konfiguration mit einem Benutzer anmelden.

### Ersten Benutzer im Projekt anlegen

Nach dem Anlegen des ersten Security-Moduls im Projekt müssen Sie zuerst einen Benutzer anlegen. Klicken Sie hierzu in den lokalen Security-Einstellungen des angelegten Security-Moduls unter dem Eintrag "Security-Eigenschaften" auf die Schaltfläche "Benutzeranmeldung" und geben Sie die Login-Daten des anzulegenden Benutzers an. Sie werden daraufhin mit dem angelegten Benutzer angemeldet und dem Benutzer wird die systemdefinierte Rolle "Administrator" zugewiesen. Diese Rolle beinhaltet vollständige Projektierungs- und Baugruppenrechte.

### Benutzer in der Benutzerverwaltung anlegen

Wenn Sie mit einem Benutzer bei der Security-Konfiguration angemeldet sind, können Sie in den globalen Security-Einstellungen unter dem Eintrag "Benutzerverwaltung" weitere Benutzer anlegen oder bestehende Benutzer löschen.

---

#### Hinweis

##### **Benutzer mit Rolle "Administrator"**

Im Projekt muss immer mindestens ein Benutzer mit vollständigen Projektierungsrechten vorhanden sein. Der Administrator, der beim ersten Aktivieren der Security-Funktionen im Projekt automatisch angelegt wird, kann nur gelöscht werden, solange mindestens ein weiterer Benutzer mit der systemdefinierten Rolle "Administrator" existiert.

---

Folgende Parameter stehen in der Benutzerverwaltung im Register "Benutzer" zur Verfügung:

Tabelle 8-2 Angaben im Register "Benutzer"

| Parameter    | Bedeutung  |
|--------------|--|
| Benutzername | Name des anzulegenden Benutzers.<br>Klicken Sie in der Spalte "Benutzername" auf den Eintrag "Neuen Benutzer hinzufügen", um einen neuen Benutzer anzulegen. |
| Passwort     | Passwort, mit dem sich der Benutzer bei der Anmeldung authentifiziert.   |
| Rolle        | Auswahl einer systemdefinierten oder benutzerdefinierten Rolle.  |
| Kommentar    | Optionale Kommentareingabe.  |

## Rollen anlegen

### Übersicht

Sie können einem Benutzer eine systemdefinierte oder benutzerdefinierte Rolle zuweisen. Die Baugruppenrechte einer benutzerdefinierten Rolle legen Sie pro Security-Modul fest.

### Systemdefinierte Rollen

Vordefiniert sind die im Folgenden aufgelisteten systemdefinierten Rollen. Den Rollen sind bestimmte Rechte zugewiesen, die auf allen Baugruppen gleich sind und die der Administrator nicht ändern oder löschen kann.

- Administrator  
Standardmäßige Rolle beim Anlegen einer Security-Konfiguration.  
Uneingeschränkte Zugriffsrechte auf alle Konfigurationsdaten und alle Security-Baugruppen.
- Standard  
Rolle mit eingeschränkten Zugriffsrechten, siehe Kapitel: Rechte verwalten (Seite 525)
- Diagnose  
Standardmäßige Rolle beim Anlegen eines neuen Benutzers.
  - Lesender Zugriff auf Konfigurationen.
  - Lesender Zugriff auf das Security-Modul in der Betriebsart "Online" für Test und Diagnose.
- Remote-Access  
Kann als Vorlage bei der Erstellung benutzerdefinierter Rollen verwendet werden.

Eine detaillierte Auflistung der Projektierungs- und Baugruppenrechte, die den systemdefinierten Rollen "Administrator", "Standard" und "Diagnose" zugeordnet sind, finden Sie in den Tabellen 1-3 bis 1-7 des Kapitels Rechte verwalten (Seite 525).

### Benutzerdefinierte Rolle

Zusätzlich zu den systemdefinierten Rollen können Sie benutzerdefinierte Rollen anlegen. Für eine benutzerdefinierte Rolle wählen Sie die Projektierungs- bzw. Baugruppenrechte und legen für jedes im Projekt verwendete Security-Modul die entsprechenden Baugruppenrechte fest. Die benutzerdefinierte Rolle weisen Sie dem entsprechenden Benutzer manuell zu.

### So erreichen Sie diese Funktion

1. Doppelklicken Sie in den globalen Security-Einstellungen auf den Eintrag "Benutzerverwaltung".
2. Wählen Sie in der Benutzerverwaltung das Register "Rollen".

Tabelle 8-3 Angaben im Register "Rollen"

| Parameter    | Bedeutung  |
|--------------|--|
| Rolle        | Frei wählbarer Rollenname.<br>Doppelklicken Sie auf den Eintrag "Neue Rolle hinzufügen", um eine neue benutzerdefinierte Rolle anzulegen.<br>Im Anschluss können Sie die Rechte für die angelegte Rolle bestimmen. |
| Beschreibung | Angabe der systemdefinierten Rolle. Bei benutzerdefinierten Rollen wird die Zeichenkette "Benutzerdefinierte Rolle" angezeigt.   |
| Kommentar    | Zusätzliche, optionale Kommentareingabe.   |

#### **Hinweis**

##### **Löschen von Rollen**

Eine benutzerdefinierte Rolle kann nur gelöscht werden, wenn sie keinem Benutzer zugewiesen ist. Weisen Sie dem Benutzer gegebenenfalls eine andere Rolle zu.

Systemdefinierte Rollen können nicht gelöscht werden.

## **Rechte verwalten**

### **So erreichen Sie diese Funktion**

1. Doppelklicken Sie in den globalen Security-Einstellungen auf den Eintrag "Benutzerverwaltung".
2. Wählen Sie in der Benutzerverwaltung das Register "Rollen".

### **Benutzerdefinierte Rolle erstellen und zuweisen**

1. Doppelklicken Sie auf den Eintrag "Neue Rolle hinzufügen".
2. Geben Sie einen Rollennamen ein.
3. Wählen Sie ggf. aus der Klappliste mit der Beschriftung "<Kopiere Rechte von>" die systemdefinierte Rolle aus, deren Rechte als Vorlage für die benutzerdefinierte Rolle verwendet werden sollen. Benutzerdefinierte Rollen sind über die Klappliste nicht auswählbar.  
Ergebnis: In der Rechtestliste der Benutzerrollen sind die Rechte ausgewählt, die der gewählten systemdefinierten Rolle zugeordnet sind.
4. Aktivieren bzw. deaktivieren Sie für jedes Security-Modul die Rechte, die der benutzerdefinierten Rolle zugewiesen werden sollen.
5. Weisen Sie die Rolle einem Benutzer im Register "Benutzer" zu.

## **Projektierungsrechte**

Projektierungsrechte sind unabhängig von Baugruppen und steuern die Berechtigungen zur Projektierung in STEP 7.

Je nach Benutzertyp stehen Ihnen die folgenden Projektierungsrechte zur Verfügung:

Tabelle 8-4 Projektierungsrechte

| Projektierungsrecht           | Administrator | Standard | Diagnose |
|-------------------------------|---------------|----------|----------|
| Security diagnostizieren      | x             | x        | x        |
| Security konfigurieren        | x             | x        | -        |
| Benutzer und Rollen verwalten | x             | -        | -        |

### Baugruppenrechte

Baugruppenrechte werden pro Baugruppe konfiguriert. In der Spalte "Dienst" wird angezeigt, auf welchen Dienst sich das jeweilige Recht bezieht. Über die Befehle "Rechte kopieren" bzw. "Rechte einfügen" des Kontextmenüs können Sie die Rechte von einer Baugruppe auf eine andere übertragen.

Je nach Benutzertyp stehen Ihnen die folgenden Baugruppenrechte zur Auswahl:

Tabelle 8-5 Baugruppenrechte CP x43-1 Advanced

| Recht innerhalb des Dienstes                                     | Administrator | Standard | Diagnose | Dienst      |
|--|---------------|----------|----------|-------------|
| Web: CP-Dateisystem formatieren *                                | x             | -        | -        | Dateisystem |
| FTP: Dateien vom CP-Dateisystem lesen                            | x             | x        | x        |             |
| FTP: Dateien auf das CP-Dateisystem schreiben                    | x             | x        | -        |             |
| FTP: Dateien (DBs) von der S7-CPU lesen **                       | x             | x        | x        | SPS         |
| FTP: Dateien (DBs) in die S7-CPU schreiben ***                   | x             | x        | -        |             |
| Applet: Variablen über projektierte Symbole lesen *              | x             | x        | -        |             |
| Applet: Variablen über projektierte Symbole schreiben *          |               |          |          |             |
| Applet: Variablen über absolute Adressen lesen *                 | x             | x        | x        |             |
| Applet: Variablen über absolute Adressen schreiben *             | x             | x        | -        |             |
| Applet: Status der Baugruppen im Rack lesen *                    | x             | x        | x        |             |
| Applet: Bestellnummern der Module im Baugruppenträger abfragen * | x             | x        | x        |             |
| SNMP: MIB-II lesen   | x             | x        | x        | SNMP        |
| SNMP: MIB-II schreiben   | x             | x        | -        |             |
| SNMP: Automation-MIB lesen                                       | x             | x        | x        |             |
| SNMP: LLDP-MIB lesen   | x             | x        | x        |             |
| SNMP: SNMPv2-MIB lesen   | x             | x        | x        |             |
| SNMP: MRP-MIB lesen  | x             | x        | x        |             |
| SNMP: MRP-MIB schreiben  | x             | x        | -        |             |
| TIA-Portal: Diagnose der Security-Baugruppe durchführen ****     | x             | x        | x        | Security    |
| Web: IP Access Control-Liste erweitern *                         | x             | -        | -        | Web         |
| Web: Auf Web-Diagnose und CP-Dateisystem zugreifen               | x             | x        | x        |             |
| Web: Test-E-Mail senden *  | x             | x        | x        |             |
| Web: Firmware aktualisieren *                                    | x             | x        | -        | Wartung     |
| Web: Nachladen von Diagnosetexten *                              | x             | x        | -        |             |

Tabelle 8-6 Baugruppenrechte CP 1628

| Recht innerhalb des Dienstes                                 | Administrator | Standard | Diagnose | Dienst   |
|--|---------------|----------|----------|----------|
| SNMP: MIB-II lesen   | x             | x        | x        | SNMP     |
| SNMP: MIB-II schreiben                                       | x             | x        | -        |          |
| SNMP: Automation-MIB lesen                                   | x             | x        | x        |          |
| SNMP: SNMPv2-MIB lesen                                       | x             | x        | x        |          |
| TIA-Portal: Diagnose der Security-Baugruppe durchführen **** | x             | x        | x        | Security |

Tabelle 8-7 Baugruppenrechte SCALANCE S

| Recht innerhalb des Dienstes                                 | Administrator | Standard | Diagnose | Dienst   |
|--|---------------|----------|----------|----------|
| Laden der Konfigurationsdateien                              | x             | x        | -        | Security |
| TIA-Portal: Diagnose der Security-Baugruppe durchführen **** | x             | x        | x        |          |

Tabelle 8-8 Baugruppenrechte CP 1543-1

| Recht innerhalb des Dienstes                                 | Administrator | Standard | Diagnose | Dienst      |
|--|---------------|----------|----------|-------------|
| FTP: Dateien vom CP-Dateisystem lesen                        | x             | x        | x        | Dateisystem |
| FTP: Dateien auf das CP-Dateisystem schreiben                | x             | x        | -        |             |
| TIA-Portal: Diagnose der Security-Baugruppe durchführen **** | x             | x        | x        | Security    |
| SNMP: Automation-MIB lesen                                   | x             | x        | x        | SNMP        |
| SNMP: IPv6-MIB lesen   | x             | x        | x        |             |
| SNMP: LLDP-MIB lesen   | x             | x        | x        |             |
| SNMP: MIB-II lesen   | x             | x        | x        |             |
| SNMP: MIB-II schreiben                                       | x             | x        | -        |             |
| SNMP: SNMPv2-MIB lesen                                       | x             | x        | x        |             |
| FTP: Dateien (DBs) von der S7-CPU lesen **                   | x             | x        | x        | SPS         |
| FTP: Dateien (DBs) in die S7-CPU schreiben ***               | x             | x        | -        |             |

- \* Um die Funktion anzuwenden, muss das Baugruppenrecht "Web: Auf Webdiagnose und CP-Dateisystem zugreifen" ebenfalls aktiviert sein.
- \*\* Um die Funktion anzuwenden, muss das Baugruppenrecht "FTP: Dateien vom CP-Dateisystem lesen" ebenfalls aktiviert sein.
- \*\*\* Um die Funktion anzuwenden, muss das Baugruppenrecht "FTP: Dateien auf das CP-Dateisystem schreiben" ebenfalls aktiviert sein.
- \*\*\*\* Um die Funktion anzuwenden, muss das Projektierungsrecht "Security diagnostizieren" ebenfalls aktiviert sein.

## Einstellen von Baugruppenrechten vor und nach dem Anlegen von Security-Modulen

Innerhalb einer benutzerdefinierten Rolle werden die Baugruppenrechte für jedes Security-Modul separat definiert. Wurde ein Security-Modul, für das Baugruppenrechte innerhalb einer Rolle eingestellt werden sollen, vor dem Hinzufügen dieser Rolle angelegt, dann werden für das Security-Modul von STEP 7 Baugruppenrechte gemäß der ausgewählten Rechtevorlage voreingestellt. Die voreingestellten Baugruppenrechte können dann beim Hinzufügen der Rolle angepasst werden. Wurde ein Security-Modul nach dem Hinzufügen einer Rolle

angelegt, dann werden für dieses Security-Modul keine Rechte voreingestellt. In diesem Fall müssen Sie die bestehende Rolle bearbeiten und für das Security-Modul nachträglich alle Baugruppenrechte selbst einstellen.

Bereits bestehende Baugruppenrechte können Sie auch durch Kopieren auf eine andere Baugruppe übernehmen und dort ggf. anpassen. Wählen Sie hierzu im Kontextmenü einer Baugruppe in den Baugruppenrechten den Menübefehl "Rechte kopieren" bzw. "Rechte einfügen".

## Konfigurationsdaten für SCALANCE M-Module generieren

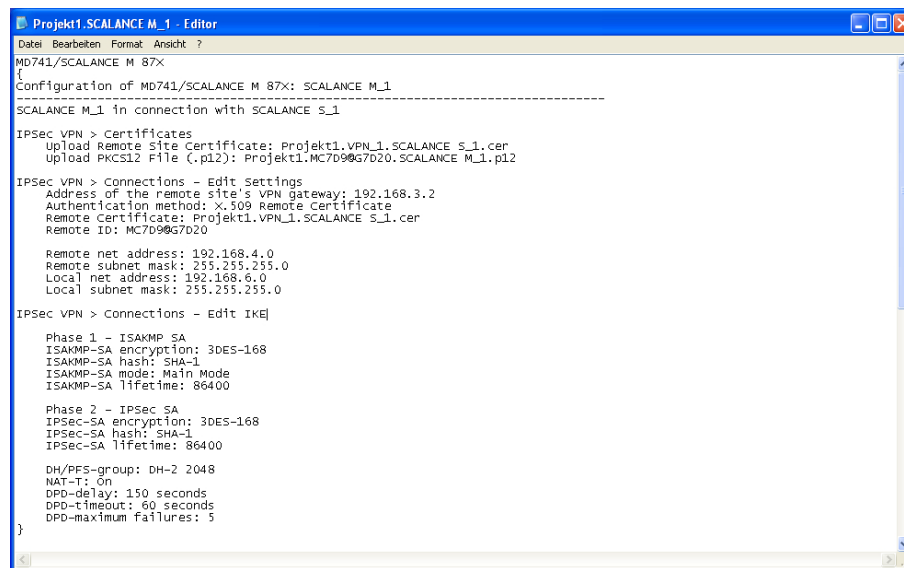
### Bezug

Sie können die VPN-Informationen zur Parametrierung eines SCALANCE M mit STEP 7 generieren. Dazu muss sich das Modul in mindestens einer VPN-Gruppe mit einem Security-Modul oder einem SOFTNET Security Client befinden. Mit den generierten Dateien können Sie dann den SCALANCE M über das Web Based Management des Geräts konfigurieren.

### Generierte Dateien

Folgende Dateitypen werden generiert:

- Exportdatei mit den Konfigurationsdaten
  - Dateityp: \*.txt-Datei im ASCII-Format
  - Enthält die exportierten Konfigurationsinformationen für das SCALANCE M einschließlich einer Information über die zusätzlich erzeugten Zertifikate.
- Modul-Zertifikat
  - Dateityp des privaten Schlüssels: \*.p12-Datei
  - Die Datei enthält das Modulzertifikat und das Schlüsselmaterial.
  - Der Zugriff ist passwortgeschützt.
- Gruppen-Zertifikat
  - Dateityp: \*.cer-Datei



```
Projekt1_SCALANCE M_1 - Editor
Datei Bearbeiten Format Ansicht ?
MD741/SCALANCE M 87X
{
Configuration of MD741/SCALANCE M 87X: SCALANCE M_1
-----
SCALANCE M_1 in connection with SCALANCE S_1
IPsec VPN > Certificates
  Upload Remote Site Certificate: Projekt1.VPN_1.SCALANCE S_1.cer
  Upload PKCS12 File (.p12): Projekt1.MC7D90G7D20.SCALANCE M_1.p12
IPsec VPN > Connections - Edit Settings
  Address of the remote site's VPN gateway: 192.168.3.2
  Authentication method: X.509 Remote Certificate
  Remote Certificate: Projekt1.VPN_1.SCALANCE S_1.cer
  Remote ID: MC7D90G7D20
  Remote net address: 192.168.4.0
  Remote subnet mask: 255.255.255.0
  Local net address: 192.168.6.0
  Local subnet mask: 255.255.255.0
IPsec VPN > Connections - Edit IKE
  Phase 1 - ISAKMP SA
  ISAKMP-SA encryption: 3DES-168
  ISAKMP-SA hash: SHA-1
  ISAKMP-SA mode: Main Mode
  ISAKMP-SA lifetime: 86400
  Phase 2 - IPsec SA
  IPsec-SA encryption: 3DES-168
  IPsec-SA hash: SHA-1
  IPsec-SA lifetime: 86400
  DH/PFS-group: DH-2 2048
  NAT-T: On
  DPD-delay: 150 seconds
  DPD-timeout: 60 seconds
  DPD-maximum failures: 5
}
```

Bild 8-1 SCALANCE M Konfigurationsdatei

## Hinweis

### Keine Übertragung an das Security Modul

Konfigurationsdateien werden nicht an das Security-Modul übertragen. Es wird eine ASCII-Datei generiert, mit der Sie das SCALANCE M konfigurieren können. Dazu muss sich der SCALANCE M in mindestens einer VPN-Gruppe mit einem anderen Security-Modul befinden.

## Gehen Sie so vor

1. Markieren Sie das Modul vom Typ "SCALANCE M".
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "SCALANCE M Konfiguration".
3. Aktivieren Sie das Kontrollkästchen "SCALANCE M Dateien generieren" und wählen Sie einen Speicherort für die Konfigurationsdateien.
4. Übersetzen Sie die Konfiguration des SCALANCE M Moduls.
5. Geben Sie im folgenden Dialog an, ob für die erstellte .p12-Datei ein eigenes Passwort erstellt werden soll.  
Wenn Sie "Abbrechen" wählen, wird als Passwort der Projektname vergeben, nicht das Projektpasswort.

Ergebnis: Die Dateien (.txt-Datei und Zertifikate) werden in dem von Ihnen angegebenen Verzeichnis abgespeichert.

## Betriebsart und Netzparameter für SCALANCE S-Module projektieren

### Übersicht

Informationen zur Projektierung von Betriebsart und Netzparametern von SCALANCE S Modulen finden Sie im Abschnitt Auto-Hotspot des Kapitels "SCALANCE S".

Die Projektierung der Netzparameter von CPs wird in den Kapiteln beschrieben, die sich mit CPs befassen.

## Firewall einrichten

### Übersicht zur Firewall

#### Bedeutung

Die Firewall-Funktionalität der Security-Module hat die Aufgabe, Netze und Stationen vor Fremdbeeinflussungen und Störungen zu schützen. Das bedeutet, dass nur bestimmte, vorher festgelegte Kommunikationsbeziehungen erlaubt werden. Nicht zugelassene Telegramme werden von der Firewall verworfen, ohne dass eine Antwort gesendet wird.

Zur Filterung des Datenverkehrs können u. a. IP-Adressen, IP-Subnetze, Portnummern oder MAC-Adressen verwendet werden. Außerdem können Sie eine Bandbreitenbegrenzung einstellen.

Die Firewall-Funktionalität kann für folgende Protokollebenen konfiguriert werden:

- IP-Firewall mit Stateful Packet Inspection (Layer 3 und 4)
- Firewall auch für Ethernet-"Non-IP"-Telegramme gemäß IEEE 802.3 (Layer 2)

Bei einem VPN-fähigen Security-Modul kann die Firewall auch für den verschlüsselten Datenverkehr (IPsec-Tunnel) eingesetzt werden. Beim Security-Modul SCALANCE S602 kann die Firewall nur für den unverschlüsselten Datenverkehr eingesetzt werden.

#### Arten von Firewall-Regeln

Firewall-Regeln beschreiben, welche Pakete in welche Richtung erlaubt bzw. verboten werden. Unterschieden wird zwischen IP-Paketfilter-Regeln und MAC-Paketfilter-Regeln.

#### Globale und lokale Firewall-Regeln

- Globale Firewall-Regeln können mehreren Security-Modulen gleichzeitig zugewiesen werden. Globale Firewall-Regeln werden in den globalen Security-Einstellungen projektieren.
- Lokale Firewall-Regeln werden in den lokalen Security-Einstellungen eines Security-Moduls projektieren.



## Dienst-Definitionen

Zusätzlich haben Sie die Möglichkeit, mit Hilfe von Dienst-Definitionen Firewall-Regeln kompakt und übersichtlich zu definieren. Dienst-Definitionen werden in den globalen Security-Einstellungen projektiert und können sowohl in den globalen als auch in den lokalen Firewall-Regeln verwendet werden.

## Automatisch erzeugte Firewall-Regeln für CP-Verbindungen

Für Verbindungen, die über CPs projektiert wurden, werden von STEP 7 automatisch Firewall-Regeln angelegt, die die Kommunikation mit dem jeweiligen Partner des CPs in der vorgegebenen Richtung (CP aktiv/passiv) freischalten. Dabei werden die Aufbaurichtungen der Verbindungen beachtet. Zur Anzeige dieser Firewall-Regeln muss bei aktiviertem erweitertem Firewall-Modus die Schaltfläche "Verbindungsregeln aktualisieren" betätigt werden. Die Firewall-Regeln werden daraufhin im erweiterten Firewall-Modus angezeigt.

Welche Firewall-Regeln automatisch erzeugt werden, erfahren Sie in folgenden Abschnitten.

- Für S7-300-/S7-400-/PC-CPs: Verbindungsbezogene automatische Firewall-Regeln (Seite 613) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".
- Für S7-1500-CPs: Verbindungsbezogene automatische Firewall-Regeln (Seite 623) im Kapitel "Security für S7-1500-CPs".

## Firewall aktivieren

In den lokalen Security-Einstellungen wird die Firewall-Funktionalität für ein bestimmtes Security-Modul über das Kontrollkästchen "Firewall aktivieren" gesteuert. Ist das Kontrollkästchen aktiviert, kann die Firewall konfiguriert werden und ist nach dem Ladevorgang wirksam. Befindet sich das Security-Modul in einer VPN-Gruppe, ist das Kontrollkästchen "Firewall aktivieren" standardmäßig aktiviert und kann nicht deaktiviert werden. Nach dem Umschalten in den erweiterten Firewall-Modus ist es nicht möglich, in den Standard-Modus zurückzuschalten. Für nähere Informationen zum Standard-Modus und erweiterten Modus der Firewall finden Sie im Kapitel:

Übersicht Lokale Firewall-Regeln (Seite 540).

## Globale Firewall-Regelsätze

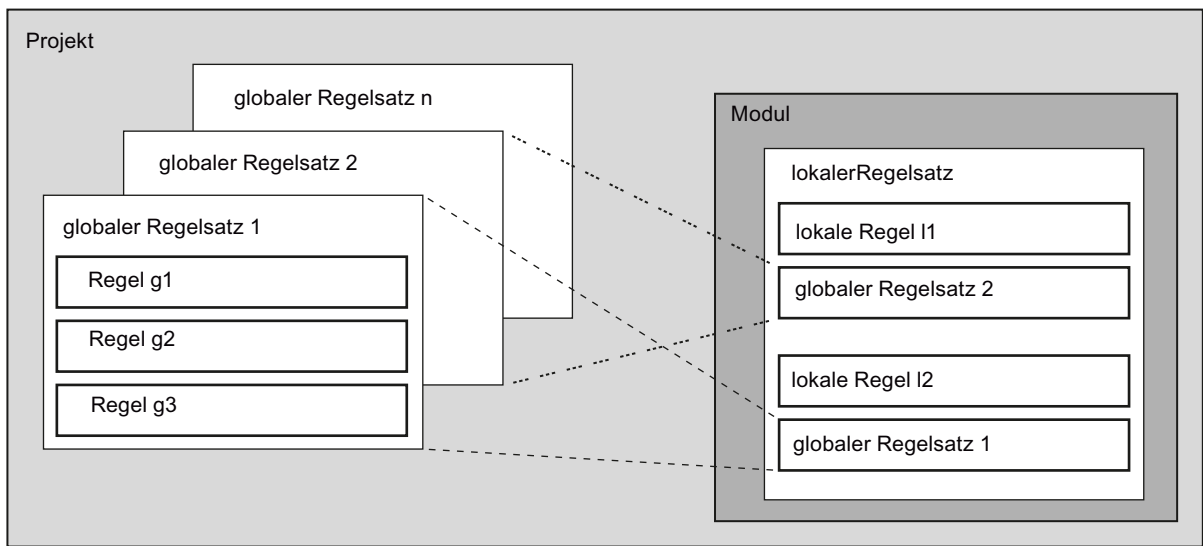
### Anwendung

Globale Firewall-Regelsätze werden modulunabhängig in den globalen Security-Einstellungen projektiert. Ein Firewall-Regelsatz besteht aus einer oder mehreren Firewall-Regeln und wird einzelnen Security-Modulen zugewiesen.

Unterschieden wird bei den globalen Firewall-Regelsätzen zwischen:

- IP-Regelsätzen
- MAC-Regelsätzen

Die folgende Darstellung verdeutlicht den Zusammenhang zwischen global definierten Regelsätzen und lokal verwendeten Regelsätzen.



### Projektierung

Bei der Projektierung von globalen Firewall-Regeln können Sie detaillierte Firewall-Einstellungen vornehmen. Sie können für einen einzelnen Teilnehmer einzelne Dienste freischalten oder für den Teilnehmer alle Dienste für den Zugriff auf die Station bzw. das Netz freigeben.

### Wann sind globale IP- und MAC-Firewall-Regeln sinnvoll?

Globale Firewall-Regeln sind dann sinnvoll, wenn Sie für mehrere Security-Module identische Filterkriterien für die Kommunikation definieren wollen.

---

#### Hinweis

##### Nur Regelsätze zuweisen, die vom Security-Modul unterstützt werden

Eine fehlerhafte Modulzuordnung kann zu unerwünschten Ergebnissen führen. Überprüfen Sie daher immer die modulspezifischen lokalen Firewall-Regeln im Ergebnis. Eine falsche Zuordnung von Firewall-Regelsätzen wird bei der automatischen Konsistenzprüfung nicht erkannt. Es werden nur die Firewall-Regelsätze korrekt übernommen, die das Security-Modul auch unterstützt. Ein globaler Firewall-Regelsatz, der eine Firewall-Regel mit der Richtung "Von: Extern" bzw. "Nach: Any" beinhaltet, kann einem CP 1628 nicht korrekt zugewiesen werden.

---

## Globale Firewall-Regelsätze - Vereinbarungen

### Globale Firewall-Regelsätze werden lokal genutzt

Folgende Vereinbarungen gelten bei der Erstellung eines globalen Firewall-Regelsatzes sowie bei der Zuweisung zu einem Modul:

- **Projektierungsansicht**  
Globale Firewall-Regelsätze werden in den globalen Security-Einstellungen projiziert.
- **Priorität**  
Lokal definierte Regeln haben standardmäßig eine höhere Priorität als globale IP- und MAC-Firewall-Regelsätze. Neu zugewiesene globale IP- und MAC-Firewall-Regelsätze werden daher in der lokalen Regelliste zunächst unten eingefügt.  
Die Priorität kann durch Verändern der Platzierung in der Regelliste verändert werden.
- **Regelsätze eingeben, ändern oder löschen**  
Globale Firewall-Regelsätze sind in der lokalen Regelliste der Firewall-Regeln bei den Moduleigenschaften nicht editierbar. Sie können dort nur angezeigt und gemäß der gewünschten Priorität platziert werden.  
Eine einzelne Regel kann in den lokalen Security-Einstellungen nicht aus einem zugeordneten Regelsatz gelöscht werden. Es kann nur der komplette Regelsatz aus der lokalen Regelliste entfernt werden. Die Firewall-Regelsätze in den globalen Security-Einstellungen bleiben davon unberührt.

### Globale Firewall-Regelsätze anlegen

#### So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Globale Firewall-Regelsätze" > "IP-Regelsätze" bzw. "MAC-Regelsätze".  
Ergebnis: Unter dem ausgewählten Eintrag werden die bisher erstellten IP-Regelsätze bzw. MAC-Regelsätze angezeigt.
2. Doppelklicken Sie auf den Eintrag "Neuen IP-Regelsatz hinzufügen" bzw. "Neuen MAC-Regelsatz hinzufügen".
3. Geben Sie die folgenden Daten ein:
  - Name: Projektweit eindeutige Bezeichnung des Regelsatzes. Der Name erscheint nach der Zuweisung des Regelsatzes in der lokalen Regelliste des Security-Moduls.
  - Beschreibung (optional): Geben Sie eine Beschreibung für den globalen Regelsatz ein.
4. Tragen Sie der Reihe nach die Firewall-Regeln in die Liste ein.  
Beachten Sie die Parameterbeschreibung in folgenden Kapiteln:  
Für IP-Regelsätze: IP-Paketfilter-Regeln definieren (Seite 541)  
Für MAC-Regelsätze: MAC-Paketfilter-Regeln definieren (Seite 544)

### Ergebnis

Sie haben den globalen Firewall-Regelsatz angelegt und können diesen nun den gewünschten Security-Modulen zuweisen.

Beachten Sie die Beschreibungen in folgendem Kapitel:


Globale Firewall-Regelsätze zuweisen (Seite 534)

## Globale Firewall-Regelsätze zuweisen

### Voraussetzung

Sie haben für die Security-Module, die Sie einem Firewall-Regelsatz zuweisen möchten, den erweiterten Firewall-Modus aktiviert.

### Vorgehensweise

1. Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Globale Firewall-Regelsätze" > "Modul einem Firewall-Regelsatz zuweisen".
2. Wählen Sie aus der Klappliste "Regelsatz" den Regelsatz aus, dem Sie das Security-Modul zuweisen möchten.  
In der rechten Tabelle werden die Security-Module angezeigt, die Sie dem ausgewählten Firewall-Regelsatz zuweisen können. In der linken Tabelle werden die Security-Module angezeigt, die dem ausgewählten Firewall-Regelsatz bereits zugeordnet sind.
3. Wählen Sie im Bereich "Verfügbare Module" die Security-Module aus, die Sie dem ausgewählten Regelsatz zuweisen möchten.
4. Klicken Sie auf die Schaltfläche , um die ausgewählten Module dem ausgewählten Regelsatz zuzuweisen.

### Ergebnis

Der globale Regelsatz wird von den zugewiesenen Security-Modulen als lokaler Regelsatz verwendet und erscheint in den lokalen Security-Einstellungen automatisch am Ende der Liste der Firewall-Regeln.

## IP-Dienste

### IP-Dienste definieren

#### So erreichen Sie diese Funktion

Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren".

### Vorgehensweise

Mit Hilfe der Definition von IP-Diensten können Sie Firewall-Regeln, die auf bestimmte Dienste angewendet werden, kompakt und übersichtlich definieren. Sie vergeben hierbei einen Namen und ordnen diesem die Dienstparameter zu.

Zusätzlich können Sie so definierte Dienste wiederum unter einem Gruppennamen zu Gruppen zusammenfassen.

Bei der Projektierung der Paketfilter-Regeln verwenden Sie dann diese Namen.

## Parameter für IP-Dienste

Die Definition der IP-Dienste erfolgt über folgende Parameter:

Tabelle 8-9 IP-Dienste: Parameter

| Parameter  | Bedeutung/Kommentar  | Auswahlmöglichkeiten / Wertebereiche  |
|------------|--|---|
| Name       | Name für den Dienst, der zur Identifikation in der Regeldefinition oder in der Gruppensammenfassung verwendet wird. Die Namen vordefinierter Dienste können nicht geändert werden. | <ul style="list-style-type: none"> <li>• Name muss mit einem Buchstaben beginnen.</li> <li>• Name darf keine Sonderzeichen enthalten.</li> <li>• Name darf maximal 20 Zeichen enthalten.</li> <li>• Name darf nicht doppelt vorkommen.</li> </ul> |
| Protokoll  | Auswahl des Protokolltyps  | <ul style="list-style-type: none"> <li>• TCP</li> <li>• UDP</li> <li>• Alle</li> </ul>  |
| Quell-Port | Es erfolgt Filterung anhand der hier angegebenen Portnummer; diese definiert den Dienstzugang beim Telegrammsender.  | Beispiele:<br>*: Port wird nicht geprüft<br>20 bzw. 21: FTP-Service   |
| Ziel-Port  | Es erfolgt Filterung anhand der hier angegebenen Portnummer; diese definiert den Dienstzugang beim Telegrammempfänger.   | Beispiele:<br>*: Port wird nicht geprüft<br>TCP 80: Web-HTTP-Service<br>TCP 102: S7-Protokoll   |

## ICMP-Dienste definieren

### So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren".
2. Selektieren Sie das Register "ICMP".

### Vorgehensweise

Mit Hilfe der Definition von ICMP-Diensten können Sie Firewall-Regeln, die auf bestimmte Dienste angewendet werden, kompakt und übersichtlich definieren. Sie vergeben hierbei einen Namen und ordnen diesem die Dienstparameter zu.

Zusätzlich können Sie so definierte Dienste unter einem Gruppennamen zu Gruppen zusammenfassen.

Bei der Projektierung der Paketfilter-Regeln verwenden Sie dann diese Namen.

### Parameter für ICMP-Dienste

| Parameter | Bedeutung/Kommentar   | Auswahlmöglichkeiten / Wertebereiche   |
|-----------|---|--|
| Name      | Frei definierbarer Name für den Dienst, der zur Identifikation in der Regeldefinition oder in der Gruppenzusammenfassung verwendet wird. Die Namen vordefinierter ICMPv6-Dienste sind nicht änderbar.   | <ul style="list-style-type: none"> <li>Name muss mit einem Buchstaben beginnen.</li> <li>Name darf keine Sonderzeichen enthalten.</li> <li>Name darf maximal 20 Zeichen enthalten.</li> <li>Name darf nicht doppelt vorkommen</li> </ul> |
| ICMPv6    | Wenn Sie dieses Kontrollkästchen aktivieren, wird der ICMP-Dienst als ICMPv6-Dienst deklariert und Sie können einen ICMPv6-spezifischen Typ und Code für den Dienst auswählen. Ein ICMPv6-Dienst kann nur in der Firewall-Regel eines Security-Moduls verwendet werden, welches IPv6 unterstützt. | <ul style="list-style-type: none"> <li>aktiviert</li> <li>deaktiviert (Standard)</li> </ul>  |
| Typ       | Typ der ICMPv4- bzw. ICMPv6-Message.  | Wenn das Kontrollkästchen "ICMPv6" deaktiviert ist, sind ICMPv4-spezifische Typen auswählbar. Wenn das Kontrollkästchen aktiviert ist, sind ICMPv6-spezifische Typen auswählbar.   |
| Code      | Code des ICMP-Typs.   | Werte sind abhängig vom gewählten Typ.   |

### Dienstgruppen anlegen

#### So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren".
2. Selektieren Sie das Register "Dienstgruppen".

#### Bildung von Dienstgruppen

Sie können mehrere Dienste durch die Bildung von Dienstgruppen zusammenfassen. Auf diese Weise können Sie komplexere Dienste aufbauen, die in den Paketfilter-Regeln dann durch einfache Namensauswahl verwendet werden können. IPv4- und IPv6-Dienste können in einer Dienstgruppe zusammengefasst werden.

Im geöffneten Register "Dienstgruppen" legen Sie Gruppen an. Anschließend ordnen Sie im Register "Gruppen-Management" die Dienste einer Gruppe zu.

### **Gehen Sie so vor**

1. Legen Sie in diesem Register zunächst Gruppen mit einem für Ihre Anforderungen zweckmäßigen Namen und einem optionalen Beschreibungstext an.
2. Wählen Sie dann das Register "Gruppen-Management". Dort können Sie den hier definierten Gruppen die zuvor festgelegten IP-Dienste zuordnen.

### **Dienstgruppen verwalten**

#### **So erreichen Sie diese Funktion**

1. Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren".
2. Selektieren Sie das Register "Gruppen-Management".

### **Bildung von Dienstgruppen**

Sie können mehrere Dienste durch die Bildung von Dienstgruppen zusammenfassen. Auf diese Weise können Sie komplexere Dienste aufbauen, die in den Paketfilter-Regeln dann durch einfache Namensauswahl verwendet werden können. IPv4- und IPv6-Dienste können in einer Dienstgruppe zusammengefasst werden.

In dem Register "Gruppen-Management " ordnen Sie einer ausgewählten Dienstgruppe Dienste zu, die Sie zuvor im Register "Dienstgruppen" angelegt haben.

### **Gehen Sie so vor**

1. Wählen Sie in diesem Register zunächst über die Klappliste "Dienstgruppen" eine zuvor im Register "Dienstgruppen" angelegte Gruppe aus.
2. Ordnen Sie dann aus dem rechten Auswahlfeld "Verfügbare Dienste" die gewünschten Dienste der Gruppe zu.

### **MAC-Dienste**

#### **MAC-Dienste definieren**

#### **So erreichen Sie diese Funktion**

Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Dienste" > "Dienste für MAC-Regeln definieren".

### **Bedeutung**

Mit Hilfe der Definition von MAC-Diensten können Sie Firewall-Regeln, die auf bestimmte Dienste angewendet werden, kompakt und übersichtlich definieren. Sie vergeben einen Namen und ordnen diesem die Dienstparameter zu.

8.1 Geräte und Netze konfigurieren

Zusätzlich können Sie so definierte Dienste unter einem Gruppennamen zu Gruppen zusammenfassen.

Bei der Projektierung der globalen oder lokalen Paketfilter-Regeln verwenden Sie dann diese Namen.

**Parameter für MAC-Dienste**

Eine MAC-Dienst-Definition wird durch protokollspezifische MAC-Parameter gebildet:

Tabelle 8-10 MAC-Dienste Parameter

| Parameter  | Bedeutung/Kommentar  | Auswahlmöglichkeiten / Wertebereiche  |
|--|--|---|
| Name   | Frei definierbarer Name für den Dienst, der zur Identifikation in der Regeldefinition oder in der Gruppenzusammenfassung verwendet wird.   | <ul style="list-style-type: none"> <li>Name muss mit einem Buchstaben beginnen.</li> <li>Name darf keine Sonderzeichen enthalten.</li> <li>Name darf maximal 20 Zeichen enthalten.</li> <li>Name darf nicht doppelt vorkommen.</li> </ul> |
| Protokoll  | Name des Protokolltyps: <ul style="list-style-type: none"> <li>ISO<br/>ISO bezeichnet Telegramme mit folgenden Eigenschaften:<br/>Lengthfield &lt;= 05DC (hex),<br/>DSAP= userdefined<br/>SSAP= userdefined<br/>CTRL= userdefined</li> <li>SNAP<br/>SNAP bezeichnet Telegramme mit folgenden Eigenschaften:<br/>Lengthfield &lt;= 05DC (hex),<br/>DSAP=AA (hex),<br/>SSAP=AA (hex),<br/>CTRL=03 (hex),<br/>OUI=userdefined,<br/>OUI-Type=userdefined</li> <li>PROFINET IO</li> </ul> | <ul style="list-style-type: none"> <li>ISO</li> <li>SNAP</li> <li>PROFINET IO</li> <li>0x (Code-Eingabe)</li> </ul>   |
| DSAP   | Destination Service Access Point: LLC-Empfänger-Adresse  |   |
| SSAP   | Source Service Access Point: LLC-Sender-Adresse  |   |
| CTRL   | LLC Control Field  |   |
| OUI  | Organizationally Unique Identifier<br>(die ersten 3 Bytes der MAC-Adresse = Hersteller Identifizierung)  |   |
| OUI-Typ  | Protokoll-Typ/-Identifizierung   |   |
| *) Die Protokolleingaben 0800 (hex) und 0806 (hex) werden nicht akzeptiert, da diese Werte für IP- bzw. ARP-Telegramme gelten. |  |   |



---

### **Hinweis**

#### **Verarbeitung für S7-CPs**

Es werden nur Einstellungen zu ISO-Frames mit DSAP=SSAP=FE (hex) verarbeitet. Andere Frame-Typen sind für S7-CPs nicht relevant und werden daher schon vor der Bearbeitung durch die Firewall verworfen.

---

### **Spezielle Einstellungen für SIMATIC NET Dienste**

Verwenden Sie für die Filterung spezieller SIMATIC NET Dienste bitte die folgenden Protokoll-Einstellungen:

- DCP (Primary Setup Tool):  
PROFINET
- SiClock:  
OUI= 08 00 06 (hex), OUI-Type= 01 00 (hex)

### **Dienstgruppen anlegen**

#### **So erreichen Sie diese Funktion**

1. Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Dienste" > "Dienste für MAC-Regeln definieren".
2. Selektieren Sie das Register "Dienstgruppen".

#### **Bildung von Dienstgruppen**

Sie können mehrere Dienste durch die Bildung von Dienstgruppen zusammenfassen. Auf diese Weise können Sie komplexere Dienste aufbauen, die in den Paketfilter-Regeln dann durch einfache Namensauswahl verwendet werden können.

Im geöffneten Register "Dienstgruppen" legen Sie Gruppen an. Anschließend ordnen Sie im Register "Gruppen-Management" die Dienste einer Gruppe zu.

#### **Gehen Sie so vor**

1. Legen Sie in diesem Register zunächst Gruppen mit einem für Ihre Anforderungen zweckmäßigen Namen und einem optionalen Beschreibungstext an.
2. Wählen Sie dann das Register "Gruppen-Management". Dort können Sie den hier definierten Gruppen die zuvor festgelegten MAC-Dienste zuordnen.

## Dienstgruppen verwalten

### So erreichen Sie diese Funktion

1. Wählen Sie in den globalen Security-Einstellungen den Eintrag "Firewall" > "Dienste" > "Dienste für MAC-Regeln definieren".
2. Selektieren Sie das Register "Gruppen-Management".

### Bildung von Dienstgruppen

Sie können mehrere Dienste durch die Bildung von Dienstgruppen zusammenfassen. Auf diese Weise können Sie komplexere Dienste aufbauen, die in den Paketfilter-Regeln dann durch einfache Namensauswahl verwendet werden können.

In dem Register "Gruppen-Management" ordnen Sie einer ausgewählten Dienstgruppe Dienste zu, die Sie zuvor im Register "Dienstgruppen" angelegt haben.

### Gehen Sie so vor

1. Wählen Sie in diesem Register zunächst über die Klappliste "Dienstgruppen" eine zuvor im Register "Dienstgruppen" angelegte Gruppe aus.
2. Ordnen Sie dann aus dem rechten Auswahlfeld "Verfügbare Dienste" die gewünschten Dienste der Gruppe zu.

## Übersicht Lokale Firewall-Regeln

### Bedeutung

Lokale Firewall-Regeln werden in den lokalen Security-Einstellungen eines Security-Moduls projektiert und gelten nur für dieses Security-Modul. Nach dem Aktivieren der Firewall-Funktionalität können Sie entweder vordefinierte Firewall-Regeln verwenden oder Firewall-Regeln im erweiterten Firewall-Modus definieren.

### Vordefinierte Firewall-Regeln verwenden

Hier greifen Sie auf einfache, vordefinierte Regeln zurück. Sie können nur dienstspezifische Regeln freischalten. Die freigeschalteten Dienste sind für alle Teilnehmer in der angegebenen Richtung zulässig. Detaillierte Informationen zur Definition von Firewall-Regeln in diesem Dialogfenster entnehmen Sie den folgenden, modulspezifischen Kapiteln:

- Für SCALANCE S: Auto-Hotspot
- Für S7-300-/S7-400-/PC-CPs: Auto-Hotspot
- Für S7-1500-CPs: Auto-Hotspot

## Firewall-Regeln im erweiterten Firewall-Modus definieren

Im erweiterten Firewall-Modus können Sie detaillierte Firewall-Einstellungen vornehmen. Sie können für einen einzelnen Teilnehmer einzelne Dienste freischalten oder für den Teilnehmer alle Dienste für den Zugriff auf die Station bzw. das Netz freigeben. Den erweiterten Firewall-Modus aktivieren Sie über das Kontrollkästchen "Firewall im erweiterten Modus aktivieren". In den lokalen Security-Einstellungen sind die Firewall-Regeln dann unter "Firewall" > "IP-Regeln" bzw. "MAC-Regeln" projektierbar. Welche Projektierungsmöglichkeiten hier im Einzelnen zur Verfügung stehen, siehe folgende Kapitel:

Für IP-Paketfilter-Regeln: IP-Paketfilter-Regeln definieren (Seite 541)

Für MAC-Paketfilter-Regeln: MAC-Paketfilter-Regeln definieren (Seite 544)

---

### Hinweis

#### **Kein Deaktivieren des erweiterten Firewall-Modus möglich**

Wenn Sie den erweiterten Firewall-Modus einmal aktiviert haben, können Sie diesen nicht mehr deaktivieren.

---

## Mengengerüst

| Anzahl Firewall-Regeln (Erweiterter Firewall-Modus) |             |
|---|-------------|
| SCALANCE S  | 256 maximal |
| S7-CPs  | 256 maximal |
| CP 1628   | 256 maximal |

## IP-Paketfilter-Regeln definieren

### Bedeutung

Mit IP-Paketfilter-Regeln filtern Sie nach IP-Telegrammen wie z. B. TCP, UDP oder ICMP.

Innerhalb einer Paketfilter-Regel können Sie auf Definitionen der IP-Dienste zurückgreifen.

## IP-Paketfilter-Regeln eintragen

Tragen Sie der Reihe nach die Firewall-Regeln in die Liste ein. Beachten Sie die folgende Parameterbeschreibung.

Tabelle 8-11 IP-Regeln: Parameter

| Parameter                | Bedeutung/Kommentar   | Auswahlmöglichkeiten / Wertebereiche   |
|--------------------------|---|--|
| Aktion                   | Zulassungsfestlegung (Freigabe/Sperre)  | <ul style="list-style-type: none"> <li>• Allow<br/>Telegramme gemäß Definition zulassen.</li> <li>• Drop<br/>Telegramme gemäß Definition sperren.</li> </ul> <p>Für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt und danach manuell angepasst wurden:</p> <ul style="list-style-type: none"> <li>• Allow*</li> <li>• Drop*</li> </ul> <p>Wenn Sie automatisch angelegte Verbindungsregeln verändern, werden diese bei der Wahl der Option "*" nicht erneut von STEP 7 erzeugt und überschrieben.</p>             |
| Von / Nach               | Auswahl der Kommunikationsrichtungen, für die die Regel gelten soll.  | <p>Werden in separaten Abschnitten beschrieben.</p> <ul style="list-style-type: none"> <li>• Für SCALANCE S Module: IP-Paketfilter-Richtungen SCALANCE S (Seite 594)</li> <li>• Für S7-300-/S7-400-/PC-CPs: IP-Paketfilter-Richtungen S7-300-/S7-400-/PC-CPs (Seite 611)</li> <li>• Für S7-1500-CPs: IP-Paketfilter-Richtungen CP 1543-1 (Seite 622)</li> </ul>  |
| IPv6 (nur für CP 1543-1) | Wenn Sie dieses Kontrollkästchen aktivieren, können Sie einen zuvor definierten ICMPv6-Dienst in der Firewall-Regel verwenden. Für CP 1543-1 V1.1 können Sie nach dem Aktivieren des Kontrollkästchens zusätzlich IPv6-Adressen in die Eingabefelder "Quell-IP-Adresse" und "Ziel-IP-Adresse" eintragen. Für CP 1543-1 V1.0 können Sie nach dem Aktivieren des Kontrollkästchens keine Quell-IP-Adresse und keine Ziel-IP-Adresse für die Firewall-Regel eintragen. | <p>Das Kontrollkästchen ist nur aktivierbar und deaktivierbar, wenn in den Eingabefeldern "Quell-IP-Adresse" und "Ziel-IP-Adresse" keine Einträge vorhanden sind.</p> <p>Wenn IPv6 in den lokalen Einstellungen des CP 1543-1 deaktiviert wurde, können Sie das Kontrollkästchen "IPv6" in den lokalen Security-Einstellungen des CPs nicht aktivieren und somit auch keine ICMPv6-Dienste sowie IPv6-Adressen in Firewall-Regeln nutzen. Bestehende Firewall-Regeln, die IPv6 nutzen, werden bei deaktiviertem IPv6 ausgegraut dargestellt.</p> |
| Quell-IP-Adresse         | Die Firewall-Regel wird auf diejenigen Telegramme angewendet, deren Sender die hier angegebene IP-Adresse hat. Wenn Sie keine IP-Adresse angeben, gilt die Firewall-Regel für alle Teilnehmer innerhalb der Kommunikationsrichtung, die Sie in Spalte "Von" ausgewählt haben.   | Weitere Informationen zu IP-Adressen finden Sie im Abschnitt IP-Adressen in IP-Paketfilter-Regeln (Seite 546).   |

| Parameter           | Bedeutung/Kommentar  | Auswahlmöglichkeiten / Wertebereiche   |
|---------------------|--|--|
| Ziel-IP-Adresse     | Die Firewall-Regel wird auf diejenigen Telegramme angewendet, deren Empfänger die hier angegebene IP-Adresse hat. Wenn Sie keine IP-Adresse angeben, gilt die Firewall-Regel für alle Teilnehmer innerhalb der Kommunikationsrichtung, die Sie in Spalte "Nach" ausgewählt haben.  |  |
| Dienst              | Name des verwendeten IP/ICMP-Dienstes oder der Dienstgruppe.<br>Sie wählen hier einen der von Ihnen im Dialog "IP-Dienste" definierten Dienste: <ul style="list-style-type: none"> <li>• IP-Dienste</li> </ul> oder <ul style="list-style-type: none"> <li>• ICMP-Dienste</li> </ul> Bevor Sie einen ICMPv6-Dienst auswählen, müssen Sie das Kontrollkästchen "IPv6" aktivieren. | Die Klappliste bietet die in den globalen Security-Einstellungen projektierten Dienste und Dienstgruppen zur Auswahl an.<br>Keine Angabe bedeutet: es wird kein Dienst geprüft, die Regel gilt für alle Dienste. |
| Bandbreite (Mbit/s) | Einstellmöglichkeit für eine Bandbreitenbegrenzung<br>Kann nur eingegeben werden, wenn bei Aktion "Allow" ausgewählt ist.<br>Ein Paket passiert die Firewall, wenn die Allow-Regel zutrifft und die zulässige Bandbreite für diese Regel noch nicht überschritten worden ist   | CP x43-1 und SCALANCE S < V3.0: 0.001 ... 100 Mbit/s<br>CP 1628 und SCALANCE S ≥ V3.0: 0.001 ... 1000 Mbit/s<br>Für globale und benutzerspezifische Regeln: 0.001 ... 100 Mbit/s                                 |
| Logging             | Ein- und Ausschalten des Loggings für diese Regel  | <ul style="list-style-type: none"> <li>• aktiviert</li> <li>• deaktiviert (Standard)</li> </ul>  |
| Nr.                 | Automatisch vergebene Nummer der Regel   |  |
| Kommentar           | Platz für eigene Erläuterung der Regel   | Ist ein Kommentar mit "AUTO" gekennzeichnet, wurde er für eine automatische Verbindungsregel angelegt. Für von Ihnen angelegte Regeln ist die Kommentareingabe optional.   |

Tabelle 8-12 Bedeutung der Einträge des Kontextmenüs

| Eintrag des Kontextmenüs  | Bedeutung  |
|---|--|
| Löschen   | Löschen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz. Hinweise zum Löschen eines global definierten, lokal zugewiesenen Regelsatzes: Indem Sie den Regelsatz hier löschen, wird nur die Zuordnung zum Security-Modul aufgehoben. |
| Als globalen Regelsatz speichern (nur für lokale Firewall-Regeln) | Kopiert die markierte Firewall-Regel und fügt sie als globalen Firewall-Regelsatz in die globalen Security-Einstellungen ein. Die aktuell für das Security-Modul projektierte Firewall-Konfiguration bleibt von diesem Vorgang unberührt.                  |
| Nach oben verschieben   | Bewegen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz um eine Position in der Liste nach oben. Dadurch wird die verschobene Regel / der verschobene Regelsatz mit einer höheren Priorität behandelt.                              |
| Nach unten verschieben  | Bewegen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz um eine Position in der Liste nach unten. Dadurch wird die verschobene Regel / der verschobene Regelsatz mit einer niedrigeren Priorität behandelt.                         |
| Dienst für IP-Regeln definieren                                   | Öffnen Sie hiermit den Dialog, in dem die IP-Dienste und Dienstgruppen verwaltet werden.   |

### MAC-Paketfilter-Regeln definieren

#### Bedeutung

Mit MAC-Paketfilter-Regeln filtern Sie nach MAC-Telegrammen.

Innerhalb einer Paketfilter-Regel können Sie auf Definitionen der MAC-Dienste zurückgreifen.

### MAC-Paketfilter-Regeln eintragen

Tragen Sie der Reihe nach die Firewall-Regeln in die Liste ein. Beachten Sie die folgende Parameterbeschreibung.

Tabelle 8-13 MAC-Regeln: Parameter

| Parameter  | Bedeutung/Kommentar  | Auswahlmöglichkeiten / Wertebereiche   |
|------------|--|--|
| Aktion     | Zulassungsfestlegung (Freigabe/ Sperre)                              | <ul style="list-style-type: none"> <li>• Allow<br/>Telegramme gemäß Definition zulassen.</li> <li>• Drop<br/>Telegramme gemäß Definition sperren.</li> </ul>   |
| Von / Nach | Auswahl der Kommunikationsrichtungen, für die die Regel gelten soll. | <p>Werden in separaten Abschnitten beschrieben.</p> <ul style="list-style-type: none"> <li>• Für SCALANCE S Module: MAC-Paketfilter-Richtungen SCALANCE S (Seite 594)</li> <li>• Für S7-300-/S7-400-/PC-CPs: MAC-Paketfilter-Richtungen S7-300-/S7-400-/PC-CPs (Seite 611)</li> <li>• Für S7-1500-CPs: MAC-Paketfilter-Richtungen CP 1543-1 (Seite 623)</li> </ul> |

| Parameter           | Bedeutung/Kommentar   | Auswahlmöglichkeiten / Wertebereiche   |
|---------------------|---|--|
| Quell-MAC-Adresse   | Die Firewall-Regel wird auf diejenigen Telegramme angewendet, deren Sender die hier angegebene MAC-Adresse hat. Wenn Sie keine MAC-Adresse angeben, gilt die Firewall-Regel für alle Teilnehmer innerhalb der Kommunikationsrichtung, die Sie in Spalte "Von" ausgewählt haben.     | MAC-Adresse im korrekten Format  |
| Ziel-MAC-Adresse    | Die Firewall-Regel wird auf diejenigen Telegramme angewendet, deren Empfänger die hier angegebene MAC-Adresse hat. Wenn Sie keine MAC-Adresse angeben, gilt die Firewall-Regel für alle Teilnehmer innerhalb der Kommunikationsrichtung, die Sie in Spalte "Nach" ausgewählt haben. |  |
| Dienst              | Name des verwendeten MAC-Dienstes oder der Dienstgruppe   | Die Klappliste bietet die projektierten Dienste und Dienstgruppen zur Auswahl an.<br>Keine Angabe bedeutet, dass kein Dienst geprüft wird und die Regel für alle Dienste gilt.   |
| Bandbreite (Mbit/s) | Einstellmöglichkeit für eine Bandbreitenbegrenzung. Kann nur eingegeben werden, wenn bei Aktion "Allow" ausgewählt ist.<br>Ein Paket passiert die Firewall, wenn die Allow-Regel zutrifft und die zulässige Bandbreite für diese Regel noch nicht überschritten worden ist.         | CP x43-1 und SCALANCE S < V3.0: 0.001 ... 100 Mbit/s<br>CP 1628 und SCALANCE S ≥ V3.0: 0.001 ... 1000 Mbit/s<br>Für globale und benutzerspezifische Regeln: 0.001 ... 100 Mbit/s |
| Logging             | Ein- und Ausschalten des Logging für diese Regel.   | <ul style="list-style-type: none"> <li>• aktiviert</li> <li>• deaktiviert (Standard)</li> </ul>  |
| Nr.                 | Automatisch vergebene Nummer der Regel  |  |
| Kommentar           | Platz für eigene Erläuterung der Regel  | Ist ein Kommentar mit "AUTO" gekennzeichnet, wurde er für eine automatische Verbindungsregel angelegt. Für von Ihnen angelegte Regeln ist die Kommentareingabe optional.         |

Tabelle 8-14 Bedeutung der Menübefehle

| Schaltfläche  | Bedeutung  |
|---|--|
| Löschen   | Löschen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz. Hinweise zum Löschen eines global definierten, lokal zugewiesenen Regelsatzes: Indem Sie den Regelsatz hier löschen, wird nur die Zuordnung zum Security-Modul aufgehoben. |
| Als globalen Regelsatz speichern (nur für lokale Firewall-Regeln) | Kopiert die markierte Firewall-Regel und fügt sie als globalen Firewall-Regelsatz in die globalen Security-Einstellungen ein. Die aktuell für das Security-Modul projektierte Firewall-Konfiguration bleibt von diesem Vorgang unberührt.                  |
| Nach oben verschieben   | Bewegen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz um eine Position in der Liste nach oben. Dadurch wird die verschobene Regel / der verschobene Regelsatz mit einer höheren Priorität behandelt.                              |

| Schaltfläche                     | Bedeutung  |
|----------------------------------|--|
| Nach unten verschieben           | Bewegen Sie hiermit die markierte Regel oder den markierten globalen Regelsatz um eine Position in der Liste nach unten. Dadurch wird die verschobene Regel / der verschobene Regelsatz mit einer niedrigeren Priorität behandelt. |
| Dienst für MAC-Regeln definieren | Öffnen Sie hiermit den Dialog, in dem die MAC-Dienste und Dienstgruppen verwaltet werden.  |

## IP-Adressen in IP-Paketfilter-Regeln

### Eingabe von IP-Adressen in IP-Paketfilter-Regeln

In IP-Paketfilter-Regeln haben Sie folgende Möglichkeiten, IP-Adressen einzugeben:

- keine Angabe  
Die Regel gilt für alle IP-Adressen.
- eine IP-Adresse  
Die Regel gilt genau für die angegebene Adresse.
- Adressband  
Die Regel gilt für alle im Adressband erfassten IP-Adressen.  
Ein Adressband wird definiert, indem die Anzahl der gültigen Bit-Stellen in der IP-Adresse angegeben wird und zwar in der Form: [IP-Adresse]/[Anzahl der zu berücksichtigenden Bits]
  - [IP-Adresse]/24 bedeutet demnach, dass nur die höchstwertigen 24 Bit der IP-Adresse in der Filterregel berücksichtigt werden. Das sind die ersten drei Stellen der IP-Adresse.
  - [IP-Adresse]/25 bedeutet, dass nur die ersten drei Stellen und das höchstwertige Bit der vierten Stelle der IP-Adresse in der Filterregel berücksichtigt werden.
- Adressbereich  
Für die Quell-IP-Adresse kann ein Adressbereich in folgendem Format eingegeben werden:  
[Start-IP-Adresse]-[End-IP-Adresse]

## IPv4-Adressen

Eine IPv4-Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind.

Tabelle 8-15 Beispiele für Adressbänder bei IPv4-Adressen

| Quell-IP-Adresse bzw. Ziel-IP-Adresse | Adressband   |                 | Anzahl Adressen *) |
|---------------------------------------|--------------|-----------------|--------------------|
|                                       | von          | bis             |                    |
| 192.168.0.0/16                        | 192.168.0.0  | 192.168.255.255 | 65.536             |
| 192.168.10.0/24                       | 192.168.10.0 | 192.168.10.255  | 256                |
| 192.168.10.0/25                       | 192.168.10.0 | 192.168.10.127  | 128                |
| 192.168.10.0/26                       | 192.168.10.0 | 192.168.10.63   | 64                 |
| 192.168.10.0/27                       | 192.168.10.0 | 192.168.10.31   | 32                 |



| Quell-IP-Adresse bzw.<br>Ziel-IP-Adresse | Adressband   |               | Anzahl<br>Adressen *) |
|--|--------------|---------------|-----------------------|
|  | von          | bis           |                       |
| 192.168.10.0/28                          | 192.168.10.0 | 192.168.10.15 | 16                    |
| 192.168.10.0/29                          | 192.168.10.0 | 192.168.10.7  | 8                     |
| 192.168.10.0/30                          | 192.168.10.0 | 192.168.10.3  | 4                     |

\*) Hinweis: Beachten Sie, dass die Netzwerkadresse und die Broadcastadresse in einem Adressband nicht als IP-Adressen von Netzwerkteilnehmern zur Verfügung stehen.

## IPv6-Adressen

IPv6-Adressen bestehen aus 8 Blöcken mit jeweils vier Hexadezimalziffern (128 Bit insgesamt). Die Blöcke sind durch einen Doppelpunkt getrennt. IPv6-Adressen können nur in IP-Paketfilter-Regeln für den CP 1543-1 V1.1 eingetragen werden.

Beispiel: fd00:ffff:ffff:ffff:ffff:ffff:2f33:8f21

Regeln / Vereinfachungen:

- Führende Nullen innerhalb eines Blocks dürfen weggelassen werden.  
Beispiel: Statt 2001:0db8:2426:08d3:1457:8a2e:0070:7344 kann auch 2001:db8:2426:8d3:1457:8a2e:70:7344 notiert werden.
- Wenn ein oder mehrere Blöcke den Wert 0 (bzw. 0000) haben, ist eine verkürzte Schreibweise möglich.  
Beispiel: Statt 2001:0db8:0:0:0:0:1428:57ab kann auch 2001:db8::1428:57ab notiert werden.  
Damit die Eindeutigkeit gewahrt bleibt, darf diese Verkürzung nur einmal innerhalb der gesamten Adresse angewendet werden.
- Dezimalnotation mit Punkt-Schreibweise  
Für die letzten 2 Blöcke bzw. 4 Byte kann die herkömmliche Dezimalnotation in Punkt-Schreibweise verwendet werden.  
Beispiel: Die IPv6-Adresse fd00::ffff.125.1.0.1 ist äquivalent zu fd00::ffff:7d01:1.
- Adressbandnotation in IP-Paketfilterregeln: Analog zu IPv4-Adressen können IPv6-Adressen ebenfalls in Form von Adressbändern notiert werden.  
Beispiel: Der Eintrag "2001:0db8:85a3:08d3:1319:8a2e:0:0/96" umfasst alle IPv6-Adressen von 2001:0db8:85a3:08d3:1319:8a2e:0:0 bis 2001:0db8:85a3:08d3:1319:8a2e:ffff:ffff.

## Modulspezifische Log-Einstellungen vornehmen

### Log-Einstellungen Übersicht

### Log-Einstellungen in der Konfiguration

Die hier vorgenommenen Log-Einstellungen werden mit der Konfiguration in das Modul geladen und werden mit dem Start des Security-Moduls wirksam.

Sie können die projektierten Paketfilter-Log-Einstellungen in den Online-Funktionen bei Bedarf einschränken. Haben Sie beispielsweise IP- und MAC-Logging projektiert, können Sie in den Online-Funktionen einstellen, dass lediglich IP-Logging angezeigt wird.

### Aufzeichnungsverfahren und Ereignisklassen

Sie können hier festlegen, welche Daten aufgezeichnet werden sollen. Dadurch aktivieren Sie die Aufzeichnung bereits mit dem Laden der Konfiguration in das Security-Modul.

Außerdem wählen Sie in der Konfiguration eine oder beide der möglichen Aufzeichnungsverfahren:

- Lokales Logging
- Netzwerk Syslog

Das Security-Modul kennt für beide Aufzeichnungsverfahren jeweils die drei folgenden Arten von Ereignissen:

- Paketfilter-Ereignisse
- Audit-Ereignisse
- System-Ereignisse

### Lokales Logging konfigurieren

#### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Log-Einstellungen" > "Lokaler Log-Speicher"

## Lokales Logging konfigurieren

Tabelle 8-16 Lokales Logging - Einstellungen für Log-Ereignisse

| Log-Ereignis               | Bedeutung   | Bemerkungen  |
|----------------------------|---|--|
| Paketfilter-Log (Firewall) | Der Paketfilter-Log zeichnet bestimmte Pakete des Datenverkehrs auf. Es werden nur solche Datenpakete geloggt, auf die eine projektierte Paket-Filter-Regel (Firewall) zutrifft, oder auf die der Basisschutz reagiert (korrupte bzw. ungültige Pakete). Voraussetzung ist, dass das Logging für die Paketfilter-Regel aktiviert ist.       | Paketfilter-Logdaten sind nicht remanent<br>Die Daten werden in einem flüchtigen Speicher des Security-Moduls abgelegt und stehen deshalb nach einem Ausschalten der Spannungsversorgung nicht mehr zur Verfügung. Zur remanenten Speicherung können Sie die im "Online & Diagnose"-Dialog angezeigten Log-Daten zusätzlich in einer Datei abspeichern.        |
| Audit-Log                  | Das Logging für Audit-Ereignisse ist immer aktiviert.<br>Die Speicherung erfolgt immer im Umlaufspeicher.<br>Der Audit-Log zeichnet automatisch fortlaufend sicherheitsrelevante Ereignisse auf, beispielsweise Benutzeraktionen wie das Ein- oder Ausschalten des Paket-Logging oder das Laden von Konfigurationen auf das Security-Modul. | Audit-Logdaten sind remanent<br>Die Daten werden in einem remanenten Speicher des Security-Moduls abgelegt und stehen deshalb auch nach einem Ausschalten der Spannungsversorgung noch zur Verfügung.<br><b>Hinweis für CPs:</b><br>Die Audit-Logdaten sind bei CPs nicht remanent. Zur Sicherung der Daten sollte deshalb ein Syslog-Server verwendet werden. |
| System-Log                 | Der System-Log zeichnet automatisch fortlaufend System-Ereignisse, wie z. B. den Start eines Prozesses oder den gescheiterten Login-Versuch eines Benutzers, auf.<br>Zur Konfiguration des Ereignisfilters und der Leitungsdiagnose wählen Sie den Eintrag "Log-Einstellungen" > "System-Ereignisse konfigurieren".                         | System-Logdaten sind nicht remanent<br>Die Daten werden in einem flüchtigen Speicher des Security-Moduls abgelegt und stehen deshalb nach einem Ausschalten der Spannungsversorgung nicht mehr zur Verfügung. Zur remanenten Speicherung können Sie die im "Online & Diagnose"-Dialog angezeigten Log-Daten zusätzlich in einer Datei abspeichern.             |

Tabelle 8-17 Lokales Logging - Speicherverfahren für die Daten-Aufzeichnung

| Speicherverfahren | Bedeutung  |
|-------------------|--|
| Umlaufspeicher    | Bei Erreichen des Puffer-Endes wird die Aufzeichnung am Puffer-Anfang mit dem Überschreiben der ältesten Einträge fortgesetzt. |
| Linearer Speicher | Die Aufzeichnung stoppt, wenn der Puffer voll ist.   |

## System-Ereignisse konfigurieren

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Log-Einstellungen" > "System-Ereignisse konfigurieren"

### Filterung der System-Ereignisse

Sie stellen in diesem Dialog für die System-Ereignisse eine Filterebene ein. Standardmäßig sind die folgenden Werte eingestellt:

- SCALANCE S: Ebene 3 (Error)
- CP: Ebene 3 (Error)

Die ausgewählte Filterebene muss die gleiche oder eine niedrigere Priorisierung haben als die für die Leitungsdiagnose eingestellte Severity, siehe Tabelle "Parametrierung der Leitungsdiagnose" (nicht für CPs).

Empfehlung: Wählen Sie als Filterebene "Error" oder einen höheren Wert, um die Aufzeichnung von allgemeinen, nicht kritischen Ereignissen zu unterbinden.

### Hinweis für CP

Wählen Sie für CPs nur Ebene 3 oder Ebene 6 aus, da für CPs nur Ereignisse dieser Ebenen erzeugt werden.

- Bei Auswahl von Ebene 3 werden die Fehlermeldungen der Ebenen 0 bis 3 ausgegeben.
- Bei Auswahl von Ebene 6 werden die Fehlermeldungen der Ebenen 0 bis 6 ausgegeben.

### Eigenschaften der System-Ereignisse – Leitungsdiagnose (Nicht für CPs)

Die Leitungsdiagnose erzeugt ein spezielles System-Ereignis. Dabei wird bei einem einstellbaren Prozentsatz fehlerhafter Telegramme ein System-Ereignis erzeugt. Diesem System-Ereignis wird die in diesem Dialog eingestellte Severity und Facility zugewiesen.

Tabelle 8-18 Parametrierung der Leitungsdiagnose

| Funktion / Option / Parameter | Bedeutung  |
|-------------------------------|--|
| Aktivieren                    | Aufzeichnung ein- und ausschalten.   |
| Limit                         | Einstellbarer Prozentsatz fehlerhafter Telegramme, bei dessen Erreichen ein System-Ereignis ausgelöst wird.                          |
| Facility                      | Sie wählen hier aus der Klappliste eine Facility, mit der das zur Aufzeichnung erfasste System-Ereignis gekennzeichnet wird.         |
| Severity                      | Über die Severity gewichten Sie die System-Ereignisse der Leitungsdiagnose im Verhältnis zur Severity der übrigen System-Ereignisse. |

---

#### Hinweis

##### Severity der System-Ereignisse der Leitungsdiagnose

Achten Sie darauf, dass Sie den System-Ereignissen der Leitungsdiagnose keine geringere Severity zuweisen, als Sie für den Filter eingestellt haben. Bei einer geringeren Severity passieren diese Ereignisse den Filter nicht und werden nicht aufgezeichnet.

---

## Netzwerk-Syslog konfigurieren

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Log-Einstellungen" > "Netzwerk-Syslog".

## Netzwerk-Syslog konfigurieren

Tabelle 8-19 Netzwerk-Syslog - Grundeinstellungen

| Option / Parameter         |   |
|----------------------------|---|
| Netzwerk-Syslog aktivieren | Die Übermittlung von Logging-Ereignissen zum Syslog-Server ein- und ausschalten.  |
| Syslog-Server              | Geben Sie hier die IP-Adresse des Syslog-Servers ein.<br>Der Syslog-Server muss vom Security-Modul aus über die angegebene IP-Adresse erreichbar sein, ggf. über die Router-Projektierung in den lokalen Security-Einstellungen unter dem Eintrag "Routing". Wenn der Syslog-Server nicht erreichbar ist, werden keine Syslog-Nachrichten versendet. Sie können diesen Betriebszustand anhand entsprechender System-Meldungen erkennen. Um das Versenden der Syslog-Informationen erneut zu aktivieren, müssen Sie ggf. die Routing-Informationen aktualisieren und einen Neustart des Security-Moduls veranlassen. |
| Modulname                  | Das Security-Modul verwendet den hier dargestellten Modulnamen als Hostnamen gegenüber dem Syslog-Server. Der Modulname kann an dieser Stelle nicht geändert werden.  |

Tabelle 8-20 Netzwerk-Syslog - Einstellungen für Log-Ereignisse

| Log-Ereignis                      | Projektiertung   | Bemerkungen   |
|-----------------------------------|--|---|
| Paketfilter-Ereignisse (Firewall) | <p>Der Paketfilter-Log zeichnet bestimmte Pakete des Datenverkehrs auf. Es werden nur solche Datenpakete geloggt, auf die eine projektierte Paket-Filter-Regel (Firewall) zutrifft, oder auf die der Basisschutz reagiert (korrupte bzw. ungültige Pakete). Voraussetzung ist, dass das Logging für die Paket-Filter-Regel aktiviert ist.</p> <p>Durch die Einstellung von Facility und Severity lassen sich Syslog-Meldungen entsprechend ihrer Herkunft und ihres Schweregrades klassifizieren. Die Zuordnung erfolgt über Klapplisten. Jedem Ereignis wird die Severity und Facility zugewiesen, die Sie hier einstellen.</p> | <p>Welchen Wert Sie hier wählen, hängt von der Auswertung im Syslog-Server ab.</p> <p>Lassen Sie den Standardwert "default" eingestellt, wird durch das Security-Modul festgelegt, mit welcher Kombination aus Facility und Severity das Ereignis angezeigt wird.</p>                           |
| Audit-Ereignisse                  | <p>Der Audit-Log zeichnet automatisch fortlaufend sicherheitsrelevante Ereignisse auf, beispielsweise Benutzeraktionen wie das Ein- oder Ausschalten des Paket-Logging oder das Laden von Konfigurationen auf das Security-Modul.</p> <p>Die Zuordnung der Severity und Facility erfolgt über Klapplisten. Jedem Ereignis wird die Severity und Facility zugewiesen, die Sie hier einstellen.</p>  | <p>Welchen Wert Sie hier für Severity und Facility wählen, hängt von der Auswertung im Syslog-Server ab.</p> <p>Lassen Sie den Standardwert "default" eingestellt, wird durch das Security-Modul festgelegt, mit welcher Kombination aus Facility und Severity das Ereignis angezeigt wird.</p> |
| System-Ereignisse                 | <p>Der System-Log zeichnet automatisch fortlaufend System-Ereignisse, wie z. B. den Start eines Prozesses oder den gescheiterten Login-Versuch eines Benutzers, auf.</p>   | <p>Zur Konfiguration des Ereignisfilters und der Leitungsdiagnose wählen Sie in den lokalen Security-Einstellungen den Eintrag "Log-Einstellungen" &gt; "System-Ereignisse konfigurieren".</p>  |

## Security-Modul als Router

### Übersicht zu Routing-Einstellungen

#### Bedeutung

Indem Sie das Security-Modul im Routing-Modus betreiben, werden die Netze am internen und externen Port zu separaten Subnetzen.

Zusätzlich haben Sie folgende Möglichkeiten:

- Spezifische Routen einstellen - einstellbar in den lokalen Security-Einstellungen unter "Routing" (nur für SCALANCE S), siehe Routen festlegen (Seite 595) im Kapitel "SCALANCE S".
- Standard-Router verwenden - einstellbar in den lokalen Security-Einstellungen unter "Externe Schnittstelle [P1] rot" bzw. "Interne Schnittstelle [P2] grün", siehe Netzparameter konfigurieren (Seite 590) im Kapitel "SCALANCE S".
- NAT/NAPT-Routing - einstellbar in den lokalen Security-Einstellungen unter "NAT / NAPT" (nicht für PC-CPs und CP 1543-1). Für die Nutzung von NAT/NAPT-Routing muss sich das Security-Modul im Routing-Modus befinden.

### **Routing-Modus aktivieren (nur für SCALANCE S Module nötig)**

Wenn sie den Routing-Modus aktiviert haben, werden die Telegramme weitergeleitet, die an eine im jeweiligen Subnetz (intern oder extern) vorhandene IP-Adresse gerichtet sind. Darüber hinaus gelten die für die jeweilige Übertragungsrichtung getroffenen Firewall-Regeln.

Für diese Betriebsart müssen Sie in den lokalen Security-Einstellungen eine interne IP-Adresse und eine interne Subnetzmaske für die Adressierung des Routers am internen Subnetz projektieren. Alle nicht zu einem Subnetz gehörenden Netzwerkanfragen werden durch das Security-Modul in ein anderes Subnetz weitergeleitet.

Hinweis: Im Gegensatz zum Bridge-Modus des Security-Moduls gehen im Routing-Modus VLAN-Tags verloren.

1. Wählen Sie in den lokalen Security-Einstellungen unter "Betriebsart" die Option "Routing-Modus".
2. Tragen Sie in den lokalen Security-Einstellungen unter "Interne Schnittstelle [P2] grün" > "Ethernet-Adressen" eine interne IP-Adresse und eine interne Subnetzmaske für die Adressierung des Routers am internen Subnetz in die Eingabefelder ein.

### **Übersicht zu NAT/NAPT**

#### **Modulspezifische Funktion**

Diese Funktion ist nicht für PC-CPs und S7-1500-CPs verfügbar.

#### **So erreichen Sie diese Funktion**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "NAT / NAPT".
3. Aktivieren Sie je nach Anforderung eine Adressumsetzung gemäß NAT (Network Address Translation) oder NAPT (Network Address Port Translation).
4. Projektieren Sie die Adressumsetzung gemäß den Angaben in den folgenden Kapiteln.

#### **Adressumsetzung mit NAT (Network Address Translation)**

NAT ist ein Protokoll zur Adressumsetzung zwischen zwei Adressräumen. Hauptaufgabe ist die Umsetzung von privaten Adressen in öffentliche, d. h. in IP-Adressen, die im Internet

8.1 Geräte und Netze konfigurieren

verwendet und auch geroutet werden. Dadurch wird erreicht, dass die Adressen des internen Netzes nach außen im externen Netz nicht bekannt werden. Die internen Teilnehmer sind im externen Netz nur über die in der Adressumsetzungsliste (NAT-Tabelle) festgelegten externen IP-Adressen sichtbar.

Handelt es sich bei der externen IP-Adresse nicht um die Adresse des Security-Moduls und ist die interne IP-Adresse eindeutig, wird dies als 1:1 NAT bezeichnet. Beim 1:1 NAT wird die interne Adresse ohne Portumsetzung auf diese externe Adresse umgesetzt. Ansonsten handelt es sich um n:1 NAT.

**Adressumsetzung mit NAPT (Network Address Port Translation)**

Die Adressumsetzung bei NAPT verändert die Zieladresse und den Zielport in einer Kommunikationsbeziehung.

Umgesetzt werden Telegramme, die vom externen Netz kommen und an die externe IP-Adresse des Security-Moduls gerichtet sind. Ist der Zielport des Telegramms identisch mit einem der Werte, der in der Spalte "Externer Port" angegeben ist, so leitet das Security-Modul die Anfrage an den internen Teilnehmer mit der IP-Adresse weiter, die in der Spalte "Interne IP-Adresse" angegeben ist. Als Zielport wird der Port verwendet, der in der Spalte "Interner Port" angegeben ist. Bei der Rückantwort setzt das Security-Modul als Quell-IP-Adresse und als Quellport die Werte ein, die im Eingabefeld "Externe IP-Adresse" bzw. in der Spalte "Externer Port" eingetragen sind.

Der Unterschied zu NAT liegt darin, dass bei NAPT auch Ports umgesetzt werden können. Es findet keine 1:1 Umsetzung, sondern eine n:1 Umsetzung von IP-Adressen statt. Eine öffentliche IP-Adresse wird unter Zusatz von Portnummern in eine Reihe von privaten IP-Adressen umgesetzt.

**Konsistenzprüfung - diese Regeln müssen Sie beachten**

Beachten Sie für die Adresszuweisung folgende Regeln, um konsistente Einträge zu erhalten:

| Prüfung / Regel   | Prüfung erfolgt |             |
|---|-----------------|-------------|
|   | lokal           | projektweit |
| Die projektierte Netz-ID des internen Subnetzes muss unterschiedlich zur Netz-ID des externen Subnetzes sein.   |                 | x           |
| Die internen IP-Adressen dürfen nicht identisch sein mit den IP-Adressen des Moduls.  |                 | x           |
| Übernehmen Sie den durch die Subnetzmaske bestimmten Anteil für die IP-Adressen: <ul style="list-style-type: none"> <li>• Die externe IP-Adresse muss im selben Subnetzbereich liegen wie die projektierte externe IP-Adresse des Security-Moduls</li> <li>• Die interne IP-Adresse muss im selben Subnetzbereich liegen wie die projektierte interne IP-Adresse des Security-Moduls</li> </ul> |                 | x           |
| Eine IP-Adresse, die in der NAT/NAPT-Adressumsetzungsliste verwendet wird, darf keine Multicast-Adresse und keine Broadcast-Adresse sein.   |                 | x           |
| Die für die NAPT-Umsetzung vergebenen externen Ports liegen im Bereich > 0 und ≤ 65535.<br>Port 123 (NTP), 443 (HTTPS), 514 (Syslog), 500+4500 (IPsec) und 161 (SNMP) sind davon ausgeschlossen.  | x               |             |



| Prüfung / Regel  | Prüfung erfolgt |             |
|--|-----------------|-------------|
|  | lokal           | projektweit |
| Die externe IP-Adresse des Security-Moduls darf in der NAT-Tabelle nur für die Richtung "nach Extern" verwendet werden.  | x               |             |
| Die interne IP-Adresse des Security-Moduls darf in der NAT-Tabelle und in der NAPT-Tabelle nicht verwendet werden.   |                 | x           |
| Duplikatsprüfung in der NAT-Tabelle<br>Eine externe IP-Adresse, die mit Richtung "Dst-NAT (von Extern)" oder "Src-NAT + Dst-NAT (Extern)" verwendet wird, darf nur einmal in der NAT-Tabelle erscheinen.   | x               |             |
| Duplikatsprüfung in der NAPT-Tabelle<br><ul style="list-style-type: none"> <li>• Eine externe Portnummer darf nur einmal eingetragen sein.</li> <li>• Die Portnummern bzw. Portbereiche der externen Ports dürfen sich nicht überschneiden.</li> </ul> | x               |             |
| Sobald der Routing-Modus aktiviert wurde, müssen dem Security-Modul die Adressparameter für die interne Schnittstelle (IP-Adresse/Subnetz) zugeordnet werden.  | x               |             |
| Interne NAPT-Ports können im Bereich $> 0$ und $\leq 65535$ liegen.  | x               |             |

## Siehe auch

Übersicht zu Routing-Einstellungen (Seite 552)

## NAT/NAPT-Routing

### NAT aktivieren

Der Eingabebereich für NAT wird aktiviert. NAT-Adressumsetzungen werden erst durch die nachfolgend beschriebene Option und Einträge in die Adressumsetzungsliste wirksam. Zusätzlich müssen Sie die Firewall passend konfigurieren.

Die IP-Adressumsetzung kann über die folgenden Schnittstellen erfolgen:

- Extern: Die Adressumsetzung erfolgt an der externen Schnittstelle

Die IP-Adressumsetzung kann in folgende Richtungen erfolgen:

- Destination-NAT (Dst-NAT): Die IP-Adressumsetzung erfolgt von extern nach intern. Vom externen Subnetz kommende Telegramme werden auf die angegebene externe IP-Adresse geprüft und mit der angegebenen internen IP-Adresse in das interne Netz weitergeleitet. Der Zugriff von extern nach intern über die externe Adresse ist möglich.
- Source-NAT (Src-NAT): Die IP-Adressumsetzung erfolgt von intern nach extern. Vom internen Subnetz kommende Telegramme werden auf die angegebene interne IP-Adresse geprüft und mit der angegebenen externen IP-Adresse in das externe Netz weitergeleitet. Der Zugriff von intern nach extern ist möglich. Am externen Netz wirkt die externe Adresse.
- Source- und Destination-NAT (Src-NAT + Dst-NAT): Die IP-Adressumsetzung kann von intern oder extern erfolgen. Der Zugriff von intern und extern ist möglich. Am externen Netz wirkt die externe Adresse.

### Eingabemöglichkeiten für Adressumsetzung an der externen Schnittstelle

Soll die Adressumsetzung an der externen Schnittstelle erfolgen, stehen Ihnen die folgenden Eingabemöglichkeiten zur Verfügung.

#### Umsetzungsart "Dst-NAT (von Extern)"

| Feld               | Mögliche Eingaben  | Bedeutung   |
|--------------------|--|---|
| Externe IP-Adresse | <ul style="list-style-type: none"> <li>IP-Adresse im externen Subnetz</li> </ul> Bei dynamischer Adressvergabe funktioniert diese Umsetzungsart nicht. | Ziel-IP-Adresse im externen Netz, über die auf eine IP-Adresse im internen Subnetz zugegriffen werden soll. Stimmt in einem Telegramm die Zieladresse mit der eingegebenen Adresse überein, wird die Adresse durch die entsprechende interne IP-Adresse ausgetauscht. |
| Interne IP-Adresse | <ul style="list-style-type: none"> <li>IP-Adresse im internen Subnetz</li> </ul>   | Die Ziel-IP-Adresse wird durch interne IP-Adresse ersetzt. Die Adressumsetzung erfolgt von extern nach intern.  |

#### Umsetzungsart "Src-NAT (nach Extern)"

| Feld               | Mögliche Eingaben  | Bedeutung  |
|--------------------|--|--|
| Externe IP-Adresse | <ul style="list-style-type: none"> <li>IP-Adresse im externen Subnetz</li> <li>IP-Adresse des Security-Moduls, falls die Option "Freigabe für alle internen Teilnehmer nach außen zulassen" nicht aktiviert ist.</li> </ul> Bei dynamischer Adressvergabe ist keine Eingabe möglich. | Eingabe der IP-Adresse, die als neue Quell-IP-Adresse verwendet werden soll.<br>Handelt es sich bei der hier eingegebenen Adresse nicht um die IP-Adresse des Security-Moduls, wird diese zur Alias-Adresse. Dies bedeutet, dass die angegebene Adresse zusätzlich als Adresse an der externen Schnittstelle registriert wird. Stellen Sie sicher, dass mit dieser Adresse kein IP-Adresskonflikt besteht. |
| Interne IP-Adresse | <ul style="list-style-type: none"> <li>IP-Adresse im internen Subnetz</li> </ul>   | Die Quell-IP-Adresse des angegebenen internen Teilnehmers wird durch die angegebene externe Adresse ersetzt.<br>Die IP-Adressumsetzung erfolgt also von intern nach extern.  |
|                    | <ul style="list-style-type: none"> <li>Subnetz bzw. IP-Adressbereich</li> </ul>  | Die Quell-IP-Adressen aus dem angegebenen Subnetz oder IP-Adressbereich werden durch die externe IP-Adresse ersetzt. Der Quellport wird ersetzt.<br>Der Bereich wird durch die Adresse des Subnetzes angegeben.  |

Zusätzlich kann die folgende Funktion aktiviert werden:

- **Freigabe für alle internen Teilnehmer nach außen zulassen**  
Indem Sie diese Option wählen, erfolgt für alle von intern nach extern gehenden Telegramme eine Umsetzung der internen IP-Adresse auf die externe Modul IP-Adresse und einer zusätzlich vom Modul vergebenen Port-Nummer. Die Verwendung der IP-Adresse der externen Schnittstelle in der Spalte "Externe IP-Adresse" ist dann nicht mehr zulässig.  
Dieses Verhalten wird anhand der zusätzlich eingeblendeten NAT-Tabelle verdeutlicht, die über den Eintrag "Freigabe für alle internen Teilnehmer nach außen zulassen" in den lokalen Security-Einstellungen aufgerufen werden kann. Dort wird mit einem Symbol "\*" in der Spalte "Interne IP-Adresse" angezeigt, dass alle von intern nach extern gerichteten Telegramme umgesetzt werden.

### Umsetzungsart "Src-NAT + Dst-NAT (Extern)"

| Feld               | Mögliche Eingaben  | Bedeutung   |
|--------------------|--|---|
| Externe IP-Adresse | <ul style="list-style-type: none"> <li>• IP-Adresse im externen Subnetz</li> </ul> Bei dynamischer Adressvergabe funktioniert diese Umsetzungsart nicht. | Erfolgt die Adressumsetzung von intern nach extern: siehe Tabelle "Src-NAT (nach Extern)" |
| Interne IP-Adresse | <ul style="list-style-type: none"> <li>• IP-Adresse im internen Subnetz</li> </ul>   | Erfolgt die Adressumsetzung von extern nach intern: siehe Tabelle "Dst-NAT (von Extern)"  |

### NAPT aktivieren

Der Eingabebereich für NAPT wird aktiviert. NAPT-Umsetzungen werden erst durch die nachfolgend beschriebene Option und Einträge in die Liste wirksam. Zusätzlich müssen Sie die Firewall passend konfigurieren.

| Feld               | Mögliche Eingaben  | Bedeutung  |
|--------------------|--|--|
| Externer Port      | Port oder Portbereich<br>Beispiel für die Eingabe eines Portbereiches: 2000:2005 | Ein Teilnehmer im externen Netz kann einem Teilnehmer im internen Subnetz ein Telegramm senden, indem er diese Portnummer verwendet. |
| Interne IP-Adresse | IP-Adresse im internen Subnetz   | IP-Adresse des angesprochenen Teilnehmers im internen Subnetz.   |
| Interner Port      | Port   | Portnummer eines Teilnehmers im internen Subnetz.  |

### Uhrzeitsynchronisation konfigurieren

### Übersicht zur Uhrzeitsynchronisation

#### Bedeutung

Zur Überprüfung der zeitlichen Gültigkeit eines Zertifikates und für die Zeitstempel von Log-Einträgen werden auf dem Security-Modul Datum und Uhrzeit geführt. Für CPs ist die Uhrzeitsynchronisation standardmäßig aktiv.

Projektierbar sind folgende Alternativen:

- SIMATIC: Empfängt das Security-Modul MMS-Uhrzeitnachrichten, so wird seine lokale Uhrzeit synchronisiert, sofern nicht das NTP-Verfahren projektiert wurde (MMS = Manufacturing Message Specification).
- NTP: Automatisches Stellen und periodischer Abgleich der Uhrzeit über einen Network Time Protocol-Server.

**Hinweis**

Die Zeitsynchronisation bezieht sich lediglich auf das Security-Modul und kann nicht zur Synchronisation von Geräten im internen Netz des Security-Moduls verwendet werden. S7-CPs können die Uhrzeit an andere Baugruppen der Station weiterleiten.

**Hinweis**

**Firewall für Kommunikation mit NTP-Server projektieren**

Wenn der NTP-Server vom Security-Modul nicht erreichbar ist, müssen Sie die Telegramme des NTP-Servers in der Firewall explizit freigeben (UDP, Port 123).

**NTP-Server definieren**

**NTP-Server in globalen Security-Einstellungen anlegen**

In den globalen Security-Einstellungen können nur gesicherte NTP-Server (secure) angelegt und CPs zugewiesen werden. Ungesicherte NTP-Server für SCALANCE S Module und CPs müssen Sie deshalb in den lokalen Security-Einstellungen anlegen.

1. Doppelklicken Sie in den globalen Security-Einstellungen auf den Eintrag "NTP".
2. Doppelklicken Sie auf den Eintrag "Neuen NTP-Server hinzufügen".
3. Geben Sie einen Namen für den NTP-Server (secure) ein.
4. Geben Sie die IP-Adresse des NTP-Servers (secure) ein.
5. Geben Sie die Verschlüsselungsparameter für den NTP-Server (secure) an.

| Eigenschaft       | Bedeutung   |
|-------------------|---|
| Schlüssel-ID      | Numerischer Wert zwischen 1 ... 65534.  |
| Authentifizierung | Wählen Sie den Authentifizierungsalgorithmus aus.   |
| Hex/ASCII         | Wählen Sie das Format für den NTP-Schlüssel aus.  |
| Schlüssel         | Geben Sie den NTP-Schlüssel mit folgenden Längen ein:<br>Hex: 22 ... 40 Zeichen<br>ASCII: 11 ... 20 Zeichen |

6. Weisen Sie einen CP dem angelegten NTP-Server (secure) zu, siehe Kapitel: Security-Modul einem NTP-Server (secure) zuweisen (Seite 562).

### **NTP-Server in den lokalen Security-Einstellungen anlegen**

1. Selektieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Uhrzeitsynchronisation".
3. Wählen Sie den gewünschten Synchronisierungsmodus aus.
4. Falls Sie den Synchronisierungsmodus "NTP" ausgewählt haben, geben Sie einen Namen und die IP-Adresse des NTP-Servers ein.
5. Falls Sie den Synchronisierungsmodus "NTP (secure)" ausgewählt haben, wählen Sie in der Spalte "Name" einen NTP-Server (secure) aus, den Sie in den globalen Security-Einstellungen angelegt haben.

### **Mengengerüst für NTP-Server**

Beim Anlegen von NTP-Servern gelten die folgenden Regeln:

- Sie können projektweit 32 NTP-Server anlegen.
- Sie können einem Security-Modul max. 4 NTP-Server zuweisen.

### **Import / Export von NTP-Servern (secure)**

Über die Menübefehle "Import" bzw. "Export" des Kontextmenüs können Sie die Schlüsselliste des aktuell selektierten NTP-Servers (secure) in den globalen Security-Einstellungen exportieren und die Datei in einen NTP-Server (secure) importieren bzw. umgekehrt.

### **Uhrzeitsynchronisation für ein Security-Modul konfigurieren**

#### **So erreichen Sie diese Funktion**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Uhrzeitsynchronisation".
3. Aktivieren Sie das Kontrollkästchen "Uhrzeitsynchronisation aktivieren".

### **Alternativen der Uhrzeitsynchronisation**

Projektierbar sind folgende Alternativen:

Tabelle 8-21 Zeitsynchronisation für CP

| Auswahlmöglichkeit | Bedeutung / Auswirkung   |
|--------------------|--|
| SIMATIC            | Empfängt das Security-Modul MMS-Uhrzeitnachrichten, so wird seine lokale Uhrzeit synchronisiert, sofern nicht das NTP-Verfahren projektiert wurde (MMS = Manufacturing Message Specification). |
| NTP                | Automatisches Stellen und periodischer Abgleich der Uhrzeit mittels eines NTP-Servers.   |
| NTP (secure)       | Automatisches Stellen und periodischer Abgleich der Uhrzeit mittels eines NTP-Servers (secure).  |

Tabelle 8-22 Zeitsynchronisation für SCALANCE S

| Auswahlmöglichkeit | Bedeutung / Auswirkung   |
|--------------------|--|
| SIMATIC            | Empfängt das Security-Modul MMS-Uhrzeitnachrichten, so wird seine lokale Uhrzeit synchronisiert, sofern nicht das NTP-Verfahren projektiert wurde (MMS = Manufacturing Message Specification). |
| NTP                | Automatisches Stellen der Uhrzeit mittels eines NTP-Servers.   |

### Modus zur Zeitsynchronisation auswählen

Gehen Sie folgendermaßen vor:

1. Wählen Sie den Synchronisierungsmodus aus.
2. Abhängig vom ausgewählten Modus stehen Ihnen folgende Einstellungsmöglichkeiten zur Verfügung:

- **SIMATIC:** Wählen Sie für S7-CPs aus, ob der S7-CP die Uhrzeit übernehmen oder auch weiterleiten soll. Beim CP 1628 wird die Uhrzeit stets weitergeleitet. Stellen Sie für CPs zudem die Richtung für das Weiterleiten der Uhrzeit ein.  
Verfügbare Richtungen:
  - Automatisch (nur für S7-CPs): Der CP nimmt die Uhrzeit aus der Station bzw. vom LAN entgegen und leitet diese an die Station bzw. zum LAN weiter. Wenn mehrere CPs in der Station betrieben werden, kann diese automatische Einstellung zu Kollisionen führen. Um dies zu vermeiden, können Sie die Richtung der Weiterleitung gezielt festlegen.
  - Von Station (nur für S7-CPs).
  - Von LAN.Bei aktivierter Weiterleitung der Uhrzeit können Sie über das Kontrollkästchen "Korrigierte Uhrzeit verwenden" festlegen, ob ein im Uhrzeitletogrammm ggf. enthaltener Korrekturfaktor verwendet werden soll. Für den CP 1628 ist diese Option standardmäßig aktiviert und kann nicht deaktiviert werden.
- **NTP:**
  - Zeitzone: Im NTP-Verfahren wird generell UTC (Universal Time Coordinated) übertragen. Dies entspricht GMT (Greenwich Mean Time). Durch die Projektierung der lokalen Zeitzone kann der Zeitoffset gegenüber UTC eingestellt werden.
  - Aktualisierungsintervall in Sekunden: Definiert den Zeitabstand der Uhrzeitanfragen in Sekunden.

---

**Hinweis****Einstellung des Aktualisierungsintervalls für CPs**

Wenn das Kontrollkästchen "Aktiviere Security-Funktionen" in den lokalen Security-Einstellungen von CPs aktiviert wird, wird die Einstellung des Aktualisierungsintervalls aus den lokalen Einstellungen des CPs in die lokalen Security-Einstellungen des CPs übernommen.

---

- Uhrzeitsynchronisation zur vollen Minute (nur für S7-300-/S7-400-/PC-CPs): Mit dieser Option können Sie entscheiden, ob die Uhrzeit genau zur vollen Minute auf den K-Bus weitergeleitet werden soll. Diese Option ist nur für einige Spezialanwendungen erforderlich.
- Uhrzeit von nicht synchronisierten NTP-Servern annehmen (nur für CPs): Hier ist einstellbar, ob das Security-Modul die Uhrzeit auch von nicht synchronisierten NTP-Servern annehmen soll.
- Uhrzeit an Station weiterleiten (nur für S7-300-/S7-400-/PC-CPs): Deaktivieren Sie diese Option, wenn die CPU die Uhrzeit eigenständig von einem NTP-Server anfordert. Sie vermeiden damit, dass die in der CPU vom NTP-Server direkt erfasste Uhrzeit von der im CP erfassten Uhrzeit wieder überschrieben wird. Durch die Weiterleitung über den CP könnte sich eine geringere Genauigkeit ergeben.
- NTP-Server: Das Anlegen von NTP-Servern in den lokalen Security-Einstellungen wird in Kapitel NTP-Server definieren (Seite 558) beschrieben.

## Security-Modul einem NTP-Server (secure) zuweisen


### Modulspezifische Funktion

Das Zuweisen von Security-Modulen an global angelegte NTP-Server (secure) ist nur für CPs möglich.

### Voraussetzung

- Sie haben in den globalen Security-Einstellungen einen NTP-Server (secure) angelegt.
- In den lokalen Security-Einstellungen des Security-Moduls, das Sie einem NTP-Server (secure) zuweisen möchten, ist als Synchronisierungsmodus "NTP (secure)" ausgewählt.

### Vorgehensweise

1. Doppelklicken Sie in den globalen Security-Einstellungen auf den Eintrag "NTP".
2. Doppelklicken Sie auf den Eintrag "Modul einem NTP-Server zuweisen".
3. Wählen Sie aus der Klappliste "NTP Server" den NTP-Server (secure) aus, dem Sie ein Security-Modul zuweisen möchten.
4. Wählen Sie im Bereich "Verfügbare Module" das Security-Modul aus, das Sie dem ausgewählten NTP-Server (secure) zuweisen möchten.
5. Klicken Sie auf die Schaltfläche , um das ausgewählte Security-Modul dem ausgewählten NTP-Server (secure) zuzuweisen.

### Ergebnis

Sie haben das Security-Modul dem NTP-Server (secure) zugewiesen. Der NTP-Server (secure) erscheint in den lokalen Security-Einstellungen automatisch in der Liste der NTP-Server.

## Security-Modul als DHCP-Server

### Modulspezifische Funktion

Die Verwendung des Security-Moduls als DHCP-Server ist nur bei SCALANCE S Modulen möglich, siehe Abschnitt: Auto-Hotspot im Kapitel "SCALANCE S".



## SNMP konfigurieren

### Modulspezifische Funktion

Die Konfiguration von SNMP ist derzeit nur für CPs möglich, siehe folgende Kapitel:

- Für S7-300-/S7-400-/PC-CPs: Auto-Hotspot
- Für S7-1500-CPs: Auto-Hotspot

## Webserver auf Security-Modul aktivieren

### Modulspezifische Funktion

Diese Funktion ist nur für CP x43-1 Advanced verfügbar, siehe Abschnitt:  
Webserver auf CP x43-1 Advanced aktivieren (Seite 616) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".

## IPsec-Tunnel: VPN-Gruppen anlegen und zuordnen

### So legen Sie mittels VPN-Gruppen IPsec-Tunnel an

### Modulspezifische Funktion

Diese Funktion ist nicht für SCALANCE S602 sowie CP 1543-1 V1.0 verfügbar. CP 1543-1 unterstützt diese Funktion erst ab der Firmware V1.1.

## Voraussetzung

---

### Hinweis

#### **Aktuelles Datum und aktuelle Uhrzeit auf den Security-Modulen**

Achten Sie bei der Verwendung von gesicherter Kommunikation (z. B. HTTPS, VPN...) darauf, dass die betroffenen Security-Module über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die verwendeten Zertifikate werden sonst als nicht gültig ausgewertet und die gesicherte Kommunikation funktioniert nicht.

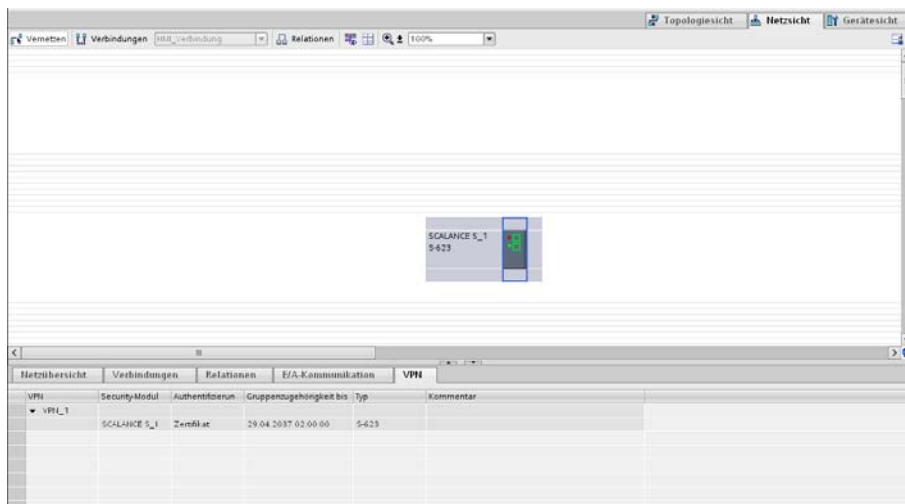
---

**So erreichen Sie diese Funktion**

1. Doppelklicken Sie in den globalen Security-Einstellungen auf den Eintrag "VPN-Gruppen" > "Neue VPN-Gruppe hinzufügen", um eine VPN-Gruppe anzulegen.
2. Doppelklicken Sie in den globalen Security-Einstellungen auf den Eintrag "VPN-Gruppen" > "Modul einer VPN-Gruppe zuweisen" und ordnen Sie der VPN-Gruppe die Security-Module und SOFTNET Security Client-Module zu, zwischen denen VPN-Tunnel aufgebaut werden sollen. Beachten Sie hierbei die Regeln zur Bildung von VPN-Gruppen. Sie finden diese Regeln im Abschnitt: Betriebsarten von VPN-Gruppen (Seite 568).

**Anzeige der VPN-Gruppen mit ihren Eigenschaften**

Wenn Sie ein Security-Modul auswählen, das sich in einer oder mehreren VPN-Gruppe(n) befindet, werden im Bereich "Netzwerkdaten" die Eigenschaften der VPN-Gruppe(n) angezeigt, in der/denen sich das Security-Modul befindet.



Folgende Eigenschaften der VPN-Gruppen werden im Register "VPN" des Bereichs "Netzwerkdaten" spaltenweise angezeigt:

| Eigenschaft/Spalte       | Bedeutung  |
|--------------------------|--|
| VPN                      | Namen der VPN-Gruppen, in denen sich das ausgewählte Security-Modul befindet             |
| Security-Modul           | Namen der zugeordneten Security-Module   |
| Authentifizierung        | Authentifizierungstyp: Preshared Key oder Zertifikat                                     |
| Gruppenzugehörigkeit bis | Datum und Uhrzeit, bis zu denen das VPN-Gruppenzertifikat des Security-Moduls gültig ist |
| Typ                      | Modellnummern der zugeordneten Security-Module   |
| Kommentar                | Kommentar  |

## Lebensdauer von Zertifikaten einstellen

Öffnen Sie den Dialog, in dem Sie das Ablaufdatum des Zertifikats eingeben können, wie folgt:

1. Markieren Sie im Register "VPN" die zu bearbeitende VPN-Gruppe.
2. Wählen Sie im Register "Eigenschaften" > "Allgemein" des Inspektorfensters den Eintrag "Authentifizierung".

---

### Hinweis

#### Ablauf eines Zertifikats

Die Kommunikation durch den VPN-Tunnel läuft nach Ablauf des Zertifikats weiter, bis der Tunnel abgebaut wird oder die SA-Lebensdauer abläuft. Weitere Informationen zu Zertifikaten finden Sie im Kapitel:  
Auto-Hotspot.

---

## Mengengerüst

| Anzahl der IPSec-Tunnel |             |
|-------------------------|-------------|
| SCALANCE S612 V2        | 64 maximal  |
| SCALANCE S612 V3        | 128 maximal |
| SCALANCE S613           | 128 maximal |
| SCALANCE S623           | 128 maximal |
| CP x43-1 Advanced       | 32 maximal  |
| CP 1628                 | 64 maximal  |
| CP 1543-1 V1.1          | 16 maximal  |

## Authentifizierungsmethoden

### Folgende Methoden stehen Ihnen zur Verfügung

Die Authentifizierungsmethode wird pro VPN-Gruppe festgelegt und bestimmt die Art der verwendeten Authentifizierung.

Es werden schlüsselbasierte oder zertifikatsbasierte Authentifizierungsmethoden unterstützt:

- **Preshared Key**  
Die Authentifizierung erfolgt über eine zuvor festgelegte Zeichenfolge, die an alle in der Gruppe befindlichen Module verteilt wird.  
Geben Sie dafür in den VPN-Gruppeneigenschaften unter "Authentifizierung" > "Allgemein" im Feld "Schlüssel" einen Preshared Key ein.
- **Zertifikat**  
Die zertifikatbasierte Authentifizierung "Zertifikat" ist die Standardeinstellung. Das Verhalten ist wie folgt:
  - Beim Anlegen einer Gruppe wird automatisch ein CA-Zertifikat als Stammzertifikat erzeugt.
  - Jedes Security-Modul, das Teilnehmer der VPN-Gruppe ist, erhält zusätzlich ein Gruppenzertifikat, das mit dem Schlüssel des CA-Zertifikats signiert ist.Sämtliche Zertifikate basieren auf dem ITU-Standard X.509v3 (ITU, International Telecommunications Union).  
Die Zertifikate werden von einer in STEP 7 enthaltenen Zertifizierungsstelle erzeugt.

---

**Hinweis**

**Einschränkung bei VLAN-Betrieb**

Bei der Übertragung von IP-Telegrammen durch den VPN-Tunnel des Security-Moduls wird kein VLAN-Tagging übertragen. Die in den IP-Telegrammen enthaltenen VLAN-Tags gehen beim Passieren der Security-Module verloren, da für die Übertragung der IP-Telegramme IPsec verwendet wird.

Standardmäßig können mit IPsec keine IP-Broadcast- bzw. IP-Multicast-Telegramme durch einen Layer 3 VPN-Tunnel übertragen werden. Durch einen Layer 2 VPN-Tunnel des Security-Moduls werden IP-Broadcast- bzw. IP-Multicast-Telegramme genau wie MAC-Pakete inklusive Ethernet-Header in UDP "verpackt" und übertragen. Daher bleibt bei diesen Paketen das VLAN-Tagging erhalten.

---

## Gruppeneigenschaften für gewählte VPN-Gruppe

### VPN-Gruppeneigenschaften

---

**Hinweis**

**IPsec-Kenntnisse erforderlich**

Um diese Parameter einstellen zu können, benötigen Sie IPsec-Kenntnisse. Wenn Sie keine Einstellungen vornehmen bzw. verändern, gelten die Standardeinstellungen.

---

In den Eigenschaften einer VPN-Gruppe sind folgende Einstellungen projektierbar:

- Authentifizierungsmethode (Eintrag: "Allgemein")
- IKE-Einstellungen (Eintrag: "Erweiterte Einstellungen Phase 1")
- IPsec-Einstellungen (Eintrag: "Erweiterte Einstellungen Phase 2")

### So erreichen Sie diese Funktion

1. Selektieren Sie im Register "VPN" des Bereichs "Netzwerkdaten" die zu bearbeitende VPN-Gruppe.
2. Wählen Sie im Register "Eigenschaften" > "Allgemein" des Inspektorfensters den Eintrag "Authentifizierung".
3. Wählen Sie aus, ob für die Authentifizierung ein Preshared Key oder ein Zertifikat verwendet werden soll. Weitere Informationen hierzu finden Sie im Kapitel: Authentifizierungsmethoden (Seite 565).

### Parameter für erweiterte Einstellungen Phase 1 - IKE-Einstellungen

Phase 1: Schlüsselaustausch (IKE = Internet Key Exchange):

Stellen Sie hier die Parameter für das Protokoll des IPsec-Schlüsselmanagements ein. Der Schlüsselaustausch erfolgt über das standardisierte Verfahren IKEv1, für das Sie folgende Protokollparameter einstellen können:

| Parameter                 | Beschreibung   |
|---------------------------|--|
| IKE-Modus                 | Schlüsselaustauschverfahren: <ul style="list-style-type: none"> <li>• Main Mode</li> <li>• Aggressive Mode</li> </ul> Der Unterschied zwischen Main- und Aggressive Mode ist die "Identity-Protection", die im Main-Mode verwendet wird. Die Identität wird im Main-Mode verschlüsselt übertragen, im Aggressive-Mode nicht.                   |
| DH-Gruppe Phase 1         | Diffie-Hellman-Schlüsselvereinbarung: <ul style="list-style-type: none"> <li>• Group 1</li> <li>• Group 2</li> <li>• Group 5</li> </ul> Diffie-Hellman-Gruppen (wählbare kryptographische Algorithmen im Oakley-Schlüsselaustausch-Protokoll).   |
| SA-Lebensdauertyp         | Phase 1 Security Association (SA): <ul style="list-style-type: none"> <li>• Time: Zeitbegrenzung in Minuten</li> </ul> Die Nutzdauer für das aktuelle Schlüsselmaterial wird zeitlich begrenzt. Nach Ablauf der Zeit wird das Schlüsselmaterial neu ausgehandelt.  |
| SA-Lebensdauer            | Numerischer Wert:<br>Wertebereich für Time: 1440 ... 2500000 Minuten (Standard: 2500000)   |
| Verschlüsselung Phase 1   | Verschlüsselungs-Algorithmus: <ul style="list-style-type: none"> <li>• DES*: Data Encryption Standard (56 Bit Schlüssellänge, Modus CBC)</li> <li>• 3DES-168: Dreifach-DES (168 Bit Schlüssellänge, Modus CBC)</li> <li>• AES-128, 192, 256: Advanced Encryption Standard (128 Bit, 192 Bit oder 256 Bit Schlüssellänge, Modus CBC)</li> </ul> |
| Authentifizierung Phase 1 | Authentifizierungsalgorithmus: <ul style="list-style-type: none"> <li>• MD5: Message Digest Algorithm 5</li> <li>• SHA1: Secure Hash Algorithm 1</li> </ul>  |

\*DES ist ein unsicherer Verschlüsselungsalgorithmus. Er sollte nur aus Gründen der Abwärtskompatibilität verwendet werden. DES wird vom CP 1543-1 V1.1 nicht unterstützt.

### Parameter für erweiterte Einstellungen Phase 2 - IPsec-Einstellungen

Phase 2: Datenaustausch (ESP = Encapsulating Security Payload)

Stellen Sie hier die Parameter für das Protokoll des IPsec-Datenaustauschs ein. Der Datenaustausch erfolgt im "Quick Mode". Die gesamte Kommunikation in dieser Phase erfolgt verschlüsselt über das standardisierte Sicherheitsprotokoll ESP, für das Sie folgende Protokollparameter einstellen können:

| Parameter                 | Beschreibung   |
|---------------------------|--|
| SA-Lebensdauertyp         | Phase 2 Security Association (SA): <ul style="list-style-type: none"> <li>• Time: Zeitbegrenzung in Minuten. Die Nutzdauer für das aktuelle Schlüsselmaterial wird zeitlich begrenzt. Nach Ablauf der Zeit wird das Schlüsselmaterial neu ausgehandelt.</li> <li>• Limit: Begrenzung des Datenvolumens in MByte</li> </ul> |
| SA-Lebensdauer            | Numerischer Wert: <ul style="list-style-type: none"> <li>• Wertebereich für Time: 60 ... 16666666 Minuten (Standard: 2880)</li> <li>• Wertebereich für Limit: 2000 ... 500000 Mbyte (Standard: 4000)</li> </ul>  |
| Verschlüsselung Phase 2   | Verschlüsselungs-Algorithmus: <ul style="list-style-type: none"> <li>• DES*: Data Encryption Standard (56 Bit Schlüssellänge, Modus CBC)</li> <li>• 3DES-168: Dreifach-DES (168 Bit Schlüssellänge, Modus CBC)</li> <li>• AES-128: Advanced Encryption Standard (128 Bit Schlüssellänge, Modus CBC)</li> </ul>             |
| Authentifizierung Phase 2 | Authentifizierungs-Algorithmus: <ul style="list-style-type: none"> <li>• MD5: Message Digest Algorithm 5</li> <li>• SHA1: Secure Hash Algorithm 1</li> </ul>   |
| Perfect Forward Secrecy   | Wählen Sie aus, ob vor jedem neuen Aushandeln einer IPsec-SA ein erneutes Aushandeln der Schlüssel mit Hilfe des Diffie-Hellman-Verfahrens erfolgen soll. Durch die Perfect Forward Secrecy wird sichergestellt, dass von den zuvor generierten Schlüsseln nicht auf die neuen Schlüssel geschlossen werden kann.          |

\*DES ist ein unsicherer Verschlüsselungsalgorithmus. Er sollte nur aus Gründen der Abwärtskompatibilität verwendet werden. DES wird vom CP 1543-1 V1.1 nicht unterstützt.

### Betriebsarten von VPN-Gruppen

#### VPN-Betriebsarten

Je nachdem, in welcher Betriebsart sich die Security-Module befinden, die einer VPN-Gruppe hinzugefügt wurden, wird zwischen unterschiedlichen Betriebsarten von VPN-Gruppen unterschieden. Die Betriebsart einer VPN-Gruppe gibt Aufschluss darüber, welche Security-Module in welchen Betriebsarten der VPN-Gruppe hinzugefügt werden können.

## Regeln für die Gruppenbildung

Beachten Sie die folgenden Regeln, wenn Sie VPN-Gruppen bilden wollen:

- Für SCALANCE S612/S613/S623  
Das erste in einer VPN-Gruppe zugeordnete Modul bestimmt, welche zusätzlichen Module hinzugefügt werden können.  
Ist das erste hinzugefügte SCALANCE S-Modul im Routing-Modus, so können zusätzlich nur SCALANCE S-Module mit aktiviertem Routing-Modus hinzugefügt werden. Ist das erste hinzugefügte SCALANCE S-Modul im Bridge-Modus, so können zusätzlich nur SCALANCE S-Module im Bridge-Modus hinzugefügt werden. Soll der Modus einer VPN-Gruppe geändert werden, so müssen alle in der Gruppe enthaltenen Module entfernt und neu hinzugefügt werden. Ein CP kann in eine Gruppe mit einem SCALANCE S im Bridge- oder Routing-Modus hinzugefügt werden.
- Für CPs  
Wird ein CP als erstes Modul in eine VPN-Gruppe eingefügt, bestimmt das nächste die Betriebsart festlegende Security-Modul die Betriebsart der VPN-Gruppe. Ein CP kann gleichzeitig mehreren VPN-Gruppen mit unterschiedlicher Betriebsart zugeordnet werden. Der CP wird dann im Mischbetrieb betrieben.
- Es ist nicht möglich, ein SCALANCE M-Modul einer VPN-Gruppe hinzuzufügen, die ein Modul im Bridge-Modus enthält.

Entnehmen Sie der folgenden Tabelle, welche Module in einer VPN-Gruppe zusammengefasst werden können:

Tabelle 8-23 Security-Module und VPN-Betriebsarten

| Modul                                    | Kann aufgenommen werden in VPN-Gruppe im... |               |
|--|---|---------------|
|  | Bridge-Modus                                | Routing-Modus |
| SCALANCE S612/S613/S623 im Bridge-Modus  | x   | -             |
| SCALANCE S612/S613/S623 im Routing-Modus | -   | x             |
| CP x43-1 Adv.                            | x   | x             |
| CP 1543-1 V1.1                           | x   | x             |
| CP 1628                                  | x   | x             |
| SOFTNET Security Client V4.0             | x   | x             |
| SCALANCE M875/MD741-1                    | -   | x             |

## Security-Modul in konfigurierte VPN-Gruppe aufnehmen

Die projektierten VPN-Gruppeneigenschaften werden für Security-Module, die in eine bestehende VPN-Gruppe aufgenommen werden, übernommen.

### Vorgehen nach Aufnahme eines Security-Moduls in eine konfigurierte VPN-Gruppe

Je nachdem, ob Sie die VPN-Gruppeneigenschaften seit dem letzten Ladevorgang geändert haben oder nicht, müssen Sie beim Vorgehen unterscheiden:

- **Fall a:** Wenn Sie die VPN-Gruppeneigenschaften nicht geändert haben und das hinzuzufügende Modul die Verbindung zu den bereits konfigurierten Modulen aktiv aufbaut:
  1. Fügen Sie das neue Security-Modul der VPN-Gruppe hinzu.
  2. Laden Sie die Konfiguration in das neue Modul.
- **Fall b:** Wenn Sie die VPN-Gruppeneigenschaften geändert haben oder das hinzuzufügende Modul die Verbindung zu den bereits konfigurierten Modulen nicht aktiv aufbaut:
  1. Fügen Sie das neue Security-Modul der VPN-Gruppe hinzu.
  2. Laden Sie die Konfiguration in alle Module, die zur VPN-Gruppe gehören.

Bereits vorhandene, in Betrieb genommene Security-Module müssen im Fall a nicht neu projiziert und geladen werden. Die laufende Kommunikation wird nicht beeinflusst oder unterbrochen.

### Einstellungen für Teilnehmer mit unbekannter IP-Adresse

Teilnehmer, bei denen zum Projektierungszeitpunkt die IP-Adresse unbekannt ist (Unknown Peers), können in eine bestehende VPN-Gruppe eingefügt werden. Da sich die Teilnehmer meist im mobilen Einsatz befinden und die IP-Adresse dynamisch beziehen (z. B. ein SOFTNET Security Client oder SCALANCE M), kann der VPN-Tunnel nur aufgebaut werden, wenn Sie die Parametereinstellungen für Phase 1 entsprechend einer der folgenden Tabellen (1-23 bis 1-27) vornehmen. Verwenden Sie andere Einstellungen, können Sie keinen VPN-Tunnel zum Endgerät aufbauen.

Tabelle 8-24 Verschlüsselungsparameter 1

| Parameter                 | Einstellung              |
|---------------------------|--------------------------|
| Authentifizierungsmethode | Zertifikat               |
| DH-Gruppe Phase 1         | Gruppe 2                 |
| SA-Lebensdauer            | 1440 ... 2500000 Minuten |
| Verschlüsselung Phase 1   | AES-256                  |
| Authentifizierung Phase 1 | SHA-1                    |

Tabelle 8-25 Verschlüsselungsparameter 2

| Parameter                 | Einstellung              |
|---------------------------|--------------------------|
| Authentifizierungsmethode | Zertifikat               |
| DH-Gruppe Phase 1         | Gruppe 2                 |
| SA-Lebensdauer            | 1440 ... 2500000 Minuten |
| Verschlüsselung Phase 1   | 3DES-168                 |
| Authentifizierung Phase 1 | SHA-1                    |



Tabelle 8-26 Verschlüsselungsparameter 3

| Parameter                 | Einstellung              |
|---------------------------|--------------------------|
| Authentifizierungsmethode | Zertifikat               |
| DH-Gruppe Phase 1         | Gruppe 2                 |
| SA-Lebensdauer            | 1440 ... 2500000 Minuten |
| Verschlüsselung Phase 1   | DES                      |
| Authentifizierung Phase 1 | MD5                      |

Tabelle 8-27 Verschlüsselungsparameter 4

| Parameter                 | Einstellung              |
|---------------------------|--------------------------|
| Authentifizierungsmethode | Preshared Key            |
| DH-Gruppe Phase 1         | Gruppe 2                 |
| SA-Lebensdauer            | 1440 ... 2500000 Minuten |
| Verschlüsselung Phase 1   | 3DES-168                 |
| Authentifizierung Phase 1 | SHA1                     |

### Zusätzliche Einschränkungen für den SOFTNET Security Client

Für den SOFTNET Security Client gelten zusätzlich die folgenden Einschränkungen:

Tabelle 8-28 Verschlüsselungsparameter für SOFTNET Security Client

| Parameter                 | Einstellung / Besonderheit                     |
|---------------------------|--|
| Verschlüsselung Phase 1   | AES-256 nur bei Windows 7 möglich              |
| SA-Lebensdauer Phase 1    | 1440 ... 2879 Minuten                          |
| SA-Lebensdauertyp         | Muss für beide Phasen identisch gewählt werden |
| Verschlüsselung Phase 2   | Kein AES-128 möglich                           |
| SA-Lebensdauer Phase 2    | 1440 ... 2879 Minuten                          |
| Authentifizierung Phase 2 | Kein MD5 möglich                               |

### Vorgehen nach Entnahme eines aktiven Teilnehmers aus einer VPN-Gruppe

Wenn Sie einen aktiven Teilnehmer aus einer bestehenden VPN-Gruppe entfernen, kann dieser immer noch eine Verbindung zu den Gruppenteilnehmern aufbauen, auch wenn Sie das Projekt erneut auf alle Teilnehmer der VPN-Gruppe geladen haben.

Soll der entfernte aktive Teilnehmer keine Verbindung mehr aufbauen können, erneuern Sie das CA-Gruppenzertifikat und laden Sie das Projekt erneut auf die Teilnehmer der VPN-Gruppe. Erneuert werden kann das Zertifikat in den Gruppeneigenschaften der VPN-Gruppe oder im Zertifikatsmanager, Register "CA".

## Interne Netzknoten konfigurieren

### Übersicht zur Konfiguration interner Netzknoten

#### Interne Netzknoten konfigurieren

Jedem Security-Modul müssen die Netzknoten im gesamten internen Netz bekannt sein, um die Authentizität eines Telegramms feststellen zu können.

Das Security-Modul muss sowohl seine eigenen internen Knoten kennen als auch die internen Knoten der Security-Module, mit denen es sich zusammen in einer VPN-Gruppe befindet. Diese Information wird auf einem Security-Modul dazu verwendet, um zu bestimmen, welches Datenpaket in welchem Tunnel übertragen werden soll.

SCALANCE S Module bieten die Möglichkeit, die Netzknoten automatisch zu erlernen oder statisch zu konfigurieren. Über die Möglichkeiten, die zum Lernen interner Netzknoten zur Verfügung stehen, entscheidet zudem die Betriebsart des Security-Moduls.

#### SCALANCE S im Bridge-Modus

Im Bridge-Modus können Sie die internen IP-/MAC-Knoten sowie die internen Subnetze konfigurieren oder alternativ das automatische Lernen interner Knoten durch den SCALANCE S zulassen.

#### SCALANCE S im Routing-Modus

Im Routing-Modus steht kein automatisches Lernen zur Verfügung. Stattdessen tragen Sie hier komplette Subnetze ein, die für die Tunnelkommunikation freigegeben werden sollen.

#### CP x43-1 Advanced und CP 1628

- CP x43-1 Adv.  
Wählen Sie aus, ob die Tunnelkommunikation zum CP (GBit-Schnittstelle) und/oder zum internen Subnetz (PROFINET-Subnetz) für VPN-Verbindungspartner im Routing-Modus (SCALANCE S / M) erlaubt ist.
- CP 1628  
Tragen Sie die NDIS-Knoten ein, die durch den Tunnel von VPN-Verbindungspartnern im Routing-Modus (SCALANCE S / M) erreichbar sein sollen.

#### Automatisches Lernen interner Netzknoten

#### Modulspezifische Funktion

SCALANCE S-Module im Bridge-Modus bieten einen Lernmodus, mit dem interne Netzknoten im Betrieb automatisch gelernt werden können. Weitere Informationen hierzu finden Sie im Abschnitt:

Lernmodus zum Lernen interner Knoten verwenden (Seite 599) im Kapitel "SCALANCE S".

## **IP-Netzknoten für SCALANCE S manuell konfigurieren**

### **Modulspezifische Funktion**

Wie Sie IP-Netzknoten für SCALANCE S Module manuell konfigurieren, siehe Abschnitt: IP-Netzknoten manuell konfigurieren (Seite 601) im Kapitel "SCALANCE S".

## **MAC-Netzknoten für SCALANCE S manuell konfigurieren**

### **Modulspezifische Funktion**

Wie Sie MAC-Netzknoten für SCALANCE S Module manuell konfigurieren, siehe Abschnitt: MAC-Netzknoten manuell konfigurieren (Seite 601) im Kapitel "SCALANCE S".

## **Interne Subnetze für SCALANCE S manuell konfigurieren**

### **Modulspezifische Funktion**

Wie Sie interne Subnetze für SCALANCE S Module konfigurieren, siehe Abschnitt: Interne Subnetze manuell konfigurieren (Seite 602) im Kapitel "SCALANCE S".

## **Zugriff auf S7-300-/S7-400-CPs für VPN-Verbindungspartner erlauben**

### **Modulspezifische Funktion**

Wie Sie den Zugriff auf S7-300-/S7-400-CPs für VPN-Verbindungspartner erlauben, siehe Abschnitt:  
Zugriff auf S7-300-/S7-400-CPs für VPN-Verbindungspartner erlauben (Seite 617) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".

## **Durch Tunnel erreichbare NDIS Knoten für PC-CPs konfigurieren**

### **Modulspezifische Funktion**

Wie Sie durch Tunnel erreichbare NDIS Knoten bei PC-CPs konfigurieren, siehe Abschnitt: Durch Tunnel erreichbare NDIS-Knoten bei PC-CPs manuell konfigurieren (Seite 617) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".

## **Modul- und verbindungsspezifische VPN-Einstellungen projektieren**

### **Voraussetzung**

Das Modul ist Teilnehmer einer VPN-Gruppe.

### Modul- und verbindungspezifische Einstellungen

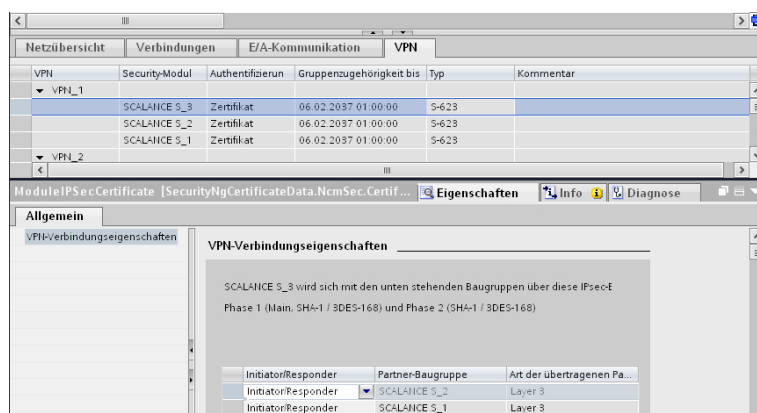
Mit modul- und verbindungspezifischen Einstellungen lassen sich spezifische VPN-Einstellungen projektieren. Modulspezifische Einstellungen werden speziell für ein Security-Modul projektiert, während verbindungspezifische Einstellungen speziell für ein Security-Modul in einer bestimmten VPN-Gruppe projektiert werden.

Folgende **modulspezifische** Eigenschaften können Sie in den lokalen Security-Einstellungen unter dem Eintrag "VPN" konfigurieren:

- Dead-Peer-Detection
- Erlaubnis zur Initiierung des Verbindungsaufbaus
- Öffentliche IP-Adresse zur Kommunikation über Internet-Gateways

Wenn Sie in der Auflistung der VPN-Gruppen im Bereich "Netzwerkdaten" ein Security-Modul auswählen, sind folgende **verbindungspezifische** VPN-Einstellungen einsehbar bzw. projektierbar:

- Erlaubnis zur Initiierung des Verbindungsaufbaus
- Partner-Baugruppen, zu denen Tunnelverbindungen bestehen
- Art der übertragenen Pakete
- Auswahl der lokalen Schnittstelle des ausgewählten Security-Moduls, die als Tunnelendpunkt fungieren soll
- Auswahl der Partner-Schnittstelle, die als Tunnelendpunkt fungieren soll



### Dead-Peer-Detection (DPD)

Standardmäßig ist DPD aktiviert.

Bei aktivierter DPD tauschen die Module in einstellbaren Zeitintervallen zusätzliche Nachrichten aus, sofern zum jeweiligen Zeitpunkt keine Kommunikation stattfindet. Hierdurch kann erkannt werden, ob die IPsec-Verbindung noch gültig ist oder eventuell neu aufgebaut werden muss. Besteht keine Verbindung mehr, werden die "Security Associations" (SA) von Phase 2 vorzeitig beendet. Bei deaktivierter DPD wird die SA erst nach Ablauf der SA-Lebensdauer beendet. Zur Einstellung der SA-Lebensdauer siehe Abschnitt: Gruppeneigenschaften für gewählte VPN-Gruppe (Seite 566).

## Erlaubnis zur Initiierung des Verbindungsaufbaus

Sie können die Erlaubnis zur Initiierung des VPN-Verbindungsaufbaus auf bestimmte Module im VPN beschränken.

Maßgebend für die Einstellung des beschriebenen Parameters ist die Vergabe der IP-Adresse für das Gateway des zu projektierenden Moduls. Bei einer statisch vergebenen IP-Adresse kann das Modul von der Gegenstelle gefunden werden. Bei dynamisch verbogebener, und sich daher ständig ändernder IP-Adresse, kann die Gegenstelle nicht ohne Weiteres eine Verbindung aufbauen.

| Modus  | Bedeutung   |
|--|---|
| Starte Verbindung zur Gegenstelle (Initiator/Responder) (Standard) | <p>Bei dieser Option ist das Modul "aktiv", d. h. es versucht, eine Verbindung zur Gegenstelle mit fester IP-Adresse herzustellen.</p> <p>Diese Option wird empfohlen, wenn Sie von dem Provider für das Gateway des hier zu projektierenden Security-Moduls eine dynamische IP-Adresse zugewiesen bekommen haben.</p> <p>Die Adressierung der Gegenstelle erfolgt über deren externe Modul IP-Adresse oder über deren projektierte WAN-IP-Adresse.</p>   |
| Warte auf Gegenstelle (Responder)                                  | <p>Bei dieser Option ist das Modul "passiv", d. h. es wartet, bis der Verbindungsaufbau von der Gegenstelle initiiert wird.</p> <p>Diese Option wird empfohlen, wenn Sie von dem Provider für das Gateway des hier zu projektierenden Security-Moduls eine statische IP-Adresse zugewiesen bekommen haben. Sie erreichen dadurch, dass Verbindungsaufbauversuche nur durch die Gegenstelle gestartet werden. Diese Gegenstelle kann beispielsweise eine dynamische WAN-IP-Adresse besitzen.</p> |

### Hinweis

Stellen Sie nicht alle Module einer VPN-Gruppe auf "Warte auf Gegenstelle", da sonst keine Verbindung aufgebaut wird.

## WAN-IP-Adresse - IP-Adressen der Module und Gateways bei einem VPN über Internet

Beim Betrieb eines VPN mit IPsec-Tunnel über das Internet sind in der Regel zusätzliche IP-Adressen für die Internet-Gateways, z. B. DSL-Router, erforderlich. Den einzelnen Security-

oder SCALANCE M-Modulen müssen die externen IP-Adressen der Partner-Module im VPN bekannt sein.

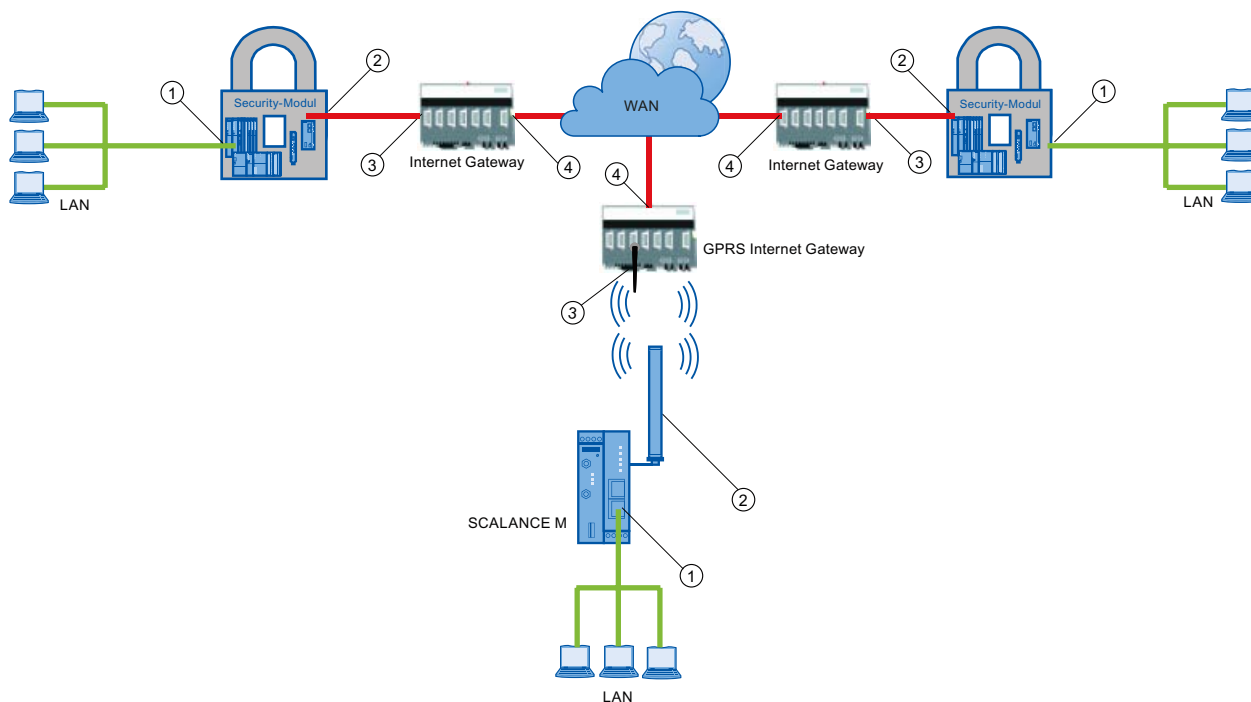
**Hinweis**

Für die Nutzung eines WAN als externes, öffentliches Netzwerk geben Sie als externe IP-Adresse Ihre vom Provider erhaltene IP-Adresse an, über welche das Security-Modul dann im WAN (Internet) erreichbar ist. Damit das Security-Modul Pakete über das WAN versenden kann, müssen Sie als "Standard-Router" Ihren DSL-Router eintragen.

Wenn Sie einen DSL-Router als Internet-Gateway nutzen, müssen an diesem mindestens die folgenden Ports freigeschaltet werden:

- Port 500 (ISAKMP)
- Port 4500 (NAT-T)

Hierzu besteht die Möglichkeit, in der Konfiguration des Moduls eine externe IP-Adresse als "WAN-IP-Adresse" zuzuordnen. Beim Laden der Modulkonfiguration werden den Gruppenteilnehmern dann die WAN-IP-Adressen der Partner-Module mitgeteilt. Tragen Sie keine WAN-IP-Adresse ein, wird die externe IP-Adresse des Moduls verwendet. Ob die externe IP-Adresse oder die WAN-IP-Adresse verwendet werden soll, können Sie in den modulspezifischen VPN-Einstellungen des Security-Moduls festlegen. Hier können Sie auch einstellen, über welche Schnittstelle die Teilnehmer einer VPN-Gruppe kommunizieren sollen und welches Security-Modul die Erlaubnis zum Verbindungsaufbau hat.



- ① IP-Adresse intern - eines Moduls
- ② IP-Adresse extern - eines Moduls
- ③ IP-Adresse eines Internet Gateways (z. B. GPRS-Gateway)
- ④ IP-Adresse (WAN-IP-Adresse) eines Internet Gateways (z. B. DSL-Router)

## Siehe auch

So legen Sie mittels VPN-Gruppen IPsec-Tunnel an (Seite 563)

## Online Funktionen - Test / Diagnose und Logging

### Diagnose und Logging in der Übersicht

Zu Test- und Überwachungszwecken verfügt das Security-Modul über Diagnose- und Logging-Funktionen.

- **Diagnosefunktionen**  
Hierunter sind verschiedene System- und Statusfunktionen zu verstehen, die Sie bei einer bestehenden Netzwerkverbindung zum Security-Modul anwenden können.
- **Logging-Funktionen**  
Hierbei geht es um die Aufzeichnung von System- und Sicherheitsereignissen sowie von Datenpaketen.

### Ereignisse mit Logging-Funktionen aufzeichnen

Welche Ereignisse aufgezeichnet werden sollen, legen Sie mit den Log-Einstellungen zum jeweiligen Security-Modul fest.

Dabei können Sie für die Aufzeichnung folgende Varianten konfigurieren:

- **Lokales Logging**  
Bei dieser Variante zeichnen Sie die Ereignisse in lokalen Puffern des Security-Moduls auf. Im "Online & Diagnose"-Dialog können Sie dann auf diese Aufzeichnungen zugreifen, diese sichtbar machen und in der Service-Station archivieren. Die Auswertung der Pufferbereiche des Security-Moduls setzt eine Netzwerkverbindung zum ausgewählten Security-Modul voraus.
- **Netzwerk-Syslog**  
Beim Netzwerk-Syslog nutzen Sie einen im Netz vorhandenen Syslog-Server. Dieser zeichnet die Ereignisse entsprechend der Konfiguration in den Log-Einstellungen des jeweiligen Security-Moduls auf.

### Log-Daten archivieren und aus Datei einlesen

Sie können die aufgezeichneten Ereignisse zur Archivierung in einer Log-Datei speichern und diese später auch ohne eine bestehende Netzwerkverbindung zum Security-Modul öffnen.

Weitere Informationen finden Sie im Kapitel:

Auto-Hotspot.

## Funktionsübersicht Online-Dialog

### Funktionen des "Online & Diagnose"-Dialogs

Das Security-Modul bietet folgende Funktionen im "Online & Diagnose"-Dialog von STEP 7:

| Eintrag im Online-Dialog   | Funktion   |
|--|--|
| Zustand  | Anzeige von Statusinformationen zum ausgewählten Security-Modul wie z.B. die aktuellen IP-Adressen der Schnittstellen sowie die laufende Uhrzeit und das Datum.  |
| System-Log   | Anzeige von geloggt System-Ereignissen, Starten und Stoppen des Loggings (nur bei einer bestehenden Online-Verbindung zu SCALANCE S-Modulen) sowie Starten und Stoppen des Auslesens von Log-Daten aus dem lokalen Puffer des Security-Moduls. |
| Audit-Log  | Anzeige von geloggt Sicherheits-Ereignissen, Starten und Stoppen des Auslesens von Log-Daten aus dem lokalen Puffer des Security-Moduls.   |
| Paketfilter-Log  | Anzeige von geloggt Daten-Paketen, Starten und Stoppen des Loggings (nur bei einer bestehenden Online-Verbindung zu SCALANCE S-Modulen) sowie Starten und Stoppen des Auslesens von Log-Daten aus dem lokalen Puffer des Security-Moduls.      |
| Kommunikationszustand (nicht für SCALANCE S602 und CP 1543-1 V1.0) | Anzeige von Statusinformationen zu VPN-Tunnelverbindungen sowie Teilnehmern von VPN-Gruppen, zu denen das ausgewählte Security-Modul gehört.   |
| Interne Knoten (nicht für SCALANCE S602 und nicht für PC-CPs)      | Anzeige der gelernten oder konfigurierten internen Netzknoten des Security-Moduls.   |
| Dynamisch aktualisierte Firewall-Regeln (Nur für S7-CPs)           | Anzeige der IP-Adressen, die über HTTP oder HTTPS dynamisch freigeschaltet oder von einem Benutzer nachgeladen wurden.   |
| Datum und Uhrzeit (nur SCALANCE S)                                 | Einstellung von Datum und Uhrzeit.   |

### Zugriffsvoraussetzungen

Damit Sie die Online-Funktionen eines Security-Moduls nutzen können, müssen folgende Voraussetzungen erfüllt sein:

- eine Netzwerkverbindung zum ausgewählten Modul besteht
- das Projekt, mit dem das Modul konfiguriert wurde, ist geöffnet
- ein Nutzer mit den nötigen Rechten muss im Projekt angemeldet sein
- eine Online-Verbindung zum Security-Modul über HTTPS besteht

### So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf das zu bearbeitende Modul.
2. Wählen Sie im Kontextmenü den Menübefehl "Online & Diagnose".  
Sobald Sie einen der Einträge für Log-Funktionen selektieren, sehen Sie im unteren Bereich den aktuellen Zustand der Aufzeichnungsfunktion des gewählten Security-Moduls: Speichereinstellungen: Umlaufspeicher / Linearer Speicher  
Der aktuelle Aufzeichnungszustand kommt aus der geladenen Konfiguration oder aus der zuvor schon einmal durchgeführten Online-Diagnose.



## Online-Einstellungen werden nicht in der Konfiguration gespeichert

Einstellungen, die Sie in der Online-Betriebsart vornehmen (z. B. Einstellungen zum Log-Speicher), werden nicht in der Konfiguration auf dem Security-Modul gespeichert. Nach einem Modul-Neuanlauf sind deshalb immer die Einstellungen aus der Projektierung wirksam.

## Funktionen der Online-Diagnose

### Statusinformationen des Security-Moduls - Eintrag "Zustand"

#### Bedeutung

Anzeige des Status des im Projekt angewählten Security-Moduls.

Tabelle 8-29 Online & Diagnose: Eintrag "Zustand"

| System- und Statusfunktionen | Bedeutung  |
|------------------------------|--|
| <b>Übersicht</b>             |  |
| Hardware-Typ                 | Der Typ des Security-Moduls.   |
| Externe IP-Adresse           | Die externe IP-Adresse des Security-Moduls.<br>Für CP 1628: Die IP-Adresse der Industrial Ethernet-Schnittstelle.<br>Für S7-CPs: Die IP-Adresse der GBit-Schnittstelle.  |
| Interne IP-Adresse           | Die interne IP-Adresse des Security-Moduls.<br>Für CP 1628: Die IP-Adresse der NDIS-Schnittstelle. Bei mehreren NDIS-Adressen wird nur eine angezeigt.<br>Für S7-CPs: Die IP-Adresse der PROFINET-Schnittstelle. |
| Seriennummer                 | Die Seriennummer des Security-Moduls.  |
| Bestellnummer                | Die MLFB-Kennzeichnung des Security-Moduls, die bei einer Bestellung verwendet wird.   |
| Firmware-Version             | Der Firmware-Ausgabestand des Security-Moduls.   |
| Betriebsart                  | Einstellung am Schnittstellenrouting   |
| Externe MAC-Adresse          | Die externe MAC-Adresse des Security-Moduls.<br>Für CP 1628: Die MAC-Adresse der Industrial Ethernet-Schnittstelle.<br>Für S7-CPs: Die MAC-Adresse der GBit-Schnittstelle.                                       |
| Interne MAC-Adresse          | Die interne MAC-Adresse des Security-Moduls.<br>Für CP 1628: Die MAC-Adresse der NDIS-Schnittstelle.<br>Für S7-CPs: Die MAC-Adresse der PROFINET-Schnittstelle.  |
| HW-Release                   | Der Hardware-Erzeugnisstand des Security-Moduls.   |

| System- und Statusfunktionen | Bedeutung  |
|------------------------------|--|
| C-PLUG                       | Anzeige, ob ein C-PLUG gesteckt ist oder nicht.  |
| <b>Lokale Zeit</b>           |  |
| Aktuelle Zeit                | <p>Datum und Uhrzeit, die auf dem Security-Modul angezeigt werden.</p> <p>Format bei Oberflächensprache "Deutsch":<br/>dd.mm.yyyy (Datum) hh:mm:ss (Uhrzeit)</p> <p>Format bei Oberflächensprache "Englisch":<br/>mm/dd/yyyy (Datum) hh:mm:ss AM/PM (Uhrzeit)</p> <p>Format bei Oberflächensprache "Französisch":<br/>dd/mm/yyyy (Datum) hh:mm:ss (Uhrzeit)</p> <p>Format bei Oberflächensprache "Italienisch":<br/>dd/mm/yyyy (Datum) hh.mm.ss (Uhrzeit)</p> <p>Format bei Oberflächensprache "Spanisch":<br/>dd/mm/yyyy (Datum) hh:mm:ss (Uhrzeit)</p> <p><b>Hinweis (nicht für CPs)</b><br/>Die lokale Zeit auf dem SCALANCE S Modul stellen Sie unter "Funktionen" &gt; "Datum und Uhrzeit" ein.</p> |
| Betriebsdauer                | <p>Zeit, die seit dem letzten Neustart des Security-Moduls vergangen ist.</p> <p>Format bei Oberflächensprache "Deutsch":<br/>dddd.hh:mm:ss</p> <p>Format bei Oberflächensprache "Englisch":<br/>dddd.hh:mm:ss</p> <p>Format bei Oberflächensprache "Französisch":<br/>dddd.hh:mm:ss</p> <p>Format bei Oberflächensprache "Italienisch":<br/>dddd.hh.mm.ss</p> <p>Format bei Oberflächensprache "Spanisch":<br/>dddd.hh:mm:ss</p>  |
| Uhrzeit Quelle               | Die Quelle, von der Datum und Uhrzeit bezogen werden.  |
| <b>Konfiguration</b>         |  |
| Erzeugt                      | <p>Datum und Uhrzeit, zu denen das Projekt zum ersten Mal erstellt wurde.</p> <p>Format bei Oberflächensprache "Deutsch":<br/>dddd.hh:mm:ss hh:mm:ss (Uhrzeit)</p> <p>Format bei Oberflächensprache "Englisch":<br/>dddd.hh:mm:ss hh:mm:ss AM/PM (Uhrzeit)</p> <p>Format bei Oberflächensprache "Französisch":<br/>dddd.hh:mm:ss hh:mm:ss (Uhrzeit)</p> <p>Format bei Oberflächensprache "Italienisch":<br/>dddd.hh.mm.ss hh.mm.ss (Uhrzeit)</p> <p>Format bei Oberflächensprache "Spanisch":<br/>dddd.hh:mm:ss hh:mm:ss (Uhrzeit)</p>   |
| Name                         | Dateiname, unter dem das Projekt zuletzt gespeichert wurde. Der Name wird von STEP 7 voreingestellt und wird dann verändert, wenn sich der Name des Projekts ändert.   |

| System- und Statusfunktionen       | Bedeutung   |
|------------------------------------|---|
| Autor                              | Name des Benutzers, der das Projekt erstellt hat. Wird aus den Projekteigenschaften übernommen.   |
| Geladen                            | Datum und Uhrzeit, zu denen das Projekt zum letzten Mal auf das Security-Modul geladen wurde.<br>Format bei Oberflächensprache "Deutsch":<br>dddd.hh:mm:ss hh:mm:ss (Uhrzeit)<br>Format bei Oberflächensprache "Englisch":<br>dddd.hh:mm:ss hh:mm:ss AM/PM (Uhrzeit)<br>Format bei Oberflächensprache "Französisch":<br>dddd.hh:mm:ss hh:mm:ss (Uhrzeit)<br>Format bei Oberflächensprache "Italienisch":<br>dddd.hh.mm.ss hh.mm.ss (Uhrzeit)<br>Format bei Oberflächensprache "Spanisch":<br>dddd.hh:mm:ss hh:mm:ss (Uhrzeit) |
| Speicherort                        | Eingetragener Ort aus den Projekteigenschaften. Der Ort wird beim Laden der Konfiguration auf das Security-Modul übertragen.  |
| <b>Dateisystem (nicht für CPs)</b> |   |
| RAM                                | Graphische und prozentuale Anzeige, wie viel RAM und Flash im Dateisystem belegt sind.  |
| Flash                              |   |

## VPN-Verbindungen des Security-Moduls - Eintrag "Kommunikationszustand"

### Modulspezifische Funktion

Diese Funktion ist nicht für SCALANCE S602 Module verfügbar.

### Bedeutung

Anzeige des Kommunikationsstatus der folgenden Netzkomponenten:

- Andere Security-Module der VPN-Gruppe, zu der das ausgewählte Security-Modul gehört
- Interne Netzknoten dieser Security-Module

Tabelle 8-30 Online & Diagnose: Eintrag "Kommunikationszustand"

| System- und Statusfunktionen          | Bedeutung  |
|---------------------------------------|--|
| Bekannte Security-Geräte oder -Module | Anzeige der Teilnehmer, mit denen sich das ausgewählte Security-Modul in einer VPN-Gruppe befindet. Außerdem wird angezeigt, ob der Tunnelzustand aktiv ist oder nicht. Damit Sie zusätzliche Informationen zu einem der Teilnehmer erhalten, markieren Sie diesen in der Liste.<br><b>Hinweis:</b><br>Projektierte, aber nicht aktive Tunnel werden ausschließlich für CPs angezeigt.                               |
| Endpunkte                             | Anzeige von Informationen zu den internen Netzknoten des Security-Moduls, das Sie in der Tabelle "Bekannte Security-Geräte oder -Module" ausgewählt haben. Für jeden internen Netzknoten wird angezeigt, ob er gelernt oder konfiguriert wurde. Zudem wird angezeigt, in welchem Subnetz sich der interne Netzknoten befindet. Bei SCALANCE S Modulen wird das Subnetz von Netzknoten nur im Bridge-Modus angezeigt. |
| Tunneleigenschaften                   | Anzeige von Eigenschaften des VPN-Tunnels, der zu dem Security-Modul aufgebaut wurde, das Sie in der Tabelle "Bekannte Security-Geräte oder -Module" ausgewählt haben.   |

### Gefundene interne Netzknoten - Eintrag "Interne Knoten"

#### Modulspezifische Funktion

Diese Funktion ist nicht für SCALANCE S602 Module und CP 1543-1 verfügbar.

#### Bedeutung

Anzeige aller gelernten und konfigurierten Netzknoten. Zudem wird angezeigt, ob der Lernmodus des Security-Moduls aktiviert oder deaktiviert ist.

### Aktualisierte Firewall-Regeln - Eintrag "Dynamisch aktualisierte Firewall-Regeln"

#### Modulspezifische Funktion

Diese Funktion ist nur für S7-CPs verfügbar, siehe Abschnitt:  
Aktualisierte Firewall-Regeln - Eintrag "Dynamisch aktualisierte Firewall-Regeln" (Seite 618) im Kapitel "Security für S7-300-/S7-400-/PC-CPs".

## Datum und Uhrzeit einstellen - Eintrag "Datum und Uhrzeit"

### Modulspezifische Funktion

Diese Funktion ist nur für SCALANCE S Module verfügbar, siehe Abschnitt:  
Datum und Uhrzeit einstellen - Eintrag "Datum und Uhrzeit" (Seite 603).

### Logging-Funktionen

## System-Ereignisse aufzeichnen - Eintrag "System-Log"

### Bedeutung

Anzeige von geloggt System-Ereignissen sowie Starten und Stoppen des Lesens von System-Ereignissen aus dem lokalen Speicher des Security-Moduls.

Das System-Log zeichnet automatisch fortlaufend System-Ereignisse wie z. B. den Start eines Prozesses auf. Anhand von Ereignisklassen ist die Aufzeichnung skalierbar.

| System- und Statusfunktionen          | Bedeutung  |
|---------------------------------------|--|
| Starte/Stoppe Logging (nicht für CPs) | Startet/Stoppt das Aufzeichnen von System-Ereignissen. Das Verfahren sowie die Ereignisklassen, die aufgezeichnet werden, werden in den lokalen Security-Einstellungen projiziert.   |
| Starte/Stoppe Lesen                   | Startet/Stoppt das Auslesen von System-Ereignissen aus dem lokalen Speicher des Security-Moduls. Aktivieren Sie das Kontrollkästchen "Log-Datei speichern", werden die aufgezeichneten Log-Daten zusätzlich als Datei gespeichert. Wählen Sie den Speicherort aus und geben Sie einen Namen für die Datei an.<br><b>Hinweis</b><br>Haben Sie das Kontrollkästchen "Log-Datei speichern" nach dem Starten des Lesevorgangs aktiviert, können die bis dahin ausgelesenen Daten nicht mehr in einer Log-Datei gespeichert werden. |
| Anzeige löschen                       | Löscht die in der Tabelle angezeigten Log-Daten.   |

Für das Öffnen von gespeicherten System-Ereignissen in Log-Dateien, siehe Kapitel:  
System-Ereignisse im Offline-Modus auswerten - Eintrag "System-Log" (Offline-Ansicht)  
(Seite 585).

## Sicherheits-Ereignisse aufzeichnen - Eintrag "Audit-Log"

### Bedeutung

Anzeige von geloggten Sicherheits-Ereignissen sowie Starten und Stoppen des Lesens von Sicherheits-Ereignissen aus dem lokalen Speicher des Security-Moduls.

Der Audit-Log zeichnet automatisch fortlaufend sicherheitsrelevante Ereignisse auf. Dazu gehören z. B. Benutzeraktionen wie das Ein- und Ausschalten des Paket-Logging.

| System- und Statusfunktionen | Bedeutung   |
|------------------------------|---|
| Starte/Stoppe Lesen          | Startet/Stoppt das Auslesen von Sicherheits-Ereignissen aus dem lokalen Speicher des Security-Moduls. Aktivieren Sie das Kontrollkästchen "Log-Datei speichern", werden die aufgezeichneten Log-Daten zusätzlich als Datei gespeichert. Wählen Sie den Speicherort aus und geben Sie einen Namen für die Datei an.<br><b>Hinweis</b><br>Haben Sie das Kontrollkästchen "Log-Datei speichern" nach dem Starten des Lesevorgangs aktiviert, können die bis dahin ausgelesenen Daten nicht mehr in einer Log-Datei gespeichert werden. |
| Anzeige löschen              | Löscht die in der Tabelle angezeigten Log-Daten.  |

Für das Öffnen von gespeicherten Sicherheits-Ereignissen in Log-Dateien, siehe Kapitel: Sicherheits-Ereignisse im Offline-Modus auswerten - Eintrag "Audit-Log" (Offline-Ansicht) (Seite 586).

## Datenpakete aufzeichnen - Eintrag "Paketfilter-Log"

### Bedeutung

Anzeige von geloggten Datenpaketen sowie Starten und Stoppen des Lesens von Paketfilter-Ereignissen.

Der Paketfilter-Log zeichnet bestimmte Pakete des Datenverkehrs auf. Es werden nur Datenpakete geloggt, auf die eine projektierte Paketfilter-Regel (Firewall) zutrifft, oder auf die der Basisschutz reagiert (korrupte bzw. ungültige Pakete). Voraussetzung ist, dass das Logging für die Paketfilter-Regel aktiviert ist.

Informationen zum Aktivieren des Logging finden Sie in Kapitel: Auto-Hotspot.

Die geloggten Daten können neben dem Lesen aus dem Puffer und dem Übertragen in die Anzeige zusätzlich zur Archivierung in einer Datei gespeichert werden.

| System- und Statusfunktionen          | Bedeutung  |
|---------------------------------------|--|
| Starte/Stoppe Logging (nicht für CPs) | Startet/Stoppt das Aufzeichnen von Datenpaketen. Das Verfahren, nach dem die Daten aufgezeichnet werden, wird in den lokalen Security-Einstellungen projektiert.   |
| Starte/Stoppe Lesen                   | Startet/Stoppt das Auslesen von aufgezeichneten Datenpaketen aus dem lokalen Speicher des Security-Moduls. Aktivieren Sie das Kontrollkästchen "Log-Datei speichern", werden die aufgezeichneten Log-Daten zusätzlich als Datei gespeichert. Wählen Sie den Speicherort aus und geben Sie einen Namen für die Datei an.<br><b>Hinweis</b><br>Haben Sie das Kontrollkästchen "Log-Datei speichern" nach dem Starten des Lesevorgangs aktiviert, können die bis dahin ausgelesenen Daten nicht mehr in einer Log-Datei gespeichert werden. |
| Anzeige löschen                       | Löscht die in der Tabelle angezeigten Log-Daten.   |
| Log-Kategorie                         | Wählen Sie aus, von welchen Datenpaketen das Logging angezeigt werden soll. Die Auswahl ist abhängig von den Einstellungen, die Sie offline in den lokalen Security-Einstellungen projektiert haben. Es werden nur die Datenpakete geloggt, für die das Logging aktiviert wurde. Wählen Sie eine Kategorie aus, für die das Logging nicht aktiviert wurde, werden für diese Kategorie auch keine Daten aufgezeichnet.  |

Für das Öffnen der gespeicherten Paketfilter-Log-Daten, siehe Kapitel: Paketfilter-Ereignisse im Offline-Modus auswerten - Eintrag "Paketfilter-Log" (Offline-Ansicht) (Seite 586).

## Log-Dateien im Offline-Modus auswerten

### System-Ereignisse im Offline-Modus auswerten - Eintrag "System-Log" (Offline-Ansicht)

#### So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf das zu bearbeitende Modul.
2. Wählen Sie im Kontextmenü den Eintrag "Online & Diagnose".
3. Wählen Sie den Eintrag "Diagnose" > "Log-Dateien (Offline-Ansicht)" > "System Log".

#### Bedeutung

Öffnen von geloggt System-Ereignissen, die Sie in der Online-Ansicht als Datei gespeichert haben.

Weitere Informationen finden Sie in Kapitel System-Ereignisse aufzeichnen - Eintrag "System-Log" (Seite 583).

## Sicherheits-Ereignisse im Offline-Modus auswerten - Eintrag "Audit-Log" (Offline-Ansicht)

### So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf das zu bearbeitende Modul.
2. Wählen Sie im Kontextmenü den Eintrag "Online & Diagnose".
3. Wählen Sie den Eintrag "Diagnose" > "Logdateien (Offline-Ansicht)" > "Audit-Log".

### Bedeutung

Öffnen von geloggten Sicherheits-Ereignissen, die Sie in der Online-Ansicht als Datei gespeichert haben.

Weitere Informationen finden Sie in Kapitel Sicherheits-Ereignisse aufzeichnen - Eintrag "Audit-Log" (Seite 584).

## Paketfilter-Ereignisse im Offline-Modus auswerten - Eintrag "Paketfilter-Log" (Offline-Ansicht)

### So erreichen Sie diese Funktion

1. Klicken Sie mit der rechten Maustaste auf das zu bearbeitende Modul.
2. Wählen Sie im Kontextmenü den Eintrag "Online & Diagnose".
3. Wählen Sie den Eintrag "Diagnose" > "Log-Dateien (Offline-Ansicht)" > "Paketfilter-Log".

### Bedeutung

Öffnen von geloggten Datenpaketen, die Sie in der Online-Ansicht als Datei gespeichert haben.

Weitere Informationen finden Sie in Kapitel Datenpakete aufzeichnen - Eintrag "Paketfilter-Log" (Seite 584).

## Ladefunktionen

### Besonderheiten beim Laden von Security-Konfigurationen

Security-Konfigurationen können die Erreichbarkeit des Security-Moduls durch den Projektierungs-PC beeinflussen. Dies ist beispielsweise der Fall, wenn in einer Konfiguration für ein Security-Modul eine Tunnelverbindung zu einem anderen Security-Modul projektiert und diese Konfiguration vom Projektierungs-PC auf das Security-Modul geladen wird. Das Security-Modul ist nach dem Laden durch den Projektierungs-PC nicht mehr erreichbar und der Erreichbarkeitstest, der durch STEP 7 standardmäßig nach dem Laden einer Konfiguration durchgeführt wird, schlägt fehl. Die daraufhin von STEP 7 ausgegebene Fehlermeldung bezieht sich ausschließlich auf den Erreichbarkeitstest; das eigentliche Laden der Konfiguration ist bei konsistenten Projektdaten sowie korrekter IP-Adressbeziehung zwischen Security-Modul und Projektierungs-PC gewährleistet.



Besonderheiten beim Laden von Konfigurationen und Firmware in SCALANCE S Module sind in folgendem Abschnitt des Kapitels "SCALANCE S" beschrieben:  
Auto-Hotspot

## **SOFTNET Security Client**

### **Einsatz des SOFTNET Security Client**

#### **Einsatzbereich - Zugriff über VPN**

Mit der PC-Software SOFTNET Security Client (SSC) sind gesicherte Fernzugriffe vom PG/PC auf Automatisierungsgeräte, die durch ein Security-Modul geschützt sind, über öffentliche Netze hinweg, möglich. Für S7-300-/S7-400-CPs sowie für den PC-CP CP 1628 wird der SOFTNET Security Client V4.0 HF1 benötigt. Mit SOFTNET Security Client  $\leq$  V4.0 sind diese CPs nicht freigegeben. Für den CP 1543-1 ist der SOFTNET Security Client nicht freigegeben.

Mittels des SOFTNET Security Clients wird ein PG/PC automatisch so konfiguriert, dass er eine gesicherte IPsec-Tunnelkommunikation im VPN (Virtual Private Network) zu einem oder mehreren Security-Modulen aufbauen kann.

Durch diese IPsec-Tunnelkommunikation ist es möglich, mit PG/PC-Applikationen wie NCM Diagnose auf gesichertem Weg auf Geräte oder Netzwerke zuzugreifen, die sich in einem durch SCALANCE S geschützten internen Netz befinden.

#### **Wie funktioniert der SOFTNET Security Client?**

Der SOFTNET Security Client liest die von STEP 7 erstellte Konfiguration ein und ermittelt aus der Datei die zu importierenden Zertifikate.

Das CA-Zertifikat und ggf. die privaten Schlüssel werden importiert und im lokalen PG/PC abgelegt.

Anschließend werden mit den Daten aus der Konfiguration Security-Einstellungen vorgenommen, damit Applikationen auf IP-Adressen hinter den Security-Modulen zugreifen können.

Ist der Lernmodus für die internen Teilnehmer bzw. Automatisierungsgeräte aktiviert, wird zunächst eine Sicherheitsrichtlinie für den gesicherten Zugriff auf Security-Module eingestellt. Danach spricht der SOFTNET Security Client die Security-Module an, um die IP-Adressen der jeweils internen Teilnehmer zu ermitteln.

Der SOFTNET Security Client trägt diese IP-Adressen in spezielle Filterlisten dieser Sicherheitsrichtlinie ein. Anschließend können Applikationen mit den Automatisierungsgeräten über VPN kommunizieren.

### **Konfigurationsdatei in STEP 7 erstellen**

#### **SOFTNET Security Client-Modul im Projekt konfigurieren**

Der SOFTNET Security Client wird im Projekt als Modul angelegt. Im Gegensatz zu den anderen Security-Modulen müssen Sie keine weiteren Eigenschaften projektieren.

Weisen Sie das SSC-Modul der oder den VPN-Gruppen zu, in denen IPsec-Tunnel zum PG/PC eingerichtet werden sollen.

**Hinweis**

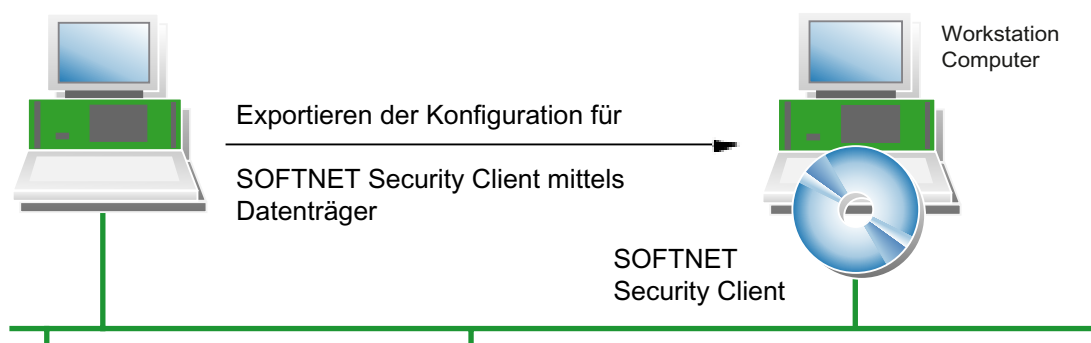
Beachten Sie die Angaben zu den Parametern im Kapitel: Security-Modul in konfigurierte VPN-Gruppe aufnehmen (Seite 569).

**Hinweis**

Wenn Sie mehrere SOFTNET Security Clients innerhalb einer Gruppe anlegen, werden keine Tunnel zwischen diesen Clients aufgebaut, sondern nur vom jeweiligen Client zu den Security-Modulen.

**Konfigurationsdateien für den SOFTNET Security Client**

Die Schnittstelle zwischen STEP 7 als Projektierwerkzeug und dem SOFTNET Security Client wird über Konfigurationsdateien bedient.



Die Konfiguration wird in folgenden Dateitypen hinterlegt:

- \*.dat
- \*.p12
- \*.cer

**Vorgehensweise**

Um die Konfigurationsdateien zu erzeugen, führen Sie in STEP 7 folgende Schritte aus:

1. Selektieren Sie in der Ansicht "Geräte & Netze" das Register "Topologiesicht" oder das Register "Netzsicht".
2. Fügen Sie aus dem Hardware-Katalog ein PC-System vom Typ "SOFTNET Security Client" in das ausgewählte Register ein.
3. Ordnen Sie den SOFTNET Security Client den VPN-Gruppen zu, in denen der PG/PC über IPsec-Tunnel kommunizieren soll.

4. Stellen Sie sicher, dass in den lokalen Security-Einstellungen des SOFTNET Security Clients unter dem Eintrag "Konfiguration des SOFTNET Security Clients" das Kontrollkästchen "SSC-Dateien generieren" aktiviert ist.
5. Wählen Sie den Speicherort für die Konfigurationsdateien.
6. Übersetzen Sie die Konfiguration des SOFTNET Security Clients, um die Konfigurationsdatei zu exportieren.
7. Wenn Sie als Authentifizierungsmethode "Zertifikat" gewählt haben, geben Sie ein Passwort für das Zertifikat der VPN-Konfiguration an. Vergeben Sie kein Passwort, wird der Projektname (nicht das Projektpasswort) als Passwort übernommen.  
Ergebnis: Der Export der Konfigurationsdateien ist abgeschlossen.
8. Übernehmen Sie die Dateien vom Typ \*.dat, \*.p12, \*.cer auf den PG/PC, auf dem Sie den SOFTNET Security Client betreiben möchten.

## SCALANCE S

### Betriebsart und Netzparameter für SCALANCE S Module projektieren

#### Betriebsart einstellen

#### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Betriebsart".

#### Betriebsart - Auswahlmöglichkeiten

Wenn sich das Security-Modul in keiner VPN-Gruppe befindet, kann die Betriebsart unter diesem Eintrag verändert werden. Die Auswahl ist für Port 1 und für Port 2 gültig.

Wenn sich das Security-Modul in einer VPN-Gruppe befindet, kann die Betriebsart nicht verändert werden.

|               |   |
|---------------|---|
| Bridge-Modus  | Für den Betrieb in flachen Netzen. Externer und interner Port befinden sich im selben IP-Subnetz. Nur der externe Port ist konfigurierbar.  |
| Routing-Modus | Externer und interner Port befinden sich in verschiedenen IP-Subnetzen. Wenn Sie den Routing-Modus aktiviert haben, müssen Sie eine interne IP-Adresse und eine interne Subnetzmaske für den internen Port des Security-Moduls projektieren.<br><b>Hinweis</b><br>Wenn Sie für das SCALANCE S-Modul den Routing-Modus aktiviert haben, können keine MAC-Firewall-Regeln definiert werden. |

## Netzparameter konfigurieren

### Bedeutung

Netzparameter wie IP-Adresse und Subnetzmaske für die Schnittstelle(n) des Security-Moduls festlegen.

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Externe Schnittstelle [P1] rot" > "Ethernet-Adressen".

#### Hinweis

##### Projektion der internen Schnittstelle bei Routing-Betrieb

Wenn Sie für das Security-Modul die Betriebsart "Routing" ausgewählt haben, müssen Sie zusätzlich eine interne IP-Adresse und eine interne Subnetzmaske für die interne Schnittstelle des Security-Moduls projektieren. Sie erreichen diese Funktion in den lokalen Security-Einstellungen unter "Interne Schnittstelle [P2] grün" > "Ethernet-Adressen".

3. Nehmen Sie die Einstellungen vor, die in der folgenden Tabelle beschrieben sind.

| Parameter        | Bedeutung   |
|------------------|---|
| IP-Adresse       | IP-Adresse für die externe Schnittstelle.<br>Die IP-Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind, beispielsweise 141.80.0.16 |
| Subnetzmaske     | Die Subnetzmaske besteht aus 4 Dezimalzahlen, die durch einen Punkt voneinander getrennt sind, beispielsweise 255.255.0.0   |
| Router verwenden | Aktivieren Sie dieses Kontrollkästchen, wenn Sie einen Standard-Router verwenden wollen und geben Sie seine IP-Adresse in das Eingabefeld "Router-Adresse" ein.                                     |

## Porteinstellungen

### Standardmäßige Einstellung "Autonegotiation"

Die Ports der Security-Module sind standardmäßig auf "Autonegotiation" eingestellt. In diesem Modus werden Übertragungsgeschwindigkeit und Duplex-Verfahren automatisch ausgewählt. Zudem wird die Autocrossing-Funktion unterstützt.

## Firewall einrichten

### Lokale Firewall-Regeln für SCALANCE S Module

### Firewall mit vordefinierten Firewall-Regeln projektieren

### Firewall mit vordefinierten IP-Regeln projektieren

#### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall".

#### Firewall standardmäßig aktiviert

Das Kontrollkästchen "Firewall aktivieren" ist standardmäßig aktiviert. Die Firewall ist also automatisch aktiviert und alle Zugriffe von extern nach intern und von intern nach extern sind gesperrt. Schalten Sie durch Anklicken der entsprechenden Kontrollkästchen die Firewall-Regeln für die jeweiligen Richtungen frei.

---

##### Hinweis

##### **Detaillierte Firewall-Einstellungen im erweiterten Firewall-Modus**

Im erweiterten Firewall-Modus können Sie Firewall-Regeln auf einzelne Teilnehmer beschränken. Um in den erweiterten Firewall-Modus zu wechseln, aktivieren Sie das Optionskästchen "Firewall im erweiterten Modus aktivieren". Für nähere Informationen zum erweiterten Firewall-Modus, siehe Kapitel: Übersicht Lokale Firewall-Regeln (Seite 540)

---

#### Firewall-Projektierung mit VPN

Befindet sich das Security-Modul in einer VPN-Gruppe, ist standardmäßig das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Dies bedeutet, dass über die externe Schnittstelle nur verschlüsselter IPsec-Datentransfer zugelassen ist.

Deaktivieren Sie das Kontrollkästchen, dann sind die getunnelte Kommunikation und zusätzlich die in den anderen Auswahlboxen angewählten Kommunikationsarten zugelassen.

Tabelle 8-31 Verfügbare Firewall-Regeln und Richtungen

| Dienst                              | Von intern nach extern | Von extern nach intern | Freigegebene Ports                   | Bedeutung   |
|-------------------------------------|------------------------|------------------------|--------------------------------------|---|
| Erlaube IP-Kommunikation            | x                      | x                      | -                                    | Der IP-Kommunikation für die ausgewählten Kommunikationsrichtungen wird zugelassen. |
| Erlaube S7-Protokoll                | x                      | x                      | TCP-Port 102                         | Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.         |
| Erlaube FTP/FTPS (expliziter Modus) | x                      | x                      | TCP-Port 20<br>TCP-Port 21           | Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.                    |
| Erlaube HTTP                        | x                      | x                      | TCP-Port 80                          | Zur Kommunikation mit einem Webserver.  |
| Erlaube HTTPS                       | x                      | x                      | TCP-Port 443                         | Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.          |
| Erlaube DNS                         | x                      | x                      | TCP-Port 53<br>UDP-Port 53           | Kommunikationsverbindung zu einem DNS-Server wird zugelassen.                       |
| Erlaube SNMP                        | x                      | x                      | TCP-Port 161/162<br>UDP-Port 161/162 | Zur Überwachung von SNMP-fähigen Netzteilnehmern.                                   |
| Erlaube SMTP                        | x                      | x                      | TCP-Port 25                          | Zum Versenden von E-Mails über einen SMTP-Server.                                   |
| Erlaube NTP                         | x                      | x                      | UDP-Port 123                         | Zur Synchronisation der Uhrzeit.  |
| Erlaube DHCP                        | x                      | x                      | UDP-Port 67<br>UDP-Port 68           | Kommunikation mit einem DHCP-Server wird zugelassen.                                |

Tabelle 8-32 Logging

| Option                                     | Aktion bei Aktivierung   |
|--|--|
| Aufzeichnen getunnelter Pakete             | Nur aktiv, wenn Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt. |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.  |
| Aufzeichnen blockierter ausgehender Pakete | Alle ausgehenden IP-Pakete, die verworfen werden, werden geloggt.  |

Die geloggt Pakete können Sie im Eintrag "Paketfilter-Log" des "Online & Diagnose"-Dialogs einsehen. Nähere Informationen hierzu finden Sie im Abschnitt Datenpakete aufzeichnen - Eintrag "Paketfilter-Log" (Seite 584).

### Firewall mit vordefinierten MAC-Regeln projektieren

#### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall".

## Firewall standardmäßig aktiviert

Das Kontrollkästchen "Firewall aktivieren" ist standardmäßig aktiviert. Die Firewall ist also automatisch aktiviert und alle Zugriffe von extern nach intern und intern nach extern sind gesperrt. Schalten Sie durch Anklicken der entsprechenden Kontrollkästchen die Firewall-Regeln für die jeweiligen Richtungen frei.

### Hinweis

#### Detaillierte Firewall-Einstellungen im erweiterten Firewall-Modus

Im erweiterten Firewall-Modus können Sie Firewall-Regeln auf einzelne Teilnehmer beschränken. Um in den erweiterten Firewall-Modus zu wechseln, aktivieren Sie das Optionskästchen "Firewall im erweiterten Modus aktivieren". Für nähere Informationen zum erweiterten Firewall-Modus, siehe Kapitel Übersicht Lokale Firewall-Regeln (Seite 540).

## Firewall-Projektierung mit VPN

Befindet sich das Security-Modul in einer VPN-Gruppe, ist standardmäßig das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Dies bedeutet, dass über die externe Schnittstelle nur verschlüsselter IPsec-Datentransfer zugelassen ist.

Deaktivieren Sie das Kontrollkästchen, dann sind die getunnelte Kommunikation und zusätzlich die in den anderen Auswahlboxen angewählten Kommunikationsarten zugelassen.

## Verfügbare MAC-Regeln und Richtungen

| Dienst                    | Von intern nach extern | Von extern nach intern | Bedeutung  |
|---------------------------|------------------------|------------------------|--|
| Erlaube MAC-Kommunikation | x                      | x                      | Der MAC-Verkehr von intern nach extern und umgekehrt ist zugelassen.                         |
| Erlaube ISO-Protokoll     | x                      | x                      | Der ISO-Verkehr von intern nach extern und umgekehrt ist zugelassen.                         |
| Erlaube SiClock           | x                      | x                      | SiClock-Uhrzeittelegramme von intern nach extern und umgekehrt sind zugelassen.              |
| Erlaube DCP               | x                      | x                      | DCP-Verkehr zur Vergabe von IP-Adressen ist von intern nach extern und umgekehrt zugelassen. |

Tabelle 8-33 Logging

| Option                                     | Aktion bei Aktivierung  |
|--|---|
| Aufzeichnen getunnelter Pakete             | Nur aktiv, wenn Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle MAC-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt. |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt.  |
| Aufzeichnen blockierter ausgehender Pakete | Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.  |

## IP-Paketfilter-Richtungen SCALANCE S

### Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den IP-Regeln des erweiterten Firewall-Modus.

### Folgende Richtungen sind verfügbar

| Auswahlmöglichkeiten / Wertebereiche |        | Security-Modul |            |         |         |
|--------------------------------------|--------|----------------|------------|---------|---------|
| Von                                  | Nach   | S602 V2/V3     | S612 V2/V3 | S613 V2 | S623 V3 |
| Intern                               | Extern | x              | x          | x       | x       |
|                                      | Tunnel | -              | x          | x       | x       |
|                                      | Any    | -              | x          | x       | x       |
| Extern                               | Intern | x              | x          | x       | x       |
|                                      | Any    | -              | -          | -       | -       |
|                                      | Tunnel | -              | -          | -       | -       |
| Tunnel                               | Intern | -              | x          | x       | x       |
|                                      | Extern | -              | -          | -       | -       |
|                                      | Any    | -              | -          | -       | -       |
| Any                                  | Intern | -              | x          | x       | x       |
|                                      | Extern | -              | -          | -       | -       |
|                                      | Tunnel | -              | -          | -       | -       |

x = Kommunikationsrichtung ist projektierbar

- = Kommunikationsrichtung ist nicht projektierbar

## MAC-Paketfilter-Richtungen SCALANCE S

### Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den MAC-Regeln des erweiterten Firewall-Modus.

### Folgende Richtungen sind verfügbar

| Auswahlmöglichkeiten / Wertebereiche |        | Security-Modul |            |         |         |
|--------------------------------------|--------|----------------|------------|---------|---------|
| Von                                  | Nach   | S602 V2/V3     | S612 V2/V3 | S613 V2 | S623 V3 |
| Intern                               | Extern | x              | x          | x       | x       |
|                                      | Tunnel | -              | x          | x       | x       |
|                                      | Any    | -              | x          | x       | x       |
| Extern                               | Intern | x              | x          | x       | x       |



| Auswahlmöglichkeiten / Wertebereiche |        | Security-Modul |          |          |          |
|--------------------------------------|--------|----------------|----------|----------|----------|
|                                      | Any    | -              | -        | -        | -        |
|                                      | Tunnel | -              | -        | -        | -        |
| Tunnel                               | Intern | -              | <b>x</b> | <b>x</b> | <b>x</b> |
|                                      | Extern | -              | -        | -        | -        |
|                                      | Any    | -              | -        | -        | -        |
| Any                                  | Intern | -              | <b>x</b> | <b>x</b> | <b>x</b> |
|                                      | Extern | -              | -        | -        | -        |
|                                      | Tunnel | -              | -        | -        | -        |

x = Kommunikationsrichtung ist projektierbar  
 - = Kommunikationsrichtung ist nicht projektierbar

## Routen festlegen

### Bedeutung

Routen für die Adressierung von Subnetzen angeben, die nicht direkt über das Security-Modul erreicht werden können.

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Routing".
3. Doppelklicken Sie auf den Eintrag "Neu hinzufügen" in der Tabelle, um eine Route hinzuzufügen.
4. Tragen Sie die folgende Werte ein:

| Parameter         | Funktion  | Beispiel-Wert |
|-------------------|---|---------------|
| Netz-ID           | Anfragen an Teilnehmer des Subnetzes mit der hier angegebenen Netz-ID und der angegebenen Subnetzmaske werden über die angegebene Router-IP-Adresse in das Subnetz geleitet.<br><br>Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse im oder außerhalb des Subnetzes liegt.<br><br>Die angegebene Netz-ID darf nicht im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls. | 192.168.11.0  |
| Subnetzmaske      | Die Subnetzmaske strukturiert das Netz. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.  | 255.255.255.0 |
| Router-IP-Adresse | IP-Adresse des Routers, über den das Subnetz erreicht wird.<br><br>Die angegebene Router-IP-Adresse muss im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls.   | 192.168.10.2  |

## Security-Modul als DHCP-Server

### Übersicht DHCP-Server

#### Übersicht

Sie können das SCALANCE S Modul am internen Netz als DHCP-Server (DHCP = Dynamic Host Configuration Protocol) betreiben. Damit ist es möglich, den am internen Netz angeschlossenen Geräten automatisch IP-Adressen zuzuweisen.

Die IP-Adressen werden hierbei entweder dynamisch aus einem von Ihnen vergebenen Adressband verteilt oder es wird gemäß Ihrer Vorgabe eine bestimmte IP-Adresse einem bestimmten Gerät zugewiesen.

#### Voraussetzungen

Sie müssen die Geräte am internen Netz so konfigurieren, dass diese die IP-Adresse von einem DHCP-Server beziehen.

Je nachdem, wie das Security-Modul konfiguriert ist, übermittelt es den Teilnehmern im Subnetz die IP-Adresse des Standard-Routers oder Sie müssen den Teilnehmern im Subnetz die Router-IP-Adresse bekannt machen.

- Router-IP-Adresse wird übermittelt  
In folgenden Fällen wird durch das DHCP-Protokoll vom SCALANCE S Modul eine Router-IP-Adresse an den Teilnehmer übermittelt:
  - Das SCALANCE S Modul ist für den Router-Betrieb konfiguriert  
Das SCALANCE S Modul übermittelt in diesem Fall die eigene IP-Adresse als Router-IP-Adresse.
  - Das SCALANCE S Modul ist nicht für den Router-Betrieb konfiguriert, es ist aber in der Konfiguration des SCALANCE S Moduls ein Standard-Router angegeben  
Das SCALANCE S Modul übermittelt in diesem Fall die IP-Adresse des Standard-Routers als Router-IP-Adresse.
- Router-IP-Adresse wird nicht übermittelt  
Tragen Sie in folgendem Fall die Router-IP-Adresse beim Teilnehmer manuell ein:
  - Das SCALANCE S Modul ist nicht für den Router-Betrieb konfiguriert und es ist in der Konfiguration kein Standard-Router angegeben.

### Siehe auch

DHCP-Server konfigurieren (Seite 597)

## DHCP-Server konfigurieren

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "DHCP-Server".
3. Wählen Sie aus, für welche Schnittstelle die DHCP-Einstellungen vorgenommen werden sollen.
4. Nehmen Sie die Adresszuordnung vor. Sie haben die beiden folgenden Möglichkeiten zur Konfiguration:
  - Statische Adresszuordnung  
Geräten mit einer bestimmten MAC-Adresse oder Client-ID werden jeweils vorgegebene IP-Adressen zugeordnet. Tragen Sie hierzu diese Geräte in die Adressliste im Eingabebereich "Statische Adresszuordnung" ein.
  - Dynamische Adresszuordnung  
Geräte, deren MAC-Adresse oder deren Client-ID nicht explizit angegeben wurde, erhalten eine beliebige IP-Adresse aus einem vorgegebenen Adressband. Aktivieren Sie hierzu das Kontrollkästchen "Dynamischen IP-Adressbereich" aktivieren". Stellen Sie im Anschluss das Adressband im Eingabebereich "Dynamischen IP-Adressbereich aktivieren" ein.

**Hinweis**

**Dynamische Adressvergabe - Verhalten nach Unterbrechung der Spannungsversorgung**

Beachten Sie, dass die dynamisch vergebenen IP-Adressen nicht gespeichert werden, wenn die Spannungsversorgung unterbrochen wird. Nach Wiederkehr der Spannungsversorgung müssen Sie daher dafür sorgen, dass die Teilnehmer erneut eine IP-Adresse anfordern.

Sie sollten daher die dynamische Adressvergabe nur für folgende Teilnehmer vorsehen:

- Teilnehmer, die im Subnetz temporär genutzt werden (wie beispielsweise Service-Geräte).
- Teilnehmer, die eine einmal zugewiesene IP-Adresse bei einer erneuten Anforderung an den DHCP-Server als "Vorzugsadresse" übermitteln (wie beispielsweise PC-Stationen).

Für dauerhafte Teilnehmer ist die statische Adresszuweisung über die Angabe einer Client-ID oder der MAC-Adresse vorzuziehen.

**Konsistenzprüfung - diese Regeln müssen Sie beachten**

Berücksichtigen Sie bei Ihrer Eingabe die nachfolgend aufgeführten Regeln.

| Prüfung / Regel  | Prüfung erfolgt <sup>1)</sup> |                        |
|--|-------------------------------|------------------------|
|  | lokal                         | projekt-/<br>modulweit |
| Die in der Adressliste im Eingabebereich "Statische Adresszuordnung" zugewiesenen IP-Adressen dürfen nicht im Bereich der dynamischen IP-Adressen liegen.  |                               | <b>x</b>               |
| IP-Adressen, MAC-Adressen und Client-IDs dürfen in der Tabelle "Statische Adresszuordnung" nur einmal vorkommen (bezogen auf das Security-Modul).  |                               | <b>x</b>               |
| Sie müssen bei den statisch zugewiesenen IP-Adressen entweder die MAC-Adresse oder die Client-ID (Rechnername) angeben.  | <b>x</b>                      |                        |
| Die Client-ID ist eine Zeichenfolge mit maximal 63 Zeichen. Es dürfen nur die folgenden Zeichen verwendet werden: a-z, A-Z, 0-9 und - (Bindestrich).<br><b>Hinweis</b><br>Bei SIMATIC S7 kann den Geräten an der Ethernet-Schnittstelle für den Bezug einer IP-Adresse über DHCP eine Client-ID zugewiesen werden.<br>Bei PCs ist die Vorgehensweise abhängig vom verwendeten Betriebssystem; empfohlen wird, hier die MAC-Adresse für die Zuordnung zu verwenden. | <b>x</b>                      |                        |
| Sie müssen bei den statisch zugewiesenen IP-Adressen die IP-Adresse angeben.   | <b>x</b>                      |                        |
| Folgende IP-Adressen dürfen nicht im Bereich des freien IP-Adressbereichs (dynamische IP-Adressen) liegen: <ul style="list-style-type: none"> <li>• alle Router-IP-Adressen im Eintrag "Routing"</li> <li>• Syslog-Server</li> <li>• Standard-Router</li> <li>• Security-Modul-Adresse(n)</li> </ul>   |                               | <b>x</b>               |

| Prüfung / Regel  | Prüfung erfolgt <sup>1)</sup> |                        |
|--|-------------------------------|------------------------|
|  | lokal                         | projekt-/<br>modulweit |
| DHCP wird vom Security-Modul an der Schnittstelle zum internen Subnetz unterstützt. Aus diesem Betriebsverhalten des Security-Moduls ergeben sich weiterhin für IP-Adressen im Bereich des freien IP-Adressbereichs (dynamische IP-Adressen) folgende Anforderungen: <ul style="list-style-type: none"> <li>• Bridge-Modus<br/>Der Bereich des freien IP-Adressbereichs muss in dem durch das Security-Modul definierten Netz liegen.</li> <li>• Routing-Modus<br/>Der Bereich des freien IP-Adressbereichs muss in dem durch das Security-Modul definierten internen Subnetz liegen.</li> </ul> |                               | <b>x</b>               |
| Der freie IP-Adressbereich muss durch die Angabe der Startadresse und der Endadresse vollständig angegeben sein. Die Endadresse muss größer als die Startadresse sein.   | <b>x</b>                      |                        |
| Die IP-Adressen, die Sie in die Adressliste im Eingabebereich "Statische Adresszuordnung" eingeben, müssen im Adressbereich des internen Subnetzes des Security-Moduls liegen.   |                               | <b>x</b>               |

**Legende:**

<sup>1)</sup> Beachten Sie die Erläuterungen im Kapitel: Konsistenzprüfung durchführen (Seite 517).

**IPSec-Tunnel: Gruppen anlegen und zuordnen****Interne Netzknoten konfigurieren****Lernmodus zum Lernen interner Knoten verwenden****Teilnehmer für die Tunnelkommunikation automatisch finden**

Ein großer Vorteil für die Konfiguration und den Betrieb der Tunnelkommunikation ist, dass SCALANCE S Module im Bridge-Modus die Teilnehmer an der internen Schnittstelle selbstständig auffinden können. Neue Teilnehmer werden vom Security-Modul im laufenden Betrieb erkannt. Die erkannten Teilnehmer werden an die Security-Module gemeldet, die zur selben Gruppe gehören. Dadurch ist der Datenaustausch innerhalb der Tunnel einer Gruppe jederzeit in beide Richtungen gewährleistet.

## Erkennbare Teilnehmer

Erkannt werden folgende Teilnehmer:

- IP-fähige Netzknoten  
IP-fähige Netzknoten werden gefunden, wenn sie eine ICMP-Antwort auf den ICMP-Subnetz-Broadcast senden.  
IP-Knoten hinter Routern sind auffindbar, wenn die Router ICMP-Broadcasts weiterleiten.
- ISO-Netzknoten  
Netzknoten, die nicht IP-fähig, jedoch über ISO-Protokolle ansprechbar sind, können ebenfalls gelernt werden.  
Voraussetzung ist, dass sie auf XID- bzw. TEST-Telegramme antworten. TEST und XID (Exchange Identification) sind Hilfsprotokolle zum Informationsaustausch auf der Layer-2-Ebene. Durch das Versenden dieser Telegramme mit einer Broadcast-Adresse können diese Netzknoten auffindbar gemacht werden.
- PROFINET-Knoten  
Mit Hilfe von DCP (Discovery and basic Configuration Protocol) werden PROFINET-Knoten gefunden.

Netzknoten, die diese Bedingungen nicht erfüllen, müssen Sie manuell konfigurieren.

Subnetze, die sich hinter internen Routern befinden, müssen Sie ebenfalls manuell konfigurieren.

## So erreichen Sie die Funktion

1. Markieren Sie das Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Knoten".

## Lernmodus ein-/ausschalten

Die Lernfunktion ist in der Konfiguration standardmäßig für jedes Security-Modul eingeschaltet.

Das Lernen kann für SCALANCE S auch völlig abgeschaltet werden. Dann müssen Sie alle internen Netzknoten, die an der Tunnelkommunikation teilnehmen sollen, manuell konfigurieren.

## Wann ist es sinnvoll, den automatischen Lernmodus auszuschalten?

Die Standardeinstellungen für das Security-Modul gehen davon aus, dass interne Netze stets sicher sind; das heißt auch, dass im Normalfall keine Netzknoten in das interne Netz zugeschaltet werden, die nicht vertrauenswürdig sind.

Das Ausschalten des Lernmodus kann sinnvoll sein, wenn das interne Netz statisch ist, d. h., wenn sich die Anzahl der internen Knoten und deren Adressen nicht ändern.

Mit Ausschalten des Lernmodus entfällt im internen Netz die Belastung des Mediums und der Knoten durch die Lerntelegramme. Auch das Security-Modul wird etwas leistungsfähiger, da es nicht durch die Bearbeitung der Lerntelegramme belastet wird.

Anmerkung: Im Lernmodus werden alle Knoten im internen Netz erfasst. Die Angaben zum VPN-Mengengerüst beziehen sich nur auf die Knoten, die im internen Netz über VPN kommunizieren.

---

#### **Hinweis**

Werden im internen Netz mehr als 128 interne Knoten betrieben, wird damit das zulässige Mengengerüst überschritten und ein nicht erlaubter Betriebszustand erzeugt. Aufgrund der Dynamik im Netzwerkverkehr kommt es dann dazu, dass interne Knoten, die bereits gelernt wurden, wieder durch neue, bis jetzt noch nicht bekannte interne Knoten ersetzt werden.

---

#### **Siehe auch**

Interne Subnetze manuell konfigurieren (Seite 602)

#### **IP-Netzknoten manuell konfigurieren**

##### **Bedeutung**

Alternativ zum Lernmodus, den Sie über das Kontrollkästchen "Lernen interner Knoten zulassen" aktivieren und über den die internen Netzknoten dynamisch vom Security-Modul gelernt werden, können Sie die zu lernenden Netzknoten im Eintrag "Interne IP-Knoten" manuell eintragen und damit für die VPN-Tunnelkommunikation freigeben. Die MAC-Adresse eines Netzknotens ist hierbei eine optionale Angabe.

##### **Voraussetzung**

- Das Security-Modul befindet sich im Bridge-Modus.
- Das Security-Modul ist Teilnehmer einer VPN-Gruppe.

##### **So erreichen Sie diese Funktion**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen "Knoten" > "Interne IP-Knoten".

#### **MAC-Netzknoten manuell konfigurieren**

##### **Bedeutung**

Alternativ zum Lernmodus, den Sie über das Kontrollkästchen "Lernen interner Knoten zulassen" aktivieren und über den die internen Netzknoten dynamisch vom Security-Modul gelernt werden, können Sie die zu lernenden Netzknoten im Eintrag "Interne MAC-Knoten" manuell eintragen und damit für die VPN-Tunnelkommunikation freigeben.

### Voraussetzung

- Das Security-Modul befindet sich im Bridge-Modus.
- Das Security-Modul ist Teilnehmer einer VPN-Gruppe.

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Knoten" > "Interne MAC-Knoten".

### Interne Subnetze manuell konfigurieren

#### Voraussetzung

- Das Security-Modul ist Teilnehmer einer VPN-Gruppe.

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Knoten" > "Interne Subnetze".

### Security-Modul im Bridge-Modus - Eintrag "Interne Subnetze"

Um interne Subnetze manuell für die VPN-Tunnelkommunikation freizugeben, müssen Sie folgende Adressparameter eingeben:

| Parameter         | Funktion   | Beispiel-Wert |
|-------------------|--|---------------|
| Netz-ID           | Netz-ID des Subnetzes, das für die VPN-Tunnelkommunikation freigegeben werden soll.<br>Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.<br>Darf nicht im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls. | 192.168.11.0  |
| Subnetzmaske      | Die Subnetzmaske strukturiert das Netz. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.   | 255.255.255.0 |
| Router-IP-Adresse | IP-Adresse des Routers, über den das freizugebende Subnetz erreicht wird.<br>Muss im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls.   | 192.168.10.2  |



## Security-Modul im Routing-Modus - Eintrag "Durch Tunnel erreichbare Subnetze"

Im Routing-Modus werden stets komplette Subnetze getunnelt. Um interne Subnetze manuell für die VPN-Tunnelkommunikation freizugeben, müssen Sie folgende Adressparameter eingeben:

| Parameter    | Funktion   | Beispiel-Wert |
|--------------|--|---------------|
| Netz-ID      | Netz-ID des Subnetzes, das für die VPN-Tunnelkommunikation freigegeben werden soll.<br>Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.<br>Darf nicht im gleichen Subnetz liegen wie die IP-Adresse des Security-Moduls. | 192.168.11.0  |
| Subnetzmaske | Die Subnetzmaske strukturiert das Netz. Anhand der Netz-ID und der Subnetzmaske erkennt der Router, ob eine Ziel-Adresse innerhalb oder außerhalb des Subnetzes liegt.   | 255.255.255.0 |
| Kommentar    | Zusätzliche, optionale Kommentareingabe.   |               |

## Online-Funktionen - Test / Diagnose und Logging

### Datum und Uhrzeit einstellen - Eintrag "Datum und Uhrzeit"

#### So erreichen Sie diese Funktion

1. Selektieren Sie das Security-Modul, dessen Uhrzeit und Datum Sie prüfen bzw. einstellen möchten.
2. Wählen Sie im Kontextmenü den Menübefehl "Online & Diagnose".
3. Wählen Sie in der Online-Diagnose den Eintrag "Funktionen" > "Datum und Uhrzeit".

#### Lokale Uhrzeit auf dem Security-Modul

In diesem Bereich können Sie die Uhrzeit und das Datum aus dem Security-Modul auslesen und einstellen. Wenn Sie auf die Schaltfläche "Übernehmen" klicken, werden dem Security-Modul die Uhrzeit und das Datum zugewiesen, die aktuell in den Eingabefeldern "Datum" und "Uhrzeit" eingetragen sind.

#### Lokale Uhrzeit auf dem PC

In diesem Bereich wird die aktuelle Uhrzeit und das aktuelle Datum des PCs angezeigt, auf dem STEP 7 installiert ist. Wenn Sie auf die Schaltfläche "Für Baugruppe übernehmen" klicken, wird dem Security-Modul die aktuelle Uhrzeit und das aktuelle Datum des PCs zugewiesen.

## Ladefunktionen

### Laden einer Konfiguration

#### Das müssen Sie vor dem Laden einer Konfiguration beachten

- **Verwendete IP-Adresse für das Laden von Konfigurationen**  
Konfigurationen werden stets über die IP-Adresse auf das Security-Modul geladen, die in den lokalen Security-Einstellungen für den internen oder externen Port projiziert ist. Diese IP-Adresse muss mit der des Security-Moduls übereinstimmen.
- **Anschlüsse**  
Prinzipiell können Sie die Konfigurationsdaten sowohl über die externe als auch über die interne Schnittstelle auf das Security-Modul laden.  
Konfigurieren Sie die Module einer VPN-Gruppe vorzugsweise über das gemeinsame externe Netz dieser Module (Geräte-Port 1). Falls sich der Konfigurationsrechner in einem internen Netz befindet, müssen in der Firewall dieses SCALANCE S Moduls die IP-Adressen der anderen Module der Gruppe explizit freigegeben und dieses Modul als erstes geladen werden.

---

#### Hinweis

##### Netzwerkadapter wählen

Wenn Sie mehrere Netzwerkadapter in Ihrem PC/PG betreiben, wählen Sie in der Klappliste "PG/PC-Schnittstelle" des Dialogs "Erweitertes Laden" zunächst den Netzwerkadapter aus, über den Sie das SCALANCE S Modul erreichen können.

---

- **Betriebszustand**  
Konfigurationen können im laufenden Betrieb der SCALANCE S Geräte geladen werden. Damit die Änderungen aus einer Konfiguration wirksam werden, muss das SCALANCE S Modul neu gestartet werden.

---

#### Hinweis

##### Besonderheiten

- Solange ein Modul noch keine IP-Parameter eingestellt hat (d. h. vor der ersten Konfiguration), darf sich zwischen Modul und Konfigurationsrechner kein Router befinden.
  - Wenn Sie einen PC von der internen auf die externe Schnittstelle des SCALANCE S umstecken, dann werden Zugriffe dieses PCs auf den SCALANCE S für ca. 20 min blockiert.
-

## Übertragen einer Firmware

### Das müssen Sie vor dem Übertragen einer neuen Firmware beachten

Zum Übertragen einer neuen Firmware auf ein Security-Modul müssen folgende Voraussetzungen erfüllt sein:

- Sie besitzen die notwendigen Rechte zur Übertragung einer Firmware, siehe Kapitel: Auto-Hotspot.
- Das Security-Modul ist mit einer IP-Adresse projektiert.

### Die Übertragung ist sicher

Das Übertragen der Firmware erfolgt über eine gesicherte Verbindung und kann deshalb auch aus dem ungeschützten Netzwerk vorgenommen werden.

Die Firmware selbst ist signiert und verschlüsselt. Damit ist sichergestellt, dass nur authentische Firmware auf das SCALANCE S-Modul geladen werden kann.

### Die Übertragung kann im laufenden Betrieb vorgenommen werden

Die Übertragung der Firmware kann im laufenden Betrieb eines SCALANCE S-Moduls erfolgen. Eine neu geladene Firmware wird erst nach einem Neustart des SCALANCE S-Moduls aktiv. Wurde die Übertragung gestört und abgebrochen, so startet die Baugruppe wieder mit dem alten Firmwarestand.

## Security für S7-300-/S7-400-/PC-CPs

### Firewall einrichten

### Lokale Firewall-Regeln für S7-300-/S7-400-/PC-CPs

### Übersicht S7-300-/S7-400-/PC-CPs

### Paketfilter-Regeln aktivieren

Wenn Sie in den lokalen Security-Einstellungen für die CPs die Security-Funktion aktivieren, sind zunächst alle Zugriffe auf und über den CP zugelassen. Um einzelne Paketfilter-Regeln zu aktivieren, aktivieren Sie das Kontrollkästchen "Firewall aktivieren". Schalten Sie anschließend die gewünschten Dienste frei. Firewall-Regeln, die aufgrund einer

Verbindungsprojektierung automatisch angelegt werden, haben Vorrang vor den manuell eingestellten Regeln.

---

**Hinweis**

**Detaillierte Firewall-Einstellungen im erweiterten Firewall-Modus**

Im erweiterten Firewall-Modus können Sie Firewall-Regeln auf einzelne Teilnehmer beschränken. Um in den erweiterten Firewall-Modus zu wechseln, aktivieren Sie das Kontrollkästchen "Firewall im erweiterten Modus aktivieren".

---

### Firewall-Projektierung mit VPN

Wird das Security-Modul einer VPN-Gruppe hinzugefügt, wird standardmäßig die Firewall aktiviert. Zudem wird das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Dies bedeutet, dass über die externe Schnittstelle nur verschlüsselter IPsec-Datentransfer zugelassen ist. Datenverkehr nach extern wird geblockt.

Deaktivieren Sie das Kontrollkästchen, dann sind die getunnelte Kommunikation und zusätzlich die in den anderen Auswahlboxen angewählten Kommunikationsarten zugelassen.

### Verbindungsregeln aktualisieren

Durch das Verändern der Verbindungsprojektierung von CPs ändern sich auch die verbindungsbezogenen Firewall-Regeln. Um die geänderten Firewall-Regeln anzuzeigen, müssen Sie erneut die Schaltfläche "Verbindungsregeln aktualisieren" betätigen. Die geänderten Firewall-Regeln werden dann im erweiterten Firewall-Modus angezeigt.

### Firewall mit vordefinierten Firewall-Regeln projektieren - CP x43-1 Advanced

### Firewall mit vordefinierten IP-Regeln projektieren - CP x43-1 Advanced

#### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Vordefinierte IP-Regeln".

Tabelle 8-34 Verfügbare Dienste und Richtungen

| Dienst                              | Von Station/<br>intern nach<br>extern | Von extern<br>nach intern | Von extern<br>nach Station | Freigegebene<br>Ports                | Bedeutung   |
|-------------------------------------|---------------------------------------|---------------------------|----------------------------|--------------------------------------|---|
| Erlaube IP-Kommunikation            | x                                     | x                         | x                          | alle                                 | Der IP-Verkehr für die ausgewählten Kommunikationsrichtungen wird zugelassen. |
| Erlaube S7-Protokoll                | x                                     | x                         | x                          | TCP-Port 102                         | Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen.   |
| Erlaube FTP/FTPS (expliziter Modus) | x                                     | x                         | x                          | TCP-Port 20<br>TCP-Port 21           | Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.              |
| Erlaube HTTP                        | x                                     | x                         | x                          | TCP-Port 80                          | Zur Kommunikation mit einem Webserver.  |
| Erlaube HTTPS                       | x                                     | x                         | x                          | TCP-Port 443                         | Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.    |
| Erlaube DNS                         | x                                     | x                         | -                          | TCP-Port 53<br>UDP-Port 53           | Kommunikationsverbindung zu einem DNS-Server wird zugelassen.                 |
| Erlaube SNMP                        | x                                     | x                         | x                          | TCP-Port 161/162<br>UDP-Port 161/162 | Zur Überwachung von SNMP-fähigen Netzteilnehmern.                             |
| Erlaube SMTP                        | x                                     | x                         | -                          | TCP-Port 25                          | Zum Versenden von E-Mails über einen SMTP-Server.                             |
| Erlaube NTP                         | x                                     | x                         | -                          | UDP-Port 123                         | Zur Synchronisation der Uhrzeit.  |

Tabelle 8-35 Logging

| Option                                     | Aktion bei Aktivierung   | Zutreffende Firewall-Regel |         |         |
|--|--|----------------------------|---------|---------|
|  |  | Aktion                     | Von     | Nach    |
| <b>IP-Log-Einstellungen</b>                |  |                            |         |         |
| Aufzeichnen getunnelter Pakete             | Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt. | Allow                      | Station | Tunnel  |
|  |  | Allow                      | Tunnel  | Station |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.  | Drop                       | Extern  | Station |

**Hinweis**

**Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln**

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

**Firewall mit vordefinierten MAC-Regeln projektieren - CP x43-1 Advanced**

**So erreichen Sie diese Funktion**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Firewall" > "Vordefinierte MAC-Regeln".

Tabelle 8-36 Verfügbare Dienste und Richtungen

| Dienst                    | Von Station nach extern | Von extern nach Station | Bedeutung   |
|---------------------------|-------------------------|-------------------------|---|
| Erlaube MAC-Kommunikation | x                       | x                       | Der MAC-Verkehr von Station nach extern und umgekehrt ist zugelassen. |
| Erlaube ISO-Protokoll     | x                       | x                       | Der ISO-Verkehr von Station nach extern und umgekehrt ist zugelassen. |

Tabelle 8-37 Logging

| Option                                     | Aktion bei Aktivierung  | Zutreffende Firewall-Regel |         |         |
|--|---|----------------------------|---------|---------|
| MAC-Log-Einstellungen                      |   | Aktion                     | Von     | Nach    |
| Aufzeichnen getunnelter Pakete             | Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle MAC-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt. | Allow                      | Station | Tunnel  |
|  |   | Allow                      | Tunnel  | Station |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt.  | Drop                       | Extern  | Station |
| Aufzeichnen blockierter ausgehender Pakete | Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.  | Drop                       | Station | Extern  |

**Hinweis**

**Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln**

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

## Firewall mit vordefinierten Firewall-Regeln projektieren - CP1628

## Firewall mit vordefinierten IP-Regeln projektieren - CP1628

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Security" > "Firewall" > "Vordefinierte IP-Regeln".

Tabelle 8-38 Verfügbare Dienste und Richtungen

| Dienst                              | Von Extern nach Station | Freigegebene Ports                   | Bedeutung   |
|-------------------------------------|-------------------------|--------------------------------------|---|
| Erlaube IP-Kommunikation            | x                       | alle                                 | Der IP-Verkehr von Extern nach Station wird zugelassen.                     |
| Erlaube S7-Protokoll                | x                       | TCP-Port 102                         | Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen. |
| Erlaube FTP/FTPS (expliziter Modus) | x                       | TCP-Port 20<br>TCP-Port 21           | Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.            |
| Erlaube HTTP                        | x                       | TCP-Port 80                          | Zur Kommunikation mit einem Webserver.                                      |
| Erlaube HTTPS                       | x                       | TCP-Port 443                         | Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.  |
| Erlaube DNS                         | x                       | TCP-Port 53<br>UDP-Port 53           | Kommunikationsverbindung zu einem DNS-Server wird zugelassen.               |
| Erlaube SNMP                        | x                       | TCP-Port 161/162<br>UDP-Port 161/162 | Zur Überwachung von SNMP-fähigen Netzteilnehmern.                           |
| Erlaube SMTP                        | x                       | TCP-Port 25                          | Zum Versenden von E-Mails über einen SMTP-Server.                           |
| Erlaube NTP                         | x                       | UDP-Port 123                         | Zur Synchronisation der Uhrzeit.  |

Tabelle 8-39 Logging

| Option                                     | Aktion bei Aktivierung   | Zutreffende Firewall-Regel |         |         |
|--|--|----------------------------|---------|---------|
|  |  | Aktion                     | Von     | Nach    |
| <b>IP-Log-Einstellungen</b>                |  |                            |         |         |
| Aufzeichnen getunnelter Pakete             | Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt. | Allow                      | Station | Tunnel  |
|  |  | Allow                      | Tunnel  | Station |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.  | Drop                       | Extern  | Station |

**Hinweis**

**Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln**

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

**Firewall mit vordefinierten MAC-Regeln projektieren - CP1628**

**So erreichen Sie diese Funktion**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Security" > "Firewall" > "MAC-Regeln".

Tabelle 8-40 Verfügbare Dienste und Richtungen

| Dienst                          | Von Station nach Extern | Von Extern nach Station | Bedeutung   |
|---------------------------------|-------------------------|-------------------------|---|
| Erlaube MAC-Ebene-Kommunikation | x                       | x                       | Der MAC-Verkehr von extern zur Station und umgekehrt ist zugelassen.            |
| Erlaube ISO-Kommunikation       | x                       | x                       | Der ISO-Verkehr von extern zur Station und umgekehrt ist zugelassen.            |
| Erlaube SiClock                 | x                       | x                       | SiClock-Uhrzeittelegramme von extern zur Station und umgekehrt sind zugelassen. |

Tabelle 8-41 Logging

| Option                                     | Aktion bei Aktivierung  | Zutreffende Firewall-Regel |         |         |
|--|---|----------------------------|---------|---------|
|  |   | Aktion                     | Von     | Nach    |
| <b>MAC-Log-Einstellungen</b>               |   |                            |         |         |
| Aufzeichnen getunnelter Pakete             | Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle MAC-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt. | Allow                      | Station | Tunnel  |
|  |   | Allow                      | Tunnel  | Station |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt.  | Drop                       | Extern  | Station |
| Aufzeichnen blockierter ausgehender Pakete | Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.  | Drop                       | Station | Extern  |



**Hinweis**

**Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln**

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

**IP-Paketfilter-Richtungen S7-300-/S7-400-/PC-CPs**

**Bedeutung**

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den IP-Regeln des erweiterten Firewall-Modus.

**Folgende Richtungen sind verfügbar**

| Auswahlmöglichkeiten / Wertebereiche |         | Security-Modul |         | Bedeutung   |
|--------------------------------------|---------|----------------|---------|---|
| Von                                  | Nach    | CP x43-1 Adv.  | CP 1628 |   |
| Intern                               | Station | x              | -       | Zugriff vom internen Netz auf die Station.                                    |
|                                      | Any     | x              | -       | Zugriff vom internen auf das externe Netz, VPN-Tunnelpartner und die Station. |
| Extern                               | Station | x              | x       | Zugriff vom externen Netz auf die Station.                                    |
|                                      | Any     | x              | -       | Zugriff vom externen auf das interne Netz und die Station.                    |
| Station                              | Intern  | x              | -       | Zugriff von der Station auf das interne Netz.                                 |
|                                      | Extern  | x              | x       | Zugriff von der Station auf das externe Netz.                                 |
|                                      | Tunnel  | x              | x       | Zugriff von der Station auf VPN-Tunnelpartner.                                |
| Tunnel                               | Station | x              | x       | Zugriff über VPN-Tunnelpartner auf die Station.                               |
|                                      | Any     | x              | -       | Zugriff von VPN-Tunnelpartnern auf das interne Netz und die Station.          |
| Any                                  | Extern  | x              | -       | Zugriff vom internen Netz und der Station auf das externe Netz.               |

**MAC-Paketfilter-Richtungen S7-300-/S7-400-/PC-CPs**

**Bezug**

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den MAC-Regeln des erweiterten Firewall-Modus.

**Folgende Richtungen sind verfügbar**

| Auswahlmöglichkeiten / Wertebereiche |         | Security-Modul |         | Bedeutung                                       |
|--------------------------------------|---------|----------------|---------|---|
| Von                                  | Nach    | CP x43-1 Adv.  | CP 1628 |   |
| Extern                               | Station | x              | x       | Zugriff vom externen Netz auf die Station.      |
| Station                              | Extern  | x              | x       | Zugriff von der Station auf das externe Netz.   |
|                                      | Tunnel  | x              | x       | Zugriff von der Station auf VPN-Tunnelpartner.  |
| Tunnel                               | Station | x              | x       | Zugriff über VPN-Tunnelpartner auf die Station. |

**Zugriffsliste projektieren**

**Modulspezifische Funktion**

Diese Funktion ist für CP 1628 nicht verfügbar.

**Bedeutung**

Mit Hilfe von IP-Zugriffslisten stellen Sie den Zugriffsschutz für bestimmte IP-Adressen ein. Bereits in den lokalen Einstellungen des CPs angelegte Listeneinträge mit den entsprechenden Rechten werden unter dem Eintrag "Firewall" > "IP-Regeln" (erweiterter Firewall-Modus) angezeigt.

---

**Hinweis**

**Geändertes Verhalten nach Aktivierung von Security**

- Nach der Aktivierung von Security für einen CP wirkt der Zugriffsschutz nur noch an der externen Schnittstelle. Damit der Zugriffsschutz auch an der internen Schnittstelle wirkt, konfigurieren Sie im erweiterten Firewall-Modus die entsprechenden Firewall-Regeln.
  - Der CP antwortet auch auf ARP-Anfragen von nicht freigegebenen IP-Adressen (Layer 2).
  - Wenn die IP-Zugriffsliste eines CPs keine Einträge enthält und Security für den CP aktiviert wird, wird die Firewall aktiviert und auf den CP kann von extern nicht mehr zugegriffen werden. Damit der CP erreichbar ist, konfigurieren Sie im erweiterten Firewall-Modus die entsprechenden Firewall-Regeln.
- 

**Auswirkung von Einträgen der IP-Zugriffsliste beim Aktivieren von Security**

Wenn Security in den lokalen Einstellungen eines CPs aktiviert wird, werden im erweiterten Firewall-Modus die entsprechenden Regeln angelegt. Wenn in der Zugriffsliste eine IP-Adresse angegeben wurde, wird für diese IP-Adresse eine Firewall-Regel "Allow" > "Extern" > "Station" angelegt. Als Quell-IP-Adresse wird hierbei die IP-Adresse aus der IP-Zugriffsliste übernommen. IP-Adressen aus einem definierten IP-Adressbereich werden ebenfalls in entsprechende Firewall-Regeln umgesetzt.

## Voraussetzungen zur Bearbeitung

Bevor die angelegten Firewall-Regeln bearbeitet werden können, muss folgende Voraussetzung erfüllt sein:

- für die Bearbeitung über STEP 7: Projektierungsrecht "Security konfigurieren".
- für die Bearbeitung über einen Webserver: Baugruppenrecht "Web: IP Access Control-Liste erweitern".

Die Voraussetzungen zur Bearbeitung der IP-Zugriffslisten außerhalb der lokalen Security-Einstellungen werden in den Kapiteln der jeweiligen CPs beschrieben.

## Verbindungsbezogene automatische Firewall-Regeln

### Bedeutung

Für Verbindungen, die über CPs projiziert wurden, werden von STEP 7 automatisch Firewall-Regeln angelegt, die die Kommunikation mit dem jeweiligen Partner des CPs in der vorgegebenen Richtung (CP aktiv/passiv) freischalten. Dabei werden die Aufbaurichtungen der Verbindungen beachtet. Zur Anzeige dieser Firewall-Regeln muss bei aktiviertem erweitertem Firewall-Modus die Schaltfläche "Verbindungsregeln aktualisieren" betätigt werden. Die Firewall-Regeln werden daraufhin im erweiterten Firewall-Modus angezeigt.

---

#### Hinweis

##### UDP-Multicast- und UDP-Broadcast-Verbindungen manuell freigeben

Für UDP-Multicast- und UDP-Broadcast-Verbindungen werden keine automatischen Firewall-Regeln angelegt. Um die Verbindungen freizugeben, fügen Sie die entsprechenden Firewall-Regeln im erweiterten Firewall-Modus manuell hinzu.

---

Je nachdem, wie der Verbindungsaufbau projiziert ist, werden die folgenden Ebene 3-Firewall-Regeln angelegt. Befindet sich das Security-Modul in einer VPN-Gruppe, wechselt die Richtung "Extern" in "Tunnel". Dies gilt nur für CPs, die VPN unterstützen.

In die Spalte "Quell-IP-Adresse" bzw. "Ziel-IP-Adresse" dieser Firewall-Regeln wird jeweils die IP-Adresse des Verbindungspartners eingetragen.

| CP->extern       | Aktion | Von     | Nach    |
|------------------|--------|---------|---------|
| aktiv            | Allow  | Station | Extern  |
|                  | Drop   | Extern  | Station |
| passiv           | Drop   | Station | Extern  |
|                  | Allow  | Extern  | Station |
| aktiv und passiv | Allow  | Extern  | Station |
|                  | Allow  | Station | Extern  |

| CP->intern | Aktion | Von     | Nach    |
|------------|--------|---------|---------|
| aktiv      | Allow  | Station | Intern  |
|            | Drop   | Intern  | Station |

| CP->intern       | Aktion | Von     | Nach    |
|------------------|--------|---------|---------|
| passiv           | Drop   | Station | Intern  |
|                  | Allow  | Intern  | Station |
| aktiv und passiv | Allow  | Intern  | Station |
|                  | Allow  | Station | Intern  |

Für Ebene-2-Verbindungen werden "Allow"-Regeln für beide Richtungen angelegt. Befindet sich das Security-Modul in einer VPN-Gruppe, wechselt die Richtung "Extern" in "Tunnel". Dies gilt nur für CPs, die VPN unterstützen.

In die Spalte "Quell-MAC-Adresse" bzw. "Ziel-MAC-Adresse" dieser Firewall-Regeln wird jeweils die MAC-Adresse des Verbindungspartners eingetragen.

| CP->extern                      | Aktion | Von     | Nach    |
|---------------------------------|--------|---------|---------|
| aktiv, passiv, aktiv und passiv | Allow  | Station | Extern  |
|                                 | Allow  | Extern  | Station |

**Hinweis**

**Verändern der Verbindungsprojektierung**

Durch das Verändern der Verbindungsprojektierung von CPs ändern sich auch die verbindungsbezogenen Firewall-Regeln. Um die geänderten Firewall-Regeln anzuzeigen, müssen Sie erneut die Schaltfläche "Verbindungsregeln aktualisieren" betätigen.

**Vereinbarungen für automatisch angelegte Firewall-Regeln**

- **Priorität**  
Die Regeln haben die höchste Priorität und werden daher in der lokalen Regelliste oben eingefügt.
- **Regeln löschen**  
Die Regeln können nicht gelöscht werden. Das Logging kann aktiviert und Dienste können zugewiesen werden. Außerdem kann eine Bandbreite und ein Kommentar eingefügt werden.
- **Aktion umstellen**  
Stellen Sie die Aktion von "Allow" auf "Drop" oder umgekehrt um, werden diese beim erneuten Systemabgleich wieder überschrieben. Sollen die vorgenommenen Änderungen bestehen bleiben, wählen Sie als Aktion "Allow\*" oder "Drop\*". In diesem Fall wird nur die IP-Adresse abgeglichen und Aktion und Richtung bleiben wie eingestellt bestehen. Einstellungen zu Logging, Dienst, Bandbreite und Kommentar bleiben bei einem erneuten Systemabgleich auch ohne das Umstellen der Aktion auf "Allow\*" bzw. "Drop\*" bestehen. Wird die projektierte Verbindung gelöscht, werden die entsprechenden Regeln aus der Liste entfernt.

**Security-Modul in VPN-Gruppe**

Standardmäßig wird das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Deaktivieren Sie das Kontrollkästchen, kann zusätzlich zur Tunnelkommunikation

zwischen Tunnelpartnern Kommunikation mit weiteren Netzwerkteilnehmern aufgebaut werden, zu welchen keine Tunnel bestehen.

- Die Kommunikation läuft ungetunnelt, wenn die Partneradresse zu einer in STEP 7 bekannten Station gehört, zu der kein VPN-Tunnel projektiert ist.
- Die Kommunikation läuft durch den VPN-Tunnel, wenn die Partneradresse ein VPN-Endpunkt ist.
- Kann nicht eindeutig zugeordnet werden, ob eine Verbindung innerhalb oder außerhalb des VPN-Tunnels laufen soll, wird die Verbindung dem VPN-Tunnel zugeordnet und ein entsprechender Hinweis wird angezeigt. Die Zuordnung kann im erweiterten Firewall-Modus, z. B. durch Ändern der "Von"-Richtung "Tunnel" auf "Extern", angepasst werden. Damit diese Anpassung bei einem erneuten Systemabgleich nicht wieder überschrieben wird, muss die Aktion "Allow\*" bzw. "Drop\*" ausgewählt werden.

---

**Hinweis**

Soll sichergestellt werden, dass ausschließlich eine Kommunikation durch den Tunnel möglich ist, müssen Sie im erweiterten Firewall-Modus entsprechende Firewall-Regeln, z. B. für interne Teilnehmer oder NDIS-Adressen, anlegen.

Um für einen CP ausschließlich getunnelte Kommunikation zuzulassen, fügen Sie eine Regel "Drop" > "Any" > "Extern" ein. Für CP 1628 fügen Sie eine Regel "Drop" > "Station" > "Extern" ein. Zusätzlich müssen Sie bereits bestehende Firewall-Regeln, die ungetunnelte Kommunikation erlauben, entfernen.

---

## SNMP konfigurieren

### Übersicht SNMP

#### Was ist SNMP?

Das Security-Modul unterstützt die Übertragung von Managementinformationen über das Simple Network Management Protocol (SNMP). Dafür ist auf dem Security-Modul ein SNMP-Agent installiert, der die SNMP-Anfragen entgegennimmt und beantwortet. Informationen über die Eigenschaften von SNMP-fähigen Geräten sind in sogenannten MIB-Dateien (Management Information Base) hinterlegt, für die der Benutzer die notwendigen Rechte haben muss.

Beim SNMPv1 wird der "Community String" mitgesendet. Der Community String ist wie ein Passwort, welches zusammen mit der SNMP-Anfrage verschickt wird. Ist der Community String korrekt, antwortet das Security-Modul mit der angeforderten Information. Ist der String falsch, verwirft das Security-Modul die Anfrage und antwortet nicht. Der Community String wird bei SNMPv1 unverschlüsselt übertragen.

Mit SNMPv3 können die Daten verschlüsselt übertragen werden.

## SNMP konfigurieren - Eintrag "SNMP"

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "SNMP".
3. Aktivieren Sie das Kontrollkästchen "SNMP aktivieren".
4. Wählen Sie eine der folgenden SNMP-Protokollversionen aus:

---

#### Hinweis

##### Verschlüsselte Datenübertragung bei SNMPv3

Um die Sicherheit zu erhöhen, sollten Sie SNMPv3 verwenden, da die Daten hierbei verschlüsselt übertragen werden.

---

- SNMPv1  
Das Security-Modul verwendet zur Steuerung der Zugriffsrechte im SNMP-Agent folgende Standardwerte für die Community Strings:  
Für Lesezugriff: public  
Für Lese- und Schreibzugriff: private  
Um den Schreibzugriff über SNMP zu aktivieren, wählen Sie die Option "Erlaube schreibenden Zugriff".
- SNMPv3  
Wählen Sie entweder nur einen Authentifizierungs-Algorithmus oder einen Authentifizierungs-Algorithmus und einen Verschlüsselungs-Algorithmus aus.  
Authentifizierungs-Algorithmus: keine, MD5, SHA-1  
Verschlüsselungs-Algorithmus: keine, AES-128, DES

---

#### Hinweis

##### Verwendung von DES vermeiden

Bei DES handelt es sich um einen unsicheren Verschlüsselungsalgorithmus. Er sollte deshalb nur aus Gründen der Abwärtskompatibilität verwendet werden.

---

5. Wenn SNMPv3 verwendet werden soll, weisen Sie einem Benutzer eine Rolle zu, bei der die entsprechenden SNMP-Rechte aktiviert sind, damit er das Modul über SNMP erreichen kann. Eine Übersicht der SNMP-Rechte finden Sie in Kapitel: Rechte verwalten (Seite 525)

## Webserver auf CP x43-1 Advanced aktivieren

### Modulspezifische Funktion

Diese Funktion ist nur für CP x43-1 Advanced verfügbar.

## **Bedeutung**

Nach der Aktivierung des Webservers erhalten Sie Zugriff auf die Webseiten der Baugruppe. In den lokalen Security-Einstellungen können Sie den Zugriff auf diese Webseiten auf das HTTPS-Protokoll beschränken. Dieser Zugriff wird über das Kontrollkästchen "Zugriff nur über HTTPS zulassen" gesteuert. Zusätzlich müssen Sie die Firewall passend konfigurieren.

## **IPSec-Tunnel: Gruppen anlegen und zuordnen**

## **Interne Netzknoten konfigurieren - Eintrag "Knoten"**

## **Zugriff auf S7-300-/S7-400-CPs für VPN-Verbindungspartner erlauben**

## **Auswahlmöglichkeiten**

Wählen Sie aus, ob die VPN-Verbindungspartner im Routing-Modus (SCALANCE S / M) auf den CP und/oder auf das interne Subnetz des CPs Zugriff haben dürfen.

## **So erreichen Sie diese Funktion**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Knoten".
3. Konfigurieren Sie den Zugriff für die VPN-Verbindungspartner des CPs im Routing-Modus (SCALANCE S / M):
  - Verbindung erlauben zum CP (GBit-Schnittstelle)
  - Verbindung erlauben zum internen Subnetz (PROFINET-Subnetz)

## **Durch Tunnel erreichbare NDIS-Knoten bei PC-CPs manuell konfigurieren**

## **Durch Tunnel erreichbare NDIS-Knoten konfigurieren**

Die internen Knoten werden gelernt und dynamisch den Routen zugeordnet. Dies betrifft die NDIS-IP-Adressen des Windows-PCs.

## **Gehen Sie so vor**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Knoten" > "Durch Tunnel erreichbare NDIS-Knoten".
3. Geben Sie die NDIS-IP-Adressen ein.

## Online-Funktionen - Test / Diagnose und Logging

### Aktualisierte Firewall-Regeln - Eintrag "Dynamisch aktualisierte Firewall-Regeln"

#### Bedeutung

Anzeige der IP-Adressen bzw. der IP-Adressbereiche, die über HTTP oder HTTPS dynamisch freigeschaltet oder von einem Benutzer nachgeladen wurden. Für die freigeschalteten IP-Adressen werden die ihnen zugewiesenen Zugriffsrechte auf den S7-CP angezeigt. Eine Aktualisierung der IP-Adressen in diesem Register kann durch folgende Ereignisse erfolgen:

- Erweiterung/Veränderung der IP Access Control-Liste
- Aktualisierung der Firewall-Regeln
- Dynamische Erweiterungen, die vom CP zur Laufzeit eingetragen werden, z.B. PROFINET IO-Devices

Da hier nur die dynamisch aktualisierten Firewall-Regeln angezeigt werden, müssen in eine vollständige Betrachtung des aktuellen Firewall-Zustands der Baugruppe auch diejenigen Firewall-Regeln einbezogen werden, die offline projektiert und auf die Station geladen wurden.

## Security für S7-1500-CPs

### Firewall einrichten

### Lokale Firewall-Regeln für S7-1500-CPs

### Übersicht Lokale Firewall-Regeln S7-1500-CPs

### Paketfilter-Regeln aktivieren

Wenn Sie in den lokalen Security-Einstellungen für die CPs die Security-Funktion aktivieren, sind zunächst alle Zugriffe auf und über den CP zugelassen. Um einzelne Paketfilter-Regeln zu aktivieren, aktivieren Sie das Kontrollkästchen "Firewall aktivieren". Schalten Sie anschließend die gewünschten Dienste frei. Firewall-Regeln, die aufgrund einer Verbindungsprojektierung automatisch angelegt werden, haben Vorrang vor den manuell eingestellten Regeln.

---

#### Hinweis

##### **Detaillierte Firewall-Einstellungen im erweiterten Firewall-Modus**

Im erweiterten Firewall-Modus können Sie Firewall-Regeln auf einzelne Teilnehmer beschränken. Um in den erweiterten Firewall-Modus zu wechseln, aktivieren Sie das Kontrollkästchen "Firewall im erweiterten Modus aktivieren".

---



## Verbindungsregeln aktualisieren

Durch das Verändern der Verbindungsprojektierung von CPs ändern sich auch die verbindungsbezogenen Firewall-Regeln. Um die geänderten Firewall-Regeln anzuzeigen, müssen Sie erneut die Schaltfläche "Verbindungsregeln aktualisieren" betätigen. Die geänderten Firewall-Regeln werden dann im erweiterten Firewall-Modus angezeigt.

## Firewall mit vordefinierten Firewall-Regeln projektieren - CP 1543-1

## Firewall mit vordefinierten IP-Regeln projektieren - CP 1543-1

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Vordefinierte IP-Regeln".

Tabelle 8-42 Verfügbare Dienste und Richtungen

| Dienst                              | Von Extern nach Station | Freigegebene Ports                   | Bedeutung   |
|-------------------------------------|-------------------------|--------------------------------------|---|
| Erlaube IP-Kommunikation            | x                       | alle                                 | Der IP-Verkehr von Extern nach Station ist zugelassen.                      |
| Erlaube S7-Protokoll                | x                       | TCP-Port 102                         | Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen. |
| Erlaube FTP/FTPS (expliziter Modus) | x                       | TCP-Port 20<br>TCP-Port 21           | Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.            |
| Erlaube HTTP                        | x                       | TCP-Port 80                          | Zur Kommunikation mit einem Webserver.                                      |
| Erlaube HTTPS                       | x                       | TCP-Port 443                         | Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.  |
| Erlaube SNMP                        | x                       | TCP-Port 161/162<br>UDP-Port 161/162 | Zur Überwachung von SNMP-fähigen Netzteilnehmern.                           |
| Erlaube Security-Diagnose           | x                       | TCP-Port 8448                        | Security-Diagnose erlauben/nicht erlauben.                                  |

Tabelle 8-43 Logging

| Option                                     | Aktion bei Aktivierung   | Zutreffende Firewall-Regel |         |         |
|--|--|----------------------------|---------|---------|
|  |  | Aktion                     | Von     | Nach    |
| <b>IP-Log-Einstellungen</b>                |  |                            |         |         |
| Aufzeichnen getunnelter Pakete             | Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt. | Allow                      | Station | Tunnel  |
|  |  | Allow                      | Tunnel  | Station |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.  | Drop                       | Extern  | Station |

**Hinweis**

**Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln**

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

**Firewall mit vordefinierten IPv6-Regeln projektieren - CP1543-1**

**Bedeutung**

Die vordefinierten IPv6-Regeln bieten die Möglichkeit, die Firewall bezüglich Diensten zu konfigurieren, in denen IPv6 verwendet wird. Durch das Freischalten einer vordefinierten IPv6-Regel in den lokalen Security-Einstellungen des CP 1543-1 V1.1 werden gleichzeitig die systemdefinierten ICMPv6-Dienste in der Firewall freigeschalten, die in den globalen Security-Einstellungen unter "Firewall" > "Dienste" > "Dienste für IP-Regeln definieren", Register "ICMP" ersichtlich sind. Die Firewall des CP 1543-1 V1.0 lässt ICMPv6-Pakete auch ohne das Freischalten einer vordefinierten IPv6-Regel passieren.

**So erreichen Sie diese Funktion**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "Firewall" > "Vordefinierte IPv6-Regeln".

Tabelle 8-44 Verfügbare Dienste und Richtungen

| Dienst                              | Von Extern nach Station | Freigegebene Ports                   | Bedeutung   |
|-------------------------------------|-------------------------|--------------------------------------|---|
| Erlaube IP-Verkehr                  | x                       | alle                                 | Der IP-Verkehr von Extern nach Station ist zugelassen.                      |
| Erlaube S7-Protokoll                | x                       | TCP-Port 102                         | Kommunikation der Netzwerkteilnehmer über das S7-Protokoll wird zugelassen. |
| Erlaube FTP/FTPS (expliziter Modus) | x                       | TCP-Port 20<br>TCP-Port 21           | Zur Dateiverwaltung und Dateizugriff zwischen Server und Client.            |
| Erlaube HTTP                        | x                       | TCP-Port 80                          | Zur Kommunikation mit einem Webserver.                                      |
| Erlaube HTTPS                       | x                       | TCP-Port 443                         | Zur gesicherten Kommunikation mit einem Webserver, z. B. zur Web-Diagnose.  |
| Erlaube SNMP                        | x                       | TCP-Port 161/162<br>UDP-Port 161/162 | Zur Überwachung von SNMP-fähigen Netzteilnehmern.                           |
| Erlaube Security-Diagnose           | x                       | TCP-Port 8448                        | Security-Diagnose über IPv6 erlauben/nicht erlauben.                        |

Tabelle 8-45 Logging

| Option                                     | Aktion bei Aktivierung   | Zutreffende Firewall-Regel |         |         |
|--|--|----------------------------|---------|---------|
|  |  | Aktion                     | Von     | Nach    |
| <b>IP-Log-Einstellungen</b>                |  |                            |         |         |
| Aufzeichnen getunnelter Pakete             | Nur aktiv, wenn das Security-Modul Teilnehmer einer VPN-Gruppe ist. Alle IP-Pakete, die über den Tunnel weitergeleitet werden, werden geloggt. | Allow                      | Station | Tunnel  |
|  |  | Allow                      | Tunnel  | Station |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden IP-Pakete, die verworfen werden, werden geloggt.  | Drop                       | Extern  | Station |

**Hinweis****Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln**

Log-Einstellungen, die unter "Vordefinierte IPv6-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

**Firewall mit vordefinierten MAC-Regeln projektieren - CP 1543-1****So erreichen Sie diese Funktion**

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie den Eintrag "Firewall" > "Vordefinierte MAC-Regeln".

Tabelle 8-46 Verfügbare Dienste und Richtungen

| Dienst                    | Von Station nach extern | Von extern nach Station | Freigegebene Ports | Bedeutung  |
|---------------------------|-------------------------|-------------------------|--------------------|--|
| Erlaube MAC-Kommunikation | x                       | x                       | -                  | Der MAC-Verkehr von extern zur Station und umgekehrt ist zugelassen.     |
| Erlaube ISO-Protokoll     | x                       | x                       | -                  | Der ISO-Verkehr von extern zur Station und umgekehrt ist zugelassen.     |
| Erlaube SiCLOCK           | x                       | x                       | -                  | Der SiCLOCK-Verkehr von extern zur Station und umgekehrt ist zugelassen. |

Tabelle 8-47 Logging

| Option                                     | Aktion bei Aktivierung   | Zutreffende Firewall-Regel |         |         |
|--|--|----------------------------|---------|---------|
|  |  | Aktion                     | Von     | Nach    |
| MAC-Log-Einstellungen                      |  |                            |         |         |
| Aufzeichnen blockierter eingehender Pakete | Alle eintreffenden MAC-Pakete, die verworfen werden, werden geloggt. | Drop                       | Extern  | Station |
| Aufzeichnen blockierter ausgehender Pakete | Alle ausgehenden MAC-Pakete, die verworfen werden, werden geloggt.   | Drop                       | Station | Extern  |

**Hinweis**

**Zusammenhang zwischen Log-Einstellungen im Standard Modus und Firewall-Regeln**

Log-Einstellungen, die unter "Vordefinierte IP-Regeln" bzw. "Vordefinierte MAC-Regeln" vorgenommen werden, wirken nicht für Firewall-Regeln, die durch eine Verbindungsprojektierung automatisch erzeugt wurden. So können beispielsweise getunnelte Telegramme einer projektierten Verbindung nicht geloggt werden. Im erweiterten Firewall-Modus kann das Logging auf die automatisch erzeugten Firewall-Regeln von Verbindungen ausgeweitet werden.

**IP-Paketfilter-Richtungen CP 1543-1**

**Bedeutung**

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den IP-Regeln des erweiterten Firewall-Modus.

**Folgende Richtungen sind verfügbar**

| Auswahlmöglichkeiten / Wertebereiche |         | Bedeutung  |
|--------------------------------------|---------|--|
| Von                                  | Nach    |  |
| Extern                               | Station | Zugriff vom externen Netz auf die Station.       |
| Station                              | Extern  | Zugriff von der Station auf das externe Netz.    |
|                                      | Tunnel  | Zugriff von der Station auf VPN-Tunnelpartner.*  |
| Tunnel                               | Station | Zugriff von VPN-Tunnelpartnern auf die Station.* |

\* Nur für CP 1543-1 V1.1.

## MAC-Paketfilter-Richtungen CP 1543-1

### Bedeutung

Auswahlmöglichkeiten für die Kommunikationsrichtungen "Von" und "Nach" in den MAC-Regeln des erweiterten Firewall-Modus.

### Folgende Richtungen sind verfügbar

| Auswahlmöglichkeiten / Wertebereiche |             | Bedeutung  |
|--------------------------------------|-------------|--|
| <b>Von</b>                           | <b>Nach</b> |  |
| Extern                               | Station     | Zugriff vom externen Netz auf die Station.       |
| Station                              | Extern      | Zugriff von der Station auf das externe Netz.    |
|                                      | Tunnel      | Zugriff von der Station auf VPN-Tunnelpartner.*  |
| Tunnel                               | Station     | Zugriff von VPN-Tunnelpartnern auf die Station.* |

\* Nur für CP 1543-1 V1.1.

## Verbindungsbezogene automatische Firewall-Regeln

### Bedeutung

Für Verbindungen, die über CPs projektiert wurden, werden von STEP 7 automatisch Firewall-Regeln angelegt, die die Kommunikation mit dem jeweiligen Partner des CPs in der vorgegebenen Richtung (CP aktiv/passiv) freischalten. Dabei werden die Aufbaurichtungen der Verbindungen beachtet. Zur Anzeige dieser Firewall-Regeln muss bei aktiviertem erweitertem Firewall-Modus die Schaltfläche "Verbindungsregeln aktualisieren" betätigt werden. Die Firewall-Regeln werden daraufhin im erweiterten Firewall-Modus angezeigt.

#### Hinweis

#### UDP-Multicast- und UDP-Broadcast-Verbindungen manuell freigeben

Für UDP-Multicast- und UDP-Broadcast-Verbindungen werden keine automatischen Firewall-Regeln angelegt. Um die Verbindungen freizugeben, fügen Sie die entsprechenden Firewall-Regeln im erweiterten Firewall-Modus manuell hinzu.

Je nachdem, wie der Verbindungsaufbau projektiert ist, werden die folgenden Ebene 3-Firewall-Regeln angelegt. Befindet sich der CP 1543-1 V1.1 in einer VPN-Gruppe, wechselt die Richtung "Extern" in "Tunnel".

In die Spalte "Quell-IP-Adresse" bzw. "Ziel-IP-Adresse" dieser Firewall-Regeln wird jeweils die IP-Adresse des Verbindungspartners eingetragen.

| CP->extern | Aktion | Von     | Nach    |
|------------|--------|---------|---------|
| aktiv      | Drop   | Extern  | Station |
|            | Allow  | Station | Extern  |

| CP->extern       | Aktion | Von     | Nach    |
|------------------|--------|---------|---------|
| passiv           | Drop   | Station | Extern  |
|                  | Allow  | Extern  | Station |
| aktiv und passiv | Allow  | Extern  | Station |
|                  | Allow  | Station | Extern  |

Für Ebene-2-Verbindungen werden "Allow"-Regeln für beide Richtungen angelegt. Befindet sich der CP 1543-1 V1.1 in einer VPN-Gruppe, wechselt die Richtung "Extern" in "Tunnel".

In die Spalte "Quell-MAC-Adresse" bzw. "Ziel-MAC-Adresse" dieser Firewall-Regeln wird jeweils die MAC-Adresse des Verbindungspartners eingetragen.

| CP->extern                      | Aktion | Von     | Nach    |
|---------------------------------|--------|---------|---------|
| aktiv, passiv, aktiv und passiv | Allow  | Station | Extern  |
|                                 | Allow  | Extern  | Station |

**Hinweis**

**Verändern der Verbindungsprojektierung**

Durch das Verändern der Verbindungsprojektierung von CPs ändern sich auch die verbindungsbezogenen Firewall-Regeln. Um die geänderten Firewall-Regeln anzuzeigen, müssen Sie erneut die Schaltfläche "Verbindungsregeln aktualisieren" betätigen.

**Vereinbarungen für automatisch angelegte Firewall-Regeln**

- **Priorität**  
Die Regeln haben die höchste Priorität und werden daher in der lokalen Regelliste oben eingefügt.
- **Regeln löschen**  
Die Regeln können nicht gelöscht werden. Das Logging kann aktiviert und Dienste können zugewiesen werden. Außerdem kann eine Bandbreite und ein Kommentar eingefügt werden.
- **Aktion umstellen**  
Stellen Sie die Aktion von "Allow" auf "Drop" oder umgekehrt um, werden diese beim erneuten Systemabgleich wieder überschrieben. Sollen die vorgenommenen Änderungen bestehen bleiben, wählen Sie als Aktion "Allow\*" oder "Drop\*". In diesem Fall wird nur die IP-Adresse abgeglichen und Aktion und Richtung bleiben wie eingestellt bestehen. Einstellungen zu Logging, Dienst, Bandbreite und Kommentar bleiben bei einem erneuten Systemabgleich auch ohne das Umstellen der Aktion auf "Allow\*" bzw. "Drop\*" bestehen. Wird die projektierte Verbindung gelöscht, werden die entsprechenden Regeln aus der Liste entfernt.

**Security-Modul in VPN-Gruppe**

Standardmäßig wird das Kontrollkästchen "Ausschließlich getunnelte Kommunikation" aktiviert. Deaktivieren Sie das Kontrollkästchen, kann zusätzlich zur Tunnelkommunikation

zwischen Tunnelpartnern Kommunikation mit weiteren Geräten aufgebaut werden, zu welchen keine Tunnel bestehen.

- Die Kommunikation läuft ungetunnelt, wenn die Partneradresse zu einer in STEP 7 bekannten Station gehört, zu der kein VPN-Tunnel projektiert ist.
- Die Kommunikation läuft durch den VPN-Tunnel, wenn die Partneradresse ein VPN-Endpunkt ist.
- Kann nicht eindeutig zugeordnet werden, ob eine Verbindung innerhalb oder außerhalb des VPN-Tunnels laufen soll, wird die Verbindung dem VPN-Tunnel zugeordnet und ein entsprechender Hinweis wird angezeigt. Die Zuordnung kann im erweiterten Firewall Modus, z. B. durch Ändern der "Von"-Richtung "Tunnel" auf "Extern", angepasst werden. Damit diese Anpassung bei einem erneuten Systemabgleich nicht wieder überschrieben wird, muss die Aktion "Allow\*" bzw. "Drop\*" ausgewählt werden.

---

**Hinweis**

Soll sichergestellt werden, dass ausschließlich eine Kommunikation durch den Tunnel möglich ist, müssen Sie im erweiterten Firewall-Modus entsprechende Firewall-Regeln anlegen.

Um für einen CP ausschließlich getunnelte Kommunikation zuzulassen, fügen Sie eine Regel "Drop" > "Any" > "Extern" ein. Zusätzlich müssen Sie bereits bestehende Firewall-Regeln, die ungetunnelte Kommunikation erlauben, entfernen.

---

## SNMP konfigurieren

### Übersicht SNMP

#### Was ist SNMP?

Das Security-Modul unterstützt die Übertragung von Managementinformationen über das Simple Network Management Protocol (SNMP). Dafür ist auf dem Security-Modul ein SNMP-Agent installiert, der die SNMP-Anfragen entgegennimmt und beantwortet. Informationen über die Eigenschaften von SNMP-fähigen Geräten sind in sogenannten MIB-Dateien (Management Information Base) hinterlegt, für die der Benutzer die notwendigen Rechte haben muss.

Beim SNMPv1 wird der "Community String" mitgesendet. Der Community String ist wie ein Passwort, welches zusammen mit der SNMP-Anfrage verschickt wird. Ist der Community String korrekt, antwortet das Security-Modul mit der angeforderten Information. Ist der String falsch, verwirft das Security-Modul die Anfrage und antwortet nicht. Der Community String wird bei SNMPv1 unverschlüsselt übertragen.

Mit SNMPv3 können die Daten verschlüsselt übertragen werden.

## SNMP konfigurieren - Eintrag "SNMP"

### So erreichen Sie diese Funktion

1. Markieren Sie das zu bearbeitende Modul.
2. Wählen Sie in den lokalen Security-Einstellungen den Eintrag "SNMP".
3. Aktivieren Sie das Kontrollkästchen "SNMP aktivieren".
4. Wählen Sie eine der folgenden SNMP-Protokollversionen aus:

---

#### Hinweis

##### Verschlüsselte Datenübertragung bei SNMPv3

Um die Sicherheit zu erhöhen, sollten Sie SNMPv3 verwenden, da die Daten hierbei verschlüsselt übertragen werden.

---

- SNMPv1  
Das Security-Modul verwendet zur Steuerung der Zugriffsrechte im SNMP-Agent folgende Standardwerte für die Community Strings:  
Für Lesezugriff: public  
Für Lese- und Schreibzugriff: private  
Um den Schreibzugriff über SNMP zu aktivieren, wählen Sie die Option "Erlaube schreibenden Zugriff".
- SNMPv3  
Wählen Sie entweder nur einen Authentifizierungs-Algorithmus oder einen Authentifizierungs-Algorithmus und einen Verschlüsselungs-Algorithmus aus.  
Authentifizierungs-Algorithmus: keine, MD5, SHA-1  
Verschlüsselungs-Algorithmus: keine, AES-128, DES

---

#### Hinweis

##### Verwendung von DES vermeiden

Bei DES handelt es sich um einen unsicheren Verschlüsselungsalgorithmus. Er sollte deshalb nur aus Gründen der Abwärtskompatibilität verwendet werden.

---

5. Wenn SNMPv3 verwendet werden soll, weisen Sie einem Benutzer eine Rolle zu, bei der die entsprechenden SNMP-Rechte aktiviert sind, damit er das Modul über SNMP erreichen kann. Eine Übersicht der SNMP-Rechte finden Sie in Kapitel: Rechte verwalten (Seite 525)



## 8.1.4 Konfigurationen erstellen

### 8.1.4.1 Wissenswertes zum Webserver

#### Einführung

Der Webserver gibt Ihnen die Möglichkeit, Ihre CPU über das Internet oder das firmeninterne Intranet zu beobachten. Auswertungen und Diagnose sind somit über große Entfernungen möglich.

Meldungen und Statusinformationen werden auf HTML-Seiten angezeigt.

#### Webbrowser

Für den Zugriff auf die HTML-Seiten der CPU benötigen Sie einen Webbrowser.

Folgende Webbrowser sind z. B. für die Kommunikation mit der CPU geeignet:

- Internet Explorer (ab Version 6.0)
- Mozilla Firefox (ab Version 1.5)
- Opera (ab Version 9.0)

#### Web-Zugriff auf die CPU über PG/PC

Um auf den Webserver zuzugreifen, gehen Sie folgendermaßen vor:

1. Verbinden Sie den Client (PG, PC) über die PROFINET-Schnittstelle mit der CPU.
2. Öffnen Sie den Webbrowser.  
Tragen Sie im Feld "Adresse" des Webbrowsers die IP-Adresse der CPU ein in der Form <http://ww.xx.yy.zz> (beispielhafte Eingabe: <http://192.168.3.141>).  
Die Startseite der CPU wird geöffnet. Von der Startseite aus können Sie zu den weiteren Informationen navigieren.

#### Weitere Informationen

Weiter Informationen über den Webserver der unterschiedlichen CPU-Familien finden Sie unter dem Stichwort "Webserver" im Informationssystem.

Informationen zum Erstellen eigener Webseiten für den Zugriff auf die CPU finden Sie unter dem Stichwort "Anwenderdefinierte Webseiten" im Informationssystem.

### 8.1.4.2 Wissenswertes zu Betriebsarten bei PROFIBUS-DP

#### Einführung

Als dezentrale Peripherie werden DP-Mastersysteme bezeichnet, die aus DP-Master und DP-Slaves bestehen, über einen Bus verbunden sind und miteinander über das Protokoll PROFIBUS-DP kommunizieren.

Im Folgenden geht es um kommunikationsfähige Baugruppen mit DP-Schnittstelle, die die Rolle DP-Master bzw. DP-Slave einnehmen können.

### Option "DP-Master" und "DP-Slave"

Kommunikationsfähige Baugruppen wie z. B. CPUs mit DP-Schnittstelle und CPs bzw. CMs mit DP-Schnittstelle haben in ihren Moduleigenschaften den Bereich "Betriebsart".

Bei S7-300-CPU's mit integrierter DP-Schnittstelle z. B. können Sie die Betriebsart "DP-Master" und "DP-Slave" einstellen. Eine CPU bzw. ein CP, der als DP-Slave parametrierbar ist, wird auch als Intelligenter DP-Slave (I-Slave) bezeichnet.

Bei S7-1500-CPU's mit integrierter DP-Schnittstelle ist nur die Betriebsart "DP-Master" möglich. Um S7-1500-CPU's als I-Slave betreiben zu können, müssen Sie das Kommunikationsmodul CM 1542-5 stecken und als DP-Slave parametrieren.

S7-1200-CPU's haben keine integrierten DP-Schnittstellen. Um eine S7-1200 als DP-Master oder als DP-Slave betreiben zu können, müssen Sie ein Kommunikationsmodul CM 1243-5 (nur DP-Master) bzw. ein Kommunikationsmodul CM 1242-5 (nur DP-Slave; I-Slave) stecken.

### Weitere Informationen

Weitere Informationen über die Dezentrale Peripherie finden Sie unter dem Stichwort "Dezentrale Peripherie" und "I-Slave" im Informationssystem.

#### 8.1.4.3 Automatisierungssysteme konfigurieren

### Baugruppen adressieren

### Adressierung von Baugruppen

### Einführung

In der Geräteübersicht sehen Sie in den Spalten E-Adresse und A-Adresse die Adressen oder Adressbereiche der Baugruppen. Daneben gibt es noch weitere Adressen, die im Folgenden erläutert werden.

### E/A-Adresse (Peripherieadresse)

E/A-Adressen (Ein/Ausgabeadressen) werden dazu benötigt, um im Anwenderprogramm Eingänge zu lesen bzw. Ausgänge zu setzen.

Ein- und Ausgangsadressen werden automatisch beim Stecken von Baugruppen auf den Baugruppenträger vergeben. Die Adresse des ersten Kanals bildet die Anfangsadresse einer Baugruppe. Die Adressen der übrigen Kanäle ergeben sich aus dieser Anfangsadresse. Das Adressende ergibt sich aus der baugruppenspezifischen Adresslänge.

## Teilnehmeradresse (z. B. Ethernet-Adresse)

Teilnehmeradressen sind Adressen von programmierbaren Baugruppen (Industrial Ethernet-Adressen). Sie werden benötigt, um die verschiedenen Teilnehmer eines Subnetzes zu adressieren, z. B. um ein Anwenderprogramm in eine CPU zu laden.

## HW-Kennung zur Identifikation von Baugruppen bzw. Submodulen

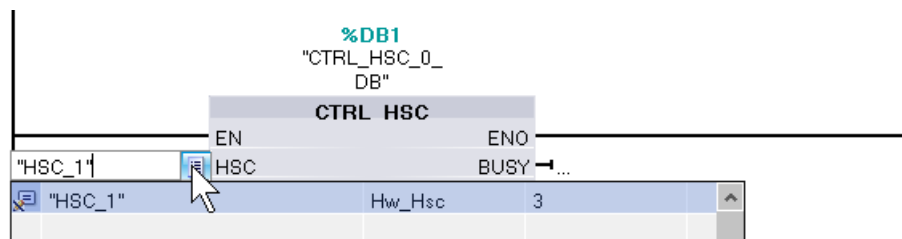
Zusätzlich zu den E-Adressen und A-Adressen wird automatisch eine HW-Kennung (HW-ID) vergeben, die zur Adressierung und Identifizierung der Baugruppe verwendet wird. Auch Submodule (Einheiten einer Baugruppe) z. B. ein integrierter Zähler erhalten eine solche HW-Kennung.

Die HW-Kennung besteht aus einer Ganzzahl und wird vom System bei Diagnosemeldungen mit ausgegeben, um die fehlerhafte Baugruppe bzw. das fehlerhafte Submodul zu lokalisieren.

Außerdem verwenden Sie die HW-Kennung für eine Reihe von Anweisungen, um die entsprechende Baugruppe zu adressieren.

Die HW-Kennung ist nicht änderbar.

Beispiel: Schneller Zähler der S7-1200 CPU identifizieren



Die HW-Kennung wird automatisch beim Stecken von Komponenten in der Geräte- oder Netzansicht vergeben und bei den PLC-Variablen (Register Systemkonstanten) eingetragen. Ein Name für die HW-Kennung wird ebenfalls automatisch vergeben. Die Systemkonstanten der PLC-Variablen sind ebenfalls nicht änderbar.

## Siehe auch

Ein- und Ausgangsadressen festlegen (Seite 629)

Adressen einer Verwendungsstelle im Programm zuweisen (Seite 630)

Einführung zum Laden einer Konfiguration (Seite 824)

## Ein- und Ausgangsadressen festlegen

Ein- und Ausgangsadressen werden automatisch voreingestellt. Sie können aber die Adresszuweisung nachträglich verändern.

Alle Adressen von Baugruppen liegen im Prozessabbild-Bereich. Das Prozessabbild wird automatisch zyklisch aktualisiert.

### Voraussetzung

Sie befinden sich in der Gerätesicht.

### Vorgehen

Um den voreingestellten Adressbereich zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Gerätesicht auf die Baugruppe, deren Anfangsadresse Sie einstellen wollen.
2. Navigieren Sie im Inspektorfenster unter "Eigenschaften" zu den "E/A-/Adressen".
3. Geben Sie unter "Anfangsadresse" die neue gewünschte Anfangsadresse ein.
4. Drücken Sie <Return> oder klicken auf ein beliebiges anderes Objekt um den geänderten Wert zu übernehmen.

Wenn Sie eine ungültige Adresse eingegeben haben, wird eine Meldung mit der nächsten freien Adresse eingeblendet.

---

#### Hinweis

Sie können die Adressen auch direkt in der Geräteübersicht ändern.

---

### Siehe auch

Eigenschaften und Parameter bearbeiten (Seite 408)

Ein- und Ausgangsadressen in der Adressübersicht (Seite 411)

### Adressen einer Verwendungsstelle im Programm zuweisen

Sie können Adressen von den E/A-Kanälen der Baugruppen direkt den Verwendungsstellen im Programm oder einer Variablentabelle zuordnen.

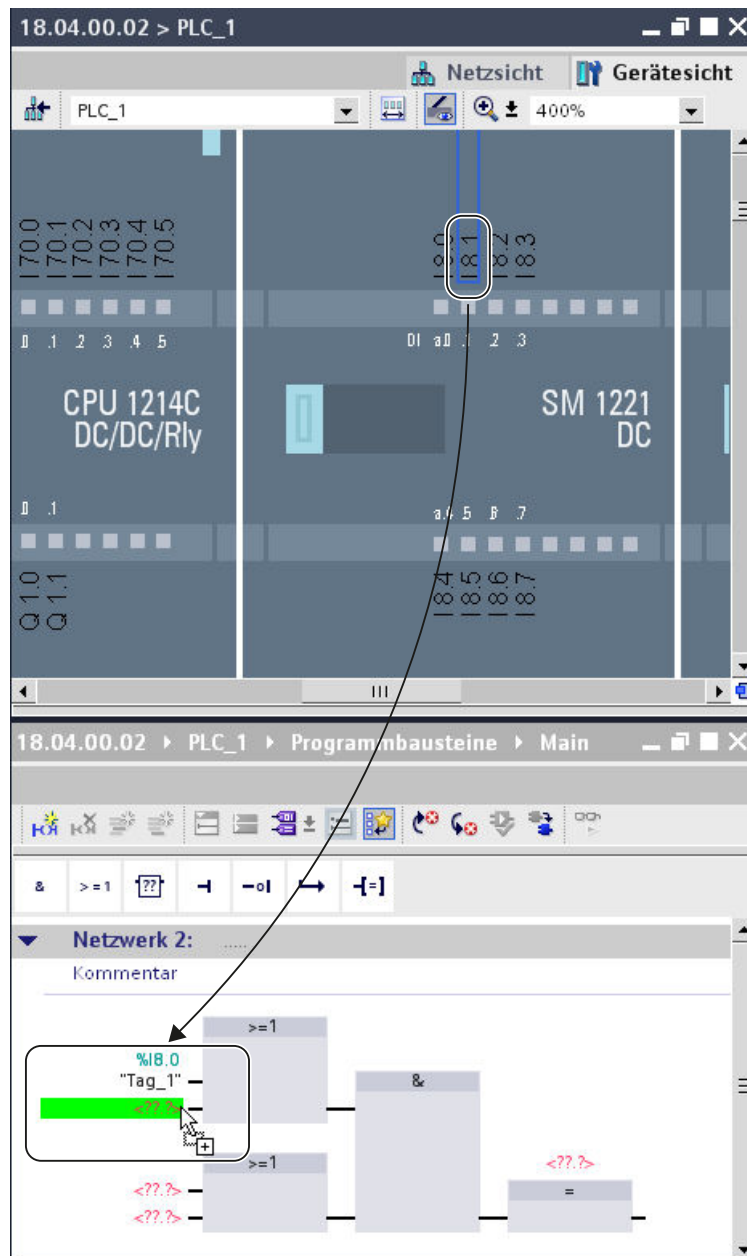
### Voraussetzung

- Die Gerätesicht des Hardware- und Netzwerkeditors ist geöffnet.
- Die Zoomstufe in der Gerätesicht muss auf mindestens 200 % eingestellt sein, damit Sie die einzelnen E/A-Kanäle sehen können.
- Das Anweisungsfenster des Programmiereditors oder eine Variablentabelle ist geöffnet.

## Vorgehen

Um E/A-Kanäle von Baugruppen den Verwendungsstellen im Programm oder einer Variablen-tabelle zuzuordnen, gehen Sie folgendermaßen vor:

1. Navigieren Sie in der Gerätesicht zu der Baugruppe mit dem gewünschten E/A-Kanal.
2. Ziehen Sie mit gedrückter Maustaste die gewünschte E/A-Adresse auf die entsprechende Verwendungsstelle des Bausteins oder in die Variablen-tabelle.



Die Adresse der Baugruppe wird der Verwendungsstelle im Programm zugewiesen, bzw. als Variable in die Variablen-tabelle eingetragen.

---

**Hinweis**

Die Variable für einen Ein- oder Ausgang eines Bausteins kann auch auf den Ein- oder Ausgang einer Baugruppe gezogen werden, um die Variable mit dem E/A-Kanal der Baugruppe zu verknüpfen.

---

## Signalboard

### Signalboard in CPU stecken

#### Einführung

Signalboards bieten Ihnen bei S7-1200 CPUs die Möglichkeit, die Anzahl der CPU-eigenen Ein- und Ausgänge zu erhöhen. Signalboards finden Sie wie alle anderen Hardware-Komponenten im Hardware-Katalog. Sie stecken Signalboards nicht wie andere Baugruppen in den Baugruppenträger, sondern direkt in einen CPU-eigenen Steckplatz.

Bei der Verwendung eines Signalboards ist folgendes zu beachten:

- Jede CPU kann nur ein Signalboard aufnehmen.
- Ein Signalboard kann nur gesteckt werden, wenn der Steckplatz der CPU frei ist.

Um ein Signalboard in eine CPU zu stecken, gibt es folgende Möglichkeiten:

- Doppelklick auf ein Signalboard im Hardware-Katalog bei freiem Steckplatz in der CPU
- Drag & Drop aus dem Hardware-Katalog auf einen freien Steckplatz in der CPU
- Kontextmenü eines Signalboards im Hardware-Katalog für "Kopieren" und "Einfügen"

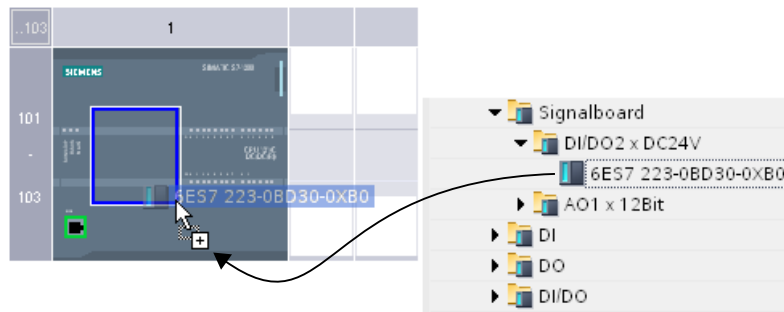
#### Voraussetzung

- Der Hardware-Katalog ist geöffnet.
- Die S7-1200 CPU hat einen freien Steckplatz für das Signalboard.

## Signalboard in CPU stecken

Um ein Signalboard in eine CPU zu stecken, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog zu dem gewünschten Signalboard.
2. Selektieren Sie das ausgewählte Signalboard.
3. Ziehen Sie das Signalboard mit Drag & Drop in den freien Steckplatz der CPU.



Sie haben nun das Signalboard in den Steckplatz der CPU gesteckt.

Wenn Sie sich in der Netzsicht befinden, können Sie auch ein Signalboard mit Drag & Drop auf ein Gerät ziehen. Wenn die CPU einen leeren Steckplatz für ein Signalboard hat, wird das Signalboard automatisch in diesen Steckplatz gesteckt.

## Konfigurationen für Webserver

### Wissenswertes zum Webserver

#### Einführung

Der Webserver gibt Ihnen die Möglichkeit, Ihre CPU über das Internet oder das firmeninterne Intranet zu beobachten. Auswertungen und Diagnose sind somit über große Entfernungen möglich.

Meldungen und Statusinformationen werden auf HTML-Seiten angezeigt.

#### Webbrowser

Für den Zugriff auf die HTML-Seiten der CPU benötigen Sie einen Webbrowser.

Folgende PC-Webbrowser sind z. B. für die Kommunikation mit der CPU geeignet:

- Internet Explorer (Version 8.0, 9.0)
- Mozilla Firefox (ab Version 17.0.1)
- Google Chrome 23.0
- Apple Safari 5.1.7 (Windows)
- Apple Safari 6.0.2 (Mac)

Folgende Webbrowser von mobilen Geräten sind ebenfalls geeignet:

8.1 Geräte und Netze konfigurieren

- Internet Explorer 6.0 und früher, für HMI-Panels
- Mobile Safari (iOS 5.0.1)
- Mobile Android Browser 2.3.4
- Mobile Google Chrome 23.0

**Informationen über den Webserver auslesen**

Aus der CPU sind die im Folgenden aufgezählten Informationen auslesbar. Die Verfügbarkeit der jeweiligen Webseiten hängt von der CPU und ihrer Firmwareversion ab.

| Seite/Information   | Beschreibung   |
|---|--|
| Intro   | Einstiegsseite für die Standard-Webseiten  |
| Start Page<br>Startseite mit allgemeinen CPU-Informationen                                      | Die Startseite liefert eine Übersicht der generellen Informationen der CPU, den CPU-Namen, den CPU-Typ und Basisinformationen zum aktuellen Betriebszustand.   |
| Identification<br>Identifikationsinformationen  | Anzeige der statischen Identifikationsinformationen, wie die Seriennummer, Bestellnummer und Versionsnummern.  |
| Diagnostic Buffer<br>Diagnoseinformation  | Anzeige des Diagnosepuffer-Inhalts mit den jüngsten Einträgen zuerst.  |
| Module Information<br>Baugruppeninformationen   | Anzeige, ob die zentral gesteckten Komponenten einer Station in Ordnung sind, ob z. B. Wartungsanforderungen vorliegen oder Komponenten nicht erreichbar sind.<br><br>Ab FW-Version 4 ist ein Firmware-Update über diese Webseite möglich.   |
| Communication<br>Kommunikation  | Anzeige der Kommunikationsverbindungen bei offener Kommunikation (OUC), Anzeige der Ressourcen sowie Adressparameter.  |
| Variable Status<br>Variablen  | Anzeige des Status von Operanden des Anwenderprogramms zum Beobachten und Ändern der Werte.  |
| Data Logs<br>(File Browser ab FW-Version 4)   | Datenarchive im CSV-Format zum Übertragen auf die Festplatte des PGs. Die Datenarchive werden mit Data Log-Anweisungen im Anwenderprogramm angelegt und mit Daten befüllt.<br><br>Ab FW-Version 4 haben Sie über die Webseite "File Browser" Zugriff auf Dateien des internen Ladespeichers und des externen Ladespeichers (Memory Card), z. B. auf den Inhalt der Verzeichnisse "DataLogs" und "Recipes". |
| User Pages<br>Anwenderseiten (falls anwenderdefinierte Webseiten projiziert und geladen wurden) | Die Anwenderwebseiten liefern eine Liste der Webseiten mit kundenspezifischen Webapplikationen.  |



## Web-Zugriff auf die CPU über PG/PC

Um auf den Webserver zuzugreifen, gehen Sie folgendermaßen vor:

1. Verbinden Sie den Client (PG, PC) über die PROFINET-Schnittstelle mit der CPU.
2. Öffnen Sie den Webbrowser.  
Tragen Sie im Feld "Adresse" des Webbrowsers die IP-Adresse der CPU ein in der Form `http://ww.xx.yy.zz` (beispielhafte Eingabe: `http://192.168.3.141`).  
Die Startseite der CPU wird geöffnet. Von der Startseite aus können Sie zu den weiteren Informationen navigieren.

## Standard-Webseiten

### Voraussetzungen für den Web-Zugriff

Im Folgenden werden die Voraussetzungen für den Zugriff auf Standard-CPU-Webseiten erläutert sowie die Auswirkungen fehlender bzw. vorhandener Projektierungsinformation dargestellt.

### Voraussetzung

Der Webserver muss gestartet sein.

Der Webserver startet erst dann, wenn Sie in den Eigenschaften der CPU, Abschnitt "Webserver" den Webserver aktiviert haben.

Beachten Sie Folgendes:

Die Webseiten werden standardmäßig über eine unsichere Verbindung übertragen und sind nicht vor Angriffen Dritter sicher. Wenn Sie die Webseiten verschlüsselt an den Browser übertragen wollen, dann verwenden Sie die URL `https://`, gefolgt von der IP-Adresse der CPU.

## Anmelden

Um lesend auf die Standard-Webseiten zu zugreifen, ist kein Anmelden notwendig. Zum Ausführen bestimmter Aktionen wie das Ändern des Betriebszustandes der CPU oder für Schreibzugriffe muss ein Benutzer als angemeldet sein.

### S7-1200-CPU bis FW-Version V3:

Für die oben genannten Aktionen müssen Sie als Benutzer "admin" angemeldet sein. Die Eingabefelder zum Anmelden sind in der linken oberen Ecke auf jeder Standard-Webseite zu finden.



The image shows a small rectangular login form with a light blue background. It contains two input fields: the top one is labeled "Name" and the bottom one is labeled "Password". To the right of the "Password" field is a blue button with the text "Log in" in white.

Wenn Sie sich als Benutzer "admin" anmelden, müssen Sie Benutzernamen und Passwort dort eingeben.

Name: admin.

Passwort: konfiguriertes CPU-Passwort (für passwortgeschützte CPU).

#### **S7-1200-CPU ab FW-Version 4:**

Sie können die Namen von Benutzern und die Passwörter frei wählen (CPU-Parameter "Webserver", Bereich "Benutzerverwaltung").

Benutzern weisen Sie Rechte zu, z. B. das Recht, Diagnose abzufragen oder die Firmware zu aktualisieren.

### **Javascript und Cookies**

Die Standard-Webseiten nutzen Javascript und Cookies. Sie sollten beides in Ihrem Webbrowser freigegeben haben.

Wenn Javascript nicht freigegeben ist, bestehen z. B. folgende Einschränkungen:

- Daten von Standard-Webseiten werden nicht automatisch aktualisiert.
- Sie können sich nicht als Benutzer anmelden.
- Felder können nicht sortiert werden (Modulinformation)

Wenn Cookies nicht freigegeben sind, können Sie sich nicht anmelden.

### **Siehe auch**

Zugriff für HTTPS (Seite 637)


### **Einstellungen für den Betrieb**

#### **Einstellungen für den Betrieb**

Um den Webserver einer S7-1200-CPU nutzen zu können, müssen Sie in der Netzsicht bzw. Gerätesicht die CPU markieren und folgende Einstellungen im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver" vor:

- Webserver aktivieren
- Zugriffe auf die CPU einschränken auf das Übertragungsprotokoll HTTPS (verschlüsselte Übertragung)  
Ein Zugriff über Port 80 wird dann gesperrt. Kommunikation ist nur über Port 443 möglich.

- Automatische Aktualisierung der Webseiten aktivieren  
Das Aktualisierungsintervall ist voreingestellt und nicht änderbar. Webseiten mit veränderlichen Inhalten (z. B. Statusinformationen oder Diagnoseinformationen) aktualisiert die CPU in regelmäßigen Abständen.
- Benutzer anlegen und verwalten  
Den Benutzern stehen ausschließlich die Optionen zur Verfügung, die den Zugriffsrechten fest zugeordnet sind.  
Je nach verwendeter CPU und Firmware können Sie unterschiedliche Benutzerrechte vergeben. Rechte, die Ihre CPU nicht unterstützt, können nicht aktiviert werden. In der Benutzerliste ist voreingestellt ein Benutzer mit Namen "Jeder" mit minimalen Zugriffsrechten, die Sie aber erweitern können. Wer ohne Passworтеingabe den Webserver nutzt, hat die Zugriffsrechte des Benutzers "Jeder".  
Sie haben die Möglichkeit, weitere Benutzer mit unterschiedlichen Zugriffsrechten zu parametrieren. Diese Benutzer müssen sich mit dem parametrisierten Benutzernamen und Passwort anmelden.

|   |
|---|
|  <b>WARNUNG</b>  |
| <b>Unautorisierter Zugriff auf die CPU über den Webserver</b>   |
| Unautorisierter Zugriff auf die CPU oder das Ändern von PLC-Variablen auf ungültige Werte kann zu Unterbrechungen des von der CPU gesteuerten Prozesses führen und Tod, schweren körperlichen Schaden bzw. Sachschaden verursachen!   |
| Weil die Aktivierung des Webserver autorisierten Personen erlaubt, z. B. Betriebszustände zu ändern, auf CPU-Daten schreibend zuzugreifen oder die Firmware zu aktualisieren, empfehlen wir, folgende Sicherheitsmaßnahmen zu berücksichtigen:  |
| <ul style="list-style-type: none"><li>• Schränken Sie wenn möglich Zugriffe auf das HTTPS-Protokoll ein.</li><li>• Legen Sie Benutzer an mit sicheren Passwörtern. Ein sicheres Passwort wird z. B. nur für eine einzige Anwendung verwendet, ist mehr als 8 Zeichen lang und besteht aus Groß- und Kleinbuchstaben sowie Sonderzeichen und Ziffern (?!+%\$1234...). Des Weiteren sollten keine gängigen Zeichenfolgen der Computertastatur oder Wörter aus dem Wörterbuch verwendet werden.<br/>Ändern Sie das Passwort in regelmäßigen Abständen.</li><li>• Erweitern Sie nicht die Rechte des Benutzers "Jeder".</li><li>• Prüfen Sie PLC-Variablen im Anwenderprogramm und schränken den Wertebereich auf zulässige Bereiche ein, da Nutzer über den Webserver ungültige Werte setzen können.</li></ul> |

## Zugriff für HTTPS

## Zugriff über HTTPS

HTTPS dient zur Verschlüsselung und Authentifizierung der Kommunikation zwischen Browser und Webserver.

Um die Datenübertragung zwischen Browser und CPU über das HTTPS-Protokoll durchzuführen, geben Sie die URL in der Form `https://ww.xx.yy.zz` in die Adresszeile Ihres Webbrowsers ein, wobei `ww.xx.yy.zz` für die IP-Adresse der CPU steht.

Für einen fehlerfreien HTTPS-Zugriff auf die CPU benötigen Sie ein gültiges und installiertes Zertifikat.

Ist kein Zertifikat installiert, wird eine Warnung angezeigt, mit der Empfehlung, die Seite nicht zu benutzen. Um die Seite sehen zu können, müssen Sie explizit eine "Ausnahme hinzufügen".

Ein gültiges Zertifikat (Certification Authority) "SIMATIC CONTROLLER" erhalten Sie als Download von der Webseite "Intro" unter "Download certificate". Wie Sie ein Zertifikat installieren, finden Sie in der Hilfe Ihres jeweiligen Webbrowsers.

### Auf Daten des CPU-Speichers zugreifen

Um auf Daten, die sich im internen oder externen Ladespeicher der CPU befinden, können Sie über eine Standard-Webseite zugreifen.

- Bei S7-1200-CPU bis einschließlich FW-Version 3 nutzen Sie die Webseite "Data Logs". Von dieser Webseite übertragen Sie Data Logs von der CPU auf ein Laufwerk Ihres PCs.
- Bei S7-1200-CPU ab FW-Version 4 nutzen Sie die Webseite "File Browser". Von dieser Webseite übertragen Sie Daten z. B. aus den Ordnern "Data Logs" oder "Recipes" auf ein Laufwerk Ihres PCs.

Abhängig vom Dateityp und den Zugriffsrechten, die Sie für Web-Server-Benutzer parametrisiert haben, können die Dateien heruntergeladen, gelöscht, umbenannt oder hochgeladen werden. Die Verzeichnisse selbst können nur erstellt, gelöscht oder umbenannt werden.

### Beispiel: Data Logs

Zum Öffnen eines Data Logs klicken Sie auf den Link des gewünschten Data Logs. Sie haben die Möglichkeit, die Datei (.csv) zu öffnen, z. B. in Microsoft Excel oder in einem anderen Programm ihrer Wahl oder die Datei zu speichern.

Besonderheit: Data Logs sind im U.S. amerikanischen CSV-Format gespeichert. Sie können die Datei nur mit der U.S.-Version von Microsoft Excel direkt öffnen. Mit anderen Länderversionen von Microsoft Excel müssen Sie die Datei importieren und dabei im Import-Assistenten als Trennzeichen "Komma" wählen.

### Herunterladen eines Data Logs

Zum Herunterladen eines Data Logs klicken Sie auf das Download-Symbol des gewünschten Data Logs. Sie haben die Möglichkeit, die Datei (.csv) zu öffnen, z. B. in Microsoft Excel oder in einem anderen Programm ihrer Wahl oder die Datei zu speichern.

### Herunterladen und Leeren bzw. Löschen eines Data Logs

Für eine CPU mit der FW-Version bis V3.0:

Zum Herunterladen und Löschen der aktuellen Einträge des Data Logs müssen Sie angemeldet sein. Klicken Sie dazu auf das Symbol "Herunterladen und Löschen" des gewünschten Data Logs. Sie haben die Möglichkeit, die Datei (.csv) zu öffnen, z. B. in Microsoft Excel oder in einem anderen Programm ihrer Wahl oder die Datei zu speichern.

Für eine CPU ab der FW-Version V4.0:

Um das Data Log zurückzusetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die CSV-Datei, z. B. mit Excel.
2. Löschen Sie die Zeilen zwischen Überschriftenzeile und der Zeile mit dem Eintrag "//END", falls diese Zeile existiert.
3. Speichern Sie die Datei auf einem Laufwerk Ihres PCs.
4. Löschen Sie auf der Webseite "File Browser" das Data Log (d. h. die CSV-Datei) und laden die vorbereitete CSV-Datei über die Schaltfläche "Upload file" der Webseite "File Browser" in die CPU.

Weitere Informationen finden Sie im Systemhandbuch zur S7-1500-CPU.

## **Anwenderdefinierte Webseiten erstellen und laden**

### **Wissenswertes zu Anwenderdefinierten Webseiten**

#### **Konzept**

Das Konzept der Anwenderdefinierten Webseiten ermöglicht Ihnen, mit einem Webbrowser auf frei gestaltete Webseiten der CPU zuzugreifen. Der Webserver der CPU stellt diese Funktion zur Verfügung.

Hinsichtlich der Gestaltung und Funktionalität der anwenderdefinierten Webseiten sind Sie nicht auf spezielle Tools angewiesen. Sie können die Seiten mit CSS im Layout anpassen, mit Javascript dynamischen Inhalt zur Verfügung stellen oder ein beliebiges Framework zur Produktion der Webseiten verwenden.

Die Gesamtheit der Dateien, die der Webserver verarbeitet, wird auch "Webapplikation" genannt.

#### **Webapplikation und Anwenderprogramm**

Mit Hilfe von speziellem HTML-Code in anwenderdefinierten Webseiten können Sie auch Daten über einen Webbrowser zur Weiterverarbeitung an das Anwenderprogramm der CPU übergeben und Daten aus dem Operandenbereich der CPU im Webbrowser anzeigen lassen.

Mit Scriptanweisungen (z. B. Javascript) können Sie Ihre Webseiten optimieren, z. B. Inhalte dynamisch ändern oder Benutzereingaben validieren.

Zur Synchronisation zwischen Anwenderprogramm und Webserver, aber auch zur Initialisierung müssen Sie die Anweisung WWW (SFC 99) im Anwenderprogramm aufrufen.

- Wenn keine Interaktion zwischen Webapplikation und Anwenderprogramm notwendig ist, z. B. wenn eine Webseite nur statische Informationen anbietet, ist nur eine Initialisierung im Anwenderprogramm notwendig.
- Wenn ein einfacher Datenaustausch zwischen PLC-Variablen und Variablen der Webapplikation notwendig ist, z. B. um den Inhalt von PLC-Variablen anzuzeigen oder in eine PLC-Variable einen Wert zu schreiben, müssen Sie die Syntax zum Lesen und Schreiben von Variablen zu berücksichtigen. Im Anwenderprogramm ist in diesem Fall ebenfalls nur eine Initialisierung z. B. im Anlauf-OB erforderlich.
- Wenn eine weitergehende Interaktion zwischen Webapplikation und Anwenderprogramm notwendig ist, müssen Sie neben der Synchronisierung zwischen Webserver und Anwenderprogramm auch Status- und Steuerungsinformation aus dem Web-Control-DB hantieren. Das ist z. B. der Fall bei Benutzereingaben, die über den Webbrowser an den Webserver übergeben werden, um von der CPU ausgewertet zu werden. Im Unterschied zum einfachen Datenaustausch beeinflusst das Anwenderprogramm unmittelbar den Zeitpunkt, zu dem die angeforderte Webseite an den Webbrowser zurückgegeben wird. In diesem Fall ist es notwendig, dass Sie sich mit dem Konzept der manuellen Fragmente und den Strukturen des Web-Control-DBs auseinandersetzen.

## Initialisierung

Anwenderdefinierte Webseiten werden in Datenbausteine "verpackt", damit die CPU sie verarbeiten kann. Aus den Quelldaten (HTML-Dateien, Bilder, Javascript-Dateien, ...) müssen Sie während der Projektierung entsprechende Datenbausteine generieren, um die Webapplikation in die CPU laden zu können. Eine besondere Rolle nimmt der Web-Control-DB ein (Default: DB 333), der Status- und Steuerungsinformationen sowie Verweise auf weitere Datenbausteine mit codierten Webseiten enthält. Datenbausteine, die codierte Webseiten enthalten, werden "Fragment-DBs" genannt.

Wenn die Datenbausteine in die CPU geladen wurden, "weiß" die CPU nicht, dass darin anwenderdefinierte Webseiten codiert sind. Durch die Anweisung "WWW" (SFC 99) z. B. im Anlauf-OB wird ihr mitgeteilt, welcher DB der Web-Control-DB ist. Nach dieser Initialisierung sind die anwenderdefinierten Webseiten über einen Webbrowser erreichbar.

## Synchronisierung

Wenn das Anwenderprogramm mit den anwenderdefinierten Webseiten Daten austauschen bzw. interagieren soll, muss die Anweisung WWW (SFC 99) im zyklischen Programmteil verwendet werden.

Beispiele für Interaktion zwischen Anwenderprogramm und Webseite:

- Empfangene Daten überprüfen
- Daten für den anfragenden Webbrowser zusammenstellen und zurücksenden

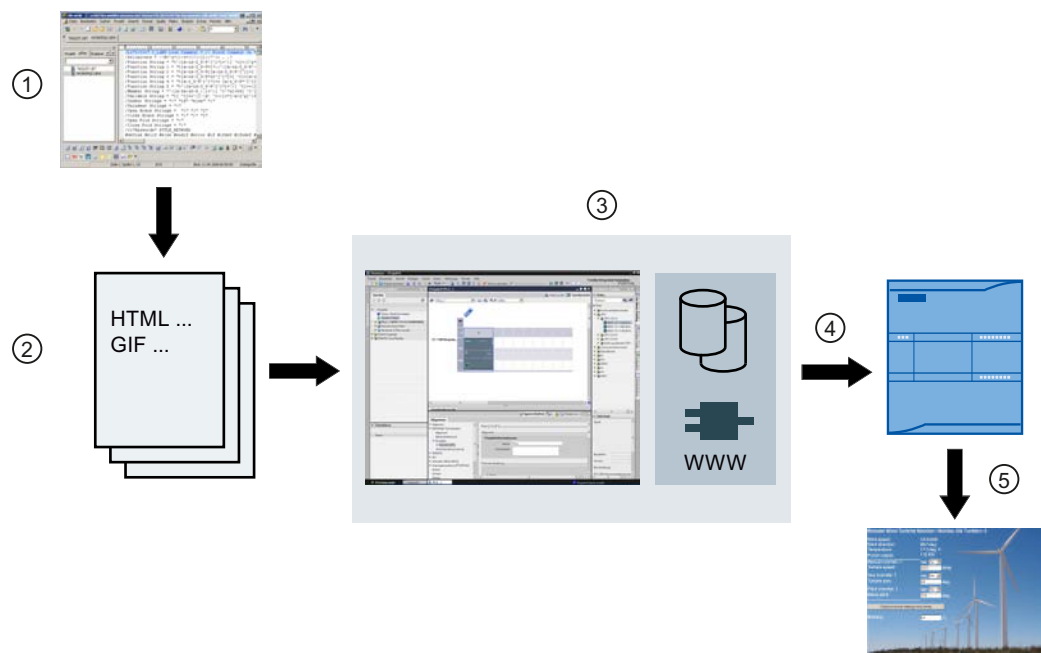
In diesem Fall muss die Statusinformation aktuell auswertbar sein und Steuerungsinformation z. B. zum Freigeben einer angeforderten Webseite dem Webserver mitgeteilt werden.

## Überblick Vorgehensweise

### Grundlegendes

In diesem Kapitel ist die grundsätzliche Vorgehensweise schrittweise erläutert, um anwenderdefinierte Webseiten zu erstellen, zu laden und in der Betriebsphase zu nutzen.

Die folgende Grafik stellt den Ablauf zur Erstellung und Anzeige von anwenderdefinierten Webseiten vereinfacht dar:



- ① Programmieren einer Webapplikation (evtl. unter Zuhilfenahme von geeigneten Tools, gegebenenfalls mit enthaltenen AWP-Kommandos für dynamische Seiten).
- ② Webapplikation besteht aus einzelnen Quelldateien, z. B. \*.html, \*.gif, \*.js, ...
- ③ Mit STEP 7:
  - Generieren der Datenbausteine (Web-Control-DB und Fragment-DBs) aus Quelldateien. Die DBs enthalten Metainformation und die komplette Webapplikation inklusive der Bilder und den dynamischen und statischen Teilen der Webapplikation. Die DBs werden unter "Systembausteine" in der Projektnavigation abgelegt.
  - Die Anweisung "WWW" im Anwenderprogramm aufrufen. Die Anweisung initialisiert den Webserver der CPU für eine Webapplikation.
  - Bei Bedarf Interaktion zwischen Webserver und Anwenderprogramm ausprogrammieren
- ④ Laden der Bausteine in die CPU.
- ⑤ Webseite im Browser aufrufen. Der Aufruf der Webseiten der CPU erfolgt über die Eingabe der IP-Adresse der CPU.

### Weitere Informationen

Weitere Informationen und Beispiele zum Webserver der S7-1200 finden Sie im Internet (<http://support.automation.siemens.com/WW/view/de/36932465>).

### Webseiten erstellen

Zur Erstellung der anwenderdefinierten Webseiten bieten sich Webdesign-Tools diverser Anbieter an. Grundsätzlich sollen die Webseiten nach den Konventionen des W3C (World Wide Web Consortium) programmiert und gestaltet werden. Im Webserver der CPU findet keine Überprüfung nach W3C Kriterien statt.

### Regeln

- Das Tool muss in der Lage sein, den HTML-Code direkt zu editieren, damit Sie AWP-Kommandos in der HTML-Seite einfügen können.  
In der CPU werden lediglich die AWP-Kommandos geparkt und z. B. durch Werte aus dem Anwenderprogramm/Prozessabbild der CPU ersetzt.
- Dateien, die AWP-Kommandos enthalten, müssen UTF-8 codiert sein. In den Metadaten der HTML-Seite setzen Sie daher das Attribut charset auf UTF-8 und speichern die Datei UTF-8-codiert.
- Dateien, die AWP-Kommandos enthalten, dürfen folgende Sequenz nicht enthalten: ]]
- Dateien, die AWP-Kommandos enthalten, dürfen außerhalb von "Variable-lesen-Bereichen" (:=<Variablenname>:) folgende Sequenz nicht enthalten: :=  
Tipp: Ersetzen Sie das erste Zeichen einer verbotenen Sequenz durch seine Zeichencodierung; für den Doppelpunkt z. B. &#58;.

Ein kleines Beispiel für eine anwenderdefinierte Webseite soll den grundsätzlichen Aufbau zeigen.

### Voraussetzung

- Die CPU muss über einen Webserver verfügen und der Webserver der CPU muss aktiviert sein
- Um als Nutzer von anwenderdefinierten Webseiten schreibend auf PLC-Variablen zugreifen zu können, müssen Sie als "admin" angemeldet sein.
- Für das Beispiel unten müssen für PLC-Variablen, die auf der Webseite angezeigt werden sollen, PLC-Variablen definiert sein. Für die erste verwendete PLC-Variable "Tank\_below\_max" ist das hier dargestellt.

|   | Name           | Datentyp | Adresse |
|---|----------------|----------|---------|
| 1 | Tank_below_max | Bool     | %I0.0   |

### Anwenderdefinierte Webseiten erstellen

Der folgende Code für eine Beispielwebseite liest Werte aus dem Prozessabbild und stellt diese in einer Tabelle dar.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">  
<html>  
  <head>  
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  
    <title>Mix</title>
```



```
</head>
<body>
  <h1>Mix</h1>
  <h2> Actual State </h2>
  <table border="1">
    <tr>
      <th>Variable</th>
      <th>State</th>
    </tr>
    <tr>
      <td>Tank below max</td>
      <td>:="Tank_below_max":</td>
    </tr>
    <tr>
      <td>Tank above min</td>
      <td>:="Tank_above_min":</td>
    </tr>
  </table>
</body>
</html>
```

## **AWP-Kommandos**

Mit AWP-Kommandos (Automation Web Programming) wird die Schnittstelle zwischen einer frei programmierbaren Webapplikation für eine CPU, die über einen Webserver verfügt, und den CPU-Daten deklariert.

Für die Entwicklung von Webapplikationen sind Sie nur den Beschränkungen unterworfen, die der Webbrowser vorgeben. Mit dem Anwenderprogramm kontrollieren Sie in einer der Programmiersprachen von STEP 7, welche CPU-Daten zu welcher Zeit im Webbrowser des Betrachters angezeigt werden. Mit Hilfe von AWP-Kommandos, die Sie innerhalb der HTML-Dateien notieren, werden Daten für die beabsichtigte Interaktion zwischen Webapplikation und Anwenderprogramm vereinbart.

AWP-Kommandos werden als HTML-Kommentare mit einer speziellen Syntax in HTML-Dateien eingefügt; sie deklarieren folgende Features:

- PLC-Variablen lesen
- PLC-Variablen schreiben
- Spezialvariablen lesen
- Spezialvariablen schreiben
- Enum-Typen definieren
- Variablen Enum-Typen zuweisen
- Fragmente definieren
- Fragmente importieren

### Syntax von AWP-Kommandos

Ein AWP-Kommando beginnt mit "`<!--AWP_`" und endet mit "`-->`". In Java-Script-Dateien sollten die Kommandos zusätzlich von Javascript-Kommentaren eingeschlossen werden ("`/*...*/`").

### Notationsregeln für PLC-Variablenamen innerhalb eines AWP-Kommandos

Die AWP-Kommandos "AWP\_In\_Variable" und "AWP\_Out\_Variable" enthalten ein Name-Attribut und optional ein Use-Attribut. Diesen Attributen ist ein PLC-Variablenname zugeordnet, über den PLC-Variablen im Browser geschrieben bzw. gelesen werden. Für den Umgang mit PLC-Variablenamen im HTML-Code gibt es folgende Regeln:

- PLC-Variablen müssen in Anführungszeichen (" ... ") eingeschlossen werden.
- In AWP-Kommandos verwendete PLC-Variablen müssen zusätzlich von einfachen Anführungszeichen (" ... ") oder mit umgekehrtem Schrägstrich (Backslash) maskierten Anführungszeichen umschlossen werden ("\" ... \").
- Wenn der PLC-Variablenname das Zeichen \ (umgekehrter Schrägstrich) enthält, dann muss dieses Zeichen mit der Escape-Sequenz \\ als normales Zeichen des PLC-Variablennamens gekennzeichnet werden.
- Wenn der PLC-Variablenname im AWP-Kommando zusätzlich mit einfachen Anführungszeichen eingeschlossen ist und innerhalb des Namens kommt das einfache Anführungszeichen (') vor, muss es auch durch die Escape-Sequenz \' als normales Zeichen gekennzeichnet werden.
- Wenn im AWP-Kommando eine absolute Adresse (Eingang, Ausgang, Merker) verwendet wird, dann wird sie von einfachen Anführungszeichen umschlossen.

| PLC-Variable     | PLC-Variable im HTML-Code  |
|------------------|--|
| "Velocity"       | <code>&lt;!-- AWP_In_Variable Name='\"Velocity\"' --&gt;</code>            |
| "abc\de"         | <code>&lt;!-- AWP_In_Variable Name='\"abc\de\"' --&gt;</code>              |
| "abc'de"         | <code>&lt;!-- AWP_In_Variable Name='\"abc\'de\"' --&gt;</code>             |
| "abc'de"         | <code>&lt;!-- AWP_In_Variable Name="abcde" Use='\"abc\'de\"' --&gt;</code> |
| "DB name".tag    | <code>&lt;!-- AWP_In_Variable Name='\"DB name\".tag' --&gt;</code>         |
| "DB name"."ta.g" | <code>&lt;!-- AWP_In_Variable Name='\"DB name\".\"ta.g\"' --&gt;</code>    |
| -                | <code>&lt;!-- AWP_Out_Variable Name='flag1' Use='M0.0' --&gt;</code>       |

### Siehe auch

- Variablen lesen (Seite 644)
- Variablen schreiben (Seite 646)
- Spezialvariablen (Seite 647)

### Variablen lesen

Anwenderdefinierte Webseiten können PLC-Variablen lesen.

Die PLC-Variable muss durch einen PLC-Variablennamen spezifiziert sein.

Diese Out-Variablen (Ausgaberichtung von der Steuerung aus betrachtet) werden mit der im folgenden beschriebenen Syntax an beliebiger Stelle im HTML-Text eingefügt.

## Syntax

`:=<varname>:`

Diese Referenzen werden im Betrieb vom Webserver durch die jeweils aktuellen Werte der PLC-Variablen ersetzt.

`<varname>` kann eine einfache globale PLC-Variable sein, aber auch ein kompletter Variablenpfad zu einem Strukturelement.

## Notationsregeln für PLC-Variablennamen

- PLC-Variablen im HTML-Code werden mit Anführungszeichen (") umschlossen, wenn sie in der Variablen-tabelle definiert sind. Bei Datenbaustein-Variablen wird der Name des Datenbausteins mit Anführungszeichen umschlossen. Wenn spezielle Zeichen in den Strukturelementen des Datenbausteins verwendet werden wie z. B. der Punkt (.) oder Leerzeichen, muss auch dieser Teil in Anführungszeichen eingeschlossen sein.
- Anführungszeichen werden nicht verwendet bei absoluten Adressen von Eingängen, Ausgängen oder Merkern.

| PLC Variable                   | PLC-Variable im HTML-Code  |
|--------------------------------|--|
| "DB_name".var_name             | := "DB_name".var_name:   |
| "DB_name".struct_name.var_name | := "DB_name".struct_name.var_name:                                     |
| "DB_name"."var.name"           | := "DB_name"."var.name":   |
| "memory"                       | := "memory":   |
| -                              | :=E0.0:  |
|                                | :=A0.0:  |
|                                | :=MW100:   |
|                                | :=%MW100:  |
| "My_Data_Block".flag1          | <!-- AWP_Out_Variable Name='flag1'<br>Use=' "My_Data_Block".flag1' --> |
|                                | ...  |
|                                | :=flag1:   |

- Wenn der PLC-Variablenname die Zeichen : (Doppelpunkt) oder \ (umgekehrter Schrägstrich) enthält, dann müssen diese Zeichen mit der Escape-Sequenz \: bzw. \\ als normales Zeichen des PLC-Variablennamens gekennzeichnet werden.

| PLC Variable | PLC-Variable im HTML-Code |
|--------------|---------------------------|
| "abc:de"     | := "abc\:de":             |
| "abc\de"     | := "abc\\de":             |

- Spezielle Zeichen "<, &, >"  
 Falls diese Zeichen im Variablennamen vorkommen (z. B. "a<b"), kann es zu Darstellungsproblemen kommen.  
 Vermeiden Sie Ausdrücke wie z. B. :=<a<b": in der HTML-Seite.  
 Um Darstellungsproblemen vorzubeugen, verwenden Sie z. B. ein AWP-Kommando mit einem Use-Ausdruck nach dem unten stehenden Muster. Das Use-Attribut definiert die PLC-Variable mit den problematischen Zeichen, das Name-Attribut definiert den Namen ohne problematische Zeichen, wie er in der HTML-Seite verwendet wird.

| PLC Variable | PLC-Variable im HTML-Code   |
|--------------|---|
| "a<b"        | <pre>&lt;!--AWP_Out_Variable Name='simplename' Use='&lt;a&lt;b"' --&gt; ... :=simplename:</pre> |

### Siehe auch

AWP-Kommandos (Seite 643)

### Variablen schreiben

Anwenderdefinierte Webseiten können Daten in die CPU schreiben.

Voraussetzung ist ein AWP-Kommando, das die zu schreibende PLC-Variable identifiziert.

Die PLC-Variable muss ferner durch einen PLC-Variablennamen spezifiziert sein.

Die In-Variablen (Eingaberichtung von der Steuerung aus betrachtet) werden auf der Browserseite gesetzt. Dies kann beispielsweise in einem Formular erfolgen.

Die Variablen werden entweder im HTTP-Header (per Cookie oder POST-Methode) oder in der URL (GET-Methode) vom Browser im HTTP-Request gesetzt und dann vom Webserver in die jeweilige PLC-Variable geschrieben.

### Syntax

Damit die In-Variablen in die CPU geschrieben werden können, ist es notwendig, die Variablen durch eine explizite AWP-Anweisung zu definieren:

```
<!-- AWP_In_Variable Name='<PLC_Varname1>' Name='<PLC_Varname2>'
Name='<PLC_Varname3>' -->
```

In einer Anweisung können - wie oben - mehrere Variablen definiert werden.

Der konkrete PLC-Variablenname wird dabei in doppelten Anführungszeichen geschrieben; z. B. <PLC\_Varname1> = "myVar".

Für den Fall, dass der Name der Variable, die Sie für die Webapplikation verwenden, nicht mit dem Namen der PLC-Variable identisch ist, kann mit dem Parameter "Use" die Zuordnung zu einer PLC-Variablen erfolgen:

```
<!-- AWP_In_Variable Name='<Webapp_Varname>' Use='<PLC_Varname>'
```

## Beispiel

Das AWP-Kommando "AWP\_In\_Variable" ist besonders für die Behandlung von Formularen unverzichtbar.

```
<form method='post' action='/awp/appl/x.html'>
  <p>
    <input name='"var1"' type='text'>
    <input value='set' name='Button1' type='submit'>
  </p>
</form>
```

Im oben definierten Formular wird mit der HTTP-Request-Methode "post" die Variable "var1" zum Webserver übertragen. Die Variable "var1" setzt der Anwender im Formularfeld. Die Variable 'Button1' hat den Wert 'set', wird aber für die CPU nicht benötigt. Damit die Variable "var1" in der CPU geschrieben werden kann, muss im gleichen Fragment folgende Anweisung aufgenommen werden:

```
<!-- AWP_In_Variable Name='"var1"' -->
```

Da PLC-Variablen mit doppelten Anführungszeichen (") umschlossen werden, muss der Name im AWP-Kommando mit einfachen Anführungszeichen (') oder mit maskierten doppelten Anführungszeichen (\") eingeschlossen sein. Um die zahlreichen Escape-Sequenzen zu vermeiden, empfehlen wir die Verwendung von einfachen Anführungszeichen.

```
<!-- AWP_In_Variable Name=' "Info".par1' -->
```

```
<!-- AWP_In_Variable Name="\ "Info".par1\" " -->
```

## Bedingungen für den Schreibzugriff im Betrieb

Damit ein Anwender von einer anwenderdefinierten Webseite aus auf PLC-Variablen schreiben kann, müssen folgende Voraussetzungen erfüllt sein:

- Die CPU ist passwort-geschützt.
- Der Anwender ist als "admin" angemeldet.

Diese Regel gilt prinzipiell für alle schreibenden Zugriffe von Webseiten auf eine CPU.

## Siehe auch

Voraussetzungen für den Web-Zugriff (Seite 635)

AWP-Kommandos (Seite 643)

## Spezialvariablen

Bei den Spezialvariablen handelt es sich hauptsächlich um die so genannten HTTP-Variablen die in den Definitionen des World Wide Web Consortium (W3C) festgelegt sind. Auch für Cookies und Server-Variablen werden Spezialvariablen verwendet.

Die AWP-Kommandos zum Lesen und Schreiben von Spezialvariablen unterscheiden sich nur durch zusätzliche Parameter von den AWP-Kommandos zum Lesen und Schreiben normaler Variablen.

### Spezialvariable lesen

Der Webserver kann PLC-Variablen lesen und sie an Spezialvariablen im HTTP-Response-Header übergeben. Sie können z. B. eine URL für eine Umleitung zu einer anderen Webseite lesen und an die Spezialvariable HEADER:Location übergeben mit Hilfe der Spezialvariable HEADER:Location.

Folgende Spezialvariablen können gelesen werden:

| Name                | Beschreibung   |
|---------------------|--|
| COOKIE_VALUE:name   | Wert des Cookies mit dem Namen: "name"   |
| COOKIE_EXPIRES:name | Ablaufzeit des Cookies mit dem Namen: "name" in Sekunden (muss zuvor gesetzt worden sein).     |
| HEADER:Status       | HTTP-Statuscode (wenn kein anderer Wert gesetzt wurde wird Statuscode 302 zurückgegeben).      |
| HEADER:Location     | Pfad für die Weiterleitung auf eine andere Seite. Statuscode 302 muss gesetzt sein.            |
| HEADER:Retry-After  | Zeit in der der Service voraussichtlich nicht verfügbar ist. Statuscode 503 muss gesetzt sein. |
| HEADER: ...         | Alle anderen Header-Variablen können ebenfalls auf diese Weise übermittelt werden.             |

Mit dem AWP-Kommando "AWP\_Out\_Variable" wird festgelegt, welche PLC-Variablen im HTTP-Header an den Webbrowser übertragen werden sollen.

Prinzipieller Aufbau:

```
<!-- AWP_Out_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

### Parameterbeschreibung

- Name: Typ und Name der Spezialvariablen
- Use (optionaler Parameter): Für den Fall, dass der Name der Spezialvariablen nicht mit dem Namen der PLC-Variablen identisch ist kann mit dem Parameter "Use" die Zuordnung zu einer PLC-Variablen erfolgen.

Beispiel:

```
<!-- AWP_Out_Variable Name="COOKIE_VALUE:siemens" Use="'info'.language' -->
```

## Spezialvariable schreiben

Prinzipiell können alle vom Webbrowser in den HTTP-Header geschriebenen HTTP-Variablen vom Anwenderprogramm der CPU ausgewertet werden. Beispiele für Variablentypen:

| Name                     | Beschreibung   |
|--------------------------|--|
| HEADER:Accept-Language   | Akzeptierte bzw. bevorzugte Sprache  |
| HEADER:Authorization     | Berechtigungsnaehweis für eine angeforderte Ressource                                  |
| HEADER:Host              | Host und Port der angeforderten Ressource  |
| HEADER>User-Agent        | Informationen zum Browser  |
| HEADER: ...              | Alle anderen Header-Variablen können ebenfalls auf diese Weise übermittelt werden      |
|                          |  |
| SERVER:current_user_id   | Zeigt an, ob ein Benutzer angemeldet ist (current_user_id=0: kein Benutzer angemeldet) |
| SERVER:current_user_name | Benutzername des angemeldeten Benutzers  |
| SERVER:GET               | Request-Methode ist GET  |
| SERVER:POST              | Request-Methode ist POST   |
|                          |  |
| COOKIE_VALUE:name        | Wert des Cookies mit dem Namen: "name"   |

Mit dem AWP-Kommando "AWP\_In\_Variable" wird festgelegt, welche Spezialvariablen im Anwenderprogramm der CPU ausgewertet werden sollen.

Prinzipieller Aufbau:

```
<!-- AWP_In_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Parameterbeschreibung:

Name: Typ und Name der Spezialvariablen

Use (optionaler Parameter): Für den Fall, dass der Name der Spezialvariablen nicht mit dem Namen der PLC-Variablen identisch ist kann mit dem Parameter Use die Zuordnung zu einer PLC-Variablen erfolgen.

## Beispiele:

```
<!-- AWP_In_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

Der Variablenname im HTTP-Header wird durch den bei Use angegebenen PLC-Variablennamen ersetzt. Das Cookie wird in die PLC-Variable "info".language geschrieben.

```
<!-- AWP_In_Variable Name='COOKIE_VALUE:siemens' Use='"info".language' -->
```

Der Variablenname im HTTP-Header wird durch den bei Use angegebenen PLC-Variablenamen ersetzt. Das Cookie wird in die PLC-Variable "info".language geschrieben.

```
<!-- AWP_In_Variable Name='"COOKIE_VALUE:siemens"' -->
```

Die HTTP-Header-Variable wird in die gleichnamige PLC-Variable geschrieben.

## Siehe auch

AWP-Kommandos (Seite 643)

## Enumerationstypen

### Enumerationstypen (Enums)

Mit Hilfe von Enums können numerische Werte aus dem PLC-Programm in Texte umgewandelt werden und umgekehrt. Die Zuordnung der numerischen Werte kann auch für mehrere Sprachen erfolgen.

### Enums anlegen

Geben Sie ein AWP-Kommando mithilfe folgender Syntax am Anfang der HTML-Datei ein:

```
<!-- AWP_Enum_Def Name="<Name des Enum Typs>" Values='0:"<Text_1>",  
1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Für z. B. deutsche Werte abzulegen als HTML-Datei im Ordner "de" des HTML-Verzeichnisses:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"an", 1:"aus", 2:"Störung"' -->
```

Für z. B. englische Werte abzulegen als HTML-Datei im Ordner "en" des HTML-Verzeichnisses:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"on", 1:"off", 2:"error"' -->
```



## Enums zuordnen

Die Zuordnung der Variablen aus dem Anwenderprogramm zu den einzelnen Enum-Texten erfolgt über ein eigenes AWP-Kommando:

```
<!-- AWP_Enum_Ref Name="<VarName>" Enum="<EnumTypeName>" -->
```

Dabei ist <VarName> der symbolische Name aus dem Anwenderprogramm und <EnumTypeName> der zuvor festgelegte Name des Enum-Typs.

---

### Hinweis

In jedem Fragment, in dem eine PLC-Variable Enum-Texte referenziert, muss diese PLC-Variable mit dem entsprechenden AWP-Kommando dem Enum-Typ-Namen zugeordnet werden.

Beachten Sie, dass zwischen einer Enum-Zuordnung und Enum-Verwendung kein AWP-Kommando zum Import von Fragmenten steht, da durch diesen Import die Enum-Zuordnung in einem anderen Fragment liegt als die Enum-Verwendung.

---

## Beispiel

Es ist ein Enum-Typ "zustand" mit den Werten "0" und "1" definiert. "0" entspricht "aus", "1" entspricht "ein":

```
<!-- AWP_Enum_Def Name="zustand" Values='0:"aus", 1:"ein"' -->
```

Im HTML-Code der auszugebenden Webseite ist der folgende Code enthalten:

```
<!-- AWP_Enum_Ref Name="betriebszustand" Enum="zustand" -->  
:=betriebszustand;
```

Das Anzeigergebnis ist je nach Wert der Variable "betriebszustand" nun nicht mehr "0" oder "1", sondern "aus" oder "ein".

## Vereinfachte Nutzung von Enumerationstypen

Bei S7-1200-CPU's ab Firmware-Version 4 ist es möglich, Enumerationen direkt in AWP-Kommandos zum Lesen und Schreiben von PLC-Variablen zu verwenden.

Sie legen wie im vorigen Abschnitt beschrieben Enums an und können dann mit Lese- und Schreib-AWP-Kommandos die Werte nutzen.

### Enums anlegen

```
<!-- AWP_Enum_Def Name="<Name des Enum Typs>" Values='0:"<Text_1>",&br/>1:"<Text_2>",&br/>... , x:"<Text_x>"' -->
```

### Enums in AWP-Schreib- und Lese-Kommandos nutzen

```
<!-- AWP_In_Variable Name='<Varname>' Enum="<EnumType>" -->  
<!-- AWP_Out_Variable Name='<Varname>' Enum="<EnumType>" -->
```

### Beispiel zum Lesen von PLC-Variablen

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is  
full", 2:"Tank is empty"' --><!-- AWP_Out_Variable Name='Alarm'  
Enum="AlarmEnum" -->...<p>The current value of "Alarm"  
is := "Alarm":</p>
```

Wenn der Wert von "Alarm" in CPU "2" ist, wird auf der HTML-Seite folgender Text angezeigt:

'The current value of "Alarm" is Tank is empty' weil die Enum-Definition den String "Tank is empty" dem numerischen Wert 2 zuordnet.

### Beispiel zum Schreiben von PLC-Variablen

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is  
full", 2:"Tank is empty"' --><!-- AWP_In_Variable Name='Alarm'  
Enum='AlarmEnum' -->...  
<form method="POST">  
<p><input type="hidden" name='Alarm' value="Tank is full" /></p>  
<p><input type="submit" value='Set Tank is full' /></p>  
</form>
```

Weil die Enum-Definition den String "Tank is full" dem numerischen Wert "1" zuordnet, wird der Wert "1" in die PLC-Variablen "Alarm" geschrieben.

## Definieren von Fragmenten

### Fragmente

Unter Fragmenten versteht man einzeln von der CPU zu bearbeitende "Sinnabschnitte" einer Webseite.

Fragmente sind meist ganze Seiten, können aber auch Einzelelemente sein, wie Dateien (z. B. Bilder) oder ganze Dokumente.

### Fragmente definieren

```
<!-- AWP_Start_Fragment Name="<Name>" [Type="<Type>"] [ID="<Id>"]  
[Mode="<Mode>"] -->
```

Der Beginn eines Fragmentes wird mit diesem Kommando festgelegt. Ein Fragment erstreckt sich bis zum Beginn des nächsten Fragments oder bis zum Ende der Datei.

- <Name> Gibt den Namen des Fragments an.  
Der Name muss mit einem Buchstaben [a-zA-Z] oder einem Unterstrich ( \_ ) beginnen. Nach diesem ersten Zeichen können Buchstaben, Unterstriche oder Zahlen [0-9] folgen.
- <Type> Gibt den Typ des Fragments an.
  - "manual" Das Anwenderprogramm wird über den Request für ein Fragment benachrichtigt, ein Eingriff in die zurückzuliefernde Webseite durch das Anwenderprogramm ist möglich.
  - "automatic" Die Seite wird automatisch bearbeitet (default).
- <id> Für das Fragment kann eine numerische ID vorgegeben werden. Wenn keine ID vergeben wird, dann wird dem Fragment automatisch eine ID zugewiesen. Bei manuellen Seiten (<Type>=manual) kann das Fragment im Anwenderprogramm der CPU mit dieser ID angesprochen werden.

---

#### **Hinweis**

Setzen Sie die ID niedrig an, da die höchste ID die Größe des Web-Control-DBs beeinflusst.

---

- <Mode> Fragmente unterstützen die beiden Modi visible und hidden.
  - "visible" Das Fragment ist ein Teil der Webseite. Dieser Mode ist voreingestellt und kann auch weggelassen werden.
  - "hidden" Das Fragment ist nicht Teil der Webseite. Das Fragment wird aber im Web-DB gespeichert und steht dem Anwenderprogramm zum Einfügen in eine angeforderte Webseite zur Verfügung. Über einen Austausch der Fragment-ID (Variable Web-Control-DB.fragment\_index) fügen Sie ein "hidden"-Fragment in die angeforderte Webseite ein.

Das Input Dokument wird durch das "AWP\_Start\_Fragment"-Kommando vollständig in Fragmente aufgeteilt. Daher ist ein "AWP\_End\_Fragment" nicht notwendig.

Ohne ein Start-Fragment-Kommando wird eine Datei als ein Fragment abgebildet, der Fragmentname leitet sich aus dem Dateinamen ab. Wird eine Datei in mehrere Fragmente zerlegt (durch "AWP\_Start\_Fragment"), so muss die Datei mit dem "AWP\_Start\_Fragment"-Kommando beginnen.

## **Importieren von Fragmenten**

Sie haben die Möglichkeit, in einer HTML-Seite ein Fragment zu deklarieren und dieses Fragment in anderen Webseiten zu importieren.

## **Beispiel**

Ein Firmen-Logo soll auf allen Webseiten einer Webapplikation angezeigt werden.

Der HTML-Code für das Fragment, das das Firmen-Logo anzeigt, existiert nur einmal. Sie können das Fragment so oft importieren und in so vielen HTML-Dateien importieren wie erforderlich.

## Syntax

```
<!-- AWP_Import_Fragment Name = "<name>"-->
```

- <name> ist der Name des zu importierenden Fragments.

## Beispiel

HTML-Code innerhalb einer Webseite, der ein Fragment deklariert:

```
<!-- AWP_Start_Fragment Name = "My_Company_Logo"-->  
<p><img src = "compay_logo.jpg"></p>
```

## Beispiel

HTML-Code innerhalb einer anderen Webseite, der das deklarierte Fragment importiert:

```
<!-- AWP_Import_Fragment Name = "My_Company_Logo"-->
```

## Datenbausteine generieren und laden

### Voraussetzung

- Sie haben alle benötigten Quelldateien (\*.html, \*.js, \*.png, ...) für die Webapplikation erstellt.
- Die Quelldateien liegen in einem Ordner; und zwar nur die Quelldateien, die für die Webapplikation benötigt werden! Es dürfen keine anderen Dateien in diesem Ordner liegen.

---

### Hinweis

#### Länge von Dateinamen und Variablennamen

Wenn Sie eine umfangreiche Webapplikation mit vielen Dateien und Verzeichnissen haben, scheitert die Generierung der Web-Datenbausteine möglicherweise. In diesem Fall wird die Generierung mit der Meldung "Überlauf Textliste..." abgebrochen. Ursache sind systeminterne Größenbeschränkungen für Verwaltungsinformationen, die im Web-Datenbaustein gespeichert werden.

Abhilfe: Verwenden Sie kurze Dateinamen und kurze Variablennamen.

---

## Vorgehen

Um aus den Quelldateien für anwenderdefinierte Webseiten in STEP 7 in Datenbausteine zu generieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU z. B. in der Gerätekonfiguration.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Webserver" die Eigenschaften für anwenderdefinierte Webseiten.
3. Als "HTML-Quelle" wählen Sie den Ordner, wo sich die Quelldateien für die Webapplikation befinden.
4. Als Start-HTML-Seite tragen Sie die HTML-Seite ein, die beim Start der Webapplikation geöffnet werden soll.
5. Wenn erforderlich, tragen Sie einen Applikationsnamen ein.
6. Als "Dateien mit dynamischem Inhalt" ergänzen Sie die Reihe von Dateinamenserweiterungen, falls erforderlich. Sie tragen nur solche Dateinamenserweiterungen ein, die auch AWP-Kommandos enthalten.
7. Die Nummer für den Web-Control-DB und für die Fragment-DB-Startnummer können Sie beibehalten, wenn Sie nicht bereits durch ihr Anwenderprogramm belegt werden.
8. Klicken Sie auf die Schaltfläche "Erzeugen", um DBs aus den Quelldateien zu erzeugen. Die erzeugten Datenbausteine werden in der Projektnavigation im Ordner "Systembausteine" (Unterordner "Webserver") abgelegt.
9. Markieren Sie in der Netzsicht die CPU, die geladen werden soll, und wählen Sie den Befehl "Laden in Gerät" im Menü "Online", um die Bausteine zu laden. Implizit wird vor dem Laden die Übersetzung der Bausteine angestoßen.  
Falls bei diesem Prozess Fehler gemeldet werden, müssen Sie diese Fehler beheben, bevor Sie die Konfiguration laden können.

## Aufbau des PLC-Programms

Ihr Anwenderprogramm muss die Anweisung "WWW" aufrufen, damit die Webapplikation, d. h. die anwenderdefinierten Webseiten, überhaupt auf den Standard-Webseiten der CPU verfügbar sind und aufgerufen werden können.

Der Web-Control-DB, den Sie aus den Quelldateien generiert haben, ist der Eingangsparameter (CTRL\_DB) für die Anweisung "WWW". Der Web-Control-DB referenziert den Inhalt der anwenderdefinierten Webseiten, der in den Fragment-DBs codiert ist, und enthält Status- und Steuerinformationen.

## Aufruf der Anweisung "WWW" im Anlaufprogramm

Wenn Sie angeforderte Webseiten nicht durch das Anwenderprogramm beeinflussen wollen, dann genügt ein einmaliger Aufruf der Anweisung "WWW" in einem Anlauf-OB. Die Anweisung initialisiert die Kommunikation zwischen dem Webserver und der CPU.

## Aufruf der Anweisung "WWW" im zyklischen Programm

Der Aufruf der Anweisung "WWW" kann auch in einem zyklisch bearbeiteten OB (z. B. dem OB 1) erfolgen. Dies bietet den Vorteil, dass auf Anfragen des Webserver im

Anwenderprogramm eingegangen werden kann. Dazu müssen manuelle Fragmente verwendet werden.

In diesem Fall müssen Sie Informationen aus dem Web-Control-DB auswerten, um die angeforderte Webseite bzw. das angeforderte Fragment zu identifizieren. Andererseits müssen Sie im Anwenderprogramm ein Bit setzen, um nach Bearbeitung der Webseiten-Anforderung die vom Webserver auszuliefernde Webseite explizit freizugeben.

Der Aufbau des Web-Control-DBs ist im folgenden Abschnitt beschrieben.

## Web-Control-DB

Der Web-Control-DB (Standardmäßig DB 333) wird von STEP 7 erzeugt und enthält Information zum Aufbau der Anwenderseiten, zum Status der Kommunikation und eventuell aufgetretenen Fehlern.

Neben dem Web-Control-DB werden noch weitere so genannte Fragment-DBs erzeugt. Diese Fragment-DBs (es kann auch nur ein Fragment-DB sein) werden im Web-Control-DB referenziert. Die Fragment-DBs enthalten die in Fragmenten codierten Webseiten und Metadaten, z. B. Bilder. Der Inhalt der Fragment-DBs ist nicht durch das Anwenderprogramm änderbar. Sie werden automatisch angelegt und dienen nur der Datenhaltung.

Auf die Status- und Steuervariablen des Web-Control-DB greifen Sie symbolisch zu.

Im Folgenden sind die Variablen des Web-Control-DBs aufgeführt, die Sie für die Statusauswertung und für die Steuerung der Interaktion benötigen.

Aus dem Web-Control-DB können Sie zweierlei Informationen gewinnen:

- Globale Statusinformation: Nicht an eine konkrete Webseitenanforderung (Request) gebunden.
- Request Status- und Steuerinformationen: Informationen zu anstehenden Requests.

## Globale Statusinformation

|  |  |
|--|--|
| "WEB-Control_DB".commandstate.init         | Aktiviert und initialisiert die Webapplikation.  |
| "WEB-Control_DB".commandstate.deactivate   | Deaktiviert die Webapplikation.  |
| "WEB-Control_DB".commandstate.initializing | Die Webapplikation wird initialisiert (Web-Control-DB lesen, etc.).  |
| "WEB-Control_DB".commandstate.error        | Webapplikation konnte nicht initialisiert werden. Der Grund ist in "WEB-Control_DB".commandstate.last_error codiert. |
| "WEB-Control_DB".commandstate.deactivating | Die Webapplikation wird beendet.   |
| "WEB-Control_DB".commandstate.initialized  | Die Webapplikation wurde initialisiert und ist bereit.   |
| "WEB-Control_DB".commandstate.last_error   | Wertetabelle der möglichen Fehler siehe nächste Tabelle.   |

| Last_error | Beschreibung   |
|------------|--|
| 1          | Fragment-DB ist inkonsistent (passt nicht zum Web-Control-DB).   |
| 2          | Eine Webapplikation mit diesem Namen existiert schon.  |
| 3          | Speicherproblem beim Initialisieren im Webserver.  |
| 4          | Die Daten im Web-Control-DB sind inkonsistent.   |
| 5          | Ein Fragment-DB ist nicht verfügbar (nicht geladen).   |
| 6          | Ein Fragment-DB hat keine AWP-Kennung.   |
| 7          | Das Enum Fragment (enthält die Texte und Informationen für die Enum-Typen) ist nicht verfügbar.          |
| 8          | Ein über die Command-Flags im Web-Control-DB angeforderte Aktion ist im aktuellen Zustand nicht erlaubt. |
| 9          | Webapplikation ist nicht initialisiert (wenn keine neue Initialisierung nach Deaktivierung).             |
| 10         | Webserver ist deaktiviert.   |
| ...        | Last_error wird zurückgesetzt, wenn die Webapplikation erfolgreich initialisiert wurde.                  |

### Request-Statusinformation

Die Request-Statusinformation ist gebunden an einen von vier möglichen Requests,  $x = [1 \dots 4]$ .

|   |   |
|---|---|
| "WEB-Control_DB".requesttab[x].idle     | Es gibt nichts zu tun.  |
| "WEB-Control_DB".requesttab[x].waiting  | Das Anwenderprogramm muss auf einen Request aus einem manuellen Fragment reagieren und die Weiterverarbeitung im Webserver explizit anstoßen. |
| "WEB-Control_DB".requesttab[x].sending  | Der Webserver ist mit der Bearbeitung des Requests/Fragments beschäftigt.   |
| "WEB-Control_DB".requesttab[x].aborting | Die TCP-Verbindung wird vom Webserver geschlossen.  |

### Request-Steuerinformation

Die Request-Steuerinformation ist gebunden an einen von vier möglichen Requests,  $x = [1 \dots 4]$ .

|   |  |
|---|--|
| "WEB-Control_DB".requesttab[x].continue | Gibt das gerade bearbeitete Fragment zum Senden frei. Die Bearbeitung des nächsten Fragments wird angestoßen.              |
| "WEB-Control_DB".requesttab[x].repeat   | Gibt das gerade bearbeitete Fragment zum Senden frei. Das Fragment wird anschließend erneut bearbeitet.                    |
| "WEB-Control_DB".requesttab[x].abort    | Schließt die TCP-Verbindung.   |
| "WEB-Control_DB".requesttab[x].finish   | Gibt das gerade bearbeitete Fragment zum Senden frei. Stoppt die weitere Bearbeitung von Fragmenten (beendet den Request). |

**Beispiel:**

Die Variable für den DB ist: "WEB-Control\_DB". Ob bei der Initialisierung der Webapplikation Fehler aufgetreten sind, können Sie anhand des Bits "WEB-Control\_DB".commandstate.error im Anwenderprogramm abfragen.

Für den Fall, dass ein Fehler aufgetreten ist, können Sie den Fehler über den Wert von "WEB-Control\_DB".commandstate.last\_error analysieren.

**Interaktion mit dem Anwenderprogramm**

Mithilfe von manuellen Fragmenten erreichen Sie, dass das Anwenderprogramm synchron auf Browsereingaben reagiert, sodass die zurückzugebende Webseite vom Anwenderprogramm aufbereitet werden kann.

**Fragmenttyp**

Um im Anwenderprogramm auf die empfangenen Daten zu reagieren, muss für das Fragment, das die Daten schreibt, der Fragmenttyp "manual" (für "manuelle Seiten") verwendet werden:  
<!-- AWP\_Start\_Fragment Name="testfrag" ID="1" Type="manual" -->

Das Übertragen der Werte an den Webserver der CPU erfolgt bei automatischen und manuellen Seiten immer auf die gleiche Weise:

**Beispiel:**

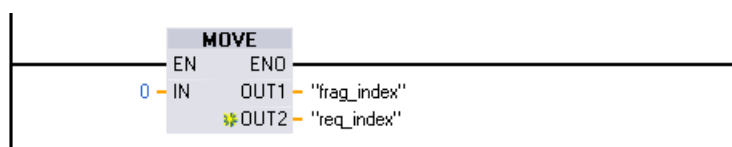
```
<form method="POST" action="">
<p>
<input type="submit" value="Set new value">
<input type="text" name="'Velocity"' size="20">
</p>
</form>
```

**Anwenderprogramm für manuelle Fragmente**

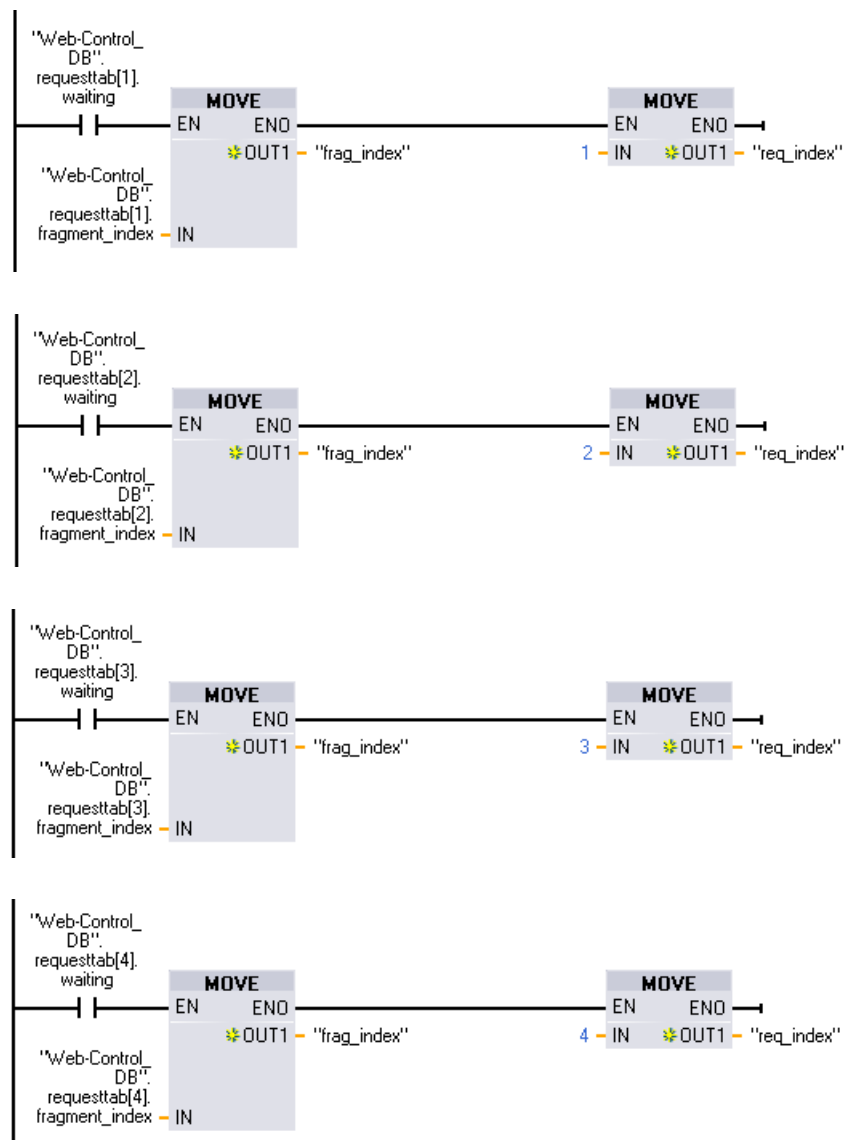
Im Anwenderprogramm der CPU muss bei der Verwendung manueller Seiten die Anweisung "WWW" zyklisch aufgerufen werden.

Um auf im Browser eingegebene Werte zu reagieren, ist es notwendig, den Request, den die manuelle Seite an den Webserver stellt, im Anwenderprogramm auszuwerten. Dazu muss der Web-Control-DB (z. B. DB 333) zyklisch auf anstehende Requests untersucht werden. Im Abschnitt "requesttab" des Web-Control-DBs befindet sich das Array, das vier Requests verwaltet. Jedes Element des Arrays enthält dabei, in einer Struktur, Informationen zum jeweiligen Request.

Ein einfaches Programmbeispiel zeigt, wie anhand der Variablen des Web-Control-DBs auf anstehende Requests geprüft wird.







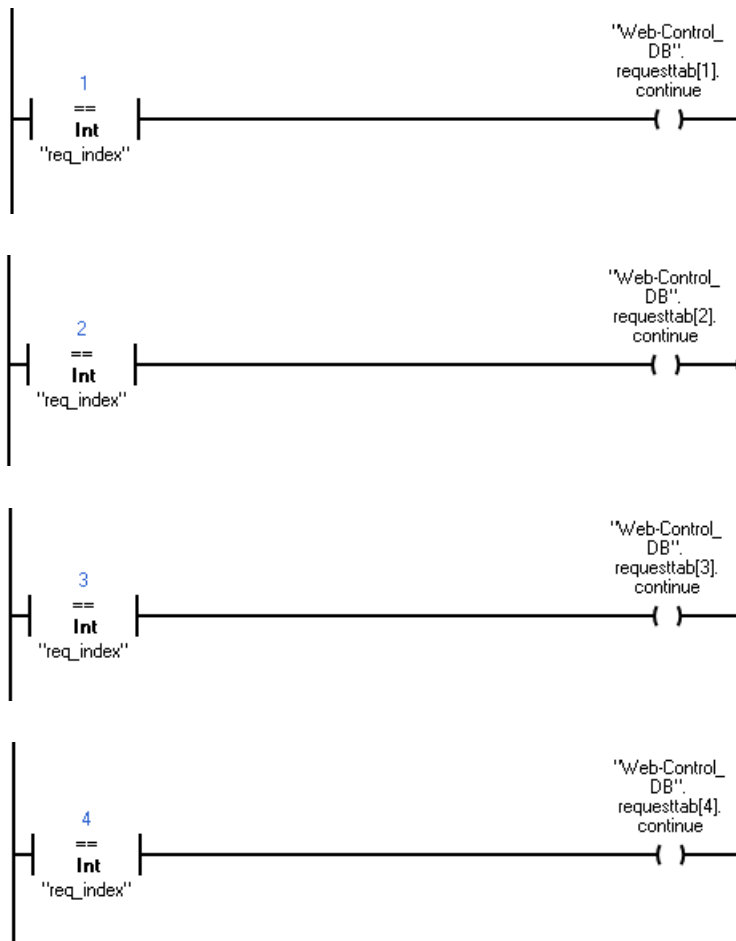
Für den Fall, dass ein Request vorliegt, schreibt dieser Programmabschnitt die Fragment-ID in die Variable #frag\_index und die Request-Nr. (Wertebereich 1-4) in die Variable #req\_index.

Mit den so gewonnenen Informationen kann die im Request übertragene Information jetzt im Programm für jede Fragment-ID gesondert bearbeitet werden (z. B. Plausibilitätsprüfung).

Wenn die Bearbeitung des Requests durch das Programm abgeschlossen ist, muss der Request beantwortet und der entsprechende Eintrag unter "requesttab" des Web-Control-DBs (z. B. DB 333) wieder zurückgesetzt werden.

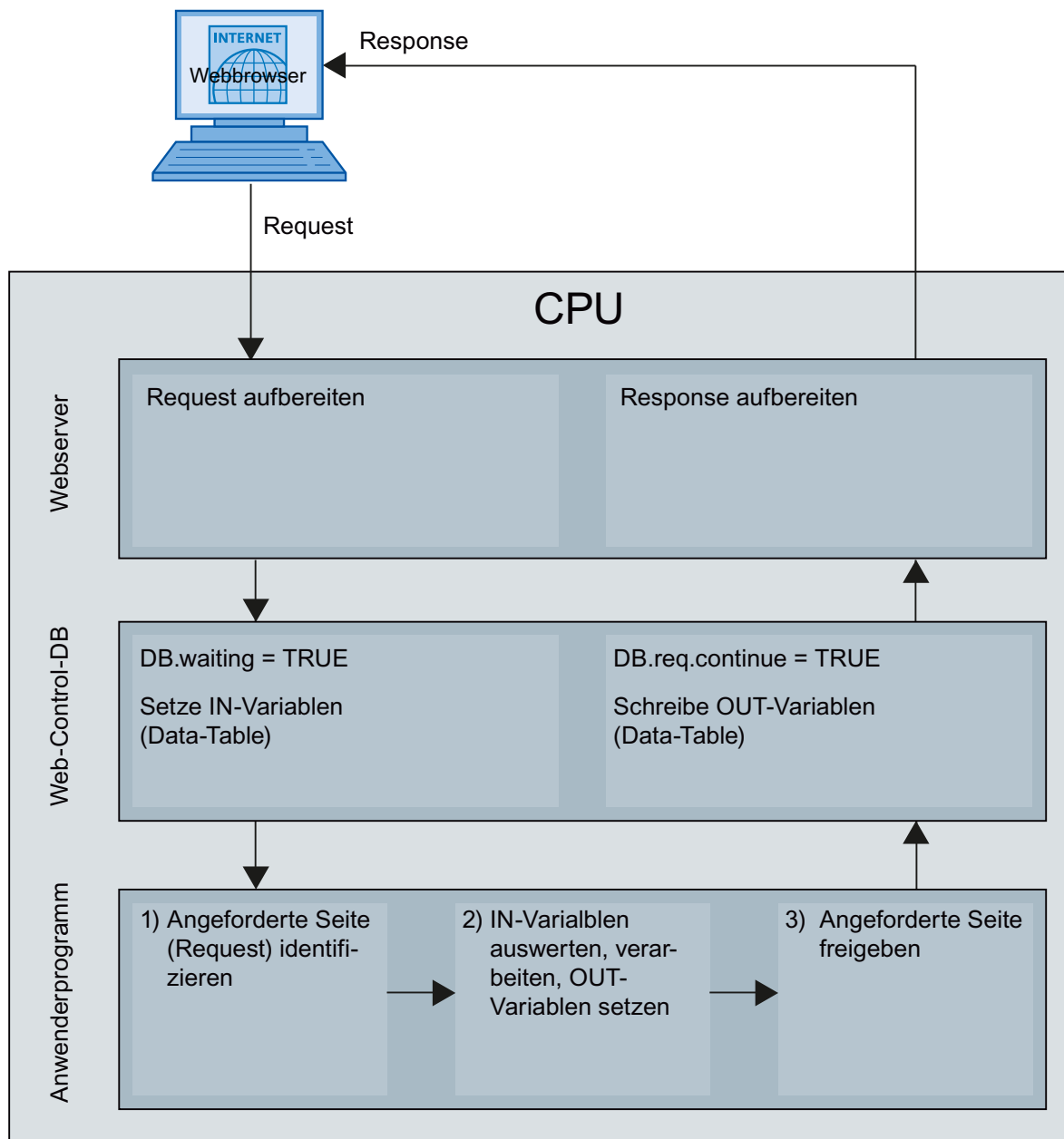
Ein einfaches Programmbeispiel zur Beantwortung von Requests:

8.1 Geräte und Netze konfigurieren



### Prinzipieller Ablauf eines Browser-Requests mit Interaktion des Anwenderprogramms

Das folgende Bild zeigt den vereinfachten, prinzipiellen Ablauf vom Webbrowser-Request über die Auswirkungen auf Web-Control-DB-Inhalte und die notwendigen Aktionen des Anwenderprogramms bis zum Zurückschicken der aufbereiteten Webseite (Response).



### Anzeigen benutzerdefinierter Webseiten im Browser

#### Webseiten im Browser anzeigen

Die Webseiten rufen Sie von den Standard-Webseiten des Webbrowsers auf.

Die Standard-Webseiten verfügen neben den anderen Links in der Navigationsleiste einen Link auf "Anwenderseiten".

Wenn Sie auf den Link "Anwenderseiten" klicken, öffnet ihr Webbrowser die Webseite, die Sie als Start-HTML-Seite konfiguriert haben.

### Anwenderdefinierte Webseiten mehrsprachig erstellen

Sie haben die Möglichkeit, Ihre anwenderdefinierten Webseiten in jeweils unterschiedlichen Landessprachen zur Verfügung zu stellen.

### Voraussetzungen

Die sprachabhängigen HTML-Seiten sind in einer Ordnerstruktur abzulegen, die Ordner mit den jeweiligen Sprachkürzeln enthält:



### Festgelegte Sprachkürzel

Die Sprachkürzel "de", "en", "fr", "es", "it" und "zh" sind festgelegt. Weitere Sprachordner bzw. anders bezeichnete Sprachordner werden nicht unterstützt.

Weitere Ordner in derselben Ordnerhierarchie für andere Dateien können Sie bei Bedarf anlegen; z. B. einen Ordner "img" für Bilder und einen Ordner "script" für Javascript-Dateien.

### Sprachumschaltung für anwenderdefinierte Webseiten

### Voraussetzungen

Die HTML-Seiten befinden sich in den vordefinierten Sprachordnern, z. B. HTML-Seiten mit deutschem Text im Ordner "de", HTML-Seiten mit englischem Text im Ordner "en".

### Konzept der Sprachumschaltung

Die Sprachumschaltung basiert auf einem vordefinierten Cookie mit Namen "siemens\_automation\_language". Wenn das Cookie auf den Wert "de" gesetzt wird, schaltet der Webserver bei der nächsten Webseiten-Anforderung bzw. Webseiten-Aktualisierung auf die Webseite aus dem Ordner "de".

Analog dazu schaltet der Webserver auf die Webseite aus dem Ordner "en" um, wenn das Cookie den Wert "en" annimmt.

## Beispiel für eine Sprachumschaltung

Das Beispiel ist folgendermaßen aufgebaut:

- In den beiden Sprachordnern "de" und "en" befinden sich die sprachabhängigen HTML-Dateien gleichen Namens, z. B. "langswitch.html". Die anzuzeigenden Texte innerhalb der beiden Dateien sind entsprechend den Ordnernamen deutsch bzw. englisch.
- Zusätzlich befindet sich ein Ordner "script" in der Ordnerstruktur, in dem sich die Javascript-Datei "lang.js" befindet. In diese Datei sind Funktionen ausgelagert, die für die Sprachumschaltung benötigt werden.

### Aufbau der Datei "langswitch.html" (Ordner "de")

Im Header der Datei sind die Metadaten "Content-Language", charset und Pfad zur Javascriptdatei festgelegt.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="de">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Sprachumschaltung Deutsche Seite</title>
<script type="text/javascript" src="script/lang.js" ></script>
</head>
```

Im Body der Datei ist die Sprachauswahl mithilfe des HTML-Elements "select" realisiert. Das select-Element leitet eine Auswahlliste ein und enthält die Optionen "de", beschriftet mit "Deutsch" und "en", beschriftet mit "Englisch"; "de" ist vorselektiert.

Mit Hilfe des event-handlers "onchange" wird die Funktion "DoLocalLanguageChange(this)" aufgerufen. Der Parameter "this" übergibt das select-Objekt mit der selektierten Option an diese Funktion. "onchange" führt bei jeder Änderung der Option zum Aufruf der Funktion.

```
<!-- Language Selection -->
<table>
  <tr>
    <td align="right" valign="top" nowrap>
      <!-- change language immediately on change of the selection
-->
      <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
        <option value="de" selected >Deutsch</option>
        <option value="en" >Englisch</option>
      </select>
    </td>
  </tr>
</table>
<!-- Language Selection End-->
```

### Aufbau der Datei "langswitch.html" (Ordner "en")

Der Header der HTML-Datei mit englischem Text ist analog zur HTML-Datei mit deutschem Text aufgebaut.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
```

```
<head>
<meta http-equiv="Content-Language" content="en">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Language switching english page</title>
<script type="text/javascript" src="script/lang.js" ></script>
```

Im Body der Datei ist die Sprachauswahl ebenfalls mit Hilfe des HTML-Elements "select" realisiert. Im Unterschied zur deutschen HTML-Datei ist die englische Option vorselektiert und der Text bzw. die Labels sind englisch.

```
<!-- Language Selection -->
<table>
  <tr>
    <td align="right" valign="top" nowrap>
      <!-- change language immediately on change of the selection
-->
      <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
        <option value="de" >German</option>
        <option value="en" selected >English</option>
      </select>
    </td>
  </tr>
</table>
<!-- Language Selection End-->
```

### Aufbau der Datei "lang.js" (im Ordner "script")

In der Javascript-Datei ist die Funktion "DoLocalLanguageChange" definiert, die ihrerseits die Funktion "SetLangCookie" mit dem Sprachauswahl-Wert aufruft. SetLangCookie setzt Cookie-Name und Cookie-Wert zusammen und setzt das Cookie schließlich über die entsprechende document.cookie-Eigenschaft. Damit der Webserver auf das Setzen des Cookies mit der Anzeige der gewünschten Sprache reagiert, muss anschließend die Webseite neu geladen werden (top.window.location.reload).

```
function DoLocalLanguageChange(oSelect) {
    SetLangCookie(oSelect.value);
    top.window.location.reload();
}

function SetLangCookie(value) {
    var strval = "siemens_automation_language=";
    // this is the cookie by which the webserver
    // detects the desired language
    // this name is required by the webserver
    strval = strval + value;
    strval = strval + "; path=/ ";
    // set path to the application, since otherwise
    // path would be set to the requesting page
    // would not get the cookie.
    // The path for user defined applications follows this
sample:
    // path=/awp/<application name>/<pagename>
```

```
// example: path=/awp/myapp/myappstartpage.htm
//(where myapp is the name of the web application
// entered in the web server properties of the cpu)
/*
use expiration if this cookie should live longer
than the current browser session
var now      = new Date();
var endtime = new Date(now.getTime() + expiration);
strval = strval + "; expires=" + endtime.toGMTString()
+ ";";
*/
document.cookie = strval;
}
```

## Weitere Konfigurationen

### Konfigurieren von weiteren Funktionen

Das Automatisierungssystem S7-1200 verfügt über zahlreiche weitere Funktionen, die als integrierte CPU-Funktionen nutzbar sind oder über steckbare Module (z. B. Kommunikationsmodule) zur Verfügung stehen. Die Beschreibung finden Sie über die unten aufgeführten Links.

### Siehe auch

Überblick über die Punkt-zu-Punkt-Kommunikation (Seite 868)

Allgemeines zu schnellen Zählern (Seite 863)

PID\_Compact V1 konfigurieren (Seite 4078)

PID\_3Step V1 konfigurieren (Seite 4111)

Motionfunktionalität der CPU S7-1200 (Seite 4127)

#### 8.1.4.4 S7-1200-CM/CP

##### S7-1200-CM/CP

##### Telecontrol S7-1200

##### CP 1242-7

##### CP 1242-7 Projektierung

##### Betriebsarten des GPRS-CP

##### Betriebsarten des CP

Der CP 1242-7 ermöglicht einer S7-1200, als GPRS-Station mit einer Zentrale oder anderen entfernten Stationen über das GSM-Netz zu kommunizieren. Für die Kommunikation über GPRS wird der CP auf eine der folgenden Betriebsarten eingestellt:

- **Telecontrol**  
Diese Betriebsart des CP ermöglicht der GPRS-Station den Datenaustausch mit folgenden Partnern:
  - **Kommunikation mit dem Telecontrol-Server**  
Diese Betriebsart des CP ermöglicht der GPRS-Station den Datenaustausch mit einem Telecontrol-Server.  
Der Telecontrol-Server ist ein am Internet angeschlossener PC mit der Applikation "TELECONTROL SERVER BASIC". Er befindet sich in der Regel in der Zentrale und dient der Überwachung und Steuerung der entfernten GPRS-Stationen. Über die integrierte OPC-Schnittstelle können Daten mit dem OPC-Client eines zentralen Leitsystems ausgetauscht werden.  
Der Telecontrol-Server-PC wird nicht in STEP 7 projektiert. Die Applikation "TELECONTROL SERVER BASIC" hat eine eigene Projektierungsoberfläche.
  - **Kommunikation mit einer anderen entfernten GPRS-Station**



Die Telegramme werden über den Telecontrol-Server geleitet.

- Kommunikation mit einer Engineering-Station (für TeleService)

Die Kommunikation mit dem Telecontrol-Server erfolgt über das GSM-Netz und das Internet.

Voraussetzungen für diese Betriebsart sind eine SIM-Karte mit freigeschaltetem GPRS-Dienst und ein für den CP erreichbarer Telecontrol-Server.

- GPRS direkt  
Diese Betriebsart des CP dient der direkten Kommunikation zwischen entfernten Stationen über das GSM-Netz. Es wird kein Telecontrol-Server benötigt.  
Damit Netzteilnehmer in öffentlichen Funknetzen direkt erreichbar sind, müssen diese über eine feste Adresse angesprochen werden können. Hierzu werden SIM-Karten mit fester IP-Adresse eingesetzt, die es ermöglichen, die Stationen direkt zu adressieren.  
Die möglichen Kommunikationsdienste und Sicherheitsfunktionen (z. B. VPN) hängen dabei vom Angebot des Netzbetreibers ab.  
Mögliche Kommunikationspartner der GPRS-Station mit CP 1242-7 in der Betriebsart "GPRS direkt" sind:
  - Ein für den CP über eine IP-Adresse erreichbarer Teilnehmer (GPRS-Station mit CP 1242-7)
  - Eine Engineering-Station (für TeleService)

## Verbindungsaufbau bei der GPRS-Kommunikation

### Verbindungsmodi

- Betriebsart "GPRS direkt"  
In der Betriebsart "GPRS direkt" gibt es keine unterschiedlichen Verbindungsmodi.
- Betriebsart "Telecontrol"  
Der CP kann für folgende Verbindungsmodi projektiert werden.
  - Verbindungsmodus "permanent"  
Es besteht eine dauerhafte TCP-Verbindung zum Telecontrol-Server. Nach dem Verbindungsaufbau besteht eine dauerhafte TCP-Verbindung zum Telecontrol-Server, auch wenn nicht permanent Daten übertragen werden.
  - Verbindungsmodus "temporär"  
Eine Verbindung zum Telecontrol-Server wird nur bei Bedarf aufgebaut.

Wenn eine TCP-Verbindung aufgebaut ist, erfolgt das Versenden von Prozessdaten, sobald die Telecontrol-Anweisungen in der CPU aufgerufen werden.

Eine Verbindung wird immer durch den CP aufgebaut. Wenn eine vom CP aufgebaute Verbindung unterbrochen wird, dann versucht der CP selbständig, die Verbindung wieder aufzubauen.

### Auslösen des Verbindungsaufbaus für permanente Stationen (Betriebsart "Telecontrol")

In der Betriebsart "Telecontrol" wird der Aufbau der permanenten Verbindung zum Telecontrol-Server beim Anlauf der Station aufgebaut. Bei Verbindungsunterbrechung kann der Aufbau der Verbindung durch eine Weck-SMS (siehe unten) veranlasst werden.

### Auslösen des Verbindungsaufbaus bei temporären Stationen (Betriebsart "Telecontrol")

Bei "temporären" Stationen kann ein Verbindungsaufbau durch folgende Ereignisse ausgelöst werden:

- Ereignis in der lokalen CPU, das programmtechnisch ausgewertet werden müssen.  
Programmtechnisch sind zwei Fälle zu unterscheiden:
  - Ereignisse, die zu einem einmaligen Verbindungsaufbau führen (z. B. Alarmer oder Kommandos eines Bedieners).
  - Ablauf eines Zeitintervalls, das zu zyklischem Verbindungsaufbau führt (z. B. einmal täglich zur Datenübertragung)
- Anforderung durch einen Kommunikationspartner (OPC-Client oder S7-Station)  
Die Anforderung des Kommunikationspartners führt zu einem Verbindungsaufbau.
- Anforderung durch eine Engineering-Station für TeleService  
Die Anforderung, welche vom Telecontrol-Server bzw. TeleService-Gateway vermittelt wird, muss programmtechnisch nicht ausgewertet werden.
- Weck-SMS des Telecontrol-Servers  
Die Weck-SMS kann spontan am Telecontrol-Server ausgelöst werden. Auch ein zyklisches Versenden kann im Telecontrol-Server projektiert werden.
- Weckruf eines Telefons  
Der Weckruf kann von einem Telefon verschickt werden, dessen Rufnummer im STEP 7-Projekt autorisiert ist. Das Telefon muss die CLIP-Funktion (Übertragung der eigenen Rufnummer) unterstützen.  
Der Verbindungsaufbau mit dem (Haupt-) Telecontrol-Server wird veranlasst.
- Weck-SMS eines Telefons  
Die Weck-SMS kann von einem Telefon verschickt werden, dessen Rufnummer im STEP 7-Projekt autorisiert ist. Das Telefon muss die CLIP-Funktion (Übertragung der eigenen Rufnummer) und das Versenden von SMS unterstützen.  
Der Verbindungsaufbau mit dem in der SMS spezifizierten Telecontrol-Server wird veranlasst.

Beim Wecken einer temporären Station werden alle Daten übertragen, wenn diese sich seit der letzten Datenübertragung geändert haben.

### Auslösen des Verbindungsaufbaus in der Betriebsart "GPRS direkt"

In der Betriebsart "GPRS direkt" wird ein Verbindungsaufbau durch folgende Ereignisse ausgelöst:

- Ereignis in der lokalen CPU, das programmtechnisch ausgewertet wird.
- Anforderung durch einen Kommunikationspartner (keine Engineering-Station)  
Die Anforderung in dem empfangenen Telegramm des Kommunikationspartners wird programmtechnisch durch Aufruf der Telecontrol-Anweisungen ausgewertet.
- Anforderung durch eine Engineering-Station für TeleService  
Die Anforderung, welche vom Telecontrol-Server bzw. TeleService-Gateway vermittelt wird, muss programmtechnisch nicht ausgewertet werden.

## Weckberechtigung durch "Autorisierte Rufnummern"

Voraussetzung dafür, dass der CP eine SMS akzeptiert, ist die Autorisierung des sendenden Kommunikationspartners mithilfe seiner Rufnummer. Diese Rufnummern werden in STEP 7 beim CP in der Liste "Autorisierte Rufnummern" projiziert.

---

### Hinweis

#### "Autorisierte Rufnummern" im STEP 7-Projekt

- Eine hier eingetragene Rufnummern berechtigt den Absender, der diese Rufnummer mit überträgt, einen Verbindungsaufbau auszulösen.
  - Wenn in der Liste ein Stern (\*) eingetragen wird, dann akzeptiert der CP SMS von allen Absendern.
  - Wenn die Liste leer ist, dann kann der CP nicht zum Verbindungsaufbau geweckt werden.
-

## Weck-SMS

Abhängig von der Verbindungsart und dem auslösenden Server bzw. dem vermittelnden TeleService-Server muss in der Weck-SMS folgender Text übertragen werden:

- Für Telecontrol-Verbindungen:
  - Text für die Weck-SMS für den Aufbau einer Verbindung zum Telecontrol-Server:  
TELECONTROL
  - Text für die Weck-SMS für den Aufbau einer Verbindung zum Haupt-Telecontrol-Server:  
TELECONTROL MAIN
  - Text für die Weck-SMS für den Aufbau einer Verbindung zum Ersatz-Telecontrol-Server:  
TELECONTROL BACKUP

Die Projektierung der Telecontrol-Server für den GPRS-CP wird in STEP 7 unter "Telecontrol-Schnittstelle > Betriebsart > Haupt- bzw. Ersatz-Telecontrol-Server" vorgenommen.

---

### Hinweis

#### Wecken durch ein Mobiltelefon

- In einer Weck-SMS kann einer der oben aufgeführten Texte verwendet werden.
  - Durch einen Weckruf verbindet sich die Station immer mit dem Haupt-Telecontrol-Server.
- 

- Für TeleService-Verbindungen:
  - Text für die Weck-SMS für den Aufbau einer Verbindung über den ersten projektierten TeleService-Server:  
TELESERVICE  
oder  
TELESERVICE 1
  - Text für die Weck-SMS für den Aufbau einer Verbindung über den zweiten projektierten TeleService-Server:  
TELESERVICE 2

Die Projektierung der TeleService-Server für den GPRS-CP wird in STEP 7 unter "Telecontrol-Schnittstelle > TeleService-Berechtigung > 1. bzw. 2. TeleService-Server" vorgenommen.

## Bevorzugte GSM-Netze

### Auswahl der bevorzugten GSM-Netze

Zur Auswahl, in welche GSM-Netze sich der CP 1242-7 bevorzugt einwählen soll, werden folgende Optionen angeboten:

- **Automatische Einwahl**  
Der CP wählt sich nach Vorgaben der SIM-Karte mit höchster Priorität in das GSM-Netz des Vertrags-Netzwerkbetreibers ein. Wenn die Einwahl in das Vertrags-Netz nicht erfolgreich ist, wählt sich der CP in andere GSM-Netze ein, mit denen der Vertrags-Netzwerkbetreiber Roaming-Verträge abgeschlossen hat und deren Zugangsdaten auf der SIM-Karte hinterlegt sind.
- **Nur Vertragsnetz**  
Der CP wählt sich nur in das GSM-Netz des Vertrags-Netzwerkbetreibers ein, dessen SIM-Karte im CP gesteckt ist. Kein Roaming.
- **Vertragsnetz und alternative Netze**  
Der CP wählt sich in erster Priorität in das Vertragsnetz ein. Wenn die Einwahl in das Vertrags-Netz nicht erfolgreich ist, wählt sich der CP in alternative GSM-Netze ein, die in der Liste der bevorzugten Netzbetreiber mit abnehmender Priorität eingetragen werden.  
Die alternativen Netze werden in die Liste als "Public Land Mobile Network" (PLMN) eingetragen. PLMN setzt sich zusammen aus Mobile Country Code (MCC) und Mobile Network Code (MNC).  
Beispiel: 26276  
Dies ist das PLMN für das Testnetz der Siemens AG mit MCC = 262 und MNC = 76.

## Uhrzeitsynchronisation NTP-Verfahren - Parametrierung

### NTP-Verfahren (NTP: Network Time Protocol)

Beim NTP-Verfahren sendet die Baugruppe in regelmäßigen Zeitabständen Uhrzeitanfragen an einen oder mehreren NTP-Server. Aus den Antworten der Server wählt der CP die genaueste Uhrzeit aus.

Der Vorteil dieses Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg.

Im NTP-Verfahren wird generell UTC (Universal Time Coordinated) übertragen. Dies entspricht GMT (Greenwich Mean Time).

## CP 1242-7 TeleService

### TeleService über GPRS

#### Kommunikationsweg bei TeleService über GPRS

Bei TeleService für entfernte S7-Stationen mit CP 1242-7 über GPRS wird die Verbindung immer über einen Vermittler zwischen Engineering-Station und entfernter S7-Station geführt.

Als Vermittler fungieren alternativ:

- Ein Telecontrol-Server (und ggf. ein Ersatz-Telecontrol-Server)  
Der Telecontrol-Server kann ein separater PC sein oder als Applikation "TELECONTROL SERVER BASIC" auf der Engineering-Station installiert sein.
- Ein TeleService-Server  
Ein TeleService-Server wird eingesetzt, wenn kein Telecontrol-Server zur Verfügung steht, das heißt, wenn der CP in der Betriebsart "GPRS direkt" arbeitet.

Der Telecontrol-Server oder der TeleService-Server kann über LAN oder Internet mit der Engineering-Station verbunden sein, von der aus die TeleService-Funktion aufgerufen wird.

#### Voraussetzungen für TeleService über GPRS

- Ein Telecontrol-Server oder ein TeleService-Server
- Das STEP 7-Projekt mit den erforderlichen Daten für die GPRS-Kommunikation

#### Hinweis zur Projektierung

Telecontrol-Server und TeleService-Server werden in STEP 7 nicht projiziert.

#### Siehe auch

Verbindungsaufbau bei der GPRS-Kommunikation (Seite 667)

Betriebsarten des GPRS-CP (Seite 666)

#### TeleService-Verbindung über GPRS aufbauen

#### Verbindungsaufbau bei TeleService über GPRS

Die Anforderung des Verbindungsaufbaus wird durch die Engineering-Station ausgelöst und durch eine Weck-SMS an die Station übertragen. Der CP 1242-7 in der S7-1200-Station baut über das GSM-Netz und das Internet eine Verbindung mit der Engineering-Station auf.

## Starten des TeleService über GPRS

Starten Sie den TeleService über GPRS folgendermaßen:

1. Selektieren Sie im Projekt auf der autorisierten Engineering-Station die entfernte S7-Station, zu der Sie eine TeleService-Verbindung über GPRS aufbauen möchten.
2. Öffnen Sie den Dialog "Online verbinden" alternativ über folgende Möglichkeiten:
  - Schaltfläche "Online verbinden"
  - Kontextmenü "Online verbinden" (rechte Maustaste)
  - Menü "Online" > "Online verbinden"Der Dialog "Online verbinden" öffnet sich.
3. Wählen Sie aus der Klappliste "Typ der PG/PC-Schnittstelle" den Schnittstellentyp "TeleService GPRS" aus.
4. Wählen Sie in der Klappliste "PG/PC-Schnittstelle" die Option "GPRS TeleService Board" aus, wenn diese nicht automatisch angezeigt wird.
5. Klicken Sie auf das "Verbinden"-Symbol neben der Klappliste "PG/PC-Schnittstelle". Der Dialog "Fernverbindung aufbauen" öffnet sich.
6. Nehmen Sie die erforderlichen Einstellungen im Dialog "Fernverbindung aufbauen" vor. Details hierzu finden Sie in den Tooltip-Kaskaden von STEP 7.

Folgende Angaben sind für einen erfolgreichen Verbindungsaufbau erforderlich:

## Notwendige Angaben für den Verbindungsaufbau mit der GPRS-Station

Die nachfolgenden Angaben sind im Dialog "Fernverbindung aufbauen" erforderlich:

- IP-Adresse oder DNS-Name des Telecontrol-Servers
- TCP-Port-Nummer des Telecontrol-Servers oder des DSL-Routers, über den die Verbindung zwischen Engineering-Station und entfernter GPRS-Station läuft.
- Server-Passwort der ES zur Authentifizierung der Engineering-Station beim Telecontrol-Server  
Nur erforderlich, wenn in der Applikation "TELECONTROL SERVER BASIC" ein gruppenspezifische Passwort projektiert wurde.
- TeleService-Benutzername  
Siehe CP-Projektierung in STEP 7.
- TeleService-Passwort  
Siehe CP-Projektierung in STEP 7.
- Zugangs-ID des CP  
Nur erforderlich, wenn mehrere CP 1242-7 in der Station vorhanden sind. Siehe CP-Projektierung in STEP 7.

## Status

### Verbindungszustände bei TeleService über GPRS

Folgende Verbindungszustände können im Dialog "Fernverbindung aufbauen" angezeigt werden:

- Nicht verbunden  
Es besteht keine Verbindung zur entfernten GPRS-Station. Der Verbindungsaufbau wurde noch nicht gestartet.

Wenn der Verbindungsaufbau durch Klick auf die Schaltfläche "Verbinden" gestartet wurde, dann werden bei erfolgreichem Verbindungsaufbau folgende Zustände der Reihe nach angezeigt:

- Verbinden mit Telecontrol-Server  
Die Engineering-Station verbindet sich mit dem Telecontrol-Server.
- Warten auf S7-Station  
Die Weck-SMS wurde an die entfernte Station abgesetzt. Warten auf Antwort der Station.
- Authentifizierung an der S7-Station  
Die S7-Station hat über GPRS und Internet eine IP-Verbindung mit der Engineering-Station aufgebaut und prüft die empfangenen Anmelde- und Authentifizierungsdaten.
- Verbunden  
Die Station hat die Verbindung mit der Engineering-Station erfolgreich aufgebaut.

Bei nicht erfolgreichem Verbindungsaufbau können folgende Zustände angezeigt werden:

- Telecontrol-Server nicht erreichbar  
Mögliche Ursachen:
  - Die Verbindung zwischen Engineering-Station und Telecontrol-Server ist unterbrochen.
  - Der Telecontrol-Server ist ausgeschaltet.
- Server-Passwort falsch  
Ursache: Im Dialog wurde das falsche Server-Passwort zur Anmeldung und Authentifizierung am Telecontrol-Server eingegeben.
- S7-Station meldet sich nicht  
Mögliche Ursachen:
  - Die GSM-Kommunikation zwischen Telecontrol-Server und Station ist gestört.
  - Die Verbindung zwischen GSM-Netz und Internet ist gestört.
  - Die Internetverbindung ist gestört.
  - Der Telecontrol-Server konnte keine Weck-SMS verschicken.
  - Der CP hat keine Weck-SMS empfangen.
  - Der SMS-Absender wurde nicht in der Liste der autorisierten Weck-Rufnummern projektiert.



- TeleService-Benutzername oder TeleService-Passwort falsch  
Mögliche Ursachen:
  - Im Dialog wurde der falsche TeleService-Benutzername oder das falsche TeleService-Passwort zur Authentifizierung am CP 1242-7 eingegeben.
  - Der TeleService-Benutzername oder das TeleService-Passwort wurde in STEP 7 nicht projiziert.
- Alle TeleService-Zugangspunkte sind belegt.
- Der CP ist dem Telecontrol-Server nicht bekannt.  
Ursache: Der CP stammt aus einem STEP 7-Projekt, das nicht dem Projekt des Telecontrol-Servers entspricht.
- Keine Ressourcen für TeleService im CP vorhanden: Bitte kontaktieren Sie die Hotline.
- Protokollfehler  
Ursache: Falsches Telegramm oder Telegramm von falschem Teilnehmer. Bitte kontaktieren Sie die Hotline.

## CP 1243-1

### CPU-Abtastzyklus

#### Aufbau des CPU-Abtastzyklus

Der Zyklus, mit dem der CP den Speicherbereich der CPU abtastet, ist aus den folgenden Phasen aufgebaut:

- **Hochpriorie Leseaufträge**  
Für Datenpunkte, welche mit der Priorität "hochprior" projiziert werden, werden die PLC-Variablen alle in jedem Abtastzyklus gelesen.
- **Spontane Schreibaufträge**  
In jedem Zyklus werden die Werte einer bestimmten Anzahl spontaner Schreibaufträge in die CPU geschrieben. Die Anzahl der Variablen, die pro Zyklus geschrieben werden, wird mit dem Parameter "Max. Anzahl der Schreibaufträge" festgelegt. Die Variablen, die diesen Wert übersteigen und somit in einem Zyklus nicht geschrieben werden, werden dann im nächsten oder einem der folgenden Zyklen geschrieben.
- **PLC-Variablen mit Änderungskontrolle**  
Die Werte von PLC-Variablen, die für die automatische Änderungskontrolle der CPU angemeldet sind, werden alle in jedem Abtastzyklus gelesen.
- **Niederpriorie Leseaufträge - anteilig**  
Für Datenpunkte, welche mit der Priorität "niederprior" projiziert werden, werden die Werte ihrer PLC-Variablen anteilig in jedem Abtastzyklus gelesen. Die Anzahl der Variablen, die pro Zyklus gelesen werden, wird mit dem Parameter "Max. Anzahl der Leseaufträge" festgelegt. Die Variablen, die diesen Wert übersteigen und somit in einem Zyklus nicht gelesen werden, werden dann im nächsten oder einem der folgenden Zyklen gelesen.
- **Systemverzögerung**  
Dies ist die Abarbeitungsdauer eines Abtastzyklus.

### Dauer des CPU-Abtastzyklus

Da für den Zyklus keine feste Zeit projektiert werden kann und da den einzelnen Phasen keine feste Anzahl an Objekten zugeordnet ist, ist die Dauer des Abtastzyklus variabel und kann sich dynamisch verändern.

Zusätzlich kann noch eine Zykluspausenzeit projektiert werden, um anderen Prozessen, die auf den Rückwandbus der Station zugreifen, genügend Zeit zu reservieren.

### Datenpunkttypen

Bei der Projektierung der vom CP 1243-1 zu übertragenden Nutzdaten wird jeder Datenpunkt einem protokollspezifischen Datenpunkttyp zugeordnet. Die vom CP unterstützten Datenpunkttypen mit den jeweils kompatiblen S7-Datentypen sind nachfolgend aufgelistet. Sie sind nach Format und Speicherbedarf gruppiert.

### CP 1243-1 (DNP3): Datenpunkttypen und Objektgruppen

Das DNP3-Protokoll spezifiziert Objektgruppen, die sich nach Datentyp und Senderichtung (**out**) bzw. Empfangsrichtung (**in**) unterscheiden.

Tabelle 8-48 Vom CP (DNP3) unterstützte Objektgruppen und Datentypen

| Format (Speicherbedarf)                                | Datenpunkttyp (Objektgruppe) | Richtung: in / out | Kompatible S7-Datentypen   |
|--|------------------------------|--------------------|--|
| • Bit  |                              |                    | • BOOL   |
|  | Binary Input (1)             | in                 |  |
|  | Binary Input Event (2)       | in                 |  |
|  | Binary Output (10)           | out                |  |
|  | Binary Output Event (11)     | out                |  |
|  | Binary Command (12)          | out                |  |
| • Byte<br>• Integer (16 bit)<br>• Integer (32 bit)     |                              |                    | • BYTE, SINT, USINT<br>• INT, UINT, WORD<br>• DINT, UDINT, DWORD |
|  | Counter Static (20)          | in                 |  |
|  | Frozen Counter (22)          | in                 |  |
|  | Counter Static Event (21)    | in                 |  |
|  | Frozen Counter Event (23)    | in                 |  |
|  | Analog Input (30)            | in                 |  |
|  | Analog Input Event (32)      | in                 |  |
|  | Analog Output Status (40)    | out                |  |
|  | Analog Output (41)           | out                |  |
|  | Analog Output Event (42)     | out                |  |
|  | Octet String (110) *         | in                 |  |
|  | Octet String Event (111) *   | in                 |  |
|  | Octet String (110) *         | out                |  |
| • Gleitpunktzahl (32 bit)<br>• Gleitpunktzahl (64 bit) |                              |                    | • REAL<br>• LREAL  |

| Format (Speicherbedarf) | Datenpunkttyp (Objektgruppe) | Richtung: in / out | Kompatible S7-Datentypen |
|-------------------------|------------------------------|--------------------|--------------------------|
|                         | Analog Input (30)            | in                 |                          |
|                         | Analog Input Event (32)      | in                 |                          |
|                         | Analog Output Status (40)    | out                |                          |
|                         | Analog Output (41)           | out                |                          |
|                         | Analog Output Event (42)     | out                |                          |
|                         | Octet String (110) *         | in                 |                          |
|                         | Octet String Event (111) *   | in                 |                          |
|                         | Octet String (110) *         | out                |                          |

\* Zusammenhängende Speicherbereiche können bis zu einer Größe von 255 Byte über Datenpunkttypen der Objektgruppen 110 (Octet String) und 111 (Octet String Event) übertragen werden.

## Übertragungsarten und Ereignisklassen

### Übertragungsarten von Datenpunktwerten

Die Werte von Datenpunkten können verschiedenen Übertragungsarten zugeordnet werden. Der Wert eines Datenpunkts wird entweder nach Abfrage des Kommunikationspartners oder spontan als Ereignis gesendet. Folgende Ereignisklassen sind möglich:

- **Übertragung nach Aufruf**  
Der Wert des Datenpunkts wird nur nach Aufruf durch den Kommunikationspartner übertragen.
- **Jeder Wert getriggert**  
Jeder einzelne geänderte Wert im Abbildspeicher des CP wird übertragen.
- **Ereignisklasse ...**  
Der Wert wird getriggert als Ereignis der Klasse 1, 2 oder 3 übertragen.
- **Aktueller Wert getriggert**  
Nur derjenige Wert eines Datenpunkts wird übertragen, der zum Zeitpunkt des Triggers, welcher die Übertragung auslöst, im Abbildspeicher des CP gespeichert ist.

### Analogwert-Vorverarbeitung

Einige CPs haben eine Analogwert-Vorverarbeitung, bei der unter anderem nachfolgend beschriebene Funktionen unterstützt werden.

### Glättungsfaktor

Schnell schwankende Analogwerte können mit Hilfe der Glättungsfunktion beruhigt werden.

Die Glättungsfaktoren werden wie bei S7-Analogeingabebaugruppen nach folgender Formel berechnet.

$$y_n = \frac{x_n + (k - 1)y_{n-1}}{k}$$

mit

$y_n$  = geglätteter Wert im aktuellen Zyklus  $n$

$x_n$  = erfasster Wert im aktuellen Zyklus  $n$

$k$  = Glättungsfaktor

Als Glättungsfaktor sind folgende Werte für die Baugruppe projektierbar.

- 1 = Keine Glättung
- 4 = Schwache Glättung
- 32 = Mittelere Glättung
- 64 = Starke Glättung

### Fehlerunterdrückungszeit

Ein Analogwert im Überlaufbereich (32767 / 7FFF<sub>n</sub>) oder Unterlaufbereich (-32768 / 8000<sub>n</sub>) wird für die Dauer der Fehlerunterdrückungszeit nicht übertragen. Dies gilt auch für Life-Zero-Eingänge. Erst nach Ablauf der Fehlerunterdrückungszeit wird der Wert im Über- bzw. Unterlaufbereich übertragen, falls er noch ansteht.

Wenn der Wert vor Ablauf der Fehlerunterdrückungszeit in den Bemessungsbereich zurückfällt, wird der aktuelle Wert sofort übertragen.

Typischer Anwendungsfall für diesen Parameter ist die Unterdrückung von Spitzenstromwerten beim Anlauf leistungsstarker Motoren, die sonst als Störung an die Leitstelle gemeldet würden.

Die Unterdrückung ist abgestimmt auf Analogwerte, die als Rohwerte von den S7-Analog-Eingangsbaugruppen erfasst werden. Diese Baugruppen liefern für alle Eingangsbereiche die genannten Werte für den Überlauf- bzw. Unterlaufbereich, auch für Life-Zero-Eingänge.

Wenn von der CPU vorverarbeitete Fertigwerte im Merkerbereich oder in einem Datenbaustein bereitgestellt werden, dann ist eine Störungsunterdrückung nur möglich bzw. sinnvoll, wenn diese Werte im Überlauf- bzw. Unterlaufbereich ebenfalls die genannten Werte von (32767 / 7FFF<sub>n</sub>) bzw. (-32768 / 8000<sub>n</sub>) annehmen. Wenn dies nicht der Fall ist, dann sollte der Parameter für vorverarbeitete Werte nicht aktiviert werden.

### Mittelwertbildung

Mit diesem Parameter werden erfasste Analogwerte als Mittelwerte übertragen.

Der aktuell anstehenden Werte eines Analogwert-Datenpunkts werden zyklisch erfasst und aufsummiert. Die Anzahl der erfassten Werte pro Zeiteinheit ist abhängig vom Lesezyklus der CPU und vom CPU-Abtastzyklus des CP. Aus den aufsummierten Werten wird der Mittelwert gebildet, sobald die Übertragung über einen Zeit-Trigger ausgelöst wird. Danach wird die Aufsummierung zur Bildung des nächsten Mittelwerts neu gestartet.

Der Mittelwert wird auch dann gebildet, wenn die Übertragung des Analogwerttelegramms durch eine Abfrage vom Kommunikationspartner ausgelöst wird. Die Dauer der Mittelungsperiode ist dann die Zeit von der letzten Übertragung (z. B. über den Trigger ausgelöst) bis zum Zeitpunkt der Abfrage. Auch nach dieser Übertragung wird die Aufsummierung zur Bildung des nächsten Mittelwertes neu gestartet.

Falls der erfasste Analogwert im Überlaufbereich (32767 / 7FFF<sub>n</sub>) oder Unterlaufbereich (-32768 / 8000<sub>n</sub>) liegt, kann dieser Wert entweder sofort für die Mittelwertbildung berücksichtigt

werden oder er wird für eine bestimmte Zeit für die Mittelwertbildung ausgeblendet. Das jeweils gewünschte Verhalten wird über den Parameter Fehlerunterdrückungszeit bestimmt:

- Fehlerunterdrückungszeit = 0  
Sobald ein Wert im Überlauf- bzw. Unterlaufbereich erfasst wird, führt dies zum sofortigen Abbruch der Mittelwertbildung. Für die laufende Mittelungsperiode wird der Wert  $32767 / 7FFF_h$  bzw.  $-32768 / 8000_h$  als ungültiger Mittelwert gespeichert und bei der nächsten Auslösung des Analogwerttelegramms übertragen. Danach wird eine neue Mittelwertbildung gestartet. Wenn der Analogwert dann immer noch im Überlauf- oder Unterlaufbereich liegt, dann wird auch dieser Wert wieder sofort als ungültiger Mittelwert gespeichert und bei der nächsten Telegrammauslösung übertragen.
- Fehlerunterdrückungszeit > 0  
Wenn der erfasste Analogwert im Überlauf- bzw. Unterlaufbereich liegt, werden die fehlerhaften Werte nicht in die Mittelwertbildung einbezogen und zwar maximal für die Dauer der Fehlerunterdrückungszeit. Wenn diese Zeitdauer überschritten wird, dann wird der Wert  $32767 / 7FFF_h$  bzw.  $-32768 / 8000_h$  als ungültiger Mittelwert gespeichert und bei der nächsten Auslösung des Analogwerttelegramms übertragen. In jeder neuen Mittelungsperiode wird identisch verfahren, d. h. fehlerhafte Werte werden wiederum für die Dauer der Fehlerunterdrückungszeit unterdrückt.

Mit der Dauer der Fehlerunterdrückungszeit lässt sich indirekt der prozentuale Anteil an ungültigen Werten pro Mittelungsperiode bestimmen. Wenn der Mittelwert z. B. im Zyklus von 15 Minuten gebildet wird und die Fehlerunterdrückungszeit ist auf 5 Minuten eingestellt, dann bedeutet dies, dass der Mittelwert nur dann als ungültig übertragen wird, wenn in der laufenden Mittelungsperiode mehr als 33 % der erfassten Analogwerte im Überlauf- oder Unterlaufbereich liegen.

#### 8.1.4.5 IPv6-Protokoll

Das Internet Protocol Version 6, nachfolgend kurz IPv6 genannt, erweitert das derzeit überwiegend verwendete Internet Protocol Version 4 (IPv4).

##### Adressformat IPv6: Notation

IPv6-Adressen bestehen aus 8 Blöcken mit jeweils vierstelligen Hexadezimalziffern (128 Bit insgesamt). Die Blöcke sind durch einen Doppelpunkt getrennt.

Beispiel:

fd00:ffff:ffff:ffff:ffff:ffff:2f33:8f21

Regeln / Vereinfachungen:

### 8.1 Geräte und Netze konfigurieren

- Führende Nullen innerhalb eines Blocks dürfen weggelassen werden.  
Beispiel: fd01:0:ffff::2d12:7d23
- Wenn ein oder mehr Blöcke den Wert 0 haben, ist eine verkürzte Schreibweise möglich.  
Die Adresse fd00:0:0:0:0:0:0:8f21 kann verkürzt auch wie folgt notiert werden:  
fd00::8f21  
Damit Eindeutigkeit gewahrt bleibt, darf diese Verkürzung nur einmal innerhalb der gesamten Adresse angewendet werden.
- Dezimalnotation mit Punkt-Schreibweise  
Für die letzten 2 Blöcke bzw. 4 Byte kann die herkömmliche Dezimalnotation in Punkt-Schreibweise verwendet werden.  
Beispiel: Die IPv6-Adresse fd00::ffff.125.1.0.1 ist äquivalent zu fd00::ffff:7d01:1

### Eingabe und Darstellung

Die Eingabe von IPv6-Adressen ist in den zuvor beschriebenen Notationen möglich. IPv6-Adressen werden immer in der selben Notation angezeigt, in der die Eingabe erfolgt ist.

#### 8.1.4.6 SCALANCE X, W und M

### Rechtlicher Hinweis

#### Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden

## Eigenschaften und Parameter bearbeiten

### Möglichkeiten zum Bearbeiten

Zum Bearbeiten von Eigenschaften und Parameter gibt es folgende Möglichkeiten:

- **Hardware- und Netzwerkeeditor**  
Nachdem Sie die Netzwerkkomponente eingefügt haben, können Sie die Eigenschaften und Parameter bearbeiten, z. B. den Gerätenamen. Weitere Informationen dazu finden Sie unter "Hardware- und Netzwerkeeditor".
- **Web Based Management (WBM)**  
Die Parameter und Eigenschaften sind über mitgelieferten HTML-Seiten (WBM-Seiten) erreichbar. Jede WBM-Seite hat eine eigene Hilfeseite, die die Eigenschaften und Parameter beschreiben. Beachten Sie für weitere Informationen das Projektierungshandbuch "SCALANCE X500 Web Based Management". Das Projektierungshandbuch finden Sie Siemens Industry Automation and Drives Service & Support im Internet unter der Beitrags-ID 67428305 (<http://support.automation.siemens.com/WW/view/de/67428305>).
- **Command Line Interface**  
Mit dem CLI können alle Konfigurationseinstellungen für das Gerät festgelegt werden. Das CLI bietet die gleichen Möglichkeiten wie das Web Based Management (WBM). Beachten Sie für weitere Informationen das Projektierungshandbuch "SCALANCE X500 Command Line Interface". Das Projektierungshandbuch finden Sie Siemens Industry Automation and Drives Service & Support im Internet unter der Beitrags-ID 67430663 (<http://support.automation.siemens.com/WW/view/de/67430663>).

### Voraussetzung für WBM und CLI

- Das Gerät verfügt über eine IP-Adresse
- Zwischen dem Gerät und dem Client-PC besteht eine Verbindung. Mit dem ping-Befehl können Sie nachprüfen, ob eine Verbindung besteht.
- Der integrierte Webserver ist aktiviert.
- Empfohlene Webbrowser:
  - Microsoft Internet Explorer Version 9.0
  - Mozilla Firefox Version ESR17
- Im Webbrowser ist JavaScript aktiviert.
- Der Webbrowser darf nicht so eingestellt sein, dass er bei jedem Zugriff auf die Seite diese neu vom Server laden soll. Die Aktualität der dynamischen Seiteninhalte wird über andere Mechanismen sichergestellt.  
Beim Internet Explorer finden Sie die Einstellung im Menü "Extras > Internetoptionen > Allgemein" im Abschnitt "Browserverlauf" über die Schaltfläche "Einstellungen". Aktivieren Sie bei "Neuere Versionen der gespeicherten Seite suchen" "Automatisch".
- Wenn eine Firewall eingesetzt wird, müssen die entsprechenden Ports freigeschaltet sein.
  - Für den Zugriff über HTTP: Port 80
  - Für den Zugriff über HTTPS: Port 443

**Darstellung des WBM auf mobilen Geräten:**

Für mobile Geräte gelten folgende minimale Voraussetzungen:

| Auflösung          | Betriebssystem                                      | Webbrowser  |
|--------------------|---|---|
| 960 x 640<br>Pixel | Android ab Version<br>4.2.1<br>iOS ab Version 6.0.2 | Chrome ab Version 18 auf Android<br>Safari ab Version 6 auf iOS |

Getestet mit folgenden Webbrowser für mobile Geräte

- Safari ab V6 auf iOS 6 (iPad Mini, iPod Touch 4. Generation)
- Chrome 25 auf Android (Galaxy Nexus 4, Galaxy Nexus 7)

---

**Hinweis**

**Seitendarstellung und Bedienung des WBM auf mobilen Geräten**

Die Darstellung und Bedienung der WBM-Seiten auf mobilen Geräten kann von der Darstellung und Bedienung derselben Seiten auf Desktop-Geräten abweichen. Einige Seiten liegen auch in einer für mobile Geräte optimierten Darstellung vor.

---

**Verfügbarkeit**

Die Verfügbarkeit der Eigenschaften und Parameter ist abhängig vom Porttyp. Folgende Typen sind möglich:

- Switch-Port
- Router-Port

**IP-Adressvergabe**

**Konfigurationsmöglichkeiten**

Im Auslieferungszustand und nach dem Zurücksetzen auf Werkseinstellungen besitzt das Gerät keine IP-Adresse.

Es gibt folgende Möglichkeiten dem Gerät eine IP-Adresse zuzuweisen:

- DHCP (Standard-Einstellung)
- Primary Setup Tool  
Beachten Sie für weitere Informationen das Projektierungshandbuch "Primary Setup Tool". Das Projektierungshandbuch finden Sie Siemens Industry Automation and Drives Service & Support im Internet unter der Beitrags-ID 19440762 (<http://support.automation.siemens.com/WW/view/de/19440762>).



- STEP 7
- CLI über die serielle Schnittstelle  
Beachten Sie für weitere Informationen das Projektierungshandbuch "SCALANCE X500 - Command Line Interface". Das Projektierungshandbuch finden Sie Siemens Industry Automation and Drives Service & Support im Internet unter der Beitrags-ID 67430663 (<http://support.automation.siemens.com/WW/view/de/67430663>).

## Voraussetzung

- In den Eigenschaften des Geräts ist "IP-Adresse auf anderem Weg beziehen" aktiviert. Weitere Informationen finden Sie unter "Adressierung von PROFINET-Geräten".

## System

## Konfiguration

Auf dieser Seite legen Sie fest, über welche Dienste auf das Gerät zu gegriffen wird. Zu einigen Diensten gibt es weitere Konfigurationsseiten, auf denen detailliertere Einstellungen möglich sind.

## Einstellungen

- **Telnet-Server**  
Aktivieren oder deaktivieren Sie den Dienst "Telnet-Server" für den unverschlüsselten Zugriff auf das CLI.
- **SSH-Server**  
Aktivieren oder deaktivieren Sie den Dienst "SSH-Server" für den verschlüsselten Zugriff auf das CLI.
- **Nur HTTPS-Server**  
Aktivieren oder deaktivieren Sie den Zugriff über HTTP.
- **SMTP-Client**  
Aktivieren oder deaktivieren Sie den SMTP-Client. Weitere Einstellungen konfigurieren Sie unter "System > SMTP-Client".
- **Syslog-Client**  
Aktivieren oder deaktivieren Sie den Systemereignis-Client. Weitere Einstellungen konfigurieren Sie unter "System > Syslog-Client".
- **DCP-Server**  
Legen Sie fest, ob auf das Gerät mit DCP (Discovery and Configuration Protocol) zugegriffen werden kann:
  - "-" (Deaktiviert)  
DCP ist deaktiviert. Geräteparameter können weder gelesen noch geändert werden.
  - Lesen/Schreiben  
Mit DCP können Geräteparameter sowohl gelesen als auch verändert werden.
  - Schreibgeschützt  
Mit DCP können Geräteparameter zwar gelesen aber nicht verändert werden.

- **Zeiteinstellung**

Wählen Sie die gewünschte Einstellung aus. Folgende Einstellungen sind möglich:

- Manuell  
Die Systemzeit wird manuell eingestellt.
- SNTP-Client  
Die Systemzeit wird über einen SNTP Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SNTP-Client".
- NTP-Client  
Die Systemzeit wird über einen NTP Server eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > NTP-Client".
- SIMATIC Time  
Die Systemzeit wird über einen SIMATIC Zeitgeber eingestellt. Weitere Einstellungen konfigurieren Sie unter "System > Systemzeit > SIMATIC Time Client".

- **SNMP**

Wählen Sie das gewünschte Protokoll. Folgende Einstellungen sind möglich:

- "-" (SNMP deaktiviert)  
Ein Zugriff auf die Geräteparameter ist über SNMP nicht möglich.
- SNMPv1/v2c/v3  
Ein Zugriff auf die Geräteparameter ist mit den SNMP Versionen 1, 2c oder 3 möglich. Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Allgemein".
- SNMPv3  
Ein Zugriff auf die Geräteparameter ist mit der SNMP Version 3 möglich. Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Allgemein".

- **SNMPv1/v2 schreibgeschützt**

Aktivieren oder deaktivieren Sie den schreibenden Zugriff auf SNMP-Variablen bei SNMPv1/v2c.

- **SNMPv1 Traps**

Aktivieren oder deaktivieren Sie das Versenden von SNMP Traps (Alarmtelegramme). Weitere Einstellungen konfigurieren Sie unter "System > SNMP > Traps".

- **SINEMA-Konfigurationsschnittstelle**  
Aktivieren oder deaktivieren Sie die SINEMA-Konfigurationsschnittstelle.
- **Konfigurationsmodus**  
Wählen Sie die gewünschte Betriebsart. Folgende Betriebsarten sind möglich:
  - **Automatisches Speichern**  
Automatischer Sicherheitsbetrieb. Ca. 1 Minute nach der letzten Parameteränderung oder beim Neustart des Geräts wird die Konfiguration automatisch abgespeichert.
  - **Test**  
Im Testmodus werden Änderungen zwar übernommen aber nicht in der Konfigurationsdatei (Startup Configuration) gespeichert.  
Um Änderungen in der Konfigurationsdatei abzuspeichern, verwenden Sie die Schaltfläche "Schreiben der Start-up-Konfiguration". Die Schaltfläche wird eingeblendet, wenn Sie den Testmodus einstellen. Zusätzlich wird im Anzeigebereich die Meldung "Testmodus aktiv - Klicken Sie zum Speichern Ihrer Einstellungen auf die Schaltfläche "Schreiben der Start-up-Konfiguration" angezeigt, sobald es ungespeicherte Änderungen gibt. Diese Meldung ist auf jeder WBM-Seite sichtbar, bis die vorgenommenen Änderungen entweder gespeichert werden oder das Gerät neu gestartet wird.

## Allgemein

### Gerät

Diese Seite enthält die allgemeinen Geräteinformationen.

### Einstellungen

- **Systemname**  
Sie können den Namen des Gerätes eintragen. Der eingetragene Name wird im Auswahlbereich angezeigt. Es sind maximal 255 Zeichen möglich. Der Systemname wird auch in der CLI-Eingabeaufforderung (Prompt) angezeigt. In der CLI-Eingabeaufforderung ist die Anzahl der Zeichen begrenzt. Der Systemname wird nach 16 Zeichen abgeschnitten.
- **Kontaktperson**  
Sie können den Namen einer Kontaktperson eintragen, die für die Verwaltung des Gerätes zuständig ist.
- **Gerätestandort**  
Sie können den Montageort des Gerätes eintragen. Der eingetragene Montageort wird im Auswahlbereich angezeigt.

---

#### Hinweis

In den Eingabefeldern wird der ASCII-Code 0x20 bis 0x7e verwendet.

Zu Beginn und am Ende der Eingabefelder "Systemname", "Kontaktperson" und "Gerätestandort" sind die Zeichen "<", ">" und "Leer" nicht erlaubt.

---

## Koordinaten

Auf dieser Seite konfigurieren Sie die geografischen Koordinaten (Breitengrad, Längengrad und die Höhe über dem Ellipsoid gemäß WGS84). Diese Felder sind reine Informationsfelder mit einer maximalen Länge von 32 Zeichen.

### Ermittlung der Koordinaten

Nutzen Sie zur Ermittlung der geografischen Koordinaten des Gerätes entsprechendes Kartenmaterial.

Die geografischen Koordinaten können auch durch einen GPS-Empfänger ermittelt werden. Meist werden die geografischen Koordinaten von diesen Geräten direkt angezeigt und müssen nur noch in die Eingabefelder dieser Seite übertragen werden.

## Einstellungen

- **Geographische Breite**  
Geben Sie die nördliche oder die südliche Breite für den Standort des Gerätes ein.  
Z. B. +49° 1' 31.67" bedeutet, dass sich das Gerät auf 49 Grad, 1 Bogenminute und 31.67 Bogensekunden nördliche Breite befindet.  
Die südliche Breite wird mit einem führenden Minuszeichen dargestellt.  
Sie können auch die Buchstaben N (nördliche Breite) oder S (südliche Breite) an die Zahlenangabe anhängen (49° 1' 31.67" N).
- **Geographische Länge**  
Geben Sie die östliche oder die westliche Länge für den Standort des Gerätes ein.  
Z. B. +8° 20' 58.73" bedeutet, dass sich das Gerät auf 8 Grad, 20 Bogenminuten und 58.73 Bogensekunden östliche Länge befindet.  
Die westliche Länge wird mit einem führenden Minuszeichen dargestellt.  
Sie können auch die Buchstaben O bzw. E (östliche Länge) oder W (westliche Länge) an die Zahlenangabe anhängen (8° 20' 58.73" E).
- **Geographische Höhe**  
Geben Sie die geografische Höhe über oder unter normal Null (Meereshöhe) in Metern ein.  
Z.B. 158 m bedeutet, dass sich das Gerät in einer Höhe von 158 m über normal Null befindet.  
Höhenangaben unterhalb von normal Null (z. B. am toten Meer) werden mit einem führenden Minuszeichen dargestellt.

## Laden & Speichern

### Hochladen und Speichern über TFTP

### Laden und speichern von Daten über einen TFTP-Server

Auf der Seite können Sie den TFTP-Server und die Dateinamen konfigurieren. Weiter bietet das WBM die Möglichkeit, Gerätedaten in einer externen Datei auf Ihrem Client-PC zu

speichern bzw. solche Daten aus einer externen Datei vom PC in die Geräte zu laden. So können Sie z. B. eine neue Firmware aus einer Datei von Ihrem Client-PC laden.

---

#### **Hinweis**

##### **Inkompatibilität zu Vorgängerversionen**

Bei der Installation einer Vorgängerversion kann es zu Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen.

##### **Inkompatibilität zu Vorgängerversionen mit gestecktem PLUG**

Bei der Installation einer Vorgängerversion kann es zu Verlust der Konfigurationsdaten kommen. In diesem Fall startet das Gerät nach der Installation der Firmware mit den Werkseinstellungen. Wenn in diesem Fall ein PLUG im Gerät gesteckt ist, hat dieser nach dem Neustart den Status "Not Accepted", da die sich auf dem PLUG weiterhin die Konfigurationsdaten der vorherigen, aktuelleren Firmware befinden. Somit kann ohne Konfigurationsdatenverlust zur vorherigen, aktuelleren Firmware zurückgekehrt werden. Falls die ursprüngliche Konfiguration auf dem PLUG nicht mehr benötigt wird, kann der PLUG manuell auf der WBM-Seite "System > PLUG" gelöscht oder neu beschrieben werden.

---

#### **Hinweis**

##### **Konfigurationsdateien und Testmodus/Automatisches Speichern**

Beim Konfigurationsmodus "Automatisches Speichern" wird eine automatische Sicherung durchgeführt, bevor die Konfigurationsdateien (ConfigPack und Config) übertragen werden. Im Testmodus werden Änderungen zwar übernommen aber nicht in den Konfigurationsdateien (ConfigPack und Config) gespeichert. Verwenden Sie die Schaltfläche "Schreiben der Start-up-Konfiguration" auf der WBM-Seite "System > Systemkonfiguration", um Änderungen in den Konfigurationsdateien abzuspeichern.

---

## **Einstellungen**

- **IP-Adresse des TFTP-Servers**  
Tragen Sie hier die IP-Adresse des TFTP-Servers ein, mit dem Sie Daten austauschen.
- **Port des TFTP-Servers**  
Tragen Sie hier den Port des TFTP-Servers ein, über den der Datenaustausch abgewickelt werden soll. Gegebenenfalls können Sie den Standardwert 69 entsprechend Ihren spezifischen Anforderungen ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Dateityp**  
Zeigt den Dateityp an.
- **Beschreibung**  
Zeigt die Kurzbeschreibung des Dateityps an.
- **Dateiname**  
Tragen Sie einen Dateinamen ein.

## Ereignisse

## Konfiguration

Auf dieser Seite legen Sie fest, wie das Gerät auf Ereignisse reagiert.

## Einstellungen

- **Meldekontakt-Verhalten**

Wählen Sie das Verhalten des Meldekontakts. Folgende Verhalten sind möglich:

- Standard  
Standardeinstellung für den Meldekontakt. Ein auftretender Fehler wird durch die Fehler-LED angezeigt und der Meldekontakt wird geöffnet. Wenn der Fehlerzustand nicht mehr besteht, erlischt die Fehler-LED und der Meldekontakt wird geschlossen.
- angepasst  
Die Funktion des Meldekontakts ist unabhängig vom auftretenden Fehler. Der Meldekontakt kann durch Benutzeraktionen beliebig geöffnet oder geschlossen werden.

- **Meldekontakt-Status**

Wählen Sie den Zustand des Meldekontakts. Folgende Zustände sind möglich:

- Geschlossen  
Meldekontakt ist geschlossen.
- Geöffnet  
Meldekontakt ist geöffnet.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **E-Mail/Trap/Protokolltabelle/Syslog/Fehler**  
Wählen Sie die gewünschte Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Ereignis**  
Die Spalte "Ereignis" enthält folgende Werte:
  - Kalt-/Warmstart  
Das Gerät wurde eingeschaltet oder vom Anwender neu gestartet.
  - Verbindungswechsel  
Dieses Ereignis tritt nur auf, wenn der Port-Status überwacht wird und sich entsprechend geändert hat, siehe "System > Fehlerüberwachung > Link Change".
  - Authentifizierungsfehler  
Dieses Ereignis tritt beim Versuch eines Zugriffs mit fehlerhaftem Kennwort auf.
  - Umschaltung Spannungsversorgung  
Dieses Ereignis tritt nur auf, wenn die Spannungsversorgungsleitungen 1 und 2 überwacht werden. Es zeigt an, dass ein Wechsel auf Leitung 1 bzw. auf Leitung 2 stattgefunden hat, siehe "System > Fehlerüberwachung > Spannungsversorgung".
  - Topologieänderung Spanning Tree  
Die STP- bzw. RSTP- oder MSTP-Topologie hat sich geändert.
  - Änderung des Fehlerstatus  
Der Fehlerstatus hat sich geändert. Der Fehlerstatus kann sich auf die aktivierte Portüberwachung, auf das Ansprechen der Meldekontakte oder die Spannungsüberwachung beziehen.
  - RMON-Alarm  
Ein Alarm oder ein Ereignis ist im Zusammenhang mit der Fernüberwachung des Systems aufgetreten.
  - Statusänderung VRRP (Nur bei Routing über VRRP)  
Der Zustand des virtuellen Routers hat sich geändert
  - Loop Detection  
Es wurde eine Schleife im Netzsegment erkannt.
  - OSPF Statusänderung  
Der Status von OSPF hat sich geändert
- **E-Mail**  
Das Gerät sendet eine E-Mail. Voraussetzung ist, dass der SMTP-Server eingerichtet und die Funktion "SMTP-Client" aktiviert ist.
- **Trap**  
Das Gerät löst einen SNMP-Trap aus. Voraussetzung ist, dass unter "System > Konfiguration" "SNMPv1 Traps" aktiviert ist.
- **Protokolltabelle**  
Das Gerät schreibt einen Eintrag in die Protokolltabelle.
- **Syslog**  
Das Gerät schreibt einen Eintrag auf den Systemprotokoll-Server. Voraussetzung ist, dass der Systemprotokoll-Server eingerichtet und die Funktion "Syslog-Client" aktiviert ist.
- **Fehler**  
Das Gerät löst einen Fehler aus. Die Fehler-LED leuchtet auf

## Severity-Filter

Stellen Sie auf dieser Seite die Schwellwertstufen für das Versenden von Systemereignisbenachrichtigungen ein.

## Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Client-Typ**  
Wählen Sie den Client-Typ, für den Sie die Einstellungen vornehmen:
  - **E-Mail**  
Versand von Systemereignismeldungen per E-Mail
  - **Protokolltabelle**  
Eintragen von Systemereignissen in die Protokolltabelle.
  - **Syslog**  
Eintragen von Systemereignissen in die Syslog-Datei
- **Severity**  
Wählen Sie die gewünschte Stufe aus. Folgende Einstellungen sind möglich:
  - **Information**  
Systemereignisse werden ab dem Severity-Level "Information" bearbeitet.
  - **Achtung**  
Systemereignisse werden ab dem Severity-Level "Achtung" bearbeitet.
  - **Kritisch**  
Systemereignisse werden ab dem Severity-Level "Kritisch" bearbeitet.

## SMTP-Client

Das Gerät bietet die Möglichkeit, beim Auftreten eines Alarmereignisses automatisch eine E-Mail (z.B. an den Netzwerkadministrator) zu senden. Die E-Mail enthält die Identifikation des absendenden Geräts, eine Beschreibung der Alarmursache in Klartext sowie einen Zeitstempel. Damit kann für Netze mit wenigen Teilnehmern eine einfache zentrale Netzüberwachung auf Basis eines E-Mail-Systems aufgebaut werden. Bei eintreffenden E-Mail-Störmeldungen kann über die Identifikation des Absenders per Browser das WBM gestartet werden, um weitere Diagnoseinformationen auszulesen.

Auf dieser Seite können Sie bis zu drei SMTP-Server und die dazugehörigen E-Mail-Adressen konfigurieren.

---

### Hinweis

Je nach Eigenschaft und Konfiguration des SMTP-Servers kann es notwendig sein, das Eingabefeld "Absender" für die E-Mails anzupassen. Informieren Sie sich beim Administrator des SMTP-Servers.

---



## Einstellungen

- **SMTP-Client**  
Aktivieren oder deaktivieren Sie den SMTP-Client.
- **Absender**
- Geben Sie den Absendernamen ein, der in der E-Mail angegeben wird, z. B. den Gerätenamen.
- **SMTP-Port**  
Ändern Sie den Port ab, wenn der SMTP-Server nicht über den Port 25 erreichbar ist.
- **IP-Adresse des SMTP-Servers**  
Geben Sie die IP-Adresse des SMTP-Servers ein.

Die Tabelle enthält folgende Spalten:

- **IP-Adresse des SMTP-Servers**  
Zeigt die IP-Adresse des SMTP-Servers.
- **Email-Adresse des Empfängers**  
Geben Sie die E-Mail-Adresse ein, an die das Gerät im Fehlerfall eine E-Mail sendet. Die E-Mail-Adresse kann die einer einzelnen Person oder einer Verteilerliste sein.

## DHCP-Client

Wenn die DHCP-Betriebsart aktiviert ist, startet der DHCP-Client bei einem konfigurierten DHCP-Server eine DHCP-Anfrage und erhält als Antwort eine IP-Adresse zugewiesen. Der Server verwaltet einen Adressbereich, aus welchem er IP-Adressen vergibt. Es ist auch möglich den Server so zu konfigurieren, dass der Client auf seine Anfrage immer dieselbe IP-Adresse zugewiesen bekommt.

## Einstellungen

- **DHCP-Client**  
Aktivieren oder deaktivieren Sie den DHCP-Client.
- **DHCP-Client Konfigurationsdatei-Anfrage (Opt. 66, 67)**  
Aktivieren Sie diese Option, wenn der DHCP-Client die Optionen 66, und 67 dazu verwenden soll, eine Konfigurationsdatei herunterzuladen und diese dann zu aktivieren.
- **DHCP-Betriebsart**  
Wählen Sie die DHCP-Betriebsart. Folgende Betriebsarten sind möglich:
  - über MAC-Adresse  
Die Identifikation läuft über die MAC-Adresse ab.
  - über ID des DHCP-Clients  
Die Identifikation läuft über eine frei definierte DHCP-Client-ID ab.
  - über Gerätename  
Die Identifikation läuft über den Gerätenamen ab. Ist der Gerätename 255 Zeichen lang, dann wird das letzte Zeichen nicht zur Identifikation benutzt.

## SNMP

### Allgemein

Auf dieser Seite treffen Sie grundlegende Einstellungen für SNMP. Aktivieren Sie die Funktionen, die Sie nutzen wollen.

### Beschreibung

Folgende Einstellungen sind möglich:

- **SNMP**  
Wählen Sie das SNMP-Protokoll. Folgende Einstellungen sind möglich:
  - "-" (Deaktiviert)  
SNMP deaktiviert.
  - SNMPv1/v2c/v3  
SNMPv1/v2c/v3 wird unterstützt.
  - SNMPv3  
Nur SNMPv3 wird unterstützt.
- **SNMPv1/v2 schreibgeschützt**  
Wenn Sie diese Option aktivieren, kann SNMPv1/v2c nur lesend auf die SNMP-Variablen zugreifen.

---

#### Hinweis

##### Community String

Verwenden Sie aus Sicherheitsgründen nicht die Standardwerte "public" oder "private". Ändern Sie die Community Strings nach der Erst-Installation.

---

- **Community String für SNMPv1/v2c - Lesen**  
Tragen Sie den Community String für den Zugriff des SNMP-Protokolls ein.
- **Community String für SNMPv1/v2c - Lesen/Schreiben**  
Tragen Sie den Community String für den lesenden und schreibenden Zugriff des SNMP-Protokolls ein.
- **SNMPv1-Traps**  
Aktivieren oder deaktivieren Sie das Senden von SNMP-Traps (Alarmtelegramme). Auf dem Register "Trap" legen Sie die IP-Adressen der Geräte fest, an die SNMP Traps gesendet werden.
- **Community String für SNMPv1/v2c - Trap**  
Tragen Sie den Community String für das Senden von SNMPv1/v2-Meldungen ein.

## Traps

Beim Eintreten eines Alarmereignisses kann ein Gerät SNMP-Traps (Alarmtelegramme) an bis zu zehn verschiedene Management-Stationen gleichzeitig senden. Es werden nur bei solchen Ereignissen Traps gesendet, die bei "Ereignisse > Konfiguration" festgelegt sind.

---

### Hinweis

SNMP-Traps werden nur dann versendet, wenn Sie bei "SNMP >Allgemein" die Einstellung "SNMPv1 Traps" aktiviert haben.

---

## Einleitung

Folgende Einstellung ist möglich:

- **IP-Adresse**  
Geben Sie die IP-Adresse der Stationen ein, an die das Gerät SNMP-Traps sendet

Die Tabelle gliedert sich in folgende Spalten:

- **IP-Adresse**  
Zeigt die IP-Adressen der Stationen an, an die das Gerät SNMP-Traps sendet.
- **Trap**  
Aktivieren oder deaktivieren Sie das Senden von SNMP-Traps. Stationen, die eingetragen, aber nicht selektiert sind, erhalten keine SNMP-Traps.

## v3-Gruppen

### Sicherheitseinstellungen und Rechtevergabe

SNMP Version 3 bietet eine Rechtevergabe, Authentifizierung und Verschlüsselung auf Protokollebene. Die Sicherheitsstufen und die Lese-/Schreibrechte werden gruppenspezifisch definiert. Für jedes Mitglied einer Gruppe gelten automatisch die entsprechenden Einstellungen.

## Beschreibung

Folgende Einstellungen sind möglich:

- **Gruppenname**  
Tragen Sie den Namen der Gruppe ein. Dieser Name muss mindestens zwei Zeichen lang sein, die maximale Länge beträgt 32 Zeichen.
- **Security-Stufe**  
Wählen Sie die Sicherheitsstufe (Authentifizierung, Verschlüsselung) aus, die für die gewählte Gruppe gültig ist. Bei den Sicherheitsstufen die folgenden Möglichkeiten:
  - Keine Auth./keine Priv.  
Keine Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./keine Priv.  
Authentifizierung aktiviert / keine Verschlüsselung aktiviert.
  - Auth./Priv.  
Authentifizierung aktiviert / Verschlüsselung aktiviert.

Die Tabelle gliedert sich in folgende Spalten:

- **Gruppenname**  
Zeigt die definierten Gruppennamen an.

---

### Hinweis

Der einmal vergebene Gruppenname und die Sicherheitsstufe können nach dem Anlegen nicht mehr geändert werden. Wenn Sie den Gruppennamen oder die Sicherheitsstufe ändern wollen, müssen Sie die Gruppe löschen und mit dem neuen Namen neu anlegen und neu konfigurieren.

---

- **Security-Stufe**  
Zeigt die konfigurierte Sicherheitsstufe an.
- **Lesen**  
Aktivieren oder deaktivieren Sie den Lesezugriff.
- **Schreiben**  
Aktivieren oder deaktivieren Sie den Schreibzugriff.

## v3-Benutzer

### Benutzerspezifische Sicherheitseinstellungen

Auf der WBM-Seite können Sie SNMPv3-Benutzer neu anlegen, ändern oder löschen. Das benutzerbasierte Sicherheitsmodell arbeitet mit dem Konzept des Benutzernamens, d. h. jedes Telegramm wird mit einer Benutzerkennung versehen. Diesen Benutzernamen und die betreffenden Sicherheitseinstellungen überprüfen sowohl der Absender wie auch der Empfänger.

## Beschreibung

Folgende Einstellungen sind möglich:

- **Benutzername**  
Geben Sie einen frei wählbaren Benutzernamen ein. Nach der Datenübernahme können Sie den Namen nicht mehr ändern.

Die Tabelle gliedert sich in folgende Spalten:

- **Benutzername**  
Zeigt den angelegten Benutzer an.
- **Gruppenname**  
Wählen Sie die Gruppe, der der Benutzer zugeordnet werden soll.
- **Authentifizierungsprotokoll**  
Wählen Sie das Authentifizierungsprotokoll fest. Nur aktivierbar, wenn die Gruppe die Funktion unterstützt.  
Folgende Einstellungen gibt es:
  - Keine
  - MD5
  - SHA
- **Privatprotokoll**  
Legen Sie fest, ob der Benutzer den DES-Algorithmus verwendet. Nur aktivierbar, wenn die Gruppe diese Funktion unterstützt.
- **Authentifizierungskennwort**  
Geben Sie in das erste Eingabefeld das Authentifizierungspasswort ein.
- **Authentifizierungskennwort-Best.**  
Bestätigen Sie das Passwort durch die Wiederholung der Eingabe.
- **Privatbereichskennwort**  
Geben Sie Ihr Verschlüsselungspasswort ein.
- **Privatbereichskennwort-Best.**  
Bestätigen Sie das Verschlüsselungspasswort durch die Wiederholung der Eingabe.

## Systemzeit

### SNTP-Client

Das SNTP (**Simple Network Time Protocol**) dient zur Zeitsynchronisation im Netzwerk. Die entsprechenden Telegramme werden von einem SNTP-Server im Netz versendet.

## Einstellungen

- **SNTP-Client**  
Aktivieren oder deaktivieren Sie die automatische Zeitsynchronisation über SNTP.
- **Zeitzone**  
Tragen Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit. Einstellungen zu Sommer bzw. Winterzeit berücksichtigen Sie bei der Angabe des Zeit-Offset in diesem Feld.
- **SNTP-Synchronisationsart**  
Wählen Sie die Synchronisationsart aus. Folgende Synchronisierungsarten sind möglich:
  - Poll  
Wenn Sie diese Protokollart wählen, werden die Eingabefelder "IP-Adresse des SNTP-Servers", "Port des SNTP-Servers" und "Poll Interval" zur weiteren Konfiguration eingeblendet. Bei dieser Synchronisationsart ist das Gerät aktiv und sendet eine Zeitabfrage an den SNTP-Server.
  - Listen  
Bei dieser Synchronisationsart ist das Gerät passiv und "hört" auf SNTP-Telegramme, die die Uhrzeit liefern.
- **IP-Adresse des SNTP-Servers**  
Tragen Sie die IP-Adresse des SNTP-Servers ein.
- **Port des SNTP-Servers**  
Tragen Sie den Port des SNTP-Servers ein.
- **Poll Interval [s]**  
Tragen Sie den Zeitabstand in Sekunden zwischen zwei Zeitanfragen ein.

## NTP-Client

Wenn die Uhrzeitsynchronisation über NTP erfolgen soll, können Sie hier die entsprechenden Einstellungen vornehmen.

## Einstellungen

- **NTP-Client**  
Aktivieren oder deaktivieren Sie die automatische Zeitsynchronisation über NTP.
- **Zeitzone**  
Tragen Sie Ihre verwendete Zeitzone im Format "+/- HH:MM" ein. Die Zeitzone bezieht sich auf UTC Standard-Weltzeit. Einstellungen zu Sommer bzw. Winterzeit berücksichtigen Sie bei der Angabe des Zeit-Offset in diesem Feld.
- **IP-Adresse des NTP-Servers**  
Tragen Sie die IP-Adresse des NTP-Servers ein.
- **Port des NTP-Servers**  
Tragen Sie den Port des NTP-Servers ein.
- **Poll Interval [s]**  
Tragen Sie den Zeitabstand in Sekunden zwischen zwei Zeitanfragen ein.

## **SIMATIC-Time-Client**

Auf dieser Seite konfigurieren Sie die Zeitsynchronisation über den SIMATIC Time Client.

### **Einstellung**

- **SIMATIC Time Client**  
Aktivieren oder deaktivieren Sie den SIMATIC Time Client.

## **PTP-Client**

Auf dieser Seite konfigurieren Sie die Zeitsynchronisation über den PTP (Precision Time Protocol).

### **Einstellung**

- **PTP-Client**  
Aktivieren oder deaktivieren Sie die Zeitsynchronisation über den PTP. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > PTP".

## **Auto-Logout**

Stellen Sie in dieser Seite die Zeiten ein, nach denen bei Inaktivität des Benutzers automatisch eine Abmeldung vom WBM oder dem CLI erfolgt.

---

### **Hinweis**

#### **Keine automatische Abmeldung vom CLI**

Wenn die Verbindung nach der eingestellten Zeit nicht beendet wird, prüfen Sie am Telnet Client die Einstellung des "Keep alive"- Mechanismus.

Ist die Intervallzeit kleiner als die projektierte Zeit, wird die Verbindung aufrecht erhalten, obwohl keine Nutzdaten übertragen werden. Z. B. Sie haben bei der automatischen Abmeldung 300 Sekunden eingestellt und bei der "Keep alive"- Funktion steht 120 Sekunden. In diesem Fall wird alle 120 Sekunden ein Paket gesendet, dass die Verbindung aufrecht erhält.

- Schalten Sie den Kepp-Alive-Mechanismus aus (Intervallzeit=0)  
oder
  - Stellen Sie die Intervallzeit so hoch ein, dass die unterlagerte Verbindung bei Inaktivität beendet wird.
-

## Einstellungen

- **Web Based Management [s]**  
Tragen Sie die Zeit in Sekunden für die automatische Abmeldung vom WBM ein. Wenn Sie den Wert 0 eingeben, ist die automatische Abmeldung deaktiviert.
- **CLI (TELNET, SSH, Serial) [s]**  
Tragen Sie die Zeit in Sekunden für die automatische Abmeldung vom CLI ein. Wenn Sie den Wert 0 eingeben, ist die automatische Abmeldung deaktiviert.

## Schaltfläche


Der Taster "Taster Select/Set" dient zum:

- Umschalten des Anzeigemodus,
- Redundanz-Manager aktivieren
- Zurücksetzen auf werkseitige Voreinstellungen,
- Definieren der Meldemaske und der LED-Anzeige,

Eine detaillierte Beschreibung der einzelnen Funktionen die am Taster bedient werden können, finden Sie in den gerätespezifischen Betriebsanleitungen.

## Einstellungen

- **Auf Werkseinstellungen zurücksetzen**  
Aktivieren oder deaktivieren Sie die Funktion "Zurücksetzen auf werkseitige Voreinstellungen" am Select/Set-Taster.

|  |
|--|
|  <b>VORSICHT</b>  |
| <b>Tasterfunktion "Auf Werkseinstellungen zurücksetzen" beim Hochlauf aktiv</b>  |
| Wenn Sie diese Funktion in ihrer Projektierung deaktiviert haben, ist die Deaktivierung nur im laufenden Betrieb gültig. Bei einem Hochlauf, z.B. nach "Stromaus", ist die Funktion bis zum Laden der Projektierung aktiv und das Gerät kann so auch unbeabsichtigt auf die Werkseinstellungen zurückgesetzt werden. Dies kann zu unerwünschten Störungen des Netzwerkbetriebs führen, da das Gerät nach diesem Vorfall erst neu projektiert werden muss. Ein gesteckter PLUG wird dabei ebenfalls gelöscht und in den Auslieferungszustand versetzt |

- **Redundanzmanager**  
Aktivieren oder deaktivieren Sie den Redundanzmanager am Select/Set-Taster.
- **Meldemaske aktivieren**  
Aktivieren oder deaktivieren Sie die Funktion "Definieren der Meldemaske über die LED-Anzeige" am Select/Set-Taster. Diese Funktion geht nur im Anzeigemodus D.

## Syslog-Client

Syslog nach RFC 3164 wird für die Übermittlung von kurzen, unverschlüsselten Textmeldungen per UDP im IP-Netz verwendet. Dazu wird ein Systemprotokoll-Server benötigt.



### Voraussetzungen für das Versenden der Protokolleinträge:

- Die Systemprotokoll-Funktion ist im Gerät aktiviert.
- Die Systemprotokoll-Funktion für das jeweilige Ereignis ist aktiviert.
- In Ihrem Netz befindet sich ein Systemprotokoll-Server, der die Log-Einträge entgegen nimmt. (Dabei es sich um eine UDP-Verbindung handelt, gibt es keine Rückmeldung an den Absender.)
- Die IP-Adresse des Systemprotokoll-Servers ist im Gerät eingetragen.

### Einstellungen

- **Syslog-Client**  
Aktivieren oder deaktivieren Sie die Systemprotokoll-Funktion.
- **IP-Adresse des Syslog-Servers**  
Tragen Sie die IP-Adresse des Systemprotokoll-Servers an.

Die Tabelle enthält folgende Spalten

- **IP-Adresse des Syslog-Servers**  
Zeigt die IP-Adresse des Systemprotokoll-Servers an.
- **Server-Port**  
Tragen Sie den verwendeten Port des Syslog-Servers ein.

### Ports

### Übersicht

Die Seite zeigt für alle Ports des Geräts die Konfiguration für den Datentransfer an.

### Einstellungen

- **Port**  
Zeigt die konfigurierbaren Ports an. Der Port setzt sich aus der Portnummer und der Slotnummer zusammen, z. B. Port 0.1 ist Slot 0, Port 1.
- **Port-Name**  
Zeigt den Namen des Ports.
- **Negotiation**  
Zeigt an, ob die automatische Konfiguration aktiviert oder deaktiviert ist.
- **Flow Ctrl. Type**  
Gibt an, ob für den Port die Flusskontrolle aktiviert oder deaktiviert ist.
- **Flow Ctrl.**  
Gibt an, ob bei diesem Port die Flusskontrolle arbeitet.
- **MTU**  
Zeigt die maximale Paketgröße an.

### 8.1 Geräte und Netze konfigurieren

- **Porttyp** (Nur bei Routing)  
Zeigt den Typ des Ports an. Folgende Typen sind möglich:
  - Switch-Port
  - Router-Port
- **Status**  
Zeigt an, ob der Port ein- oder ausgeschaltet ist. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.

## Konfiguration

Auf dieser Seite konfigurieren Sie die Ports des Geräts

## Einstellungen

- **Port**  
Wählen Sie die zu konfigurierenden Port aus. Der Port setzt sich aus der Portnummer und der Slotnummer zusammen, z. B. Port 0.1 ist Slot 0, Port 1.
- **Status**  
Legen Sie fest, ob der Port ein oder ausgeschaltet ist.
  - enabled  
Der Port ist eingeschaltet. Der Datenverkehr ist nur über einen eingeschalteten Port möglich.
  - disabled  
Der Port ist ausgeschaltet, aber die Verbindung besteht noch.
  - Link down  
Der Port ist ausgeschaltet und die Verbindung zum Partnergerät ist abgebaut.
- **Port-Name**  
Tragen Sie hier einen Namen für den Port ein.
- **Übertragungsmodus**  
Zeigt die Übertragungsgeschwindigkeit und das Übertragungsverfahren des Ports an. Die Einstellungen für "Autonegotiation" und "Übertragungsrate" legen Sie in den Port-Optionen fest.

---

### Hinweis

Damit der Port und der Partner-Port miteinander kommunizieren können, müssen die Einstellungen auf beiden Seiten übereinstimmen.

---

- **Negotiation**  
Zeigt an, ob die automatische Anschlusskonfiguration zum Partner-Port aktiviert oder deaktiviert ist.

- **Flow Ctrl. Type**  
Gibt an, ob bei diesem Port die Flusskontrolle arbeitet.

---

**Hinweis**

**Ein-/Ausschalten der Flusskontrolle bei Auto Negotiation**

Die Flusskontrolle kann nur bei ausgeschalteter Funktion "Auto Negotiation" aktiviert oder deaktiviert werden. Die Funktion kann danach wieder aktiviert werden.

---

- **Flow Ctrl.**  
Gibt an, ob bei diesem Port die Flusskontrolle arbeitet.
- **MTU**  
Tragen Sie die Paketgröße ein.
- **Porttyp** (Nur bei Routing)  
Wählen Sie die Art des Ports.
  - Switch-Port
  - Router-Port

## Veränderung der Port-Konfiguration

---

**Hinweis**

Optische Ports arbeiten immer mit dem Übertragungsverfahren Vollduplex und mit maximaler Übertragungsgeschwindigkeit. Deshalb können Sie bei optischen Ports folgende Einstellungen nicht vornehmen:

- Automatische Konfiguration
  - Übertragungsgeschwindigkeit
  - Übertragungsverfahren
- 

**Hinweis**

Das Gerät verhindert oder reduziert bei Überlastung eines Ports durch verschiedene Automatismen die Rückwirkung auf andere Ports und Prioritätsklassen (Class of Service). Dies kann auch bei aktivierter Flusskontrolle dazu führen, dass Telegramme verworfen werden.

Port-Überlastungen treten auf, wenn das Gerät mehr Telegramme empfängt, als es senden kann, z.B. infolge unterschiedlicher Übertragungsgeschwindigkeiten.

---

## Fehlerüberwachung

### Spannungsversorgung

Konfigurieren Sie, ob die Spannungsversorgung durch das Meldesystem überwacht werden soll. Je nach Hardware-Variante gibt es ein oder zwei Spannungsanschlüsse (Versorgung 1 / Versorgung 2). Bei redundanter Spannungsversorgung konfigurieren Sie die Überwachung für jede einzelne Zuleitung getrennt.

Wenn an einem überwachten Anschluss (Versorgung1 oder Versorgung 2) keine oder eine zu geringe Spannung anliegt, wird ein Fehler durch das Meldesystem signalisiert.

---

**Hinweis**

Die zulässigen Betriebsspannungsgrenzen entnehmen Sie der Kompaktbetriebsanleitung des Geräts.

---

Ein Fehler führt zum Aufleuchten der Fehler-LED am Gerät. Abhängig von der Konfiguration kann der Fehler einen Trap, eine E-Mail oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

## Einstellungen

- **Versorgung 1**  
Aktivieren oder deaktivieren Sie die Überwachung des Spannungsanschlusses 1.
- **Versorgung 2**  
Aktivieren oder deaktivieren Sie die Überwachung des Spannungsanschlusses 2.

## Link Change

Auf dieser Seite konfigurieren Sie, ob bei einer Zustandsänderung einer Netzwerkverbindung eine Fehlermeldung ausgelöst wird.

Bei aktivierter Verbindungsüberwachung wird ein Fehler signalisiert,

- wenn an einem Port ein Link vorhanden sein soll und dieser fehlt.
- oder wenn an dem Port kein Link vorhanden sein soll und ein Link erkannt wird.

Ein Fehler führt zum Auslösen des Meldekontakts und zum Aufleuchten der Fehler-LED am Gerät. Abhängig von der Konfiguration kann der Fehler einen Trap, eine E-Mail oder einen Eintrag in der Ereignisprotokoll-Tabelle auslösen.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Einstellung**  
Wählen Sie aus der Klappliste die Einstellung. Sie haben folgende Möglichkeiten:
  - "-" (Deaktiviert)
  - Up
  - Down
  - Keine Änderung:  
Einstellung in der Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten

- **Port**  
Zeigt die verfügbaren Ports an.
- **Einstellung**  
Wählen Sie die Einstellung aus. Sie haben folgende Möglichkeiten:
  - Up  
Die Fehlerbehandlung wird beim Übergang in den aktiven Zustand des Ports ausgelöst.  
(Von "Link down" nach "Link up")
  - Down  
Die Fehlerbehandlung wird beim Übergang in den inaktiven Zustand des Ports ausgelöst.  
(Von "Link up" nach "Link down")
  - "-" (Deaktiviert)  
Die Fehlerbehandlung wird nicht ausgelöst.

## Redundanz

Auf dieser Seite konfigurieren Sie, ob bei einer Zustandsänderung einer Netzwerkverbindung eine Fehlermeldung ausgelöst wird.

## Einstellung

- **Redundanz-Verlust (nur HRP)**  
Aktivieren oder deaktivieren Sie die Verbindungsüberwachung. Wenn die Redundanz der Verbindung verloren geht, wird ein Fehler signalisiert.

## PoE

### Allgemein

Auf dieser Seite legen Sie für die Spannungsversorgung (PSE) die maximale Leistung fest. Ist die Spannungsversorgung redundant ausgeführt, werden die Werte beider Spannungsversorgungen angezeigt.

## Einstellung

- **PSE**  
Zeigt die Nummer der Spannungsversorgung an.
- **Schwellenwert für Leistung [%]**  
Wenn die von den Endgeräten verbrauchte Leistung größer ist als der hier angegebene Prozentanteil, wird ein Ereignis ausgelöst.

## Port

### Einstellungen für die Ports

Für jeden einzelnen PoE-Port können Sie festlegen, ob eine Spannungsversorgung über Ethernet erfolgen soll. Außerdem können Sie für jeden angeschlossenen Verbraucher eine Priorität festlegen. Geräte, für die eine hohe Priorität festgelegt wurde, werden im Bedarfsfall gegenüber anderen bei der Spannungsversorgung bevorzugt.

## Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Port**  
Zeigt die konfigurierbaren PoE-Ports an.  
Der Port setzt sich aus der Portnummer und der Slotnummer zusammen, z. B. Port 0.1 ist Slot 0, Port 1.
- **Einstellung**  
Schalten Sie die PoE-Spannungsversorgung für diesen Port frei oder unterbrechen Sie sie.

- **Priorität**  
Wählen Sie die Priorität mit der dieser Port bei der Spannungsversorgung berücksichtigt wird.  
Es gibt folgende Einstellungsmöglichkeiten mit aufsteigender Relevanz:
  - niedrig  
niedrige Priorität
  - hoch  
mittlere Priorität
  - kritisch  
hohe PrioritätIst für zwei Ports die gleiche Priorität vorgegeben, wird im Bedarfsfall der Port mit der niedrigeren Portnummer bevorzugt.
- **Typ**  
Hier können Sie eine Zeichenkette eingeben, die das angeschlossene Gerät näher beschreibt.

## Layer 2

### Konfiguration

Auf dieser Seite werden die Funktionen des Layer 2 konfiguriert. Zu einigen Funktionen gibt es weitere Konfigurationsseiten, auf denen detailliertere Einstellungen möglich sind. Auf den Konfigurationsseiten können Sie auch die Einstellungen prüfen.

### Einstellungen

- **Protokoll-basiertes VLAN**  
Aktivieren oder deaktivieren Sie protokoll-basiertes VLAN. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN".
- **Subnetz-basiertes VLAN**  
Aktivieren oder deaktivieren Sie subnetz-basiertes VLAN. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN".
- **Dynamic MAC Aging**  
Aktivieren oder deaktivieren Sie den Mechanismus "Aging". Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Dynamic MAC Aging".
- **Redundanz-Typ**  
Folgende Einstellungen gibt es:
  - **"-" (Deaktiviert)**  
Die Redundanzfunktion ist deaktiviert.
  - **Ring**  
Aktiviert Ring-Redundanz. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Ring-Redundanz > Ring".
  - **STP**  
Wenn Sie diese Option auswählen, legen Sie bei "Redundanzverfahren" den gewünschten Redundanz-Modus fest.

- **Redundanzverfahren**

Wenn Sie bei "Redundanz-Typ" "STP" auswählen, stehen Ihnen folgende Auswahlmöglichkeiten zur Verfügung:

- **STP**

Aktiviert Spanning Tree Protocol (STP). Typische Rekonfigurationszeiten bei Spanning Tree liegen zwischen 20 und 30 Sekunden. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > MSTP".

- **RSTP**

Aktiviert Rapid Spanning Tree Protocol (RSTP). Wenn an einem Port ein Spanning Tree-Telegramm erkannt wird, fällt dieser Port von RSTP auf Spanning Tree zurück. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > MSTP".

---

**Hinweis**

Bei RSTP (Rapid Spanning Tree-Protokoll) kann es zu kurzzeitiger Schleifenbildung mit Telegrammverdoppelung oder zu Telegrammüberholungen kommen. Wenn das in Ihrem Anwendungsfall nicht akzeptabel sein sollte, müssen Sie das langsamere Standardverfahren Spanning Tree benutzen.

---

- **MSTP**

Aktiviert Multiple Spanning Tree Protocol (MSTP). Weitere Einstellungen konfigurieren Sie unter "Layer 2 > MSTP".

Wenn Sie bei "Redundanz-Typ" "Ring" auswählen, stehen Ihnen folgende Auswahlmöglichkeiten zur Verfügung:

- **Automatische Redundanz-Erkennung**

Wählen Sie diese Einstellung, um eine automatische Konfiguration des Ring-Redundanzverfahrens vorzunehmen. Bei diesem Modus stellt das Gerät automatisch fest, ob sich ein Gerät mit der Rolle "HRP-Manager" im Ring befindet. Ist dies der Fall, so nimmt das Gerät die Rolle "HRP-Client" ein. Wird kein HRP-Manager gefunden, so handeln alle Geräte mit der Einstellung "Automatische Redundanz-Erkennung" oder "MRP Auto-Manager" untereinander aus, welches Gerät die Rolle "MRP-Manager" einnimmt. Dabei wird immer das Gerät mit der niedrigsten MAC-Adresse zum "MRP-Manager". Die übrigen Geräte stellen sich automatisch auf das Ring-Redundanzverfahren "MRP-Client" ein.

- **MRP Auto-Manager**

automatischer Medienredundanz-Manager

- **MRP-Client**

Medienredundanz-Client

- **HRP-Client**

High Speed Redundancy Protocol-Client

- **HRP-Manager**

High Speed Redundancy Protocol Manager

- **Standby**

Aktivieren oder deaktivieren die Funktion "Standby-Redundanz". Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Ring-Redundanz > Standby".

- **Passive Listening**

Aktivieren oder deaktivieren Sie Funktion "Passive Listening".



- **RMON**  
Wenn Sie dieses Optionskästchen aktivieren, ermöglicht Remote Monitoring (RMON), Diagnosedaten im Gerät zu sammeln, aufzubereiten und über SNMP von einer Netzwerkmanagement-Station, die ebenfalls RMON unterstützt, auszulesen. Diese Diagnosedaten, wie zum Beispiel portbezogene Lastverläufe, ermöglichen es, Probleme im Netzwerk frühzeitig zu erkennen und zu beseitigen.
- **Dynamisches Multicast**  
Folgende Einstellungen sind möglich:
  - **"-" (Deaktiviert)**
  - **IGMP Snooping**  
Aktiviert IGMP (Internet Group Management Protocol). Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Multicast > IGMP".
  - **GMRP**  
Aktiviert GMRP (GARP Multicast Registration Protocol). Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Multicast > GMRP".

---

**Hinweis**

GMRP und IGMP können nicht gleichzeitig betrieben werden.

---

- **GVRP**  
Aktivieren oder deaktivieren Sie "GVRP" (GARP VLAN Registration Protocol). Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN > GVRP".
- **Mirroring**  
Aktivieren oder deaktivieren Sie die Port-Spiegelung. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mirroring > Port".
- **Loop Detection**  
Aktivieren oder deaktivieren Sie die Schleifenerkennung. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Loop Detection".
- **PTP**  
Legen Sie fest, wie das Gerät PTP-Nachrichten verarbeiten soll. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > PTP".
  - **aus**  
Das Gerät verarbeitet keine PTP-Nachrichten. PTP-Nachrichten werden aber nach den Regeln des Switchs weitergeleitet.
  - **transparent**  
Das Gerät übernimmt die Funktion einer Transparent Clock und leitet PTP-Nachrichten an andere Teilnehmer weiter, wobei es Eintragungen in das Korrekturfeld der PTP-Nachricht vornimmt.

## QoS

### CoS-Map

Auf dieser Seite werden CoS-Prioritäten bestimmten Warteschlangen (Traffic Queues) zugeordnet.

## Einstellungen

- **CoS**  
Zeigt CoS-Priorität der eingehenden Pakete an.
- **Warteschlange**  
Wählen Sie die Weiterleitungs-Warteschlange (Sendepriorität) aus, welcher der COS-Priorität zugeordnet wird.  
Je höher die Nummer der Warteschlange desto höher die Sendepriorität.

## DSCP-Map

Auf dieser Seite werden DSCP-Einstellungen verschiedenen Warteschlangen (Traffic Queues) zugeordnet.

## Einstellungen

- **DSCP**  
Zeigt DSCP-Priorität der eingehenden Pakete an.
- **Warteschlange**  
Wählen Sie die Weiterleitungs-Warteschlange (Sendepriorität) aus, welcher dem DSCP-Wert zugeordnet wird. Je höher die Queue-Nummer desto höher die Sendepriorität.

## Lastbegrenzung

### Begrenzung der Transferrate eingehender und ausgehender Daten

Auf dieser Seite konfigurieren Sie die Lastbegrenzung (maximale Anzahl von Datenpaketen pro Sekunde) für die einzelnen Ports. Sie können festlegen, für welche Kategorie von Telegrammen diese Grenzwerte gelten sollen.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Limit Ingress Unicast (DLF) / Limit Ingress Broadcast / Limit Ingress Multicast**  
Wählen Sie die gewünschte Einstellung aus.
  - aktiviert: Aktiviert die Funktion.
  - deaktiviert: Deaktiviert die Funktion
  - Keine Änderung: Einstellung in der Tabelle 2 bleibt unverändert
- **Ingress-Gesamtübertragungsrate [pkts/s]**  
Legen Sie die maximale Anzahl der eingehenden Pakete fest, die vom Gerät verarbeitet werden. Wenn "keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle unverändert.

- **Egress-Übertragungsrate [kb/s]**  
Legen Sie die Datenrate für alle ausgehenden Telegramme fest. Wenn "Keine Änderung" eingetragen ist, bleibt der Eintrag in der Tabelle unverändert
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt den Port an, auf den sich die Einstellungen beziehen.
- **Limit Ingress Unicast (DLF)**  
Aktivieren oder deaktivieren Sie die Datenrate zur Begrenzung der eingehenden Unicast-Telegramme mit nicht auflösbarer Adresse (Destination Lookup Failure).
- **Limit Ingress Broadcast**  
Aktivieren oder deaktivieren Sie die Datenrate zur Begrenzung der eingehenden Broadcast-Telegramme.
- **Limit Ingress Multicast**  
Aktivieren oder deaktivieren Sie die Datenrate zur Begrenzung der eingehenden Multicast-Telegramme.
- **Ingress-Gesamtübertragungsrate [pkts/s]**  
Legen Sie die maximale Anzahl der eingehenden Pakete fest, die vom Gerät verarbeitet werden.
- **Egress-Übertragungsrate [kb/s]**
- Legen Sie die Datenrate für alle ausgehenden Telegramme fest.

---

**Hinweis****Rundungen der Werte, Abweichung vom Sollwert**

Beachten Sie bei der Eingabe der Werte für Übertragungsraten, dass das WBM auf korrekte Werte rundet.

Sind Werte für Ingress-Gesamtübertragungsrate und Egress-Übertragungsrate konfiguriert, können die tatsächlichen Werte im Betrieb die eingestellten Werte um 10% unterschreiten oder überschreiten.

---

## VLAN

### Allgemein

Auf dieser Seite definieren Sie das VLAN und legen die Verwendung der Ports fest.

---

**Hinweis****Ändern der Agent VLAN ID**

Wenn der Konfigurations-PC direkt über Ethernet mit dem Gerät verbunden ist und Sie die Agent VLAN ID ändern, ist nach der Änderung das Gerät über Ethernet nicht mehr erreichbar.

---

## Wichtige Regeln für VLANs

Berücksichtigen Sie bei der Konfiguration und beim Betrieb Ihrer VLANs folgende Regeln:

- Telegramme mit der VLAN ID "0" werden wie ungetaggte Telegramme behandelt, behalten jedoch ihren Prioritätswert.
- Alle Ports am Gerät senden standardmäßig Telegramme ohne VLAN-Tag, um sicher zu gehen, dass der Endteilnehmer diese Telegramme empfangen kann.
- Bei SCALANCE X-Geräten ist an allen Ports die VLAN ID 1 voreingestellt.
- Ist an einem Port ein Endteilnehmer verbunden, dann sollen ausgehende Telegramme ohne Tag versendet werden (statischer Zugriffs-Port). Befindet sich jedoch an dem Port ein weiterer Switch, so ist das Telegramm mit einem Tag zu versehen (Trunk Port).

## Einstellungen

- **VLAN-ID**  
Tragen Sie die VLAN-ID ein.

Die Tabelle gliedert sich in folgende Spalten:

- **VLAN-ID**  
Zeigt die VLAN ID an. Die VLAN-ID wird beim Anlegen eines neuen Datensatzes einmalig vergeben und ist danach nicht mehr änderbar. Zur Änderung muss der gesamte Datensatz gelöscht und neu angelegt werden.
- **Name**  
Tragen Sie einen Namen für das VLAN ein. Der Name hat nur informativen Charakter und keine Auswirkungen auf die Konfiguration.

- **Status**  
Zeigt die Statusart des Eintrags in der Portfiltertabelle an. Dabei bedeutet statisch, dass die Adresse vom Anwender statisch eingetragen wurde. Die Angabe GVRP bedeutet, dass die Konfiguration über ein GVRP-Telegramm registriert wurde. Voraussetzung dafür ist allerdings, dass GVRP für das Gerät aktiviert wurde.
- **Liste der Ports**  
Legen Sie die Verwendung des Ports fest. Folgende Möglichkeiten gibt es:
  - "-"  
Der Port ist kein Mitglied des angegebenen VLANs. Bei der Neudefinition sind alle Ports mit der Kennung "-" belegt.
  - M  
Der Port ist Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden mit dem entsprechenden VLAN-Tag weitergeleitet.
  - R  
Der Port ist Mitglied des VLANs. Die Registrierung erfolgt über ein GVRP-Telegramm.
  - U (Großbuchstabe)  
Der Port ist ungetaggttes Mitglied des VLANs. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet. Von diesem Port werden Telegramme ohne VLAN-Tag gesendet.
  - u (Kleinbuchstabe)  
Der Port ist ungetaggttes Mitglied des VLANs, jedoch ist das VLAN nicht als Port-VLAN konfiguriert. In diesem VLAN gesendete Telegramme werden ohne VLAN-Tag weitergeleitet.
  - F  
Der Port ist kein Mitglied des angegebenen VLANs und es ist nicht möglich, dass das VLAN dynamisch über GVRP an diesem Port registriert werden kann. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > VLAN > Port-basiertes VLAN".

## GVRP

### Konfiguration der GVRP-Funktion

Über ein GVRP-Telegramm kann sich ein anderes Gerät am Port des Geräts für eine bestimmte VID registrieren. Ein anderes Gerät kann z. B. ein Endteilnehmer oder ein Switch sein. Das Gerät kann außerdem GVRP-Telegramme über diesen Port senden.

### Einstellungen

- **GVRP**  
Aktivieren oder deaktivieren Sie die Funktion "GVRP".

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Aktiviert das Senden von GVRP-Telegrammen.
  - Deaktiviert  
Deaktiviert das Senden von GVRP-Telegrammen.
  - Keine Änderung  
Keine Änderung in Tabelle 2.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports an.
- **Einstellung**  
Aktivieren oder deaktivieren Sie das Senden von GVRP-Telegrammen.

## Port-basiertes VLAN

### Verarbeitung empfangener Telegramme

Auf dieser Seite legen Sie die Konfiguration der Port-Eigenschaften für den Telegrammempfang fest.

### Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **Port**  
Zeigt an, dass die Einstellungen für alle Ports gültig sind.
- **Priorität / Port VID / erlaubte Telegramme Frames / Ingress Filterung**
- Wählen Sie die Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports und Link-Aggregationen an. Der Port setzt sich aus der Portnummer und der Slotnummer zusammen, z. B. Port 0.1 ist Slot 0, Port 1.
- **Priorität**  
Wählen Sie die Priorität aus, mit der ungetaggte Telegramme versehen werden.  
Die CoS-Priorität (Class of Service), die im VLAN-Tag verwendet wird. Wird ein Telegramm ohne Tag empfangen, wird ihm diese Priorität zugeordnet. Diese Priorität legt fest, wie dieses Telegramm im Vergleich zu anderen Telegrammen weiterhin bearbeitet wird.  
Es gibt insgesamt acht Prioritäten, mit den Werten 0 bis 7, wobei 7 der höchsten Priorität entspricht (IEEE 802.1p Port Priority).
- **Port VID**  
Wählen Sie die VLAN-ID aus. Nur die VLAN-IDs sind wählbar, die Sie auf der Seite "VLAN > Allgemein" definiert haben.  
Wenn ein empfangenes Telegramm kein VLAN-Tag hat, so wird es um ein Tag mit der hier angegebenen VLAN-ID ergänzt und entsprechend den Regeln am Port gesendet.
- **Erlaubte Telegrammtypen**  
Legen Sie fest, welche Arten von Telegrammen akzeptiert werden. Es gibt folgende Alternativen:
  - Nur getaggte Telegramme  
Das Gerät verwirft alle ungetaggten Telegramme. Andernfalls gelten die Weiterleitungsregeln entsprechend der Konfiguration.
  - Alle  
Das Gerät leitet alle Telegramme weiter
- **Ingress Filterung**  
Legen Sie fest, ob die VID von empfangenen Telegrammen ausgewertet wird.  
Folgende Möglichkeiten gibt es:
  - Aktiviert  
Die VLAN-ID empfangener Telegramme bestimmt die Weiterleitung: Für die Weiterleitung eines VLAN-getaggten Telegramms muss der Empfangsport Mitglied im selben VLAN sein. Am Empfangsport werden Telegramme aus unbekanntem VLANs verworfen.
  - Deaktiviert  
Alle Telegramme werden weitergeleitet.

## Protokoll-basierte VLAN-Gruppe

Auf dieser Seite legen Sie Gruppen fest, denen ein Protokoll zugeordnet wird.

## Einstellungen

- **Protokoll-basiertes VLAN**  
Aktivieren oder deaktivieren Sie die protokoll-basierte VLAN-Zuordnung.
- **Protokollwert**  
Tragen Sie den hexadezimalen Protokollwert ein.  
Einige Beispiele sind:
  - Profinet: 88:92
  - IP: 08:00
  - Novell: 81:37
  - netbios: f0:f0
  - appletalk: 80:9b
- **Gruppen-ID**  
Tragen Sie die ID der Gruppe ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Protokollwert**  
Zeigt den Protokollwert an.
- **Gruppen-ID**  
Zeigt die Gruppen ID an.

## Protokoll-basierter VLAN-Port

Auf dieser Seite legen Sie fest, welches Protokoll und welches VLAN dem einzelnen Port zugeordnet wird.

## Einstellungen

- **Port**  
Wählen Sie in den gewünschten Port aus. Alle verfügbaren Ports und die Link-Aggregationen werden angeboten.
- **Gruppen-ID**  
Wählen Sie in der Klappliste die Gruppen-ID aus. Die ID legen Sie unter "Layer 2 > VLAN > Protokoll-basierte VLAN-Gruppe" fest.

Die Tabelle gliedert sich in folgende Spalten:

- **Port**  
Alle verfügbaren Ports und die Link-Aggregationen werden angezeigt.
- **Gruppen-ID**  
Zeigt die Gruppen ID an, die dem Port zugeordnet ist.
- **VLAN-ID**  
Wählen Sie die gewünschte VLAN-ID aus, die Sie dem Port zuordnen.

## Ipv4 Subnetz-basiertes VLAN

Auf dieser Seite legen Sie fest, welche VLAN-ID dem Subnetz zugeordnet wird.



## Einstellungen

- **Subnetz-basiertes VLAN**  
Aktivieren oder deaktivieren Sie die subnetz-basierte VLAN-Zuordnung
- **Port**  
Wählen Sie den Port aus. Alle verfügbaren Ports und die Link-Aggregationen werden angeboten.
- **Subnetz**  
Tragen Sie die Netzadresse ein.  
Beispiel: 192.168.10.0 für das Netz 192.168.10.x mit den Teilnehmern 192.168.10.1 bis 192.168.10.254.

Die Tabelle gliedert sich in folgende Spalten:

- **Port**  
Alle verfügbaren Ports und die Link-Aggregationen werden angezeigt.
- **Subnetz**  
Zeigt das Subnetz an, das dem Port zugeordnet ist.
- **VLAN-ID**  
Wählen Sie die VLAN-ID, die Sie dem Port bzw. dem Subnetz zuordnen.

## Mirroring

### Allgemein

Auf dieser Seite legen Sie die Grundeinstellungen für die die Port-Spiegelung fest.

## Einstellungen

- **Mirroring**  
Aktivieren oder deaktivieren Sie die Spiegelung des Datenverkehrs.
- **Monitor-Barrier**  
Aktivieren oder deaktivieren Sie die Option, um die Kommunikation über den Monitor Port einzuschränken.

---

### Hinweis

#### Monitor-Barrier

Wenn Sie Monitor-Barrier einschalten, wird der Datenverkehr auf dem Ziel-Port automatisch geblockt (Broadcast, Multicast, Unicast, DCP-forwarding, LLDP), sodass nur noch der gespiegelte Verkehr anliegt. Um anderen Datenverkehr wiederzuzulassen, müssen Sie diesen konfigurieren. Die vorherigen Zustände dieser Optionen werden nach Beendigung von Monitor Barrier nicht wiederhergestellt und müssen neu konfiguriert werden.

---

- **Aktiviert**  
Der Monitor-Port ist vom normalen Switching von Telegrammen ausgenommen.
- **Deaktiviert**  
Für die Kommunikation über den Monitor-Port gibt es keine Einschränkung.

Die Tabelle hat folgende Spalten:

- **Session-ID**  
Aktiviert oder deaktivieren Sie am gewünschten Port das Mithören der eingehenden Pakete.
- **Session-Typ**  
Legen Sie fest, welcher Datenverkehr gespiegelt wird. Folgende Möglichkeiten gibt es:
  - ' '   
keine
  - Port-basiert  
Port-basiertes Mirroring. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mirroring > Port".
  - VLAN  
VLAN-basiertes Mirroring. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mirroring > VLAN".
  - MAC ACL  
Mirroring der MAC Access Control List. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mirroring > MAC Flow".
  - IP ACL  
Mirroring der IP Access Control List. Weitere Einstellungen konfigurieren Sie unter "Layer 2 > Mirroring > IP Flow".
- **Status**  
Zeigt an, ob die Spiegelung aktiv ist oder nicht.
- **Ziel-Port**  
Wählen Sie den Ziel-Port, auf den in dieser Session gespiegelt wird.

## Port

### Ports spiegeln

Einen Port spiegeln bedeutet, dass der Datenverkehr an einem Port (gespiegelter Port) des IE-Switches auf einen anderen Port (Monitor-Port) kopiert wird. Sie können einen oder mehrere Ports auf einen Monitor-Port spiegeln.

Wird am Monitor-Port ein Protokollanalysator angeschlossen, kann damit der Datenverkehr am gespiegelten Port aufgezeichnet werden, ohne dass die Verbindung dort unterbrochen wird. Dadurch ist eine rückwirkungsfreie Untersuchung des Datenverkehrs möglich.

Voraussetzung hierfür ist, dass am gleichen Gerät ein freier Port als Monitor-Port zur Verfügung steht.

---

#### **Hinweis**

Wenn die maximale Datenrate des gespiegelten Ports höher ist als die des Monitor-Ports, kann es zu Datenverlusten kommen und der Monitor-Port gibt dann nicht mehr alle Abläufe am gespiegelten Port wieder. Auf einen Monitor-Port können mehrere Ports gleichzeitig gespiegelt werden.

Das Spiegeln eines Ports funktioniert nicht über Switch-Core-Grenzen.

Sie müssen die Portspiegelung ausschalten, wenn Sie an den Monitor-Port ein normales Endgerät anschließen.

---

### **Einstellungen**

- **Session-ID**  
Wählen Sie die Session-ID aus. Nur Session-IDs vom Session-Typ "Port-basiert" sind verfügbar.
- **Port**  
Zeigt den Port an, der überwacht werden soll.
- **Ingress-Mirroring**  
Aktivieren oder deaktivieren Sie am gewünschten Port die Spiegelung der eingehenden Pakete.
- **Egress-Mirroring**  
Aktivieren oder deaktivieren Sie am gewünschten Port die Spiegelung der ausgehenden Pakete.

### **VLAN**

#### **VLAN-Quellen der Port-Spiegelungsfunktion**

Auf dieser Seite bestimmen Sie das VLAN, dessen eingehender Datenverkehr auf den Monitor-Port gespiegelt wird.

### **Einstellungen**

- **Session-ID**  
Wählen Sie die Session-ID aus. Nur Session-IDs vom Session-Typ "VLAN" sind verfügbar.

Die Tabelle gliedert sich in folgende Spalten:

- **VLAN-ID**  
Zeigt die VLAN-ID an. Die VLAN-ID kann nur beim Anlegen eines neuen Datensatzes einmalig vergeben werden und ist danach nicht mehr änderbar. Zur Änderung muss der gesamte Datensatz gelöscht und neu angelegt werden.
- **Ingress-Mirroring**  
Aktivieren oder deaktivieren Sie die Spiegelung der eingehenden Telegramme.

## MAC Flow

Der MAC ACL Filter bestimmt, welche Daten am Monitor-Port bereitgestellt werden.

### Einstellungen

- **Session-ID**  
Wählen Sie die Session-ID aus. Nur Session-IDs vom Session-Typ "MAC ACL" sind verfügbar.
- **ACL-Filternummer**  
Zeigt die Nummer des ACL-Filters an. Den MAC ACL-Filter konfigurieren Sie unter "Sicherheit > Port ACL MAC".
- **Ingress-Mirroring**  
Zeigt an, ob eingehende Pakete gespiegelt werden.

---

#### Hinweis

#### Regeln

Eine Regel, die für das Ingress-Mirroring ausgewählt wurde, wird nur dann aktiv, wenn sie auch auf mindestens einem Port als Port Ingress-Regeln konfiguriert wurde. Die Port Ingress-Regeln konfigurieren Sie unter "Sicherheit > Port MAC IP > Port Ingress-Regeln".

---

- **Quell-MAC-Adresse**  
Zeigt die MAC-Adresse des Absenders an.
- **Ziel-MAC-Adresse**  
Zeigt die MAC-Adresse des Empfängers an.
- **Egress-Port**  
Zeigt eine Liste aller Egress-Ports, für die diese Regel gilt.
- **Ingress-Port**  
Zeigt eine Liste aller Ingress-Ports, für die diese Regel gilt.

## IP Flow

Der IP ACL Filter bestimmt, welche Daten auf den Monitor-Port gespiegelt werden.

### Einstellungen

- **Session-ID**  
Wählen Sie die Session-ID aus. Nur Session-IDs vom Session-Typ "IP ACL" sind verfügbar.
- **ACL-Filternummer**  
Zeigt die Nummer des ACL-Filters an. Den IP ACL-Filter konfigurieren Sie unter "Sicherheit > Port ACL IP".

- **Ingress-Mirroring**  
Zeigt an, ob eingehende Pakete gespiegelt werden.

---

**Hinweis**

**Regeln**

Eine Regel, die für das Ingress-Mirroring ausgewählt wurde, wird nur dann aktiv, wenn sie auch auf mindestens einem Port als Port Ingress-Regeln konfiguriert wurde. Die Port Ingress-Regeln konfigurieren Sie unter "Sicherheit > Port ACL IP > Port Ingress-Regeln".

---

- **Quell-IP-Adresse**  
Zeigt die IP-Adresse des Absenders an.
- **Quell-Subnetzmaske**  
Zeigt die Subnetzmaske des Absenders an.
- **Ziel-IP-Adresse**  
Zeigt die IP-Adresse des Empfängers an.
- **Ziel-Subnetzmaske**  
Zeigt die Subnetzmaske des Empfängers an.
- **Egress-Port**  
Zeigt eine Liste aller Egress-Ports, für die diese Regel gilt.
- **Ingress-Port**  
Zeigt eine Liste aller Ingress-Ports, für die diese Regel gilt.

## Dynamic MAC Aging

Das Gerät lernt automatisch die Quelladressen der angeschlossenen Teilnehmer.

Diese Information wird dazu benutzt, um Datentelegramme gezielt an die betroffenen Teilnehmer weiterzuleiten. Die Netzlast für die anderen Teilnehmer wird reduziert. Erhält ein Gerät innerhalb einer bestimmten Zeitspanne kein Telegramm, dessen Quelladresse mit einer gelernten Adresse übereinstimmt, dann wird die gelernte Adresse gelöscht. Dieser Mechanismus wird als "Aging" bezeichnet. Durch Aging wird verhindert, dass Telegramme fehlgeleitet werden, wenn z.B. ein Endgerät (beispielsweise ein Programmiergerät) an einen anderen Port angeschlossen wird. Wenn die Option nicht aktiviert ist, löscht ein Gerät gelernte Adressen nicht automatisch.

## Einstellungen

- **Dynamic MAC Aging**  
Aktivieren oder deaktivieren Sie die Funktion zum automatischen Aging von gelernten MAC-Adressen:
- **Aging Time [s]**  
Tragen Sie die Zeitspanne in Sekunden ein. Nach dieser Zeitspanne wird eine gelernte Adresse gelöscht, wenn das Gerät keine weiteren Telegramme von dieser Absenderadresse mehr empfängt.

## Ring-Redundanz

### Ring

Auf dieser Seite ist die gewünschte Betriebsart für die schnelle Ring-Redundanz wählbar. Das Ring-Redundanzverfahren "Automatische Redundanz-Erkennung" ist bei Auslieferung des Gerätes voreingestellt.

---

#### **Hinweis**

Die Ring Redundanz kann nicht eingeschaltet werden, wenn auf dem Gerät noch ein (M/R/S-) Spanning Tree aktiviert ist.

---

## Einstellungen

- **Ring-Redundanz**  
Aktivieren oder deaktivieren Sie die Ring-Redundanz.
- **Ring-Redundanzverfahren**  
Legen Sie die Betriebsart der Ring-Redundanz fest. Folgende Möglichkeiten gibt es:
  - **"-" (Deaktiviert)**  
Redundanz-Funktion ist deaktiviert.
  - **Automatische Redundanz-Erkennung**  
Wählen Sie diese Einstellung, um eine automatische Konfiguration des Ring-Redundanzverfahrens vorzunehmen. Bei diesem Modus stellt das Gerät automatisch fest, ob sich ein Gerät mit der Rolle "HRP-Manager" im Ring befindet. Ist dies der Fall, so nimmt das Gerät die Rolle "HRP-Client" ein. Wird kein HRP-Manager gefunden, so handeln alle Geräte mit der Einstellung "Automatische Redundanz-Erkennung" oder "MRP Auto-Manager" untereinander aus, welches Gerät die Rolle "MRP-Manager" einnimmt. Dabei wird immer das Gerät mit der niedrigsten MAC-Adresse zum "MRP-Manager". Die übrigen Geräte stellen sich automatisch auf das Ring-Redundanzverfahren "MRP-Client" ein.
  - **MRP Auto-Manager**  
automatischer Medienredundanz-Manager
  - **MRP-Client**  
Medienredundanz-Client.
  - **HRP-Client**  
High Speed Redundancy Protocol-Client.
  - **HRP-Manager**  
High Speed Redundancy Protocol-Manager

### Ring-Ports

Legen Sie die Ports fest, die bei der Medienredundanz in Ring-Topologien als Ring-Ports verwendet werden sollen.

---

### Hinweis

Mit dem Wiederherstellen der Werkseinstellungen wird das voreingestellte Ring-Redundanzverfahren "Automatische Redundanz-Erkennung" aktiv.

Außerdem wird auch die Ringportkonfiguration auf die werkseitig voreingestellten Ports zurückgesetzt. Wurden zuvor andere Ports als Ring-Ports verwendet, so kann bei entsprechendem Anschluss ein zuvor korrekt konfiguriertes Gerät kreisende Telegramme und damit den Ausfall des Datenverkehrs verursachen.

---

## Standby

### Standby-Manager

Der Standby-Manager erlaubt die redundante Kopplung von zwei HRP-Ringen. Dazu müssen innerhalb eines Rings zwei benachbarte Geräte als Standby-Partner konfiguriert werden. Aktivieren sie den Standby-Manager für beide Standby-Partner und wählen Sie, über welchen Port das Gerät mit dem zu koppelnden Ring verbunden ist. Als "Name der Standby-

Verbindung" muss für beide Partner ein eindeutiger Name im Ring vergeben werden, mit dem die beiden zusammengehörenden Geräte als Standby-Partner identifiziert werden.

---

**Hinweis**

Um die Funktion zu nutzen, muss HRP aktiviert sein.

---

## Einstellungen

- **Standby**  
Aktivieren oder deaktivieren Sie den Standby-Manager.

---

**Hinweis**

Sind zwei Geräte über Standby-Funktion gekoppelt, muss die Funktion "Standby" an beiden Geräten aktiviert sein.

---

**Hinweis**

Der Standby-Manager erfordert immer einen aktivierten HRP-Client.

---

- **Name der Standby-Verbindung**  
Tragen Sie den Namen für die Standby-Verbindung ein. Durch diesen Namen wird das Master-/Slave-Gerätepaar definiert. Beide müssen im gleichen Ring liegen. Hierzu wird der gleiche Name auf zwei Geräten eines Rings eingetragen.

---

**Hinweis**

Achten Sie darauf, dass der Standby-Name (für ein Gerätepaar) nur einmal im Netz verwendet wird.

---

- **Standby-Master-Betriebsart für Gerät vorgeben**  
Wenn aktiviert, wird das Gerät unabhängig von seiner MAC-Adresse als Standby-Master konfiguriert.
  - Wenn bei keinem der beiden Geräte, bei denen die Standby-Funktion eingeschaltet ist, die Einstellung aktiviert, dann übernimmt im fehlerfreien Zustand das Gerät mit der höheren MAC-Adresse die Funktion des Standby-Masters.
  - Wenn die Einstellung bei beiden Geräten aktiviert ist, oder wenn die Einstellung nur von einem Gerät unterstützt wird, dann wird der Standby-Master ebenfalls anhand der MAC-Adresse ausgewählt.

Wichtig ist diese Art der Zuordnung insbesondere bei einem Gerätetausch. Abhängig von den MAC-Adressen kann das bisherige Gerät mit Slave-Funktion die Rolle des Standby-Masters übernehmen.

Die Tabelle hat folgende Spalten:



- **Port**  
Zeigt den Port an, auf den sich die Einstellung bezieht.
- **Einstellung**  
Legen Sie fest, welche Ports Standby-Ports sind. Die Standby-Ports sind an der Umleitung des Datenverkehrs beteiligt.  
Im ungestörten Fall sind nur die Standby-Ports des Masters aktiv und übernehmen den Datenverkehr in die angeschlossenen HRP-Ringe (-Linien). Fällt der Master oder die Ethernet-Verbindung (Link) eines der Standby-Ports des Masters aus, werden alle Standby-Ports des Masters abgeschaltet und die Standby-Ports des Slaves aktiviert. Damit wird wieder eine funktionierende Ethernet-Verbindung in die angeschlossenen Netzsegmente (HRP-Ringe/-Linien) hergestellt.

## MSTP

### Allgemein

#### Allgemeine Einstellungen von MSTP

Auf dieser Seite konfigurieren Sie die Einstellungen für das MSTP. Standardmäßig ist Rapid Spanning Tree aktiviert, das über einen Schalter in den MSTP, RSTP oder STP kompatiblen Modus gesetzt werden kann.

In der jeweiligen Konfigurationsseite dieser Funktionen sind detailliertere Einstellungen möglich.

Je nach Kompatibilitäts-Modus können Sie in der jeweiligen Konfigurationsseite die entsprechende Funktion konfigurieren.

### Einstellungen

- **MSTP**  
Aktivieren oder deaktivieren Sie MSTP.
- **Protokoll-Kompatibilität**  
Wählen Sie den Kompatibilitäts-Modus von MSTP aus. Z. B. wenn Sie RSTP wählen, dann verhält sich MSTP wie ein RSTP.  
Folgende Einstellungen gibt es:
  - STP
  - RSTP
  - MSTP

### CIST Allgemein

Auf dieser Seite konfigurieren Sie CIST.

## Einstellungen

- **Bridge-Priorität**  
Anhand der Bridge-Priorität wird festgelegt, welches Gerät Root-Bridge wird. Die Bridge mit der höchsten Priorität wird Root-Bridge. Je kleiner der Wert, desto höher die Priorität. Wenn in einem Netz mehrere Geräte die gleiche Priorität besitzen, wird das Gerät Root-Bridge, dessen MAC-Adresse den niedrigsten Zahlenwert hat. Beide Parameter, Bridge-Priorität und MAC-Adresse, bilden zusammen die Bridge-Kennung. Da die Root-Bridge alle Wegeänderungen verwaltet, sollte sie wegen der Laufzeit der Telegramme möglichst zentral angeordnet sein.
- **Bridge Hello Time [s]**  
Jede Bridge versendet regelmäßig Konfigurationstelegramme (BPDUs). Der Zeitabstand zwischen zwei solchen Telegrammen ist die Hello-Time. Der Standardwert für diesen Parameter beträgt 2 Sekunden.
- **Bridge Forward Delay [s]**  
Neue Konfigurationsinformationen werden von einer Bridge nicht sofort, sondern erst nach dem im Parameter festgelegten Zeitraum angewendet. So wird sichergestellt, dass der Betrieb entsprechend der neuen Topologie erst gestartet wird, wenn alle Bridges die notwendigen Informationen haben.
- **Bridge Max Age**  
Bridge Max Age definiert das maximale "Alter", welches eine empfangene BPDU haben darf, um vom Switch als gültig akzeptiert zu werden.
- **Bridge Max Hop Count**  
Dieser Parameter gibt an, wie viele MSTP-Teilnehmer eine BPDU passieren darf. Wird eine MSTP-BPDU empfangen, deren Hop Count den hier konfigurierten Wert übersteigt, wird sie verworfen.
- **Name der Region**  
Tragen Sie den Namen der MSTP-Region ein, zu der dieses Gerät gehört. Defaultmäßig ist hier die MAC-Adresse des Gerätes eingetragen. Dieser Wert muss auf allen Geräten die zur selben MSTP-Region gehören gleich sein.
- **Version der Region**  
Tragen Sie die Versionsnummer der MSTP-Region ein, in der sich das Gerät befindet. Dieser Wert muss auf allen Geräten die zur selben MSTP-Region gehören gleich sein.

## CIST Port

### Konfiguration der MSTP-CIST-Ports

In der Tabelle wird beim Aufruf der Seite der aktuelle Stand der Konfiguration der Port-Parameter angezeigt.

Klicken Sie zur Konfiguration in die entsprechenden Felder der Port-Tabelle.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **MSTP-Status**  
Wählen Sie aus der Klappliste die Einstellung. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Port ist im Spanning-Tree integriert.
  - Deaktiviert  
Port ist im Spanning-Tree nicht integriert.
  - Keine Änderung  
Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports an.
- **MSTP-Status**  
Legen Sie fest, ob der Port im Spanning-Tree integriert ist oder nicht.

---

### Hinweis

Wenn Sie die Option "MSTP Status" für einen Port deaktivieren, kann es zur Schleifenbildung kommen. Die Topologie muss beachtet werden.

---

- **Priorität**  
Tragen Sie die Priorität des Ports ein. Die Priorität wird nur dann ausgewertet, wenn die Pfadkosten gleich sind.  
Der Wert muss durch 16 teilbar sein. Wenn der Wert nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
- **erm. Kosten**  
Tragen Sie die Wegekostenberechnung ein. Wenn Sie den Wert "0" eintragen, wird im Feld "Pfadkosten" der automatisch ermittelte Wert angezeigt.

- **Edge-Typ**

Legen Sie die Art des Edge-Port fest. Sie haben folgende Möglichkeiten:

- "-"  
Edge-Port ist deaktiviert. Der Port wird wie ein "no-Edge-Port" behandelt.
- Admin  
Wählen Sie diese Option, wenn sich an diesem Port ein immer Endgerät befindet. Sonst wird bei jeder Verbindungsänderung eine Rekonfiguration des Netzwerks ausgelöst.
- Auto  
Wählen Sie diese Option, wenn an diesem Port automatisch erkannt werden soll, ob ein Endgerät angeschlossen ist. Beim ersten Verbindungsaufbau wird der Port wie ein "no Edge Port" behandelt.
- Admin/Auto  
Wählen Sie diese Optionen, wenn Sie an diesem Port eine Kombination aus beiden betreiben. Beim ersten Verbindungsaufbau wird der Port als Edge Port behandelt.

- **P.t.P. Type**

Wählen Sie die gewünschte Option aus. Die Auswahl ist abhängig vom eingestellten Port.

- "-"  
Punkt zu Punkt wird automatisch ermittelt. Steht der Port auf Halbduplex, wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.
- P.t.P.  
Auch bei Halbduplex wird von einer Punkt zu Punkt-Verbindung ausgegangen.
- Shared Media  
Auch bei einer Vollduplexverbindung wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.

---

**Hinweis**

Punkt zu Punkt-Verbindung bedeutet eine direkte Verbindung zwischen zwei Geräten. Eine Shared Media-Verbindung ist z.B. eine Verbindung zu einem Hub.

---

- **P.t.P. Type**  
Wählen Sie in der Klappliste die gewünschte Option aus. Die Auswahl ist abhängig vom eingestellten Port.
  - "\_"  
Punkt zu Punkt wird automatisch ermittelt. Steht der Port auf Halbduplex, wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.
  - P.t.P.  
Auch bei Halbduplex wird von einer Punkt zu Punkt-Verbindung ausgegangen.
  - Shared Media  
Auch bei einer Vollduplexverbindung wird nicht von einer Punkt zu Punkt-Verbindung ausgegangen.

---

**Hinweis**

Punkt zu Punkt-Verbindung bedeutet eine direkte Verbindung zwischen zwei Geräten. Eine Shared Media-Verbindung ist z.B. eine Verbindung zu einem Hub.

---

- **Hello Time**  
Tragen Sie das Intervall ein, nach der die Bridge Konfigurations BPDUs sendet

---

**Hinweis**

Die portspezifische Einstellung der Hello-Time ist nur mit MSTP kompatiblen Modus möglich.

---

## MST Allgemein

### Multiple Spanning Tree-Konfiguration

Bei MSTP können zusätzlich zu RSTP mehrere VLANs in einem LAN mit eigenen RSTP-Bäumen verwaltet werden.

### Einstellungen

- **MSTP-Instanz-ID**  
Tragen Sie die Nummer der MSTP-Instanz ein.  
Die Tabelle gliedert sich in folgende Spalten:
- **MSTP-Instanz-ID**  
Zeigt die Nummer der MSTP-Instanz an.
- **Root-Adresse**  
Zeigt die MAC-Adresse der Root-Bridge an
- **Root-Priorität**  
Zeigt die Priorität der Root-Bridge an.

- **Bridge-Priorität**  
Tragen Sie die Bridge-Priorität ein. Der Wert für die Bridge-Priorität ist ein ganzzahliges Vielfaches von 4096.
- **VLAN-ID**  
Tragen Sie die VLAN-ID ein. Sie können hier auch Bereiche mit Start-ID, "-", End-ID angeben. Mehrere Bereiche oder IDs werden durch "," separiert.

## MST Port

### Konfiguration der Multiple Spanning Tree Port Parameter

Auf dieser Seite stellen Sie die Parameter für die Ports der konfigurierten Multiple Spanning Tree Instanzen ein.

### Einstellungen

- **MSTP-Instanz-ID**  
Wählen Sie die ID der MSTP-Instanz aus.

Die Tabelle gliedert sich in folgende Spalten:

- **Port**  
Zeigt alle verfügbaren Ports und Link-Aggregationen an.
- **MSTP-Instanz-ID**  
ID der MSTP-Instanz.
- **MSTP-Status**  
Aktivieren oder deaktivieren Sie MSTP für diesen Port.
- **Priorität**  
Tragen Sie die Priorität des Ports ein. Die Priorität wird nur dann ausgewertet, wenn die Pfadkosten gleich sind.  
Der Wert muss durch 16 teilbar sein. Wenn der Wert nicht durch 16 teilbar ist, wird der Wert automatisch angepasst.
- **erm. Kosten**  
Tragen Sie in das Eingabefeld die Wege-Kostenberechnung ein. Wenn Sie hier 0 eintragen, wird im nächsten Feld "Path Cost" der automatisch ermittelte Wert angezeigt.

### Erweiterte Kompatibilität für Passive Listening

#### Aktivieren der Funktion

Auf dieser Seite aktivieren Sie die erweiterte Kompatibilität für Passive Listening.

## Einstellungen

- **Erweiterte Kompatibilität für Passive Listening**  
Aktivieren oder deaktivieren Sie diese Funktion für das gesamte Gerät.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Aktiviert die Funktion für alle Ports des Gerätes.
  - Deaktiviert  
Deaktiviert die Funktion für alle Ports des Gerätes.
  - Keine Änderung  
Keine Änderung in Tabelle 2.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt den Port des Gerätes an.
- **Einstellung**  
Aktivieren oder deaktivieren Sie die Funktion für diesen Port.

## Loop Detection

Mit der Funktion "Loop Detection" legen Sie fest, für welche Ports Schleifenerkennung aktiviert werden soll. Von den betreffenden Ports werden spezielle Testtelegramme, die Loop Detection-Telegramme gesendet. Wenn diese Telegramme wieder zum Gerät zurück gesendet werden, dann liegt eine Schleife ("Loop") vor.

Von einem "Local Loop" unter Beteiligung dieses Gerätes spricht man, wenn die Telegramme an einem anderen Port desselben Gerätes wieder empfangen werden. Wenn die ausgesendeten Telegramme wieder am gleichen Port empfangen werden, ist eine Schleife "Remote Loop" an anderen Netzkomponenten aufgetreten.

---

### Hinweis

Eine Schleife ist ein Fehler im Netzaufbau, der beseitigt werden muss. Die Schleifenerkennung kann helfen den Fehler schneller zu finden, behebt ihn jedoch nicht. Die Schleifenerkennung ist nicht dazu geeignet, die Netzwerkverfügbarkeit durch den gezielten Einbau von Schleifen zu erhöhen.

---

### Hinweis

Beachten Sie, dass die Schleifenerkennung nur auf Ports möglich ist, die nicht als Ring-Port oder Standby-Port konfiguriert wurden.

---

## Einstellungen

- **Loop Detection**  
Aktivieren oder deaktivieren Sie die Schleifenerkennung.
- **VLAN Loop Detection**  
Aktivieren oder deaktivieren Sie die Schleifenerkennung bei VLAN.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind
- **Schwellenwert / Remote Reaction / Local Reaction**  
Legen Sie die gewünschten Einstellungen fest.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports an.
- **Einstellung**  
Legen Sie fest, wie der Port mit Loop Detection-Telegrammen verfahren soll.

---

### Hinweis

Durch die Testtelegramme entsteht zusätzliche Netzlast. Wir empfehlen, nur einzelne Switches, z. B. an den Abzweigungen vom Ring, als "sender" zu konfigurieren und die anderen als "forwarder".

---

- sender  
Loop Detection-Telegramme werden ausgesendet und weitergeleitet.
  - forwarder  
Loop Detection-Telegramme von anderen Geräten werden weitergeleitet.
  - gesperrt  
Die Weiterleitung der Loop-Detection-Telegramme wird blockiert.
- **Schwellenwert**  
Legen Sie fest, nach wie vielen empfangenen Loop Detection-Telegrammen von einer Schleife ausgegangen wird.



- **Remote Reaction**  
Legen Sie fest, wie der Port bei Auftreten einer Remote-Loop reagieren soll. Wählen Sie aus der Klappliste eine der beiden Optionen:
  - keine Aktion: Eine Schleife hat keine Auswirkungen auf den Port.
  - deaktivieren: Der Port wird geblockt.
- **Local Reaction**  
Legen Sie fest, wie der Port bei Auftreten eines Local Loop reagieren soll. Wählen Sie aus der Klappliste eine der beiden Optionen:
  - keine Aktion: Eine Schleife hat keine Auswirkungen auf den Port.
  - deaktivieren: Der Port wird geblockt

## Link Aggregation

### Bündelung von Netzwerk Verbindungen für Redundanz und höhere Bandbreite

Die Link-Aggregationen nach IEEE 802.3ad erlaubt es, mehrere Verbindungen zwischen benachbarten Geräten zu bündeln, um so höhere Bandbreiten zu erreichen und zusätzlich für Ausfallsicherheit zu sorgen.

Hierbei werden Ports auf beiden Partnergeräten in Verbindungsbündel eingebunden und dann die Geräte über diese Ports miteinander verbunden. Um Ports (also Links) korrekt einem Partnergerät zuzuordnen, wird das Link-Aggregation Control Protocol (LACP) aus dem Standard IEEE 802.3ad verwendet.

Bis zu 8 Link-Aggregationen können definiert werden. Jeder Link-Aggregation können max. 8 Ports zugewiesen werden.

## Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Port**  
Zeigt die virtuelle Port-Nummer dieser Link-Aggregation an. Die Bezeichnung wird intern von der Firmware vergeben.
- **Link Aggregation-Name**  
Tragen Sie einen Namen für die Link-Aggregation fest Dieser Name kann bei der Konfiguration vom Benutzer angegeben werden. Der Name ist nicht zwingend notwendig, kann aber hilfreich sein, die verschiedenen Link-Aggregationen zu unterscheiden.
- **Status**  
Aktivieren oder deaktivieren Sie die Link-Aggregation.
- **MTU**  
Legen Sie die Paketgröße fest.

- **LACP**
  - An  
Aktiviert das Senden von LACP-Telegrammen.
  - Aus  
Deaktiviert das Senden von LACP-Telegrammen.
- **Frame Distribution**

Legen Sie die Art der Verteilung von Paketen auf die einzelnen Links einer Aggregation fest.

  - Ziel- & Quell-MAC-Adresse  
Die Verteilung basiert auf einer Kombination der Ziel- und Quell-MAC-Adresse.
  - Ziel- & Quell-IP-Adresse  
Die Verteilung basiert auf einer Kombination der Ziel- und Quell-IP-Adresse.
- **Port**

Zeigt die Ports an, die zu dieser Link-Aggregation gehören. Dabei können aus der Klappliste folgende Werte gewählt werden:

  - "-" (Deaktiviert)  
Link-Aggregation ist deaktiviert.
  - "a" (Aktiv)  
Der Port sendet LACP-Telegramme und nimmt nur an der Link-Aggregation teil, wenn LACP-Telegramme empfangen werden.
  - "p" (Passiv)  
Der Port nimmt nur an der Link-Aggregation teil, wenn LACP-Telegramme empfangen werden.
  - "o" (Ein)  
Der Port nimmt an der Link-Aggregation teil und sendet keine LACP-Telegramme.

---

**Hinweis**

Innerhalb einer "Link-Aggregation" sind immer nur Ports folgender Konfiguration möglich:

- alle Ports mit "o"
  - alle Ports mit "a" oder "p".
- 

## DCP-Weiterleitung

### Anwendungen

Das DCP-Protokoll wird von STEP 7 und dem PST-Tool für die Konfiguration und Diagnose verwendet. In der Werkseinstellung ist DCP auf allen Ports aktiviert, d.h. empfangene DCP-Telegramme werden auf allen Ports weitergeleitet. Mit dieser Option haben Sie die Möglichkeit das Aussenden der Telegramme für einzelne Ports auszuschalten, um z.B. einzelne Netzbereiche von der Konfiguration per PST-Tool abzuschotten, bzw. um das gesamte Netz in kleinere Teilnetze für die Konfiguration und Diagnose zu unterteilen.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports an.
- **Einstellung**  
Legen Sie fest, ob der Port DCP-Telegramme ausgangsseitig blocken oder weiterleiten soll. Folgende Möglichkeiten gibt es:
  - **Weiterleiten**  
An diesem Port werden DCP-Telegramme weitergeleitet.
  - **Blockieren**  
An diesem Port werden ausgangsseitig keine DCP-Telegramme weitergeleitet. Ein Empfangen ist jedoch über diesen Port weiterhin möglich.

## LLDP

### Link Layer Discovery Protocol (LLDP)

PROFINET benutzt das LLDP-Protokoll für die Topologie-Diagnose. In der Werkseinstellung ist LLDP auf der Schnittstelle aktiviert, d. h. LLDP-Telegramme werden gesendet und empfangen. LLDP wird nur von der Ethernet-Schnittstelle und von der SFP-Schnittstelle P1 unterstützt.

Auf dieser WBM-Seite aktivieren oder deaktivieren Sie das Senden oder Empfangen von LLDP-Telegrammen auf der Schnittstelle.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt den Port an.
- **Einstellung**  
Legen Sie die LLDP-Funktionalität fest. Folgende Möglichkeiten gibt es:
  - Rx  
Dieser Port kann LLDP-Telegramme nur empfangen.
  - Tx  
Dieser Port kann LLDP-Telegramme nur senden.
  - Rx & Tx  
Dieser Port kann LLDP-Telegramme empfangen und senden.
  - "-" (Deaktiviert)  
Dieser Port kann LLDP-Telegramme weder empfangen noch senden.

## Unicast

### Filter

#### Adressfilterung

Diese Seite zeigt den aktuellen Inhalt der Unicast-Filtertabelle. In dieser Tabelle sind die Quelladressen von Unicast-Adresstelegrammen aufgeführt. Einträge können entweder dynamisch erfolgen, wenn ein Teilnehmer ein Telegramm an einen Port sendet oder statisch durch Parametrierung seitens des Anwenders.

Auf dieser Seite definieren Sie auch die statischen Unicast-Filter.

#### Einstellungen

- **VLAN-ID**  
Wählen Sie die VLAN-ID aus, in dem Sie eine neue MAC-Adresse statisch konfigurieren. Wenn nichts vorgegeben wird, ist "VLAN1" als Grundeinstellung parametrierung.

- **MAC-Adresse**  
Tragen Sie hier die MAC-Adresse ein.

Die Tabelle enthält folgende Spalten:

- **1. Spalte**  
Aktivieren Sie in der zu löschenden Zeile das Optionskästchen.
- **VLAN-ID**  
Zeigt die VLAN-ID, die dieser MAC-Adresse zugeordnet ist.
- **MAC-Adresse**  
Zeigt die MAC-Adresse des Teilnehmers, die das Gerät gelernt hat oder die der Anwender projiziert hat.

- **Status**  
Zeigt den Status jedes Adresseintrags:
  - Elernt  
Die angegebene Adresse wurde durch Empfang eines Telegramms dieses Teilnehmers gelernt und werden nach Ablauf der Aging Time wieder gelöscht, sollten keine weiteren Pakete dieses Teilnehmers empfangen werden.
  - Statisch  
Vom Anwender projiziert. Statische Adressen sind permanent gespeichert, d.h. sie werden nach Ablauf der Aging Time oder beim Neustart des Switchs nicht gelöscht.
  - Ungültig  
Diese Werte werden nicht ausgewertet.
- **Port**  
Zeigt an, über welchen Port der Teilnehmer mit der angegebenen Adresse erreichbar ist. Vom Gerät empfangene Telegramme, deren Zieladresse mit dieser Adresse übereinstimmt, werden an diesen Port weitergegeben.

---

**Hinweis**

Für Unicast-Adressen können Sie nur **einen** Port angeben.

---

## Gesperrte Ports

### Aktivierung der Zugangskontrolle

Auf dieser Seite können Sie einzelne Ports für unbekannte Teilnehmer sperren.

Wenn die Port Lock-Funktion aktiviert ist, werden Pakete an diesem Port, die von unbekanntem MAC-Adressen kommen, sofort verworfen. Die Pakete von bekannten Teilnehmern werden vom Port angenommen.

Da Ports mit aktivierter Port Lock-Funktion auch keine MAC-Adressen lernen, werden gelernte Adressen auf diesen Ports nach Aktivieren der Port Lock-Funktion automatisch ausgetragen.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie aus der Klappliste die Einstellung. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Aktiviert die Port Lock-Funktion.
  - Deaktiviert  
Deaktiviert die Port Lock-Funktion.
  - Keine Änderung  
Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports an.
- **Einstellung**  
Aktivieren oder deaktivieren Sie die Zugriffsteuerung für den Port.

## Blockierung

### Sperrung der Weiterleitung von unbekanntem Unicast-Telegrammen

Auf der Seite wird das Weiterleiten von unbekanntem Unicast-Telegrammen für einzelne Ports gesperrt.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Blocken von Unicast-Telegrammen ist aktiviert.
  - Deaktiviert  
Blocken von Unicast-Telegrammen ist deaktiviert.
  - Keine Änderung  
Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports an.
- **Einstellung**  
Aktivieren oder deaktivieren Sie das Blockieren von Unicast-Telegrammen.

---

### Hinweis

#### Ring-Redundanz / Standby

Wenn Ring Redundanz oder Standby aktiviert sind, werden die hierfür konfigurierten Ports vom Blockieren ausgenommen.

---

## Multicast

## Gruppen

## Multicast-Anwendungen

In der Mehrzahl der Fälle wird ein Telegramm mit einer Unicast-Adresse an einen bestimmten Empfänger gesendet. Wenn eine Anwendung die gleichen Daten an mehrere Empfänger senden soll, kann das zu sendende Datenvolumen reduziert werden, indem die Daten über eine Multicast-Adresse an alle gesendet werden. Für manche Anwendungen gibt es feste Multicast-Adressen (NTP, IETF1-Audio, IETF1-Video usw.).

## Reduzierung der Netzlast

Im Gegensatz zu Unicast-Telegrammen bewirken Multicast-Telegramme eine höhere Last für das Gerät. Denn generell werden Multicast-Telegramme an allen Ports versendet. Es gibt drei Möglichkeiten, die Last durch Multicast-Telegramme zu reduzieren:

- Statischer Eintrag der Adressen in die Multicast-Filtertabelle.
- Dynamischer Eintrag der Adressen durch Mithören von IGMP-Parametriertelegrammen (IGMP-Konfiguration).
- Aktive dynamische Vergabe von Adressen durch GMRP-Telegramme.

Alle genannten Verfahren haben zur Folge, dass Multicast-Telegramme nur an solche Ports versendet werden, für die eine entsprechende Adresse eingetragen ist.

Unter "Multicast " werden die aktuell in der Filtertabelle eingetragenen Multicast-Telegramme mit ihren Zielports angezeigt. Die Einträge können dynamisch (das Gerät hat sie gelernt) oder statisch (der Anwender hat sie parametrier) erfolgt sein.

## Einstellungen

- **VLAN-ID**  
Wählen Sie die VLAN-ID, die der MAC-Multicast-Adresse zugeordnet wird.
- **MAC-Adresse**  
Geben Sie eine neu zu projektierende MAC-Multicast-Adresse ein.

Die Tabelle gliedert sich in folgende Spalten:

- **VLAN-ID**  
Zeigt VLAN-ID des VLANs an, dem die MAC-Multicast-Adresse zugeordnet ist.
- **MAC-Adresse**  
Zeigt MAC-Multicast-Adresse an, die das Gerät gelernt hat oder die der Anwender projektiert hat.
- **Status**  
Zeigt den Status jedes Adress-Eintrags. Dabei sind folgende Angaben möglich:
  - **Statisch**  
Die Adresse wurde vom Anwender statisch eingetragen. Statische Adressen sind permanent gespeichert, d.h. sie werden nicht nach Ablauf der Aging Time oder beim Neustart des Gerätes gelöscht. Sie müssen vom Anwender gelöscht werden.
  - **IGMP**  
Der Zielport für diese Adresse wurde über IGMP-Konfiguration ermittelt.
  - **GMRP**  
Der Zielport für diese Adresse wurde über ein empfangenes GMRP-Telegramm registriert.



## IGMP

### Festlegung der IGMP Snooping Aging Time

Auf dieser Seite können Sie die Aging-Time für die IGMP-Konfiguration festlegen. Nach Ablauf dieser Zeit werden durch IGMP erzeugte Einträge aus der Adresstabelle gelöscht, wenn diese nicht durch ein neues IGMP-Telegramm aktualisiert werden.

Die Festlegung gilt für alle Ports, eine portspezifische Konfiguration ist nicht möglich.

IE-Switches unterstützen neben "IGMP Snooping" auch die IGMP Querier-Funktion. Ist "IGMP Snooping" aktiviert, so werden IGMP-Telegramme ausgewertet und mit diesen Informationen die Multicast-Filtertabelle aktualisiert. Ist zusätzlich IGMP Query aktiviert, so versenden IE-Switches auch IGMP-Anfragen, die bei IGMP-fähigen Teilnehmern Antworten auslösen.

### Einstellungen

- **IGMP Snooping**  
Aktivieren oder deaktivieren Sie IGMP (Internet Group Management Protocol). Die Funktion ermöglicht die Zuordnung von IP-Adressen zu Multicast-Gruppen. Wenn die Option aktiviert ist, werden IGMP-Einträge in die Tabelle aufgenommen und IGMP-Telegramme weitergeleitet.
- **IGMP Snooping Aging Time**  
Tragen Sie in dieses Feld den Wert für die Aging Time in Sekunden ein.
- **IGMP Querier**  
Aktivieren oder deaktivieren Sie "IGMP Querier". Das Gerät verschickt IGMP-Anfragen.

## GMRP

### Aktivierung von GMRP

Durch Aktivieren der entsprechenden Optionen legen Sie individuell für jeden Port fest, ob GMRP angewendet wird. Wenn für einen Port "GMRP" deaktiviert ist, werden für ihn keine Registrierungen durchgeführt und er kann keine GMRP-Telegramme versenden.

### Einstellungen

- **GMRP**  
Aktivieren oder deaktivieren Sie die GMRP-Funktion.

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Aktiviert das Versenden von GRMP-Telegrammen.
  - Deaktiviert  
Deaktiviert das Versenden von GRMP-Telegrammen.
  - Keine Änderung  
Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports und die Link-Aggregationen an.
- **Einstellung**
- Aktivieren oder deaktivieren Sie GMRP für den Port oder für die Link-Aggregation.

## Blockierung

Auf der Seite wird das Weiterleiten von unbekanntem Multicast-Telegrammen für einzelne Ports gesperrt.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Blocken von Multicast-Telegrammen ist aktiviert.
  - Deaktiviert  
Blocken von Multicast-Telegrammen ist deaktiviert.
  - Keine Änderung  
Tabelle 2 bleibt unverändert
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports an.
- **Einstellung**  
Aktivieren oder deaktivieren Sie das Blockieren von Multicast-Telegrammen.

## Broadcast

### Sperrung der Weiterleitung von Broadcast-Telegrammen

Auf dieser Seite kann das Weiterleiten von Broadcast-Telegrammen für einzelne Ports gesperrt werden.

---

#### Hinweis

Einige Kommunikationsprotokolle funktionieren nur mit Unterstützung von Broadcast. In diesen Fällen kann das Sperren zum Ausfall der Datenkommunikation führen. Sperren Sie Broadcast nur, wenn Sie sicher sind, dass Sie darauf verzichten können.

---

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Das Blocken von Broadcast-Telegrammen ist aktiviert.
  - Deaktiviert  
Das Blocken von Broadcast-Telegrammen ist deaktiviert.
  - Keine Änderung  
Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, wird die Einstellung für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Alle verfügbaren Ports und die Link-Aggregationen werden angezeigt.
- **Einstellung**  
Aktivieren oder deaktivieren Sie das Blockieren von Broadcast-Telegrammen.

## PTP

### Allgemein

Das Precision Time Protocol (PTP) nach IEEE 1588v2 ermöglicht die Uhrzeitsynchronisation von Geräten, die an den Ports eines Geräts angeschlossen sind. Diese Geräte leiten die Synchronisationstelegramme nach dem Verfahren "Transparent Clock" (TC) durch das Netz. Es werden die Korrekturmechanismen "End-to-End" und "Peer-to-Peer" unterstützt.

Um IEEE 1588v2 zu nutzen, müssen Sie diese Funktion aktivieren und jeden Port, der im Synchronisationspfad liegt, sowie Ports, die durch Redundanzmechanismen geblockt sind, konfigurieren. IEEE 1588v2 ist auch nutzbar bei Redundanzmechanismen im Ring wie HRP, Standby-Kopplung von Ringen, MRP und RSTP.

### Einstellung

- **Betriebsart 1588**  
Es gibt die folgenden Einstellmöglichkeiten:
  - **aus**  
Das Gerät verarbeitet keine PTP-Nachrichten. PTP-Nachrichten werden aber nach den Regeln des Geräts weitergeleitet.
  - **transparent**  
Das Gerät übernimmt die Funktion einer Transparent Clock und leitet PTP-Nachrichten an andere Teilnehmer weiter, wobei es Eintragungen in das Korrekturfeld der PTP-Nachricht vornimmt.

### TC Allgemein

#### 1588 Transparent Clock

Auf dieser Seite legen Sie die allgemeinen Einstellungen zu PTP fest.

### Einstellungen

- **Delay Mechanismus**  
Legen Sie fest, mit welchem Delay-Mechanismus das Gerät arbeiten soll:
  - End-to-End (Delay-Request-Response-Mechanismus wird verwendet)
  - Peer-to-Peer (Peer-Delay-Mechanismus wird verwendet)
- **Domänen-Nummer**  
Tragen Sie die Identifikations-Nummer für die Uhrzeitdomäne ein. Die Synchronisation erfolgt nur für die Geräte innerhalb der Domäne. Das Gerät ignoriert PTP-Nachrichten mit einer abweichenden Domänen-Nummer. Ein SCALANCE-Gerät kann nur einer Synchronisations-Domäne zugeordnet sein.

### TC-Port

Auf dieser Seite legen Sie die Ports fest, die PTP-Nachrichten verarbeiten können.

## Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung / Transport-Mechanismus**  
Wählen Sie die Einstellung aus. Wenn "Keine Änderung" ausgewählt ist, bleibt der Eintrag in der Tabelle 2 unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt die verfügbaren Ports an.
- **Einstellung**  
Aktivieren oder deaktivieren Sie PTP. Wenn aktiviert, verarbeitet der Port PTP-Nachrichten.
- **Fehlerflag**  
Zeigt den Fehlerstatus im Bezug auf PTP.
  - **wahr**  
Es ist ein Fehler aufgetreten.
  - **falsch**  
An diesem Port sind keine Fehler aufgetreten.
- **Transport-Mechanismus**  
Legen Sie das Protokoll für die Übertragung der PTP-Nachrichten fest. Dieses Protokoll muss vom jeweiligen Kommunikationspartner des Ports unterstützt werden
  - Ethernet
  - UDP IPv4

## Layer 3

### Konfiguration

Die Seite enthält die Übersicht über die Layer 3-Funktionen des Geräts. Um die Funktionen "Routing", "VRRP" und "OSPF" zu nutzen, benötigt das Geräte einen KEY-PLUG.

### Einstellungen

- **Routing** (nur bei Layer 3 verfügbar)  
Aktivieren oder deaktivieren Sie die Funktion Routing.
- **DHCP Relay Agent**  
Aktivieren oder deaktivieren Sie den DHCP Relay Agenten. Weitere Einstellungen konfigurieren Sie unter "Layer 3 > DHCP Relay Agent".

- **VRRP** (nur bei Layer 3 verfügbar)  
Aktivieren oder deaktivieren Sie Routing über VRRP. Um VRRP zu verwenden, aktivieren Sie zuerst die Funktion "Routing". Weitere Einstellungen konfigurieren Sie unter "Layer 3 > VRRP".
- **OSPF** (nur bei Layer 3 verfügbar)  
Aktivieren oder deaktivieren Sie Routing über OSPF. Um OSPF zu verwenden, aktivieren Sie zuerst die Funktion "Routing". Weitere Einstellungen konfigurieren Sie unter "Layer 3 > OSPF".

## Subnetze

### Übersicht

#### Subnetz

Die Seite zeigt die Subnetze für die ausgewählte Schnittstelle. Wenn an einer Schnittstelle mehrer Subnetze verfügbar sind, ist der erste Eintrag dieser Schnittstelle vom Adresstyp "Primär". Alle weiteren Subnetze sind vom Adresstyp "Sekundär". Alle weiteren Subnetze werden auf dieser Seite angelegt und unter "Layer 3 > Subnetze > Konfiguration" konfigurieren Sie die Einstellungen.

#### Einstellungen

Für Subnetze vom Adresstyp "Sekundär" sind folgende Einstellungen möglich:

- **Schnittstelle**  
Wählen Sie die gewünschte Schnittstelle aus, an die Sie ein weiteres Subnetz projektieren.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**  
Zeigt die Schnittstelle an.
- **TIA-Schnittstelle**  
Zeigt an, ob der Eintrag eine TIA-Schnittstelle ist.
- **Schnittstellen-Name**  
Zeigt den Namen der Schnittstelle.
- **IP-Adresse**  
Zeigt die IP-Adresse des Subnetzes an.
- **Subnetzmaske**  
Zeigt die Subnetzmaske an.
- **Adresstyp**  
Zeigt den Adresstyp an. Folgende Werte sind möglich:
  - Primär  
Das erste Subnetz der Schnittstelle.
  - Sekundär  
Alle weiteren Subnetze der Schnittstelle.

- **Methode der IP-Adressezuweisung**  
Zeigt an, wie die IP-Adresse zugeordnet wird.
  - Statisch  
Die IP-Adresse ist statisch. Die IP-Einstellungen tragen Sie bei "IP-Adresse" und "Subnetzmaske" ein.
  - Dynamisch (DHCP)  
Das Gerät bezieht eine dynamische IP-Adresse von einem DHCP-Server.

## Konfiguration

Auf dieser Seite konfigurieren Sie das Subnetz. Das Subnetz legen Sie unter "Layer 3 > Subnetze > Übersicht" an.

## Einstellungen

- **Schnittstelle (Name)**  
Wählen Sie die gewünschte Schnittstelle aus.
- **Schnittstellen-Name**  
Geben Sie den Bezeichner für die Schnittstelle ein, z. B. eth0; P3.
- **DHCP**  
Aktivieren oder deaktivieren Sie den DHCP-Client für diese Schnittstelle.
- **IP-Adresse**  
Geben Sie die IP-Adresse des Subnetzes ein. Die IP-Adresse darf nicht mehrfach verwendet werden.
- **Subnetzmaske**  
Geben Sie die Subnetzmaske des zu erstellenden Subnetzes ein. Subnetze an unterschiedlichen Schnittstellen dürfen sich nicht überlappen.
- **Adresstyp**  
Zeigt den Adresstyp an. Folgende Werte sind möglich:
  - Primär  
Das erste Subnetz der Schnittstelle.
  - Sekundär  
Alle weiteren Subnetze der Schnittstelle.
- **TIA-Schnittstelle**  
Aktivieren oder deaktivieren Sie die Einstellung.

## Routen

### Statische Route

Auf dieser Seite legen Sie statische Routen an. Statische Routen werden bei automatischen Anpassungen nicht berücksichtigt und müssen von Hand angepasst werden.

## Einstellungen

- **Zielnetzwerk**  
Tragen Sie die Netzwerkadresse des Ziels ein.
- **Subnetzmaske**  
Tragen Sie die dazugehörige Subnetzmaske ein.
- **Gateway**  
Tragen Sie die IP-Adresse des nächsten Gateways ein.

Die Tabelle gliedert sich in folgende Spalten:

- **Zielnetzwerk**  
Zeigt die Netzwerkadresse des Ziels an.
- **Subnetzmaske**  
Zeigt die dazugehörige Subnetzmaske an.
- **Gateway**  
Zeigt die IP-Adresse des nächsten Gateways an.
- **Metrik**  
Tragen Sie die Metrik für die Route ein. Beim Erstellen der Route wird automatisch "not used" eingetragen. Die Metrik entspricht der Güte einer Verbindung, z. B. Geschwindigkeit, Kosten. Bei mehreren gleichen Routen wird die Route mit dem kleinsten Metrik-Wert benutzt.

## DHCP Relay Agent

### Allgemein

Wenn der DHCP-Server in einem anderen Netz befindet, kann das Gerät den DHCP-Server nicht erreichen. Der DHCP Relay Agent vermittelt zwischen einem DHCP-Server und dem Gerät. Dazu gibt der DHCP Relay Agent die Portnummer des Geräts zusammen mit der DHCP-Anfrage an den DHCP-Server weiter. Wenn ein DHCP-Server nicht erreichbar ist, kann das Gerät auf einen anderen DHCP-Server ausweichen.

## Einstellungen

- **DHCP Relay-Agent (Opt. 82)**  
Aktivieren oder deaktivieren Sie den DHCP Relay Agent.
- **Server-IP-Adresse**  
Tragen Sie die IP-Adresse des DHCP-Servers ein.

Die Tabelle gliedert sich in folgende Spalten:



- **Server-IP-Adresse**  
Zeigt die IP-Adresse des DHCP-Servers an.

## Option

### Parameter des DHCP Relay Agent

Auf dieser Seite können Sie Parameter für den DHCP-Server festlegen, z. B. die Circuit ID. Die Circuit ID beschreibt die Herkunft der DHCP-Anfrage, z. B. welcher Port die DHCP-Anfrage empfangen hat.

Die DHCP-Server legen Sie auf dem Register "General" fest.

## Einstellungen

- **Circuit ID Router-Index**  
Wenn aktiviert, wird der erzeugten Circuit ID der Router Index hinzugefügt.
- **Circuit ID Empfänger-VLAN-ID**  
Wenn aktiviert, wird der erzeugten Circuit ID die VLAN ID hinzugefügt.
- **Circuit ID Empfänger-Port**  
Wenn aktiviert, wird der erzeugten Circuit ID der Empfangsport hinzugefügt.

---

### Hinweis

Sie müssen mindestens eine Option auswählen.

---

- **Schnittstelle**  
Wählen Sie die gewünschte Schnittstelle aus.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**  
Zeigt die Schnittstelle an.
- **Remote-ID-Typ**  
Wählen Sie die Art der Geräteerkennung aus. Sie haben folgende Möglichkeiten:
  - IP-Adresse  
Als Geräteerkennung wird die IP-Adresse des Geräts verwendet.
  - MAC-Adresse  
Als Geräteerkennung wird die MAC-Adresse des Geräts verwendet.
  - Benutzerdefinierter Text  
Wenn Sie "Benutzerdefinierter Text" verwenden, können Sie bei "Remote-ID" den Gerätenamen als Geräteerkennung eintragen.
- **Remote-ID**  
Tragen Sie den Gerätenamen ein. Das Feld ist nur editierbar, wenn Sie bei "Remote-ID-Typ" den Eintrag "Benutzerdefinierter Text" auswählen.

- **Circuit ID-Type**  
Wählen Sie aus der Klappliste die Art der Circuit ID aus. Sie haben folgende Möglichkeiten:
  - Voreinstellung  
Die Circuit ID wird automatisch erstellt, basierend auf Router Index, VLAN ID oder Port.
  - Benutzerdefinierte Nummer  
Wenn Sie "Benutzerdefinierte Nummer" verwenden, können Sie bei "Circuit ID" die ID eingeben.
- **Circuit ID**  
Tragen Sie die Circuit ID ein. Das Feld ist nur editierbar, wenn Sie bei "Circuit ID-Type" den Eintrag "Benutzerdefinierte Nummer" auswählen.

## VRRP

### Router

Auf dieser Seite legen Sie virtuelle Router an. Weitere Parameter konfigurieren Sie unter "Layer 3 > VRRP > Konfiguration".

---

#### Hinweis

- VRRP ist nur bei Layer 3 verfügbar.
  - VRRP ist nur im Zusammenhang mit VLAN-Interfaces nutzbar. Router-Ports werden nicht unterstützt.
  - Aktivieren Sie "VRRP", um VRRP zu konfigurieren
- 

## Einstellungen

Folgende Einstellungen gibt es:

- **VRRP**  
Aktivieren oder deaktivieren Sie das Routing über VRRP.
- **Rückmeldung auf Ping-Anfragen virtueller Schnittstellen.**  
Wenn aktiviert, antworten auch die virtuellen IP-Adressen auf den Ping.
- **Schnittstelle**  
Wählen Sie die Schnittstelle, die als virtueller Router fungiert.
- **VRID**  
Tragen Sie in das Eingabefeld die ID des virtuellen Routers ein. Diese ID definiert die Gruppe der Router, die einen virtuellen Router (VR) bilden. In der Gruppe ist diese gleich. Für andere Gruppen kann sie nicht mehr verwendet werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**  
Zeigt die Schnittstelle an, die als virtueller Router fungiert.
- **VRID**  
Zeigt die ID des virtuellen Routers an.

- **Primäre IP-Adresse**  
Zeigt die vorrangige IP-Adresse auf diesem VLAN an. Der Eintrag 0.0.0.0 bedeutet, dass die "Primary" Adresse auf diesem VLAN verwendet. Ansonsten sind alle IP-Adressen, die auf diesem VLAN unter "Layer 3 > Subnetz" konfiguriert wurden, gültige Werte.
- **Priorität**  
Zeigt die Priorität des virtuellen Routers an.  
Die 255 erhält automatisch der aktuelle Master-Router. Alle anderen Prioritäten können frei auf die VRRP-Router verteilt werden. Je größer eine Priorität ist, desto eher wird der VRRP-Router zum "Master".
- **Advert. Intern**  
Zeigt das Intervall an, in dem der Master-Router VRRP-Pakete verschickt.
- **Unterbrechen**  
Zeigt die Bevorrechtigung eines Routers beim Rollenwechsel zwischen Backup und Master an.
  - ja  
Dieser Router ist beim Rollenwechsel bevorrechtigt.
  - nein  
Dieser Router ist beim Rollenwechsel nicht bevorrechtigt.

## Konfiguration

### Einleitung

Auf dieser Seite konfigurieren Sie den virtuellen Router.

---

#### Hinweis

VRRP ist nur bei Layer 3 verfügbar.

---

### Beschreibung der angezeigten Werte

Die Seite enthält folgende Felder:

- **Schnittstelle / VRID**  
Wählen Sie die ID des zu konfigurierenden virtuellen Routers aus.
- **Status**  
Aktivieren oder deaktivieren Sie die Funktion "Status".

- **Primäre IP-Adresse**  
Wählen Sie die vorrangige IP-Adresse aus. Wenn der Router zum Master-Router wird, verwendet der Router diese IP-Adresse.

---

**Hinweis**

Wenn Sie auf diesem VLAN nur ein Subnetz konfigurieren, ist keine Angabe erforderlich. Der Eintrag ist 0.0.0.0.

Wenn Sie mehrere Subnetze am VLAN konfigurieren und Sie möchten, dass eine bestimmte IP-Adresse als Quelladresse für VRRP-Pakete genutzt wird, wählen Sie aus der Klappliste die entsprechende IP-Adresse. Ansonsten wird die vorrangige IP-Adresse verwendet.

---

- **Master**  
Wenn diese Option aktiviert ist, wird die vorrangige IP-Adresse bei "Zugeordnete IP-Adresse" eingetragen. Damit wird die vorrangige IP-Adresse des VRRP-Routers als virtuelle IP-Adresse des virtuellen Master-Routers verwendet. Die Backup-Router in dieser Gruppe, müssen die Option deaktivieren und die IP-Adresse des Routers bei "Zugeordnete IP-Adresse" verwenden.
- **Priorität**  
Tragen Sie die Priorität dieses virtuellen Routers ein.  
Die 255 erhält immer der aktuelle Master-Router. Alle anderen Prioritäten können frei auf die redundanten Router verteilt werden. Je größer eine Priorität ist, desto eher wird der Router zum "Master".
- **Advertisement-Intervall**  
Tragen Sie das Zeitintervall in Sekunden ein, nach dem ein Master-Router erneut ein VRRP-Paket verschickt.
- **Router mit niedrigerer Priorität unterbrechen**  
Erlauben sie die Bevorrechtigung zum Rollenwechsel zwischen Backup und Master basierend auf dem Auswahlprozess.

## Adressübersicht

### Übersicht

Diese Seite zeigt an, welche IP-Adressen der virtuelle Router überwacht.

---

**Hinweis**

VRRP ist nur bei Layer 3 verfügbar.

---

## Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Schnittstelle**  
Zeigt die Schnittstelle an, die als virtueller Router fungiert.
- **VRID**  
Zeigt die ID dieses virtuellen Routers an.
- **Anzahl der Adressen**  
Zeigt die Anzahl der IP-Adressen an.
- **Zugeordnete IP-Adresse (1) ... Zugeordnete IP-Adresse (4)**  
Zeigt die Router IP-Adressen, die durch diesen virtuellen Router überwacht werden. Wenn ein Router die Rolle des Masters übernimmt, wird die Routingfunktion all dieser IP-Adressen durch diesen Router übernommen.

## Adresskonfiguration

---

### Hinweis

VRRP ist nur bei Layer 3 verfügbar.

---

## Einstellungen

Folgende Einstellungen sind möglich:

- **Schnittstelle / VRID**  
Wählen Sie den gewünschten virtuellen Router aus.
- **Zugeordnete IP-Adresse**  
Geben Sie die IP-Adresse ein, die der virtuelle Router überwachen soll.

Die Tabelle gliedert sich in folgende Spalten:

- **Zugeordnete IP-Adresse**  
Zeigt die IP-Adressen an, die der virtuelle Router überwacht.

## OSPFv2

## Konfiguration

## Einleitung

Auf dieser Seite konfigurieren Sie das Routing über OSPF.

---

### Hinweis

OSPF ist nur bei Layer 3 verfügbar.

---

## Einstellungen

- **OSPFv2**  
Aktivieren oder deaktivieren Sie das Routing über OSPF.
- **Router-ID**  
Tragen Sie eine Bezeichnung einer der OSPF-Schnittstellen ein. Die Bezeichnung wird im IP-Adressformat eingegeben und muss nicht mit der eigenen IP-Adresse übereinstimmen.
- **Maximale Anzahl externer LSAs**  
Um die Einträge von externen LSAs in der Datenbank zu begrenzen, tragen Sie die maximale Anzahl der External LSAs ein.
- **Exit-Intervall [s]**  
Tragen Sie das Intervall ein, nach dem der OSPF-Router wieder versucht aus dem Overflow-Status herauszukommen. Eine 0 bedeutet, dass der OSPF-Router erst nach einem Hochlauf wieder versucht den Overflow Status zu verlassen.
- **Weiterverteilung von Routen**  
Legen Sie fest, welche bekannten Routen über OSPF weitergegeben werden. Folgende Routenarten gibt es:
  - Voreinstellung
  - Verbunden
  - Statisch

---

### Hinweis

Die Einstellungen lassen sich nur an einem AS Border Router aktivieren. Besonders die Aktivierung von "Voreinstellung" und "Statisch" kann zu Problemen führen, wenn sie an zu vielen Stellen im Netz aktiviert werden, z. B. Forwarding Loops.

---

- **OSPFv2 - RFC1583-Kompatibilität**  
Aktivieren Sie die Option, wenn sie noch alte OSPF-Router in Betrieb haben, die nicht zu RFC 2328 kompatibel sind.
- **AS Border-Router**  
Legen Sie fest, ob der Router ein AS Border Router ist. Ein AS Border Router vermittelt zwischen mehreren autonomen Systemen, z. B. wenn Sie ein zusätzliches RIP Netzwerk haben. Ein AS Border Router ist auch nötig, um statische Routen hinzuzufügen und zu verteilen.

## Areas

### Übersicht

Ein autonomes Netzwerk (Autonomous System) kann in kleinere Bereiche (Areas) unterteilt werden.

Auf dieser Seite können Sie die Bereiche des Routers ansehen, anlegen, ändern oder löschen.

---

**Hinweis**

OSPF ist nur bei Layer 3 verfügbar.

---

## Einstellungen

- **Area ID**  
Geben Sie die ID des Bereichs ein. Für alle Router eines Bereichs wird die Datenbank synchronisiert.  
Eingabeformat: x.x.x.x  
x = 0 ... 255

Die Tabelle enthält folgende Spalten:

- **Area ID**  
Zeigt die ID des Bereichs an.
- **Area Typ**  
Wählen Sie in der Klappliste den Bereichstyp aus.
  - Normal
  - Stub
  - NSSA
  - Backbone
- **Summary**  
Legen Sie fest, ob für diesen Bereich Summary LSAs erzeugt werden.
  - Summary: Summary LSAs werden in den Bereich verschickt.
  - No Summary: Summary LSAs werden nicht in den Bereich verschickt.
- **Metrik**  
Zeigt die Kosten für die OSPF-Schnittstelle an.

## Area Range

### Anlegen einer neuen OSPFv2 Bereichsumfanges.

Auf dieser Seite können Netzwerke unter einer Area ID zusammengefasst werden. Das Verfahren wird nur bei Area Border Routern angewandt. Damit propagiert ein Area Border Router nach aussen lediglich eine Route für jeden Adressbereich.

---

**Hinweis**

OSPF ist nur bei Layer 3 verfügbar.

---

## Einstellungen

- **Area ID**  
Wählen Sie die Area ID aus. Die ID konfigurieren Sie unter "Layer 3 > OSPF > Areas".
- **Subnetz-Adresse**  
Tragen Sie die Adresse des Netzwerks ein, das zusammengefasst wird.
- **Subnetzmaske**  
Tragen Sie die Subnetzmaske des Netzwerks ein, das zusammengefasst wird.

Die Tabelle enthält folgende Spalten:

- **Area ID**  
Zeigt die ID des Bereichs an.
- **Subnetz-Adresse**  
Zeigt die Adresse des Netzwerks an, das mit anderen Netzwerken zusammengefasst wird.
- **Subnetzmaske**  
Zeigt die Subnetzmaske des Netzwerks an, das mit anderen Netzwerken zusammengefasst wird.
- **veröffentlichen**  
Aktivieren Sie diese Option, um das zusammengefasste Netzwerk zu veröffentlichen.

## Schnittstellen

## Übersicht

Auf dieser Seite können Sie OSPF-Schnittstellen konfigurieren.

---

### Hinweis

OSPF ist nur bei Layer 3 verfügbar.

---

## Einstellungen

- **IP-Adresse**  
Wählen Sie die IP-Adresse der OSPF-Schnittstelle aus.

Die Tabelle enthält folgende Spalten:

- **IP-Adresse**  
Zeigt die IP-Adresse der OSPF-Schnittstelle an.
- **Area ID**  
Wählen Sie die Area ID, mit der die OSPF-Schnittstelle verbunden ist.
- **OSPF-Status**  
Legen Sie fest, ob OSPF an der Schnittstelle aktiv ist.
  - Aktiviert: OSPF ist an der Schnittstelle aktiviert.
  - Deaktiviert: OSPF ist an der Schnittstelle deaktiviert



- **Metrik**  
Tragen Sie die Kosten für die OSPF-Schnittstelle ein.
- **Priorität**  
Tragen Sie die Router-Priorität ein. Die Priorität spielt nur bei der Auswahl des designierten Router eine Rolle. Dieser Parameter kann auf Routern des gleichen Subnetzes unterschiedlich gewählt werden.
- **Trans. Verzög.**  
Tragen Sie die gewünschte Verzögerung beim Versenden einer Verbindungsaktualisierung ein.
- **Retrans. Verzög.**  
Tragen Sie die Zeit ein, nach der ein OSPF-Paket erneut übertragen wird, wenn keine Bestätigung empfangen wurde.
- **Hello-Intervall**  
Tragen Sie den Abstand zwischen zwei Hello-Paketen ein.
- **Dead-Intervall**  
Tragen Sie das Intervall ein, nach dem der Nachbar-Router als "ausgefallen" markiert wird, wenn in dieser Zeit keine Hello-Pakete mehr von ihm empfangen wurden.

## Schnittstellen-Authentifizierung

### Konfigurieren der Schnittstellen Anmeldung

Auf dieser Seite definieren Sie die Authentifizierung der Schnittstelle.

---

#### Hinweis

OSPF nur bei Layer 3 verfügbar.

---

## Einstellungen

- **OSPF-Schnittstelle**  
Wählen Sie die OSPF-Schnittstelle aus, für die Sie die Authentifizierung konfigurieren wollen.
- **Authentifizierungs-Typ**  
Wählen Sie die Authentifizierungsmethode der OSPF-Schnittstelle aus. Sie haben folgende Möglichkeiten:
  - Keine: Keine Authentifizierung
  - Einfach: Authentifizierung über ein unverschlüsseltes Passwort
  - MD5: Authentifizierung über MD5
- **Kennwort**  
Tragen Sie für die "einfache Authentifizierung" das Passwort ein.

- **Bestätigung**  
Bestätigen Sie das eingegebene Passwort.
- **Authentifizierungsschlüssel-ID**  
Tragen Sie für die MD5-Authentifizierung die ID ein, unter der das Passwort als Schlüssel verwendet wird. Da die Schlüssel-ID mit dem Protokoll übertragen wird, muss bei allen benachbarten Routern der gleiche Schlüssel unter der gleichen Schlüssel-ID gespeichert werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Authentifizierungsschlüssel-ID**  
Nur editierbar, wenn Sie die Authentifizierungsmethode MD5 einstellen. Nur dort können mehrere Schlüssel verwendet werden.
- **MD5-Schlüssel**  
Tragen Sie den MD5-Schlüssel ein.
- **MD5-Schlüsselbestätigung**  
Bestätigen Sie den eingegebenen Schlüssel.
- **Neueste Schlüssel-ID**  
Gibt an, ob der MD5-Schlüssel die neueste Schlüssel-ID ist.

## Virtual Links

### Übersicht

Jeder Area Border Router muss aus protokolltechnischen Gründen Zugang zum Backbone-Bereich erhalten. Wenn ein Router nicht direkt am Backbone-Bereich angeschlossen ist, wird eine virtuelle Verbindung dorthin angelegt.

---

#### Hinweis

OSPF ist nur bei Layer 3 verfügbar.

---

#### Hinweis

Beachten Sie, dass beim Anlegen einer virtuellen Verbindung sowohl der Transit-Bereich als auch der Backbone-Bereich bereits konfiguriert sein müssen.

Die virtuelle Verbindung muss auf beiden Seiten gleich konfiguriert werden.

---

### Einstellungen

- **Nachbar-Router-ID**  
Tragen Sie die ID des Nachbar-Routers auf der anderen Seite der virtuellen Verbindung ein.
- **Transit Area ID**  
Wählen Sie die ID des Bereichs, der die beiden Router verbindet.

Die Tabelle enthält folgende Spalten:

- **Transit Area ID**  
Zeigt die ID an, über die die beiden Router verbunden sind.
- **Nachbar-Router-ID**  
Zeigt die ID des Nachbar-Routers auf der anderen Seite der virtuellen Verbindung an.
- **Virt. Link Status**  
Legen Sie fest, in welchem Zustand sich die virtuelle Verbindung befindet. Folgende Zustände sind möglich:
  - down: Die virtuelle Verbindung ist inaktiv.
  - Punkt-zu-Punkt: Die virtuelle Verbindung ist aktiv.
- **Trans. Verzög.**  
Tragen Sie die gewünschte Verzögerung bei dem Versenden eines Link Update Paketes ein.
- **Retrans. Verzög.**  
Tragen Sie die Zeit ein, nach der ein Paket erneut übertragen wird, wenn keine Bestätigung empfangen wurde.
- **Hello-Intervall**  
Tragen Sie den Abstand zwischen zwei Hello-Paketen ein.
- **Dead-Intervall**  
Tragen Sie das Intervall ein, nach dem der Nachbar-Router als "ausgefallen" gilt, wenn in dieser Zeit keine Hello-Pakete mehr von ihm empfangen wurden.

## Virtual Links-Authentifizierung

### Konfigurieren der virtuellen Link Anmeldung

Auf dieser Seite definieren Sie die Authentifizierung für die virtuellen Links.

---

#### Hinweis

OSPF ist nur bei Layer 3 verfügbar.

---

## Einstellungen

- **Virtual Link (Area/Nachbar)**  
Wählen Sie den virtuellen Link aus, für die Sie die Authentifizierung konfigurieren wollen.
- **Authentifizierungs-Typ**  
Wählen Sie die Authentifizierungsmethode der OSPF-Schnittstelle aus. Sie haben folgende Möglichkeiten:
  - Keine: Keine Authentifizierung
  - Einfach: Authentifizierung über ein unverschlüsseltes Passwort.
  - MD5: Authentifizierung über MD5
- **Kennwort**  
Tragen Sie für die "einfache Authentifizierung" das Passwort ein.

- **Bestätigung**  
Bestätigen Sie das eingegebene Passwort.
- **Authentifizierungsschlüssel-ID**  
Tragen Sie für die MD5-Authentifizierung die ID ein, unter der das Passwort als Schlüssel verwendet wird. Da die Schlüssel-ID mit dem Protokoll übertragen wird, muss bei allen benachbarten Routern der gleiche Schlüssel unter der gleichen Schlüssel-ID gespeichert werden.

Die Tabelle gliedert sich in folgende Spalten:

- **Authentifizierungsschlüssel-ID**  
Nur editierbar, wenn Sie die Authentifizierungsmethode MD5 einstellen. Nur dort können mehrere Schlüssel verwendet werden.
- **MD5-Schlüssel**  
Tragen Sie den MD5-Schlüssel ein.
- **MD5-Schlüsselbestätigung**  
Bestätigen Sie den eingegebenen Schlüssel.
- **Neueste Schlüssel-ID**  
Gibt an, ob der MD5-Schlüssel die neueste Schlüssel-ID ist.

## Sicherheit

### AAA

### Allgemein

Auf dieser Seite konfigurieren Sie die Anmeldung.

### Einstellungen

- **Login-Authentifizierung**  
Legen Sie fest, wie die Anmeldung erfolgt:
  - Lokal  
Anmeldung mit lokalem Benutzernamen und Passwort.
  - Radius  
Anmeldung über RADIUS-Server.
- **802.1X - Reauthentifizierung**  
Wenn Sie dieses Optionskästchen aktivieren, wird ein authentifizierter 802.1X Supplicant zyklisch zu einer Reauthentifizierung gezwungen.

## Radius-Client

### Authentifizierung über einen externen Server

Das Konzept von RADIUS basiert auf einem externen Authentifizierungs-Server. Für ein Endgerät ist der Zugang zum Netzwerk erst möglich, nachdem das Gerät die Anmeldedaten beim Authentifizierungs-Server verifiziert hat. Sowohl das Endgerät als auch der Authentifizierungs-Server müssen das EAP-Protokoll (Extensive Authentication Protocol) unterstützen.

Jede Spalte der Tabelle enthält die Zugangsdaten für je einen Server. In der Suchreihenfolge wird der primäre Server zuerst angefragt. Ist der primäre Server nicht erreichbar, werden in der eingetragenen Reihenfolge sekundäre Server angefragt.

Wenn keiner der Server antwortet, findet keine Authentifizierung statt. Der Client erhält keinen Zugriff auf das Netzwerk, obwohl ein Link am Port angezeigt wird.

### Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Server-IP-Adresse**  
Tragen Sie die IP-Adresse des RADIUS-Servers ein.
- **Server-Port**  
Tragen Sie den Eingangs-Port auf dem RADIUS-Server ein. Standardmäßig ist der Eingangs-Port 1812 eingestellt.
- **Zugangskennung**
- Tragen Sie Ihre Zugangskennung ein.
- **Best. Zugangskennung**  
Tragen Sie die Zugangskennung zur Bestätigung erneut ein.
- **Erneute Verbindungen max.**  
Tragen Sie die maximale Anzahl der Anfrageversuche ein, bevor ein anderer konfigurierter RADIUS-Server angefragt wird oder die Anmeldung für gescheitert erklärt wird. Standardmäßig ist 3 eingestellt.
- **Primärer Server**  
Legen Sie fest, ob dieser Server der primäre Server ist. Sie können aus den Optionen "ja" oder "nein" auswählen.
- **Status**  
Aktivieren oder deaktivieren Sie den RADIUS-Servers.

---

#### **Hinweis**

Sie können auf dieser Seite maximal zwei Server konfigurieren.

---

## 802.1x-Authentifizierung

### Aktivierung der Authentifizierung für einzelne Ports

Durch Aktivieren der entsprechenden Optionen legen Sie individuell für jeden Port fest, ob der Netzzugriffsschutz nach IEEE 802.1x auf diesem Port aktiviert ist.

### Einstellungen

Die Tabelle 1 gliedert sich in folgende Spalten:

- **1. Spalte**  
Zeigt an, dass die Einstellungen für alle Ports der Tabelle 2 gültig sind.
- **Einstellung**  
Wählen Sie die Einstellung aus. Folgende Einstellungsmöglichkeiten haben Sie:
  - Aktiviert  
Zugriffsschutz ist aktiviert.
  - Deaktiviert  
Zugriffsschutz ist deaktiviert.
  - Keine Änderung  
Tabelle 2 bleibt unverändert.
- **In Tabelle übernehmen**  
Wenn Sie auf die Schaltfläche klicken, werden die Einstellungen für alle Ports der Tabelle 2 übernommen.

Die Tabelle 2 gliedert sich in folgende Spalten:

- **Port**  
Zeigt den Port an, auf den sich die Einstellung bezieht.
- **Einstellung**  
Aktivieren oder deaktivieren Sie die Authentifizierung für den gewünschten Port. Ports, für die diese Konfiguration nicht möglich ist, werden grau dargestellt und Sie können die Einstellungen nicht verändern.

### Port ACL MAC

### Regel-Konfiguration

Auf dieser Seite legen Sie die ACL-Regeln für das MAC-basierte ACL fest.

## Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Regelnummer**  
Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Quell-MAC-Adresse**  
Tragen Sie die Unicast MAC-Adresse der Quelle ein.  
**Ziel-MAC-Adresse**  
Tragen Sie die Unicast MAC-Adresse des Ziels ein.
- **Aktion**  
Wählen Sie das Verhalten. Folgendes ist möglich:
  - Weiterleiten  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- **Egress-Port**  
Zeigt eine Liste aller Egress-Ports, für die diese Regel gilt.
- **Ingress-Port**  
Zeigt eine Liste aller Ingress-Ports, für die diese Regel gilt.

---

### Hinweis

Wenn Sie bei Quell- und/oder Ziel-MAC-Adresse die Adresse "00:00:00:00:00:00" eingeben gilt die so erstellte Regel für alle Quell- bzw. Ziel-MAC-Adressen.

---

## Port Ingress-Regeln

Auf dieser Seite legen Sie fest, nach welcher ACL-Regel eingehende Telegramme vom Port behandelt werden.

## Einstellungen

- **Ports**  
Wählen Sie den gewünschten Port aus.
- **Regel hinzufügen**  
Wählen Sie die ACL-Regel, die dem Port zugewiesen wird. Die ACL-Regel legen Sie auf der Seite "Regel-Konfiguration" fest.
- **Hinzufügen**  
Um dem Port die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**  
Wählen Sie die zu löschende ACL-Regel.
- **Entfernen**  
Um die ACL-Regel vom Port zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Reihenfolge**  
Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**  
Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Quell-MAC-Adresse**  
Zeigt die Unicast MAC-Adresse der Quelle an.
- **Ziel-MAC-Adresse**  
Zeigt die Unicast MAC-Adresse des Ziels an.
- **Aktion**  
Wählen Sie das Verhalten. Folgendes ist möglich:
  - Weiterleiten  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.

## Port Egress-Regeln

Auf dieser Seite legen Sie fest, nach welcher ACL-Regel ausgehende Telegramme vom Port behandelt werden.

## Beschreibung

- **Ports**  
Wählen Sie den gewünschten Port aus.
- **Regel hinzufügen**  
Wählen Sie die ACL-Regel, die dem Port zugewiesen wird. Die ACL-Regel legen Sie auf der Seite "Regel-Konfiguration" fest.
- **Hinzufügen**  
Um dem Port die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**  
Wählen Sie die zu löschende ACL-Regel.
- **Entfernen**  
Um die ACL-Regel vom Port zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Reihenfolge**  
Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**  
Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.



- **Quell-MAC-Adresse**  
Zeigt die Unicast MAC-Adresse der Quelle an.
- **Ziel-MAC-Adresse**  
Zeigt die Unicast MAC-Adresse des Ziels an.
- **Aktion**  
Wählen Sie das Verhalten. Folgendes ist möglich:
  - Weiterleiten  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.

## Port ACL IP

### Regel-Konfiguration

Auf dieser Seite legen Sie die Regeln für die IP-basierte ACL fest.

### Einstellungen

Die Tabelle gliedert sich in folgende Spalten:

- **Regelnummer**  
Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Quell-IP-Adresse**  
Geben Sie die IP-Adresse der Quelle ein.
- **Quell-Subnetzmaske**  
Geben Sie die Subnetzmaske ein, in der sich die Quelle befindet.
- **Ziel-IP-Adresse**  
Tragen Sie die IP-Adresse des Ziels ein.
- **Ziel-Subnetzmaske**  
Geben Sie die Subnetzmaske ein, in der sich das Ziel befindet.
- **Aktion**  
Wählen Sie aus der Klappliste das Verhalten. Folgendes ist möglich:
  - Weiterleiten  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.
- **Egress-Port**  
Zeigt eine Liste aller Egress-Ports, für die diese Regel gilt.
- **Ingress-Port**  
Zeigt eine Liste aller Ingress-Ports, für die diese Regel gilt.

## Port Ingress-Regeln

Auf dieser Seite legen Sie fest, nach welcher ACL-Regel eingehende Telegramme vom Port behandelt werden.

### Einstellungen

- **Ports**  
Wählen Sie den gewünschten Port aus.
- **Regel hinzufügen**  
Wählen Sie die ACL-Regel, die dem Port zugewiesen wird. Die ACL-Regel legen Sie auf der Seite "Regel-Konfiguration" fest.
- **Hinzufügen**  
Um dem Port die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**  
Wählen Sie die zu löschende ACL-Regel.
- **Entfernen**  
Um die ACL-Regel vom Port zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Reihenfolge**  
Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**  
Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Quell-IP-Adresse**  
Zeigt Sie die IP-Adresse der Quelle an.
- **Quell-Subnetzmaske**  
Zeigt die Subnetzmaske an, in der sich die Quelle befindet.
- **Ziel-IP-Adresse**  
Tragen Sie die IP-Adresse des Ziels ein.
- **Ziel-Subnetzmaske**  
Tragen Sie die Subnetzmaske ein, in der sich das Ziel befindet.
- **Aktion**  
Wählen Sie das Verhalten. Folgendes ist möglich:
  - Weiterleiten  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.

## Port Egress-Regeln

Auf dieser Seite legen Sie fest, nach welcher ACL-Regel ausgehende Telegramme vom Port behandelt werden.

## Einstellungen

- **Ports**  
Wählen Sie den gewünschten Port aus.
- **Regel hinzufügen**  
Wählen Sie die ACL-Regel, die dem Port zugewiesen wird. Die ACL-Regel legen Sie auf der Seite "Regel-Konfiguration" fest.
- **Hinzufügen**  
Um dem Port die ACL-Regel fest zu zuordnen, klicken Sie auf die Schaltfläche "Hinzufügen". Die Konfiguration wird in der Tabelle angezeigt.
- **Regel entfernen**  
Wählen Sie die zu löschende ACL-Regel.
- **Entfernen**  
Um die ACL-Regel vom Port zu entfernen, klicken Sie auf die Schaltfläche "Entfernen".

Die Tabelle gliedert sich in folgende Spalten:

- **Reihenfolge**  
Zeigt die Reihenfolge der ACL-Regeln an.
- **Regelnummer**  
Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **Quell-IP-Adresse**  
Zeigt Sie die IP-Adresse der Quelle an.
- **Quell-Subnetzmaske**  
Zeigt die Subnetzmaske an, in der sich die Quelle befindet.
- **Ziel-IP-Adresse**  
Tragen Sie die IP-Adresse des Ziels ein.
- **Ziel-Subnetzmaske**  
Tragen Sie die Subnetzmaske ein, in der sich das Ziel befindet.
- **Aktion**  
Wählen Sie aus der Klappliste das Verhalten. Folgendes ist möglich:
  - Weiterleiten  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm weitergeleitet.
  - Discard  
Wenn das Telegramm die ACL-Regel erfüllt, wird das Telegramm nicht weitergeleitet.

## ACL-Management

Auf dieser Seite können Sie die Sicherheit Ihres Geräts erhöhen. Um festzulegen, welche Station mit welcher IP-Adresse auf Ihrem Gerät zugreifen darf, konfigurieren Sie die IP-Adresse oder auch ein ganzes Adress-Band.

Sie können einstellen, mit welchen Protokollen und über welche Ports die Station auf dem Gerät zugreifen darf. Sie definieren, in welchem VLAN die Station liegen darf. Damit wird gewährleistet, dass nur bestimmte Stationen innerhalb eines VLANs Zugriff auf das Gerät haben.

---

#### Hinweis

Beachten Sie, eine fehlerhafte Konfiguration kann dazu führen, dass Sie nicht mehr auf das Gerät zugreifen können.

---

## Einstellungen

- **ACL-Management**  
Aktivieren oder deaktivieren Sie die Funktion.
- **IP-Adresse**  
Geben Sie die IP-Adresse oder die Netzadresse ein, für welche die Regel gelten soll. Wenn Sie die IP-Adresse 0.0.0.0 verwenden, gelten die Einstellungen für alle IP-Adressen.
- **Subnetzmaske**  
Geben Sie die Subnetzmaske ein. Die Subnetzmaske 255.255.255.255 ist für eine bestimmte IP-Adresse. Möchten Sie ein Subnetz zulassen, geben Sie z. B. für ein C-Subnetz 255.255.255.0 ein. Die Subnetzmaske 0.0.0.0 gilt für alle Subnetze.

Die Tabelle gliedert sich in folgende Spalten:

- **Regelnummer**  
Zeigt die Nummer der ACL-Regel an. Beim Erstellen der Regel wird eine neue Zeile mit einer eindeutigen Nummer angelegt.
- **IP-Adresse**  
Zeigt die IP-Adresse an.
- **Subnetzmaske**  
Zeigt die Subnetzmaske an.
- **Zulässige VLANs**  
Geben Sie die Nummer des VLANs ein, in dem sich das Gerät befindet. Nur die Station kann auf das Gerät zugreifen, wenn es sich in diesem konfigurierten VLAN befindet. Bleibt dieses Eingabefeld leer, gibt es keine Einschränkung.
- **SNMP**  
Legen Sie fest, ob die die Station bzw. die IP-Adresse über das Protokoll SNMP auf das Gerät zugreifen darf.
- **TELNET**  
Legen Sie fest, ob die die Station bzw. die IP-Adresse über das Protokoll TELNET auf das Gerät zugreifen darf.
- **HTTP**  
Legen Sie fest, ob die Station bzw. die IP-Adresse über das Protokoll HTTP auf das Gerät zugreifen darf.
- **HTTPS**  
Legen Sie fest, ob die Station bzw. die IP-Adresse über das Protokoll HTTPS auf das Gerät zugreifen darf.

- **SSH**  
Legen Sie fest, ob die Station bzw. die IP-Adresse über das Protokoll SSH auf das Gerät zugreifen darf.
- **Px.y**  
Legen Sie fest, ob die Station bzw. die IP-Adresse über diesen Port auf dieses Gerät zugreifen darf.

#### **8.1.4.7 PROFIBUS DP konfigurieren**

### **Grundlagen zum Konfigurieren eines DP-Mastersystems**

#### **Dezentrale Peripherie**

Als dezentrale Peripherie werden DP-Mastersysteme bezeichnet, die aus DP-Master und DP-Slaves bestehen, über einen Bus verbunden sind und miteinander über das Protokoll PROFIBUS-DP kommunizieren.

#### **Firmware-Version der S7-1200-CPU**

Die Nutzung der PROFIBUS-Funktionalitäten bei der S7-1200 setzt CPUs ab Firmware-Version 2.0 voraus.

#### **Konfiguration der dezentralen Peripherie**

Weil DP-Master bzw. DP-Slaves unterschiedliche Geräte sind, sind hier nur die grundsätzlichen Vorgehensweisen zum Konfigurieren erläutert. Die Konfiguration der dezentralen Peripherie ist aber weitgehend identisch mit dem Vorgehen bei der Konfiguration eines zentralen Aufbaus.

#### **Anlegen des DP-Mastersystems in der Netzsicht**

Nachdem Sie per Drag & Drop aus dem Hardware-Katalog einen DP-Master und einen DP-Slave (z. B. ein CM 1243-5 und ein CM 1243-5) in der Netzsicht platziert haben, verbinden Sie die beiden Geräte mit einem PROFIBUS-Subnetz.

#### **Weitere Informationen**

Beachten Sie weitere Informationen zum Funktionsumfang in den Handbüchern des jeweiligen Geräts.

## DP-Slaves im Hardware-Katalog

### DP-Slaves im Hardware-Katalog

DP-Slaves finden Sie im Hardware-Katalog in dem Ordner "Dezentrale Peripherie". Dort finden Sie kompakte und modulare DP-Slaves:

- Kompakte DP-Slaves  
Baugruppen mit integrierten digitalen/analogen Ein- und Ausgängen, z. B. ET 200L
- Modulare DP-Slaves  
Anschaltungsbaugruppen mit zugeordneten S7-Baugruppen, z. B. ET 200M

Die verwendbaren DP-Slaves hängen von dem vorhandenen DP-Master und der gewünschten Funktionalität ab.

### I-Slaves im Hardware-Katalog

Ein DP-Slave, den Sie als intelligenten DP-Slave projektieren können, ist beispielsweise das CM 1242-5. Sie finden es im Hardware-Katalog an folgender Stelle:

- CM 1242-5  
"PLC > SIMATIC S7 1200 > Kommunikationsmodul > PROFIBUS"

## DP/DP-Koppler im Hardware-Katalog

### Einführung

Ein DP/DP-Koppler dient dazu, zwei PROFIBUS DP-Netze als Netzübergang (Gateway) miteinander zu verbinden, und so Daten vom DP-Master des einen Netzes zum DP-Master des anderen Netzes zu übertragen.

Die maximale Größe der übertragbaren Daten beträgt 244 Byte Eingangsdaten und 244 Byte Ausgangsdaten.

### DP/DP-Koppler im Hardware-Katalog

DP/DP-Koppler als Gateway zwischen zwei DP-Mastersystemen finden Sie im Hardware-Katalog in dem Ordner "Weitere Feldgeräte > PROFIBUS DP > Netzübergänge".

### Projektieren des DP/DP-Kopplers

DP/DP-Koppler werden an beiden PROFIBUS-Netzen in jeweils eigenen Mastersystemen projiziert.

Die Ein- und Ausgangsbereiche beider Netze müssen dabei aufeinander abgestimmt sein. Die Ausgangsdaten der einen Seite des DP/DP-Kopplers werden als Eingangsdaten der jeweils anderen Seite übernommen und umgekehrt.

## Konfigurationen mit PROFIBUS-DP

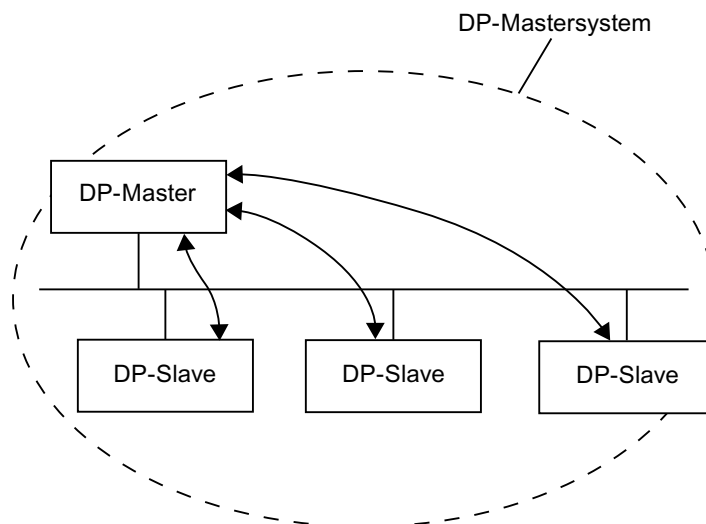
### Konfigurationen mit einfachen DP-Slaves

#### Datenaustausch zwischen DP-Master und DP-Slave

In der Konfiguration mit einfachen DP-Slaves erfolgt der Datenaustausch zwischen DP-Master und einfachen DP-Slaves, also mit E/A-Baugruppen über den DP-Master. Der DP-Master ruft nacheinander jeden innerhalb des DP-Mastersystems projizierten DP-Slave, der in seiner Aufrufliste (Polling-Liste) enthalten ist, ab. Er überträgt dabei die Ausgangsdaten an die Slaves und erhält deren Eingangsdaten zurückgeliefert.

#### Mono-Master-System

Die Konfiguration mit nur einem DP-Master wird auch als Mono-Master-System bezeichnet. Ein einziger DP-Master ist mit seinen zugehörigen DP-Slaves an einem physikalischen PROFIBUS DP-Subnetz angeschlossen.



### Konfigurationen mit intelligenten DP-Slaves

#### Definition

DP-Slaves mit eigenem Programm zur Vorverarbeitung werden als intelligente DP-Slaves (I-Slaves) bezeichnet.

Das CM 1242-5 ist ein intelligenter DP-Slave.

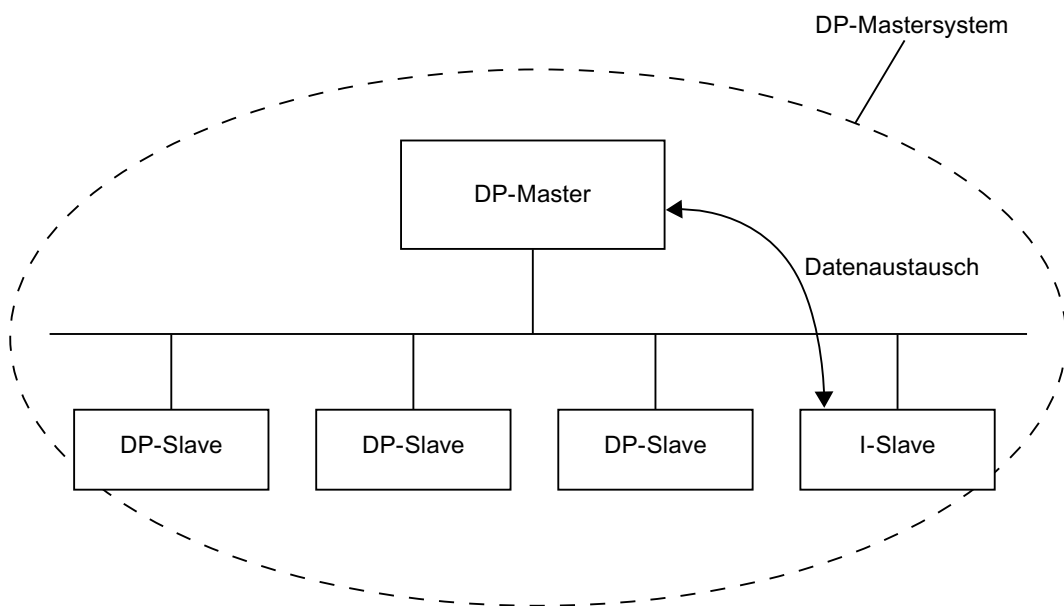
#### Datenaustausch I-Slave <> DP-Master

Ein überlagertes Automatisierungssystem behandelt die in Teilaufgaben zerlegte Automatisierungsaufgabe. Die Teilaufgaben werden in den unterlagerten

Automatisierungssystemen bearbeitet. Die zu erledigenden Steuerungsaufgaben werden eigenständig und effizient als Programme der Vorverarbeitung in den CPUs abgearbeitet.

Bei Konfigurationen mit intelligenten DP-Slaves greift der DP-Master nicht auf die E/A-Baugruppen des intelligenten DP-Slaves zu, sondern nur auf den Operandenbereich der CPU des I-Slaves. Der Operandenbereich darf nicht für reale E/A-Baugruppen im I-Slave belegt werden. Diese Zuordnung muss bei der Projektierung des I-Slaves erfolgen.

Die Adressen der zwischen Master und Slave auszutauschenden Daten werden im Transferbereich des I-Slave projiziert.



## Dezentrale Peripheriesysteme konfigurieren

### Tipp: Schnelles Konfigurieren von Mastersystemen

Wenn das DP-Mastersystem viele DP-Slaves hat, weisen Sie alle per Drag & Drop platzierten DP-Slaves in einem Schritt einem Master zu.

### Voraussetzung

DP-Master und DP-Slaves sind in der Netzsicht platziert.

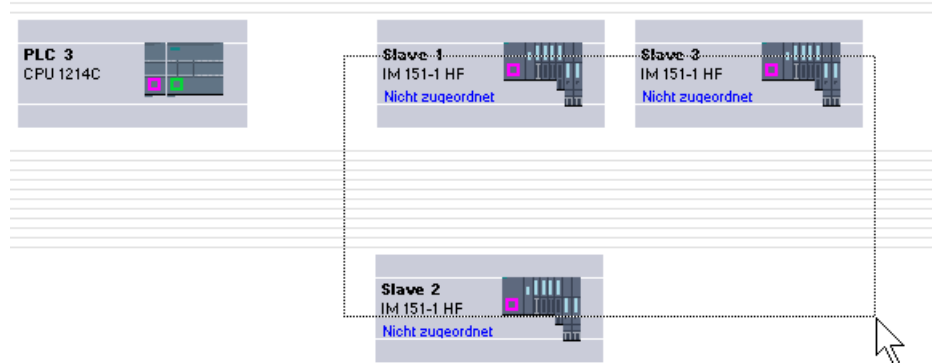
### DP-Slaves einem DP-Mastersystem zuweisen

Gehen Sie folgendermaßen vor:

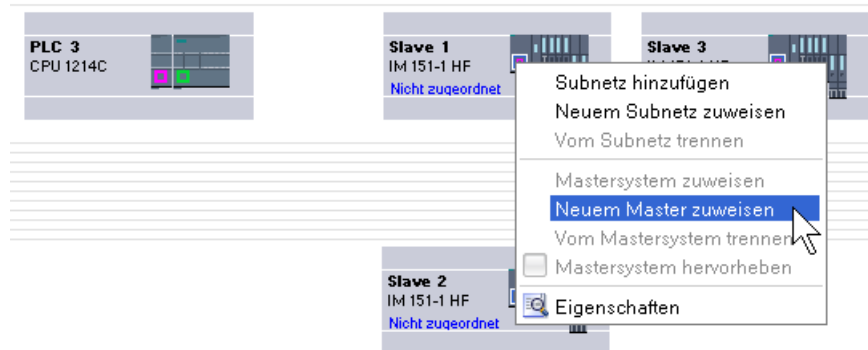
1. Wählen Sie einen passenden Zoomfaktor, so dass Sie möglichst viele DP-Slaves in der Netzsicht sehen.
2. Ordnen Sie die DP-Slaves in maximal zwei Reihen an.



3. Wählen Sie mit dem Mauszeiger alle DP-Schnittstellen aus (nicht die Geräte!). Das gelingt nur, wenn Sie außerhalb des ersten DP-Slaves mit dem Ziehen des Mauszeigers beginnen und beim letzten DP-Slave die Maustaste loslassen (Auswahl mit dem Lasso).



4. Wählen Sie das Kontextmenü "Neuem Master zuweisen" und markieren Sie im Folgedialog die entsprechende DP-Schnittstelle des DP-Masters.



5. Die DP-Slaves werden automatisch mit dem DP-Master vernetzt und bilden mit ihm zusammen ein DP-Mastersystem.

#### Hinweis

Bei einem hervorgehobenem DP-Mastersystem können Sie im Hardware-Katalog auf einen DP-Slave doppelklicken und damit schnell weitere DP-Slaves hinzuzügeln. Der DP-Slave wird dadurch automatisch in das hervorgehobene DP-Mastersystem aufgenommen.

## DP-Mastersystem anlegen

### Einführung

Um ein DP-Mastersystem anzulegen, benötigen Sie einen DP-Master und wenigstens einen DP-Slave. Sobald Sie einen DP-Master mit einem DP-Slave verbinden, erfolgt eine Master-Slave-Kopplung.

## DP-Master

Als DP-Master können Sie folgende Geräte einsetzen:

- CM 1243-5

## Voraussetzung

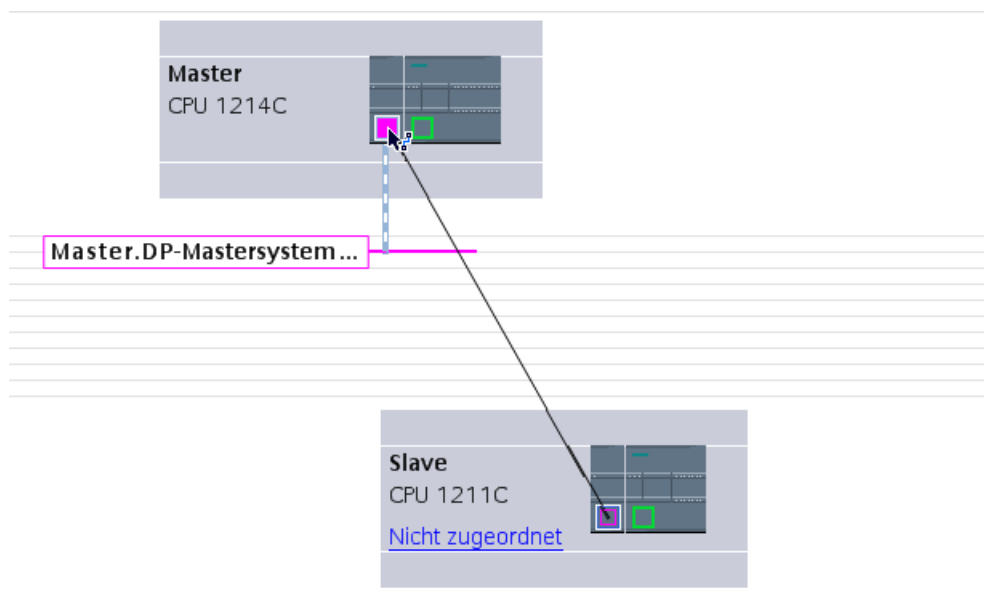
- Sie befinden sich in der Netzsicht.
- Der Hardware-Katalog ist geöffnet.

## Vorgehen

Um ein DP-Mastersystem anzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen DP-Master aus dem Hardware-Katalog.
2. Ziehen Sie den DP-Master auf die freie Fläche in der Netzsicht.
3. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle des DP-Masters.
4. Wählen Sie aus dem Kontextmenü "Mastersystem anlegen".  
Es wird ein DP-Mastersystem mit einem DP-Master als einzigem Teilnehmer angelegt.

Wenn Sie die DP-Schnittstelle eines DP-Slave mit der DP-Schnittstelle des DP-Masters verbinden, wird der DP-Slave in das Mastersystem aufgenommen.



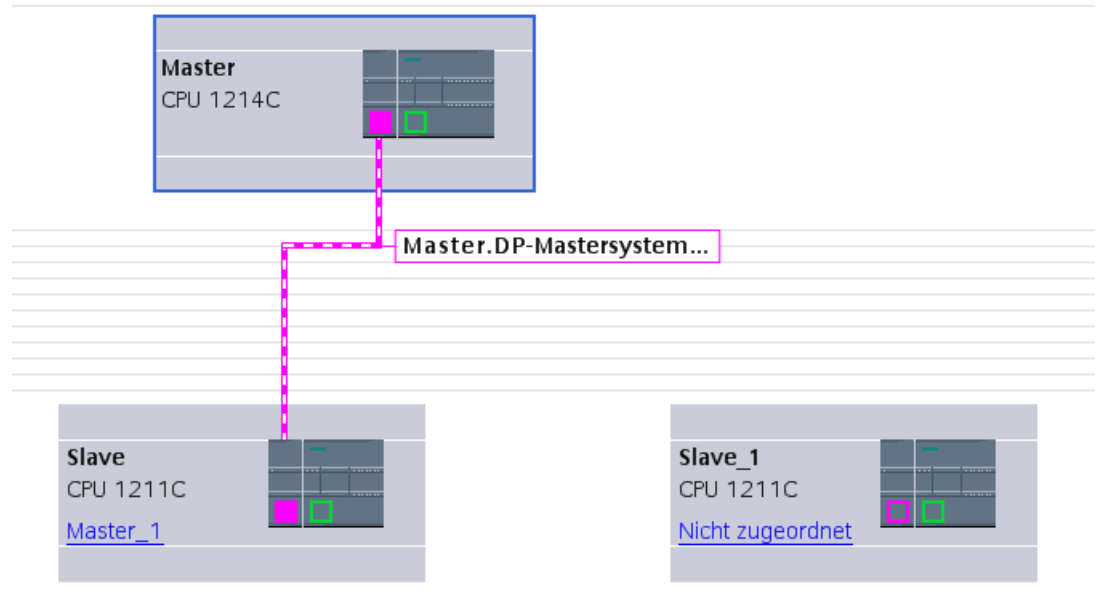
Wenn Sie in der Netzsicht bereits einen DP-Master und einen DP-Slave platziert haben, können Sie diese mit Drag & Drop verbinden und dadurch ein DP-Mastersystem erstellen. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie auf die DP-Schnittstelle des DP-Masters oder DP-Slave.
2. Ziehen Sie mit gedrückter Maustaste eine Verbindung von der selektierten DP-Schnittstelle auf die DP-Schnittstelle des gewünschten Kommunikationspartners.

Es wird ein Subnetz mit einem DP-Mastersystem zwischen DP-Master und DP-Slave erzeugt.

### Anzeige des DP-Masters im DP-Slave

Wenn Sie einen DP-Slave mit einem DP-Master verbinden, dann wird der Name des DP-Masters als Hyperlink am DP-Slave angezeigt. Wenn Sie auf den Hyperlink klicken, wird der zugeordnete DP-Master selektiert.

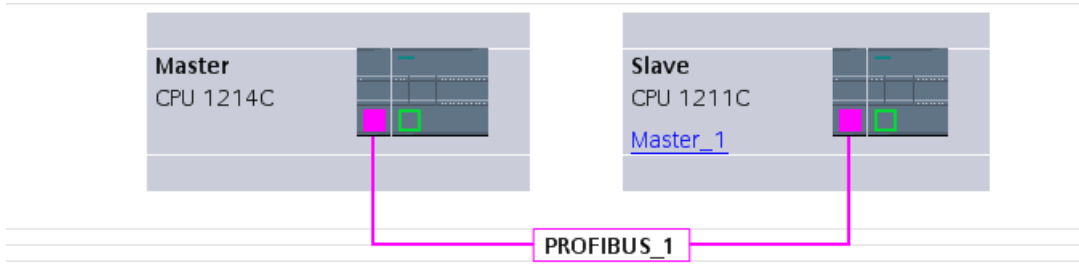


### Hervorheben des DP-Mastersystems

Wenn Sie ein neues DP-Mastersystem angelegt haben, wird das DP-Mastersystem hervorgehoben dargestellt. Dadurch erkennen Sie schnell, welche Geräte zu dem DP-Mastersystem gehören. Sie können ein DP-Mastersystem auch hervorheben, indem Sie den Mauszeiger über ein Subnetz ziehen. Es werden daraufhin die Namen der vorhandenen DP-Mastersysteme angezeigt. Wenn Sie mit der Maus auf eines der angezeigten DP-Mastersysteme klicken, wird das entsprechende DP-Mastersystem hervorgehoben.

Um die Hervorhebung eines DP-Mastersystems wieder auszublenden, stehen Ihnen mehrere Möglichkeiten zu Verfügung:

- Heben Sie ein anderes Mastersystem hervor.
- Klicken Sie auf die Pin-Nadel mit der Bezeichnung des Mastersystems in der rechten oberen Ecke der Netzsicht.



## DP-Mastersysteme und -Schnittstellen bearbeiten

### Einführung

Nachdem Sie ein DP-Mastersystem angelegt haben, haben Sie auch die Möglichkeit, das DP-Mastersystem von seinen Komponenten zu trennen. Daraus können sich Subnetze mit DP-Slaves aber ohne DP-Master ergeben.

Das Bearbeiten der Schnittstellen eines DP-Masters ist meist nicht notwendig.

Am DP-Mastersystem können Sie den Namen und die Nummer ändern.

### Trennung von Master oder Slaves vom DP-Mastersystemen

Wenn Sie einen PROFIBUS-CP als DP-Master mit Mastersystem projiziert haben, dann können Sie das DP-Mastersystem vom DP-Master trennen. Das Gerät ist dann nicht mehr an das DP-Mastersystem angeschlossen.

Das Trennen des Subnetzes von einem DP-Master hat zur Folge, dass das Mastersystem nicht mehr existiert, da es keinem DP-Master mehr zugeordnet ist. Die einzelnen DP-Slaves sind aber weiterhin miteinander über das Subnetz verbunden.

Wenn Sie die DP-Slaves löschen oder vom Mastersystem trennen, dann bleibt das Mastersystem am DP-Master erhalten.

### Voraussetzung

- Sie befinden sich in der Netzsicht.
- Es besteht ein DP-Mastersystem mit einem DP-Master und mindestens einem DP-Slave.

### DP-Master vom DP-Mastersystem trennen

Um das DP-Mastersystem zu trennen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle des DP-Masters.
2. Wählen Sie aus dem Kontextmenü "Vom Mastersystem trennen".

Der selektierte DP-Master wird vom DP-Mastersystem getrennt. Es bleibt ein Subnetz mit den DP-Slaves zurück.

### **DP-Master dem DP-Mastersystem hinzufügen**

Um einem Subnetz wieder einen DP-Master zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle eines DP-Masters.
2. Wählen Sie aus dem Kontextmenü "Mastersystem anlegen".
3. Ziehen Sie das neue DP-Mastersystem auf die DP-Schnittstellen der DP-Slaves.

Der DP-Master bildet mit den DP-Slaves wieder ein DP-Mastersystem.

### **Eigenschaften eines DP-Mastersystems bearbeiten**

Um die Eigenschaften eines DP-Mastersystems zu bearbeiten, gehen Sie folgendermaßen vor:

1. Bewegen Sie den Mauszeiger auf ein Subnetz mit DP-Mastersystem.
2. Klicken Sie in der erscheinenden Meldung über die vorhandenen DP-Mastersysteme auf das DP-Mastersystem, welches Sie bearbeiten wollen. Das DP-Mastersystem wird nun farblich hervorgehoben.
3. Klicken Sie auf das hervorgehobene DP-Mastersystem.
4. Bearbeiten Sie im Inspektorfenster unter "Eigenschaften > Allgemein" die Attribute des DP-Mastersystems.

---

#### **Hinweis**

Wenn Sie ein Subnetz anklicken, ohne dass ein DP-Mastersystem hervorgehoben ist, können Sie im Inspektorfenster unter "Eigenschaften" die Eigenschaften des ganzen Subnetzes bearbeiten.

---

### **DP-Slaves am Mastersystem einfügen und konfigurieren**

In der Netzsicht fügen Sie verschiedene DP-Slaves direkt per Drag & Drop oder Doppelklick aus dem Hardware-Katalog ein.

### **DP-Slaves**

Bei der Projektierung von DP-Slaves werden diese unterschieden nach:

- Kompakte DP-Slaves  
(Baugruppen mit integrierten digitalen/analogen Ein- und Ausgängen, z. B. ET 200L)
- Modulare DP-Slaves  
(Anschaltungsbaugruppen mit zugeordneten S5- oder S7-Baugruppen, z. B. ET 200M)
- Intelligente DP-Slaves (I-Slaves)  
(CM 1242-5 oder ET 200S mit IM 151-7 CPU)

## Regeln

- In einem DP-Mastersystem verwenden Sie nur einen DP-Master und einen oder mehrere DP-Slaves.
- Sie dürfen nicht mehr DP-Slaves in einem DP-Mastersystem projektieren als für den betreffenden DP-Master zulässig ist.

---

### Hinweis

Beachten Sie bei der Konfigurierung des DP-Mastersystems die technischen Daten des DP-Masters (max. Anzahl Teilnehmer, max. Anzahl Steckplätze, max. Anzahl Nutzdaten). Möglicherweise können Sie aufgrund der Begrenzung der Nutzdaten nicht die maximale Anzahl von Teilnehmern verwenden!

---

## Voraussetzung

- Sie befinden sich in der Netzsicht.
- Ein DP-Mastersystem ist angelegt.

## DP-Slave am DP-Mastersystem einfügen

Um einen DP-Slave aus dem Hardware-Katalog am DP-Mastersystem einzufügen, gehen Sie folgendermaßen vor:

1. Wählen Sie einen DP-Slave aus dem Hardware-Katalog.
2. Ziehen Sie den DP-Slave per Drag & Drop vom Hardware-Katalog in die Netzsicht.
3. Ziehen Sie eine Verbindung von der DP-Schnittstelle des DP-Masters oder von dem hervorgehobenen DP-Mastersystem zu der DP-Schnittstelle des neuen DP-Slave.

Es wird automatisch ein DP-Mastersystem erstellt und der DP-Slave wird automatisch mit dem DP-Master verbunden.

---

### Hinweis

Bei einem hervorgehobenem DP-Mastersystem können Sie im Hardware-Katalog einen Doppelklick auf den gewünschten DP-Slave durchführen. Der DP-Slave wird dadurch automatisch in das hervorgehobene DP-Mastersystem aufgenommen.

---

### **DP-Slave vom DP-Mastersystem trennen**

Um einen DP-Slave vom DP-Mastersystem zu trennen, gehen Sie folgendermaßen vor:

1. Klicken Sie innerhalb der Netzsicht mit der rechten Maustaste auf die DP-Schnittstelle des DP-Slaves.
2. Wählen Sie aus dem Kontextmenü die Art der Trennung vom DP-Mastersystem:
  - "Vom Subnetz trennen": Die PROFIBUS-Verbindung wird getrennt und das Gerät ist nicht mehr an das DP-Mastersystem oder ein Subnetz angeschlossen.
  - "Vom Mastersystem trennen": Der DP-Slave ist weiterhin an das Subnetz angeschlossen, aber nicht mehr dem DP-Mastersystem als DP-Slave zugeordnet.

Der selektierte DP-Slave wird vom DP-Mastersystem getrennt.

### **DP-Slave einem neuen DP-Mastersystem zuweisen**

Um einen vorhandenen DP-Slave einem neuen DP-Mastersystem zuzuweisen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die DP-Schnittstelle des DP-Slave.
2. Wählen Sie im Kontextmenü "Neuem Master zuweisen".  
Es ist dabei unerheblich, ob der DP-Slave bereits einem anderen DP-Mastersystem zugeordnet ist.
3. Wählen Sie aus der Auswahlliste den DP-Master aus, an dessen DP-Mastersystem Sie den DP-Slave anbinden wollen.  
Der selektierte DP-Slave wird einem neuen DP-Mastersystem zugewiesen.

Analog dazu können Sie einen DP-Slave über die Funktion "Neuem Subnetz zuweisen" auch an ein anderes Subnetz anbinden, ohne ihn dadurch an ein bestehendes DP-Mastersystem zu koppeln.

### **DP-Slave konfigurieren**

Um einen DP-Slave zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wechseln Sie in die Gerätesicht des DP-Slave.
2. Wählen Sie die gewünschte Baugruppen aus.
3. Konfigurieren Sie den DP-Slave im Inspektorenfenster.

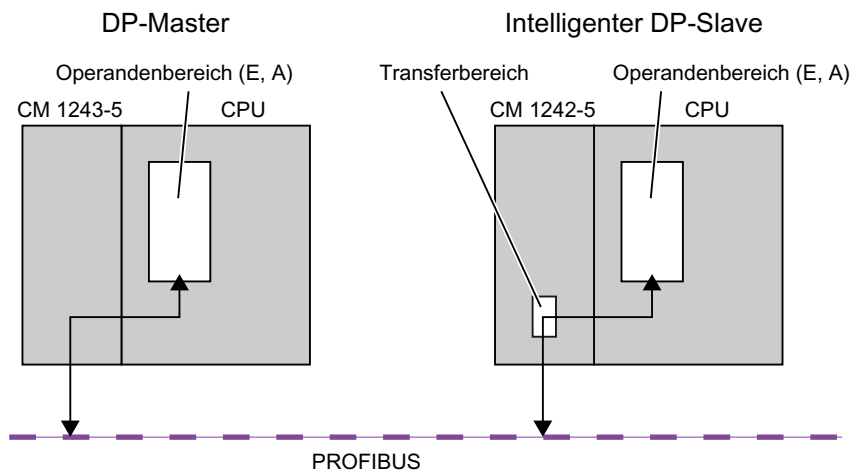
### **Intelligente DP-Slaves konfigurieren**

#### **I-Slave in ein DP-Mastersystem einfügen**

#### **Einführung**

Merkmal eines intelligenten DP-Slave (I-Slave) ist, dass Ein-/Ausgangsdaten nicht unmittelbar von einem realen Ein-/Ausgang dem DP-Master zur Verfügung gestellt werden, sondern von

einer vorverarbeitenden CPU. Diese vorverarbeitende CPU bildet zusammen mit dem CP den I-Slave.



### Unterschied: DP-Slave - Intelligenter DP-Slave

Bei einem DP-Slave greift der DP-Master auf die dezentralen Ein-/Ausgänge zu.

Bei einem intelligenten DP-Slave greift der DP-Master nicht auf die angeschlossenen Ein-/Ausgänge des intelligenten DP-Slaves zu, sondern auf einen Übergabebereich im E/A-Adressraum der vorverarbeitenden CPU. Das Anwenderprogramm der vorverarbeitenden CPU muss für den Austausch der Daten zwischen Operandenbereich und Ein-/Ausgängen sorgen.

---

#### Hinweis

Die konfigurierten E/A-Bereiche für den Datenaustausch zwischen DP-Master und DP-Slaves dürfen nicht von E/A-Baugruppen verwendet werden.

---

### Anwendungen

Konfigurationen mit intelligenten DP-Slaves: Datenaustausch I-Slave <> DP-Master

### Vorgehen

Um einen I-Slave in ein DP-Mastersystem einzufügen, gehen Sie folgendermaßen vor:

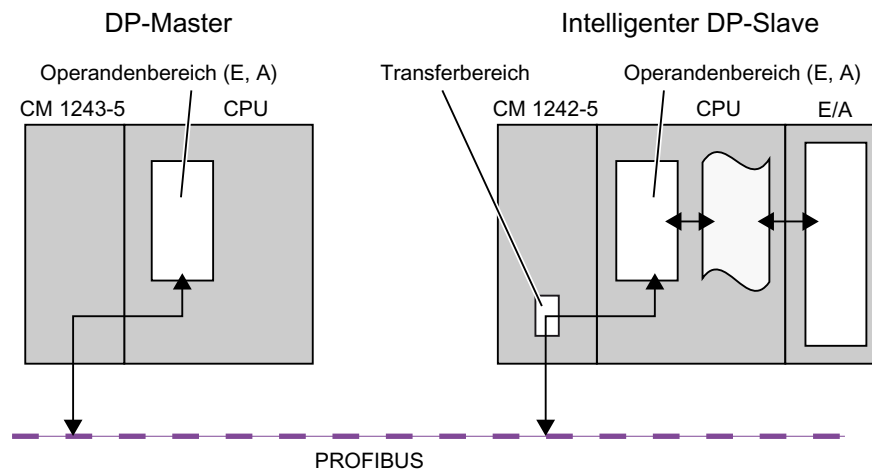
1. Ziehen Sie in der Netzsicht aus dem Hardware-Katalog einen CM 1242-5 als I-Slave in eine Station und einen CM 1243-5 als DP-Master in eine Station.
2. Ziehen Sie eine Verbindungslinie zwischen den DP-Schnittstellen der beiden Geräte. Dadurch verbinden Sie den I-Slave mit einem DP-Master in einem DP-Mastersystem. Ergebnis: Sie haben nun ein DP-Mastersystem mit einem DP-Master und einem I-Slave aufgebaut.



## Zugriff auf Daten des I-Slave konfigurieren

### Datenzugriff

Für den CM 1242-5 als I-Slave gilt: Die Adressen für den Übergabebereich der Daten und die Adresse für die E/A-Baugruppen im I-Slave sind unterschiedlich. Die Anfangsadresse, die eine E/A-Baugruppe belegt, kann daher nicht mehr für den Übergabespeicher verwendet werden. Wenn also der übergeordnete DP-Master auf Daten einer E/A-Baugruppe im I-Slave zugreifen soll, müssen Sie diesen Datenaustausch zwischen E/A-Baugruppe und Übergabebereich im Anwenderprogramm des I-Slave konfigurieren.



### Projektierung des Übergabebereichs beim CM 1242-5 (Transferbereich)

Beim CM 1242-5 wird der Übergabebereich für den zyklischen PROFIBUS-Datenaustausch in der Parametergruppe "PROFIBUS-Schnittstelle > Betriebsart > I-Slave-Kommunikation" als Transferbereich projektiert.

### Direkter Datenzugriff von CPU zu CPU

Direkter Datenzugriff von CPU zu CPU über PROFIBUS wird von den S7-1200-PROFIBUS-CMs nur über die PUT/GET-Dienste unterstützt.

## DP-Slaves der Dezentralen Peripherie konfigurieren

### ET 200S konfigurieren

#### Steckplatzregeln für die Konfiguration einer ET 200S

Für das Konfigurieren der ET 200S gelten die folgende Regeln:

- Stecken Sie die Module der ET 200S lückenlos nebeneinander.
- Steckplatz 1: nur für Powermodule PM-E oder PM-D

- Links neben einem Elektronikmodul (EM): Nur ein EM oder ein Powermodul (PM-E oder PM-D)
- Links neben einem Motorstarter (MS): Nur ein MS oder ein Powermodul PM-D, PM-D Fx (1..x..4) oder PM-X
- Links neben einem PM-X: Nur ein Motorstarter oder ein PM-D
- Erlaubt sind maximal 63 Module und ein Interfacemodul IM

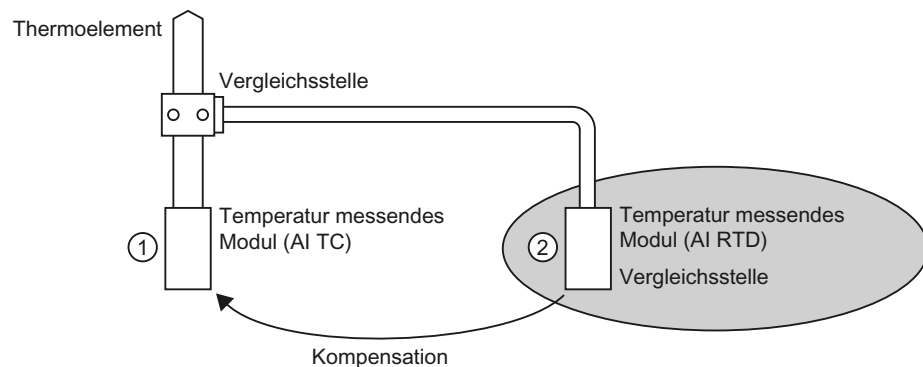
**Hinweis**

Beachten Sie die korrekte Zuordnung von PM-E-Spannungsbereich und EM-Spannungsbereich!

**Vergleichsstelle parametrieren**

Eine Vergleichsstelle ist der Anschluss eines Thermoelements an eine Zuleitung (i. d. R. im Klemmenkasten). Die hier durch Temperatureinfluss auftretende Spannung verfälscht den Temperaturwert, den das Modul misst.

Bei ET 200S kann ein Kanal des Moduls AI RTD als Vergleichsstelle parametrieren werden. Mit der von diesem Modul gemessenen Temperatur an der Vergleichsstelle können andere AI TC-Module ihre Messwerte kompensieren.



- ① Parametrierung des AI TC:  
→ Auswahl der verwendeten Vergleichsstelle
- ② Parametrierung des AI RTD:  
→ Aktivierung der Vergleichsstelle  
→ Festlegung Steckplatz und Kanal des AI RTD

**Besonderheiten bei der Parametrierung von Vergleichsstellen**

Die Parametrierung von Vergleichsstellen wird Ihnen exemplarisch an der Verwendung eines Widerstandsthermometers Pt 100 "Klimabereich" zum Erfassen der Vergleichsstellentemperatur gezeigt.

Um die Vergleichsstelle zu parametrieren, gehen Sie folgendermaßen vor:

1. Platzieren Sie in der Gerätesicht der ET 200S ein analoges Elektronikmodul, z. B. 2AI RTD HF.
2. Selektieren Sie die Baugruppe im Baugruppenträger.
3. Stellen Sie im Inspektorfenster unter "Eigenschaften > Eingänge" einen Kanal für die Vergleichsstellen-Funktion auf den Messbereich "RTD-4L Pt 100 Kl." ein.
4. Selektieren Sie die ET 200S.
5. Aktivieren Sie im Inspektorfenster unter "Eigenschaften > Baugruppenparameter > Vergleichsstellen" das Optionskästchen "Vergleichsstelle" und geben Sie den Steckplatz und die Kanalnummer der relevanten RTD-Baugruppe an.
6. Platzieren Sie das analoge Elektronikmodul für Temperaturmessung mittels Thermoelement (TC-Modul) und parametrieren Sie es mit der Vergleichsstellennummer des RTD-Moduls.

### Weitere Informationen

Beachten Sie für weitere Informationen zu den verschiedenen Typen und Einsatzmöglichkeiten von Modulen in ET 200S die Betriebsanleitung und das Gerätehandbuch "Dezentrales Peripheriesystem ET 200S".

Beachten Sie für weitere Informationen zur Analogwertverarbeitung die Dokumentation zum Dezentralen Peripheriesystem ET 200S.

### Adressen packen

### Einführung

DP-Slaves und IO-Devices der Familie ET 200S werden konfiguriert wie andere modulare DP-Slaves und IO-Devices. Zusätzlich zu den üblichen Funktionen modularer DP-Slaves und IO-Devices gibt es bei ET 200S die Funktion "Adressen packen":

Digitale Elektronikmodule mit einem benötigten Adressraum von 2 oder 4 Bit belegen beim Einfügen in der Gerätesicht zunächst pauschal 1 Byte. Der wirklich belegte Adressraum kann aber nach dem Konfigurieren durch die Funktion "Adressen packen" zusammengeschoben werden.

|             | Vorbelegung | Nach "Adressen packen" |
|-------------|-------------|------------------------|
| Modul       | E-Adresse   | E-Adresse              |
| 2DI (2 Bit) | Byte 10     | 10.0...10.1            |
| 4DI (4 Bit) | Byte 11     | 10.2...10.5            |

### Voraussetzung

- Sie befinden sich in der Gerätesicht.
- Eine ET 200S, z. B. IM 151-1, ist vorhanden.
- Ein paar digitale Elektronikmodule, z. B. 2DI AC120V ST befinden sich in den Steckplätzen.

## Adressen packen

Um Adressen zu packen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Elektronikmodule, deren Adressen gepackt werden sollen. Für das Selektieren von mehreren Elektronikmodulen haben Sie folgende Möglichkeiten:
  - Klicken Sie die Elektronikmodule bei gleichzeitigem Drücken von <Shift> oder <Strg> einzeln an.
  - Klicken Sie außerhalb des Baugruppenträgers und ziehen Sie die Maus um die zu selektierenden Elektronikmodule.
2. Klicken Sie im Kontextmenü der selektierten Elektronikmodule auf "Adressen packen".

Es werden die Adressbereiche für Eingänge, Ausgänge und Motorstarter getrennt gepackt. In den Spalten E-Adresse und A-Adresse der Geräteübersicht sehen Sie die gepackten Adressen.

## Adressbildung und Struktur bei gepackten Adressen

Wenn Sie die Funktion "Adressen packen" verwenden, werden die Adressen der selektierten Elektronikmodule nach folgenden Regeln gepackt:

- Der Anfang des Adressbereichs ist durch die niedrigste Adresse der selektierten Elektronikmodule festgelegt: X.0.
- Wenn die Bitadresse nicht "0" ist, dann wird automatisch die nächste (freie) Byteadresse herangezogen, ab welcher der markierte Bereich eingeschoben werden kann: (X+n).0.
- Wenn kein zusammenhängender Bereich mehr vorhanden ist, wird automatisch in vorhandene Adresslücken gepackt.

Elektronikmodule mit gepackten Adressen und derselben Byteadresse bilden eine Packgruppe.

## Adressen entpacken

Um Adressen zu entpacken, gehen Sie folgendermaßen vor:

1. Selektieren Sie ein oder mehrere Elektronikmodule mit gepackter Adresse.
2. Klicken Sie im Kontextmenü der selektierten Elektronikmodule auf "Adressen entpacken".

Packgruppen der selektierten Elektronikmodule werden aufgelöst und die gepackten Adressen der entsprechenden Elektronikmodule werden entpackt.

Wenn Sie aus einer Packgruppe Elektronikmodule löschen oder verschieben oder auf einen freien Steckplatz innerhalb einer Packgruppe Elektronikmodule einfügen, wird die Packgruppe ebenfalls aufgelöst und die gepackten Adressen werden entpackt.

Die Anfangsadressen der entpackten Elektronikmodule werden auf die jeweils nächsten freien Byte-Adressen gelegt.

## **Besonderheiten bei Elektronikmodulen mit gepackten Adressen**

Für ein Elektronikmodul mit einer gepackten Adresse gelten folgende Besonderheiten:

- Aus CPU-Sicht ist keine Steckplatzzuordnung für das Elektronikmodul möglich. Aus diesem Grund gibt die Anweisung GADR\_LGC (SFC 5) für den tatsächlichen Steckplatz des Elektronikmoduls die Fehlerinformation W#16#8099 "Steckplatz nicht projektiert" aus.
- Die Anweisung LGC\_GADR (SFC 49) und die SZL-ID W#16#xy91 "Baugruppenzustandsinformation" für ein Elektronikmodul können nicht ausgewertet werden.
- Das Elektronikmodul erhält eine zusätzliche Diagnoseadresse über die DPV1-Funktionalität, weil aus CPU-Sicht Alarmer aufgrund der gepackten Adresse sonst nicht zugeordnet werden können.
- Der "Ziehen/Stecken-Alarm" ist nicht möglich, da sich die Funktionen "Adressen packen" und "Ziehen/Stecken-Alarm" gegenseitig ausschließen.

## **Optionenhandling mit Reservemodulen konfigurieren**

Über das Optionenhandling können Sie die ET 200S mit PROFIBUS-Schnittstelle für zukünftige Erweiterungen (Optionen) einrichten. Im hier beschriebenen Abschnitt wird das Optionenhandling mit Reservemodulen beschrieben.

Dazu montieren, verdrahten, projektieren und programmieren Sie den geplanten Maximalausbau der ET 200S und verwenden statt der erst später erforderlichen Elektronikmodule bei der Montage zunächst preisgünstige Reservemodule (138-4AA00 oder 138-4AA10).

---

### **Hinweis**

Die ET 200S kann mit der Stammverkabelung vollständig vorverdrahtet werden, da ein Reservemodul keine Verbindung zu den Klemmen des Terminalmoduls und damit zum Prozess hat.

---

## **Voraussetzung**

- ET 200S Interfacemodul
  - IM 151-1 STANDARD (ab 6ES7 151-1AA03-0AB0)
  - IM 151-1 FO STANDARD (ab 6ES7 151-1AB02-0AB0)
- Powermodul mit Optionenhandling
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

## Vorgehen

Um das Optionenhandling einzuschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Gerätesicht das IM 151-1 und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Optionenhandling" das Optionskästchen "Optionenhandling".
2. Aktivieren Sie nun die nummerierten Optionskästchen für die Steckplätze, die zunächst mit Reservemodulen statt mit Elektronikmodulen bestückt sind.
3. Selektieren Sie in der Gerätesicht das Powermodul und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Adressen" das Optionskästchen "Optionenhandling". Reservieren Sie für die Steuer- und Rückmeldeschnittstelle den dafür benötigten Adressraum im Prozessabbild der Ausgänge (PAA) und im Prozessabbild der Eingänge (PAE).

Zu einem späteren Zeitpunkt können die montierten Reservemodule durch die projektierten Module ersetzt werden, ohne die Projektierung nachholen zu müssen.

---

### Hinweis

Die Adressen für diese Schnittstellen werden reserviert, sobald Sie Optionenhandling im Powermodul aktiviert haben. Die Funktion "Optionenhandling" muss auch im DP-Slave (Interfacemodul IM 151-1 STANDARD) aktiviert sein. Wenn sie nicht aktiviert ist, werden die reservierten Adressen für die Steuer- und Rückmeldeschnittstelle wieder frei gegeben.

Beachten Sie, dass durch wiederholtes Aktivieren und Deaktivieren des Optionenhandlings sich die Adresse der Steuer- und Rückmeldeschnittstelle ändern kann.

Das Optionenhandling darf nur bei genau einem Powermodul PM-E DC24..48V oder PM-E DC24..48V/AC24..230V aktiviert sein.

---

## Weitere Informationen

Beachten Sie für weitere Informationen über die Belegung und Bedeutung der Bytes im Prozessabbild, das Optionenhandling am PROFIBUS und den Einsatz von Reservemodulen die Dokumentationen zum Dezentralen Peripheriesystem ET 200S.

## Funktionsweise des Optionenhandlings im Anlauf

Wenn "Anlauf bei Soll- ungleich Istausbau" gesperrt ist, dann läuft die ET 200S auch dann an, wenn statt des projektierten Elektronikmoduls ein Reservemodul steckt und das Optionenhandling für diesen Steckplatz eingeschaltet ist.

## Funktionsweise des Optionenhandlings im Betrieb

Im Betrieb unterscheidet sich die Funktionsweise des Optionenhandlings wie folgt:

- Optionenhandling für einen Steckplatz eingeschaltet:  
Auf diesem Steckplatz darf sich das Reservemodul (Option) oder das projektierte Elektronikmodul befinden. Wenn sich ein anderes Modul auf diesem Steckplatz befindet, wird eine Diagnose gemeldet (kein Modul bzw. falsches Modul).
- Optionenhandling für einen Steckplatz ausgeschaltet:  
Auf diesem Steckplatz darf sich nur das projektierte Elektronikmodul befinden. Bei jedem anderen Modul wird eine Diagnose gemeldet (kein Modul bzw. falsches Modul).

## Ersatzwerte des Reservemoduls

- Ersatzwert für digitale Eingänge: 0
- Ersatzwert für analoge Eingänge: 0x7FFF

## Steuern und Auswerten im Anwenderprogramm

Die ET 200S verfügt über eine Steuer- und Rückmeldeschnittstelle für die Funktion "Optionenhandling".

Die Steuerschnittstelle liegt im Prozessabbild der Ausgänge (PAA). Jedes Bit in diesem Adressbereich steuert einen der Steckplätze 2 bis 63:

- Wert des Bits = 0: Es gilt die Parametrierung des Optionenhandlings. Reservemodule sind erlaubt.
- Wert des Bits = 1: Die Parametrierung des Optionenhandlings ist aufgehoben. Reservemodule werden auf diesem Steckplatz nicht akzeptiert.

Die Rückmeldeschnittstelle liegt im Prozessabbild der Eingänge (PAE). Jedes Bit in diesem Adressbereich gibt Auskunft über das tatsächlich steckende Modul auf den Steckplätzen 1 bis 63:

- Wert des Bits = 0: Auf dem Steckplatz befinden sich das Reservemodul, ein falsches Modul oder ein gezogenes Modul.
- Wert des Bits = 1: Auf dem Steckplatz befindet sich das projektierte Modul.

## Siehe auch

Welche Module unterstützen Optionenhandling? (<http://support.automation.siemens.com/WW/view/de/22564754>)

### Optionenhandling ohne Reservemodule konfigurieren

Über das Optionenhandling können Sie die ET 200S für zukünftige Erweiterungen (Optionen) auch ohne die Montage von Reservemodulen einrichten. Im hier beschriebenen Abschnitt wird das Optionenhandling ohne Reservemodulen beschrieben.

---

#### Hinweis

##### ET 200S mit PROFINET-Schnittstelle

Diese Beschreibung bezieht sich auf ET 200S mit PROFIBUS-Schnittstelle. Das Optionenhandling für ET 200S mit PROFINET-Schnittstelle funktioniert prinzipiell so wie hier beschrieben ohne Reservemodule. Statt der hier aufgeführten DP-Interfacemodule sind PN-Interfacemodule zu verwenden. Weitere Informationen zum Optionenhandling bei ET 200S mit PROFINET-Schnittstelle finden Sie in den entsprechenden Gerätehandbüchern.

---

### Voraussetzung

- ET 200S Interfacemodul
  - IM 151-1 HIGH FEATURE (ab 6ES7151-1BA02)
  - IM 151-1 STANDARD (ab 6ES7 151-1AA05-0AB0)
- Powermodul mit Optionenhandling
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V



## Vorgehen

Um das Optionenhandling einzuschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Gerätesicht das IM 151-1 und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Optionenhandling" das Optionskästchen "Optionenhandling".
2. Selektieren Sie in der Gerätesicht das Powermodul und aktivieren Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Adressen" das Optionskästchen "Optionenhandling". Reservieren Sie für die Steuer- und Rückmeldeschnittstelle den dafür benötigten Adressraum im Prozessabbild der Ausgänge (PAA) und im Prozessabbild der Eingänge (PAE).
3. Konfigurieren Sie den Maximalausbau des Slaves. Die Aktivierung/Deaktivierung von Optionen wird über das Anwenderprogramm gesteuert.

---

### Hinweis

Die Adressen für diese Schnittstellen werden reserviert, sobald Sie Optionenhandling im Powermodul aktiviert haben. Die Funktion "Optionenhandling" muss auch im DP-Slave (Interfacemodul IM 151-1) aktiviert sein. Wenn sie nicht aktiviert ist, werden die reservierten Adressen für die Steuer- und Rückmeldeschnittstelle wieder frei gegeben.

Beachten Sie, dass durch wiederholtes Aktivieren und Deaktivieren des Optionenhandlings sich die Adresse der Steuer- und Rückmeldeschnittstelle ändern kann.

Das Optionenhandling darf nur bei genau einem Powermodul PM-E DC24..48V oder PM-E DC24..48V/AC24..230V aktiviert sein.

---

## Weitere Informationen

Beachten Sie für weitere Informationen über die Belegung und Bedeutung der Bytes im Prozessabbild, das Optionenhandling am PROFIBUS und den Einsatz von Reservemodulen die Dokumentationen zum Dezentralen Peripheralsystem ET 200S.

## Steuern und Auswerten im Anwenderprogramm

Die ET 200S verfügt über eine Steuer- und Rückmeldeschnittstelle für die Funktion "Optionenhandling".

Die Steuerschnittstelle liegt im Prozessabbild der Ausgänge (PAA). Jedes Bit in diesem Adressbereich steuert einen der Steckplätze 1 bis 63:

- Wert des Bits = 0: Steckplatz ist in der Istkonfiguration nicht vorhanden.
- Wert des Bits = 1: Steckplatz ist in der Istkonfiguration vorhanden.

Die Rückmeldeschnittstelle liegt im Prozessabbild der Eingänge (PAE). Jedes Bit in diesem Adressbereich gibt Auskunft über das tatsächlich steckende Modul auf den Steckplätzen 1 bis 63:

- Wert des Bits = 0: Steckplatz gehört zu einer nicht vorhandenen Option oder Modulstatus ist nicht in Ordnung.
- Wert des Bits = 1: Auf dem Steckplatz befindet sich das projektierte Modul.

**Siehe auch**

Beispielapplikation für ET 200S, Optionenhandling ohne Reservemodule (<http://support.automation.siemens.com/WW/view/de/29430270>)

**ET 200S im DPV1-Modus konfigurieren**

Mit PROFIBUS DPV1 stehen Ihnen erweiterte PROFIBUS-Funktionalitäten zur Verfügung.

**Voraussetzung**

- Sie befinden sich in der Netzsicht.
- Ein DP-Master mit DPV1-Funktionalität ist vorhanden.
- Eine Master-Slave-Verbindung mit PROFIBUS ist aufgebaut.

**Vorgehen**

Um den DP-Slave auf DPV1 zu schalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den DP-Slave.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Baugruppenparameter" aus der Klappliste "DP-Alarm-Mode" die Betriebsart "DPV1".

oder

1. Selektieren Sie den DP-Master.
2. Selektieren Sie in der E/A-Kommunikationstabelle die Zeile mit der Verbindung zwischen dem DP-Master und dem gewünschten DP-Slave.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Baugruppenparameter" aus der Klappliste "DP-Alarm-Mode" die Betriebsart "DPV1".

**Besonderheiten**

Es gibt Abhängigkeiten der Parameter untereinander, die im Folgenden dargestellt sind:

| Parameter                            | Betriebsart DPV0               | Betriebsart DPV1  |
|--------------------------------------|--------------------------------|---|
| Betrieb bei Soll- ungleich Istausbau | Uneingeschränkt bedienbar      | Uneingeschränkt bedienbar   |
| Diagnosealarm (OB 82)                | Nicht bedienbar, nicht gesetzt | Uneingeschränkt bedienbar   |
| Prozessalarm (OB 40 bis 47)          | Nicht bedienbar, nicht gesetzt | Uneingeschränkt bedienbar   |
| Ziehen/Stecken-Alarm (OB 83)         | Nicht bedienbar, nicht gesetzt | Nur bedienbar, wenn Adressen nicht gepackt sind.<br>Wenn Ziehen/Stecken-Alarm aktiviert wird, dann wird automatisch "Anlauf bei Soll-ungleich Istausbau" aktiviert. |

## Alarmer bei Modulen mit gepackten Adressen

Wenn das Modul Alarmer auslösen kann und die Bitadresse wegen gepackter Adressen ungleich 0 ist, dann müssen Sie im Adressdialog der ET 200S eine Diagnoseadresse vergeben.

Die Diagnoseadresse ist notwendig für die Zuordnung eines DPV1-Alarms zum Modul als Alarmauslöser. Nur wenn ein Modul diese "ungepackte" Adresse hat, kann die CPU einen Alarm zuordnen und Informationen zum Alarm in der Startinformation des Alarm-OB bzw. im Diagnosepuffer hinterlegen. Eine "gepackte" Adresse kann die CPU hierfür nicht verwenden.

Aus Sicht der Alarmverarbeitung (Alarm-OB) hat das Modul dann die zugeordnete Diagnoseadresse, für die Verarbeitung der Ein- und Ausgangsdaten im Anwenderprogramm hat das Modul die gepackten Adressen!

---

### Hinweis

Wenn die Adressen des Moduls gepackt sind, ist der Ziehen/Stecken-Alarm für die ET 200S gesperrt!

---

## GSD-Dateien verwenden

### GSD-Revisionen

#### Wissenswertes zu GSD-Revisionen

Die Eigenschaften von DP-Slaves werden durch GSD-Dateien für Projektierungstools verfügbar gemacht.

Funktionserweiterungen im Bereich der dezentralen Peripherie haben Auswirkungen auf die GSD-Spezifikation, z. B. führen sie zur Definition neuer Schlüsselwörter.

Daraus ergibt sich eine Versionierung der Spezifikation. Die Version der Spezifikation, nach der sich eine GSD-Datei richtet, heißt bei GSD-Dateien "GSD-Revision".

Die GSD-Revision ist als Schlüsselwort "GSD\_Revision" in GSD-Dateien ab GSD-Revision 1 obligatorisch enthalten. GSD-Dateien ohne dieses Schlüsselwort werden daher von Projektierungstools als GSD-Revision "0" interpretiert.

GSD-Dateien können bis GSD-Revision 5 interpretiert werden. Damit werden DP-Slaves unterstützt, die z. B. folgende Funktionen unterstützen:

- Diagnosemeldungen für Alarmblöcke
- Taktsynchronität und Äquidistanz (Isochron-Modus)
- SYNC/FREEZE
- Uhrzeitsynchronisation für DP-Slaves

## GSD-Datei installieren

### Einführung

In einer GSD-Datei (Gerätestammdaten-Datei) sind alle Eigenschaften eines DP-Slaves hinterlegt. Falls Sie einen DP-Slave projektieren möchten, der nicht im Hardware-Katalog vorhanden ist, müssen Sie die vom Hersteller mitgelieferte GSD-Datei installieren. Über GSD-Dateien installierte DP-Slaves werden im Hardware-Katalog angezeigt und können dann selektiert und konfiguriert werden.

### Voraussetzung

- Der Hardware- und Netzwerkeitor ist geschlossen.
- Sie haben Zugriff auf die benötigten GSD-Dateien in einem Verzeichnis auf der Festplatte.

### Vorgehen

Um eine GSD-Datei zu installieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien installieren".
2. Wählen Sie im Dialog "Gerätebeschreibungsdateien installieren" das Verzeichnis aus, in dem die GSD-Dateien liegen.
3. Wählen Sie aus der Liste der angezeigten GSD-Dateien eine oder mehrere Dateien aus.
4. Klicken Sie auf die Schaltfläche "Installieren".
5. Um eine Protokolldatei für die Installation zu erstellen, klicken Sie auf die Schaltfläche "Protokolldatei speichern".  
Eventuell auftretende Probleme bei der Installation lassen sich anhand der Protokolldatei nachvollziehen.

Sie finden den neuen über die GSD-Datei installierten DP-Slave in einem neuen Ordner im Hardware-Katalog.

---

#### Hinweis

Die Installation einer GSD-Datei kann nicht rückgängig gemacht werden.

---

## GSD-basierten DP-Slave konfigurieren

DP-Slaves, die Sie über die Installation einer GSD-Datei eingefügt haben, können Sie wie üblich über den Hardware-Katalog auswählen und in der Netzsicht einfügen. Wenn Sie die Baugruppen der GSD-basierten DP-Slaves einfügen wollen, müssen Sie einige Besonderheiten beachten.

### Voraussetzung

- Sie haben einen DP-Slave über eine GSD-Datei installiert.
- Sie haben die Kopfbaugruppe in der Netzsicht wie üblich eingefügt.

- Die Geräteübersicht in der Gerätesicht ist geöffnet.
- Der Hardware-Katalog ist geöffnet.

## Vorgehen

Um die Baugruppen eines GSD-basierten DP-Slaves hinzuzufügen, gehen Sie folgendermaßen vor:

1. Navigieren Sie im Hardware-Katalog zu den Baugruppen des GSD-basierten DP-Slaves. GSD-basierte DP-Slaves, auch DP-Normslaves genannt, befinden sich im Hardware-Katalog im Ordner "Weitere Feldgeräte".
2. Selektieren Sie die gewünschte Baugruppe.
3. Ziehen Sie die Baugruppe mit Drag & Drop auf einen freien Platz der Geräteübersicht.
4. Selektieren Sie die Baugruppe in der Geräteübersicht, um deren Parameter zu bearbeiten.

Sie haben die Baugruppe nun auf einen freien Steckplatz des GSD-basierten DP-Slaves gesteckt und können dessen Parameter bearbeiten.

---

### Hinweis

In dem grafischen Bereich der Gerätesicht können Sie nur den GSD-basierten DP-Slave sehen. Die hinzugefügten Baugruppen von GSD-basierten DP-Slaves finden Sie immer nur in der Geräteübersicht.

---

## Sollkonfiguration

Bei Baugruppen mit einer einstellbaren Sollkonfiguration können Sie diese im Inspektorfenster unter "Eigenschaften > Sollkonfiguration" ändern.

### 8.1.4.8 Konfigurationen für PROFINET IO

#### Wissenswertes zu PROFINET IO

#### Was ist PROFINET IO?

#### PROFINET IO

PROFINET als ethernet-basierter Automatisierungsstandard der PROFIBUS Nutzerorganisation e.V. (PNO) definiert ein herstellerübergreifendes Kommunikations-, Automatisierungs- und Engineering-Modell.

## Zielsetzung

Zielsetzung von PROFINET ist:

- eine durchgängige Kommunikation über Feldbus und Ethernet
- eine offene, verteilte Automatisierung
- die Verwendung offener Standards

## Architektur

Die PROFIBUS Nutzerorganisation e.V. (PNO) hat für die PROFINET-Architektur folgende Teilaspekte vorgesehen:

- Kommunikation zwischen Steuerungen als Komponenten in verteilten Systemen.
- Kommunikation zwischen Feldgeräten wie z. B. Peripheriegeräten und Antrieben.

## Umsetzung durch Siemens

Die Forderung nach "Kommunikation zwischen Steuerungen als Komponenten in verteilten Systemen" ist durch "Component Based Automation" (CBA) umgesetzt. Mit Component Based Automation erstellen Sie eine verteilte Automatisierungslösung auf Basis vorgefertigter Komponenten und Teillösungen.

Die Forderung nach "Kommunikation zwischen Feldgeräten" ist von Siemens mit "PROFINET IO" umgesetzt. Wie bei PROFIBUS DP ist die vollständige Projektierung und Programmierung der beteiligten Komponenten mit dem Totally Integrated Automation Portal möglich.

Um die Projektierung der Kommunikation zwischen Feldgeräten mit PROFINET IO geht es in den folgenden Abschnitten.

## Übersicht über die RT-Klassen

### RT-Klassen in PROFINET IO

PROFINET IO ist ein skalierbares Echtzeit-Kommunikationssystem auf Basis der Ethernet-Technologie. Der skalierbare Ansatz drückt sich in verschiedenen Echtzeitklassen aus:

- **RT:** Übertragung der Daten in priorisierten Ethernet-Telegrammen, nicht taktsynchron. Die benötigte Bandbreite fällt in den Bereich der freien Bandbreite für die TCP/IP-Kommunikation.
- **IRT:** Taktsynchrone Übertragung der Daten mit hoher Stabilität für zeitkritische Anwendungen (z. B. Motion-Control). Die benötigte Bandbreite fällt in den Bereich der für zyklische Daten reservierte Bandbreite.

Je nach Gerät werden nicht alle Echtzeitklassen unterstützt.

## Anbindung bestehender Bussysteme

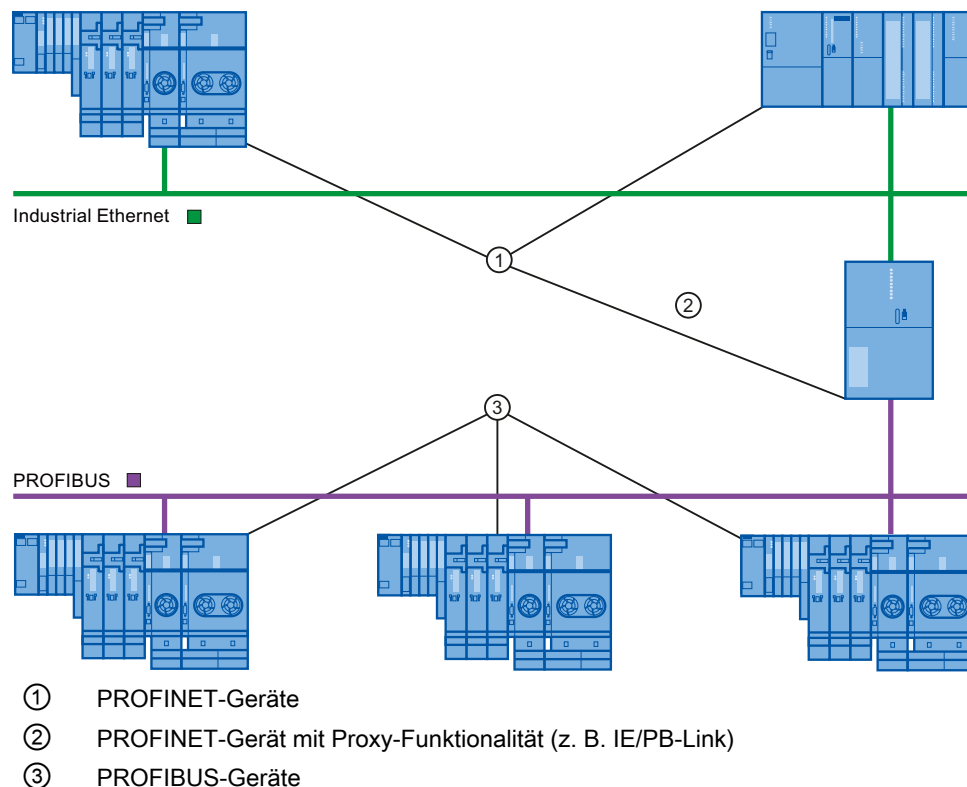
### Kopplung von PROFINET und PROFIBUS

PROFINET IO und PROFIBUS DP können über folgende Wege miteinander gekoppelt werden:

- über Industrial Ethernet:  
Zur Verbindung der beiden Netztypen Industrial Ethernet (Leitebene) und PROFIBUS (Zellebene/Feldebene) verwenden Sie z. B. das IE/PB Link.
- über Industrial Wireless LAN:  
PROFIBUS-Geräte können Sie z. B. drahtlos an PROFINET IO über ein Wireless-LAN/PB-Link koppeln. Dadurch können Sie bestehende PROFIBUS-Konfigurationen in PROFINET integrieren.

AS-Interface-Geräte können Sie mit einem IE/AS-i Link PN IO an die Schnittstelle eines PROFINET-Geräts koppeln. Dadurch können Sie das bereits bestehende AS-i-Netzwerk in PROFINET integrieren.

Das folgende Bild zeigt die Anbindung eines PROFIBUS-Subnetzes über ein PROFINET-Gerät mit Proxy-Funktionalität.



### PROFINET-Gerät mit Proxy-Funktionalität als Stellvertreter eines PROFIBUS-Geräts

Das PROFINET-Gerät mit Proxy-Funktionalität ist der Stellvertreter eines PROFIBUS-Geräts am Ethernet. Die Proxy-Funktionalität ermöglicht es, dass ein PROFIBUS-Gerät nicht nur mit seinem Master, sondern mit allen Teilnehmern am PROFINET kommunizieren kann.

Bestehende PROFIBUS-Systeme können bei PROFINET problemlos mit Hilfe der Proxy-Funktionalität in die PROFINET-Kommunikation eingebunden werden.

Wenn Sie beispielsweise ein PROFIBUS-Gerät über einen IE/PB-Link an PROFINET anbinden, nimmt der IE/PB-Link stellvertretend für die PROFIBUS-Komponenten die Kommunikation über PROFINET auf.

## Konfiguration mit IE/PB-Link PN IO

## Konfiguration mit IE/PB-Link PN IO

Um PROFIBUS DP-Konfigurationen an PROFINET IO anzubinden, können Sie das IE/PB-Link PN IO verwenden.

Aus Sicht der CPU hängen die PROFIBUS-DP-Slaves am gleichen Netz wie das IE/PB Link PN IO. Diese Slaves haben denselben Gerätenamen und IP-Adresse wie das IE/PB-Link PN IO, aber unterschiedliche Gerätenummern. Außerdem weisen sie noch eine spezifische PROFIBUS-Adresse auf.

In den Eigenschaften des IE/PB-Links werden neben den PROFINET-Gerätenummern auch die PROFIBUS-Adressen der angeschlossenen DP-Slaves angezeigt, da dieses Gerät zwei Adressierungsschemas besitzt.

## Hantieren von Gerätenummern und PROFIBUS-Adressen am Mastersystem

Beim Plazieren wird dieselbe Ziffer für die PROFINET-Gerätenummer und die PROFIBUS-Adresse vergeben.

Einen Überblick über die verwendete Gerätenummern und PROFIBUS-Adressen eines IE/PB-Links bekommen Sie im Inspektorfenster unter "Eigenschaften Allgemein > PROFINET Gerätenummer". Dort können Sie auch die Gerätenummern ändern. Außerdem können Sie einstellen, dass die Gerätenummer und die PROFIBUS-Adresse immer identisch sein sollen oder nicht. Wenn die Option "PROFINET-Gerätenummer=PROFIBUS-Adresse" aktiviert ist, entfällt für Sie das Nachführen der Gerätenummer, wenn sich die PROFIBUS-Adresse ändert.

Die PROFIBUS-Adresse können Sie in den Eigenschaften des PROFIBUS-Geräts ändern.

## Einschränkungen

Am PROFIBUS-Subnetz eines IE/PB-Link gelten für DP-Slaves in der oben beschriebenen Konfiguration folgende Einschränkungen:

- Kein IE/PB-Link steckbar
- Kein DP/PA-Link steckbar
- Kein Y-Link steckbar
- Nicht CiR-fähig
- Keine redundanten Slaves steckbar
- Keine Taktsynchronisation / Äquidistanz projektierbar
- SYNC/FREEZE-Anweisungen ("DPSYC\_FR") einer CPU am Ethernet-Subnetz für DP-Slaves hinter dem IE/PB-Link werden nicht unterstützt.



## Konfiguration mit IWLAN/PB-Link

### Maximale Anzahl Devices an einem IWLAN Segment

Wenn ein Ethernet-Subnetz als Funknetz aufgebaut ist (IWLAN = Industrial Wireless LAN), dann ist zyklischer Datenaustausch zwischen IO-Controllern und IO-Devices über eine Funkstrecke möglich.

Auf der einen Seite der Funkstrecke befinden sich fest installierte Access Points (z. B. SCALANCE W 788) und auf der anderen Seite mobilen Stationen (mit z. B. IWLAN/PB-Links mit PROFIBUS Teilnehmern).

Wenn der Aktionsradius der mobilen Stationen groß ist, müssen möglicherweise mehrere Access Points (SCALANCE W 788) installiert werden. Jeder Access Point bildet mit seiner Funkreichweite ein Segment, so dass das IWLAN aus einer Reihe von Segmenten aufgebaut ist.

Die mobilen Geräte "auf der anderen Seite" der Funkstrecke mit ihren IWLAN/PB-Links können sich entlang der Segmente bewegen.

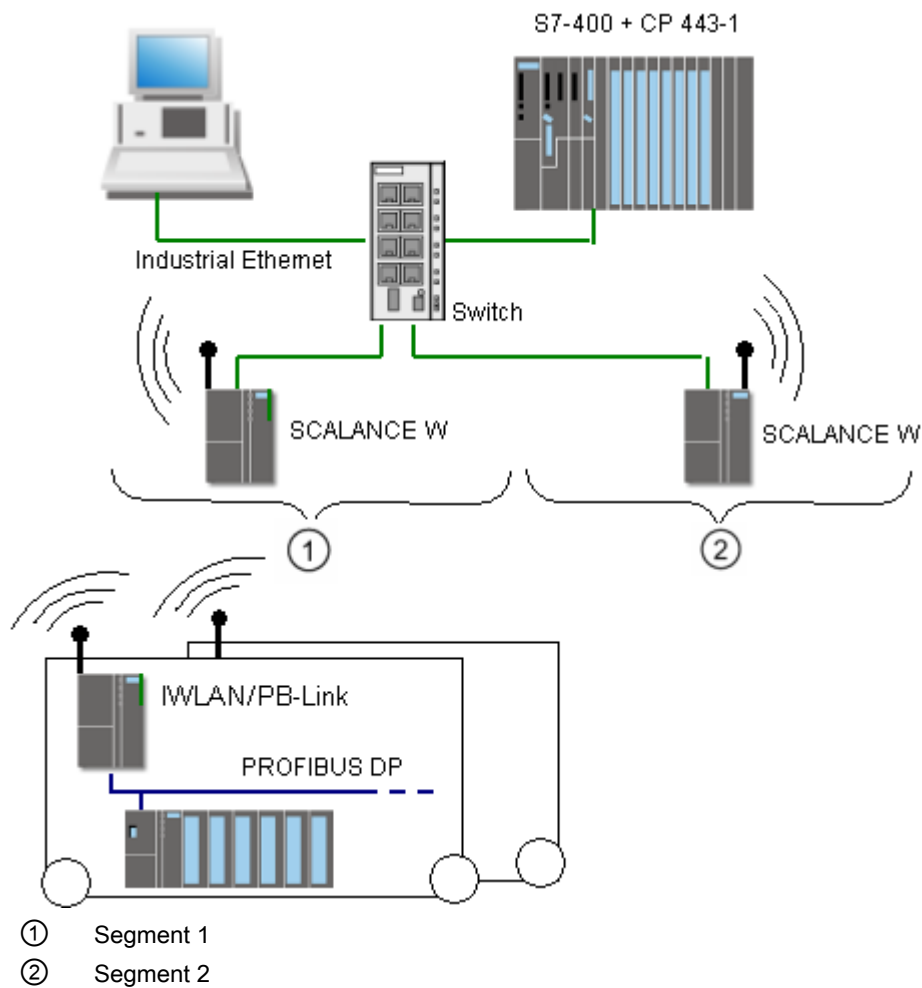
### Besonderheit

Wenn mehrere IWLAN/PB-Links sich innerhalb eines Segments befinden, dann müssen sie sich die Bandbreite, die für die Funkübertragung zur Verfügung steht, teilen. Das führt zu einer Vergrößerung der Aktualisierungszeit für diese Geräte.

### Beispiel

Im Beispiel unten befinden sich zwei IO-Devices (IWLAN/PB-Links) innerhalb eines Segments.

Wenn sich nicht mehr als maximal zwei IWLAN/PB-Links zur gleichen Zeit in einem IWLAN Segment aufhalten, dann tragen Sie eine "2" ein.



## PROFINET IO konfigurieren

### Adressierung von PROFINET-Geräten

#### Adressen- und Namensvergabe für PROFINET-Geräte

In diesem Kapitel erfahren Sie, welche Adress- und Namenskonventionen für PROFINET-Geräte gelten.

#### IP-Adressen

Alle PROFINET-Geräte beherrschen das TCP/IP-Protokoll und benötigen daher für den Betrieb am Ethernet eine IP-Adresse.

Die IP-Adressen können Sie in den Baugruppen-Eigenschaften festlegen. Wenn das Netz Teil eines bestehenden Ethernet-Firmennetzes ist, dann erfragen Sie diese Daten von Ihrem Netzwerkadministrator.

Die IP-Adressen der IO-Devices werden automatisch vergeben und normalerweise erst im Anlauf der CPU den IO-Geräten zugewiesen. Die IP-Adressen der IO-Devices haben immer dieselbe Subnetzmaske wie der IO-Controller und werden - ausgehend von der IP-Adresse des IO-Controllers - in aufsteigender Reihenfolge vergeben.

## Gerätenamen

Bevor ein IO-Device von einem IO-Controller angesprochen werden kann, muss es einen Gerätenamen haben. Bei PROFINET ist diese Vorgehensweise gewählt worden, weil Namen einfacher zu handhaben sind als komplexe IP-Adressen.

Sowohl IO-Controller als auch IO-Devices haben einen Gerätenamen. Bei aktivierter Option "PROFINET-Gerätename automatisch generieren" wird der Gerätename automatisch aus dem projektierten Namen des Geräts abgeleitet (CPU, CP oder IM):

- Der PROFINET-Gerätename setzt sich aus dem Namen des Geräts (z. B. der CPU), dem Namen der Schnittstelle (nur bei mehreren PROFINET-Schnittstellen) und optional dem Namen des IO-Systems zusammen:  
<CPU-Name>.<Name der Schnittstelle>.<IO-Systemname>  
Diesen Namen können Sie nicht direkt ändern. Sie ändern den PROFINET-Gerätenamen indirekt, indem Sie den Namen der betreffenden CPU, des CPs oder der IM in den Allgemein-Eigenschaften des Moduls ändern. Der PROFINET-Gerätename wird z. B. auch in der Liste der erreichbaren Teilnehmer angezeigt. Wenn Sie den PROFINET-Gerätenamen unabhängig vom Namen des Moduls einstellen wollen, müssen Sie die Option "PROFINET-Gerätename automatisch generieren" deaktivieren.
- Aus dem PROFINET-Gerätenamen wird ein "konvertierter Name" erzeugt. Das ist der Gerätename, der dann tatsächlich in das Gerät geladen wird.  
Der PROFINET-Gerätename wird nur dann konvertiert, wenn er nicht den Regeln von IEC 61158-6-10 entspricht. Auch diesen Namen können Sie nicht direkt ändern.

## Regeln für den konvertierten Namen

Im Folgenden sind die Regeln für den konvertierten Namen aufgelistet. Wenn der konvertierte Name sich **nicht** vom Namen der Baugruppen unterscheiden soll, dann muss der Name der Baugruppe diesen Regeln genügen.

- Der Name besteht aus einem oder mehreren Namensbestandteilen (engl: labels), die durch einen Punkt [.] getrennt sind.
- Beschränkung auf 240 Zeichen insgesamt (Kleinbuchstaben, Ziffern, Bindestrich oder Punkt)
- Ein Namensbestandteil innerhalb des Gerätenamens, d. h. eine Zeichenkette zwischen zwei Punkten, darf maximal 63 Zeichen lang sein.
- Ein Namensbestandteil besteht aus den Zeichen [a-z, 0-9].
- Der Gerätename darf nicht mit dem Zeichen "-" beginnen und auch nicht mit diesem Zeichen enden.
- Der Gerätename darf nicht mit Ziffern beginnen.
- Der Gerätename darf nicht die Form n.n.n.n haben (n = 0, ... 999).
- Der Gerätename darf nicht mit der Zeichenfolge "port-xyz" oder "port-xyz-abcde" beginnen (a, b, c, d, e, x, y, z = 0, ... 9).

### Beispiel für Gerätenamen

device-1.machine-1.plant-1.vendor

Wenn Sie diesen Namen z. B. für eine CPU vergeben, konvertiert STEP 7 ihn nicht, da er den oben beschriebenen Regeln entspricht.

### Gerätenummer

Neben dem Gerätenamen wird beim Stecken eines IO-Devices automatisch auch eine Gerätenummer vergeben, die Sie ändern können.

### Teilnehmerzahl in einem PROFINET-Subnetz

In einem PROFINET-Subnetz wird die zulässige maximale Anzahl der Teilnehmer während der Projektierung überwacht.

### Siehe auch

Gerätename und IP-Adresse zuweisen (Seite 798)

Remanenz von IP-Adressparametern und Gerätenamen (Seite 805)

### Gerätename und IP-Adresse zuweisen

#### Erstmaliges Zuweisen von IP-Adresse und Subnetzmaske bei einem IO-Controller

Dazu gibt es verschiedene Möglichkeiten: Bei der Parametrierung der PROFINET-Schnittstelle müssen Sie einstellen:

- IP-Adresse ist im Projekt eingestellt.
- IP-Adresse wird auf anderem Weg bezogen.

| Zuweisung einer IP-Adresse   | Bemerkungen   |
|--|---|
| Option: IP-Adresse im Projekt einstellen:<br>Mit Hardware-Konfiguration laden            | Mit dem Laden der Hardware-Konfiguration in den IO-Controller (z. B. CPU) wird auch die IP-Adresse und der Gerätename geladen.  |
| Option: IP-Adresse im Projekt einstellen:<br>Online über PROFINET-Schnittstelle zuweisen | Schließen Sie Ihr PG/PC am gleichen Netz an, an dem das betreffende PROFINET-Gerät angeschlossen ist. Die Schnittstelle des PGs/PCs muss auf TCP/IP (Auto) eingestellt sein. Lassen Sie sich die Erreichbaren Teilnehmer anzeigen. Wählen Sie das Zielgerät über seine MAC-Adresse aus und weisen Sie dessen projektierte IP-Adresse zu, bevor Sie die Hardware-Konfiguration inklusive der projektierten IP-Adresse laden (IP-Adresse ist dann remanent hinterlegt). |

| Zuweisung einer IP-Adresse   | Bemerkungen   |
|--|---|
| Option: IP-Adresse im Projekt einstellen:<br>Online über MPI/PROFIBUS-Schnittstelle zuweisen   | Wenn Ihr PROFINET-Gerät über eine MPI- oder PROFIBUS DP-Schnittstelle verfügt, dann schließen Sie Ihren PG/PC über die MPI- oder PROFIBUS DP-Schnittstelle direkt an das PROFINET-Gerät an. Beim Laden der Hardware-Konfiguration wird die projektierte IP-Adresse übernommen.  |
| Option "IP-Adresse auf anderem Weg beziehen":<br><ul style="list-style-type: none"> <li>• Online zuweisen</li> <li>• Zuweisen über Anwenderprogramm</li> <li>• Übergeordneter IO-Controller (nur bei I-Devices)</li> </ul> | Wenn Sie diese Option in den Eigenschaften der PROFINET-Schnittstelle gewählt haben, kann die IP-Adresse über den Online- und Diagnose-Editor, mit dem Primary Setup Tool, oder auch durch das Anwenderprogramm (Anweisung "IP_CONF") vergeben werden.<br><br>Achten Sie bei einer S7-1200-CPU darauf, dass der Zugriff auf die CPU nicht durch ein Passwort geschützt ist. Im Fall einer schreibgeschützten CPU kann keine IP-Adresse und auch kein Geräte name auf einem anderen Weg zugewiesen werden. |

### Inbetriebnahme einer PROFINET-Schnittstelle

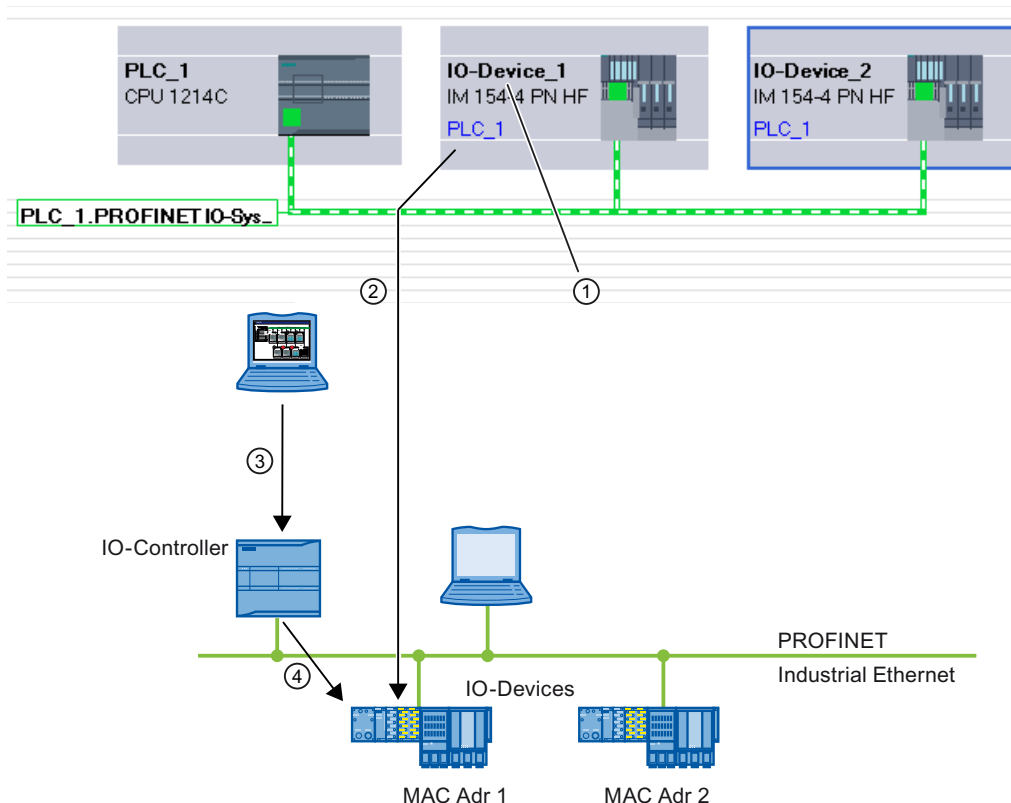
Weitere Details, wie Sie eine PROFINET-Schnittstelle in Betrieb nehmen, finden Sie auch in den Betriebsanleitungen der PROFINET-Geräte der SIMATIC-Gerätefamilie.

### Vergabe des Gerätenamens für IO-Devices bei aktivierter Option "Gerätetausch ohne Wechselmedium/PG"

Für IO-Devices mit aktivierter Option "Gerätetausch ohne Wechselmedium/PG" ist keine Vergabe des Gerätenamens im Falle eines Gerätetauschs notwendig.

### Vergabe von Gerätenamen und Adresse bei einem IO-Device (Ausnahme Gerätetausch ohne Wechselmedium/PG)

Die folgende Illustration veranschaulicht, wie die Vergabe des Gerätenamens und der Adresse abläuft.



- ① Jedes Gerät bekommt einen Namen, STEP 7 vergibt automatisch eine IP-Adresse.
- ② Aus dem Namen erzeugt STEP 7 einen PROFINET-Gerätenamen, den Sie online einem IO-Device (MAC-Adresse) zuordnen und der ins Gerät geschrieben wird.
- ③ Die Projektierung laden Sie in den IO-Controller.
- ④ Der IO-Controller vergibt im Anlauf die entsprechende IP-Adresse an das IO-Device mit dem zugewiesenen PROFINET-Gerätenamen.

Jedem IO-Device wird ein Gerätenamen zugeordnet. Sie haben die Möglichkeit, Name und IP-Adresse nachträglich manuell zu ändern.

Ihnen stehen grundsätzlich zwei Wege zur Verfügung, die projizierten Daten in das PROFINET IO-Device zu laden:

- **Offline mit Micro Memory Card:**  
Legen Sie die projektierten Daten (Gerätename z. B. turbo-3) für das IO-Device auf der Micro Memory Card im PG/PC ab. Verwenden Sie dazu den Befehl "SIMATIC Card Reader > Gerätenamen auf Memory Card schreiben" im Menü "Projekt".  
Stecken Sie dann die Micro Memory Card in das IO-Device. Das IO-Device übernimmt automatisch seinen projektierten Gerätenamen.
- **Online mit PG/PC:**  
Schließen Sie den PG/PC über die PROFINET-Schnittstelle an das Ethernet-Subnetz an. Markieren Sie in der Netzsicht das Subnetz und wählen Sie den Kontextmenübefehl "Gerätename vergeben".  
Im Dialog "PROFINET-Gerätename vergeben" wählen Sie die passende PG/PC-Schnittstelle aus, um sich mit dem Ethernet-Subnetz zu verbinden.  
In der oberen Klappliste sind alle projektierten PROFINET-Gerätenamen wählbar. Wählen Sie dort einen PROFINET-Gerätenamen aus zu markieren Sie in der unteren Tabelle das IO-Device, das diesen Gerätenamen erhalten soll. Sie können die Anzeige der Geräte in der Tabelle nach verschiedenen Kriterien filtern.  
Über die Schaltfläche "LED blinken" können Sie das Gerät leicht identifizieren.  
Der IO-Controller erkennt das IO-Device über dessen Gerätenamen und vergibt an das IO-Device automatisch die projektierte IP-Adresse.

### IP-Adressvergabe für spezielle IO-Devices

Spezielle IO-Devices, z. B. SCALANCE X, S7-300 CPs, unterstützen die Option, die IP-Adresse nicht vom IO-Controller im Anlauf zuzuweisen. In diesem Fall ist die IP-Adresse auf anderem Wege zu vergeben. Weitere Informationen finden Sie in dem Gerätehandbuch des betreffenden PROFINET-Geräts der SIMATIC-Gerätefamilie.

### Voraussetzung für weitere Verfahren der Vergabe von IP-Adresse und Gerätename

Wenn das IO-Device, wie oben beschrieben, die IP-Adresse bzw. Gerätename nicht vom IO-Controller beziehen soll, gehen Sie wie folgt vor:

1. Wählen Sie die Geräte- oder Netzsicht.
2. Öffnen Sie die Eigenschaften des betreffenden PROFINET-Geräts.
3. Wählen Sie die Option "IP-Adresse auf anderem Weg beziehen" bzw. "Gerätenamen auf anderem Weg beziehen".

### Regeln

Wird die Option "IP-Adresse / Gerätenamen auf anderem Weg beziehen" in einem PROFINET Gerät verwendet, beachten Sie Folgendes:

- Der Subnetz-Teil der IP-Adresse des IO-Devices muss mit dem Subnetz-Teil der IP-Adresse des IO-Controllers übereinstimmen.
- Das entsprechende PROFINET-Gerät kann nicht als Netzübergang verwendet werden.

### Siehe auch

Die Namensvergabe über "Erreichbare Teilnehmer" aufrufen (Seite 1004)

## Beispiel für die Vergabe des Gerätenamens

In diesem Beispiel weisen Sie einem PROFINET IO-Controller und einem PROFINET IO-Device Gerätenamen zu. Die Gerätenamen sollen zur leichten Zuordnung auch den Namen des PROFINET IO-Systems enthalten.

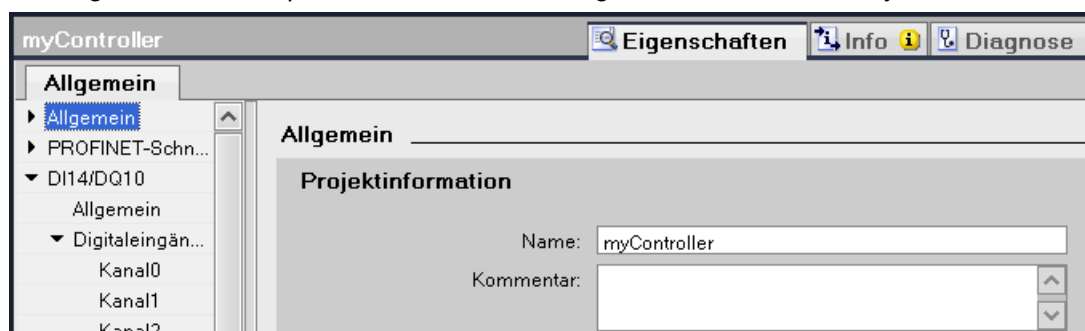
### Voraussetzung

- Sie befinden sich in der Netzsicht.
- Eine CPU 1214C (ab V2.0) ist in der Netzsicht vorhanden.
- Ein Interfacemodul IM 151-3PN ist vorhanden.
- Die PROFINET-Schnittstellen beider Baugruppen sind vernetzt.

### Vorgehen

Um die Namen zu vergeben, gehen Sie folgendermaßen vor:

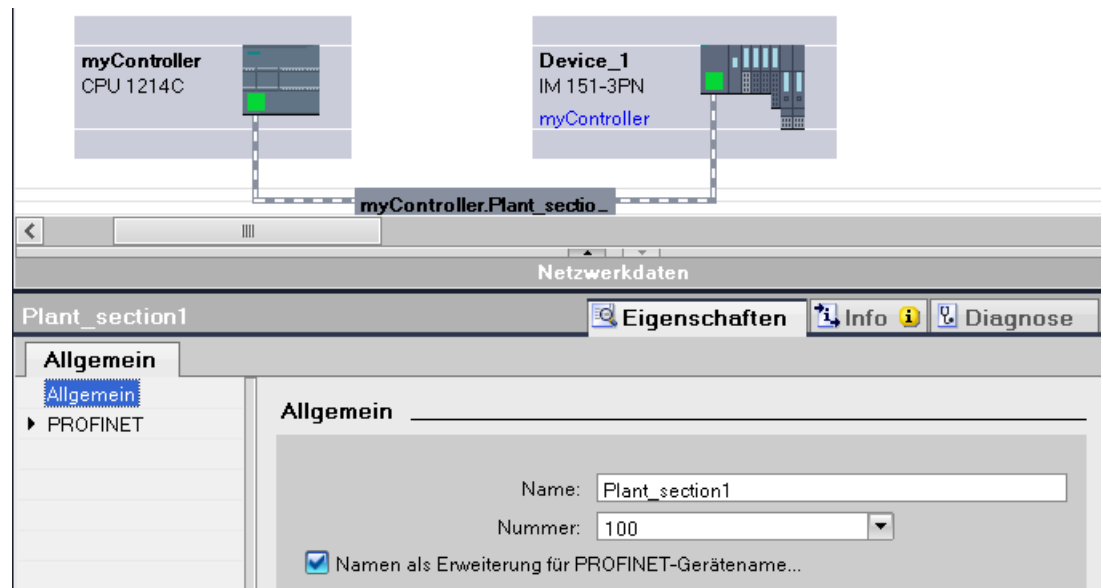
1. Selektieren Sie die CPU.  
Achten Sie darauf, dass Sie nur die CPU selektiert haben und nicht das komplette Gerät!
2. Vergeben Sie im Inspektorfenster, Bereich "Allgemein", den Namen "myController".



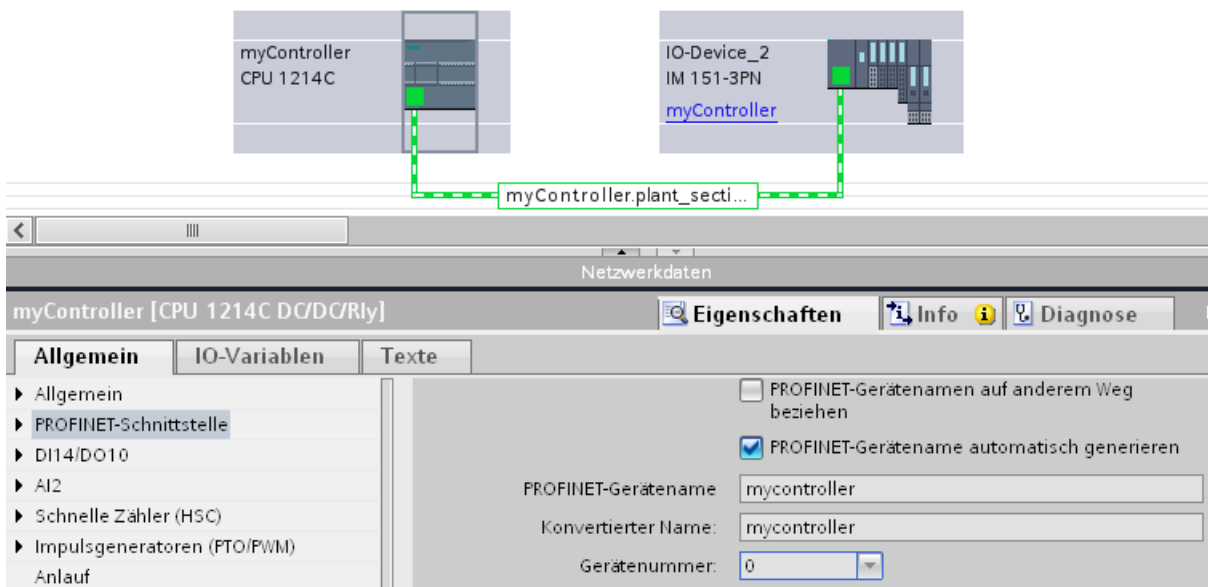
3. Selektieren Sie das Interfacemodul.  
Achten Sie darauf, dass Sie nur das Interfacemodul selektiert haben und nicht das komplette Gerät ET 200S!
4. Vergeben Sie im Inspektorfenster, Bereich "Allgemein", den Namen "Device\_1"
5. Klicken Sie mit der rechten Maustaste auf das PROFINET IO-System und wählen sie den Befehl "Eigenschaften".



6. Vergeben Sie für das IO-System den Namen "Plant\_section1" und aktivieren Sie das Optionskästchen "Namen als Erweiterung für PROFINET-Gerätenamen verwenden".



7. Die automatisch gebildeten PROFINET-Gerätenamen finden Sie bei markiertem Gerät im Inspektorfenster, Bereich "PROFINET-Schnittstelle".



Der PROFINET-Gerätename entspricht dem Namen der Baugruppe (hier mit Erweiterung um den Namen des IO-Systems) mit dem Unterschied, dass nur Kleinschreibung verwendet wird. Hintergrund: Bei der Speicherung des Namens wird nicht zwischen Groß- und Kleinschreibung unterschieden ("case insensitive"). Wenn Sie den Gerätenamen unabhängig vom Namen der Baugruppe bestimmen wollen, müssen Sie die Option "PROFINET-Gerätename automatisch generieren" deaktivieren. In diesem Fall ist der PROFINET-Gerätename editierbar. Darunter wird der konvertierte Name angezeigt. Das ist der Name, der automatisch aus dem PROFINET-Gerätenamen erzeugt wird und den DNS-Konventionen genügt. Wenn

Sie mit STEP 7 arbeiten, benötigen Sie diesen Namen nicht. Dieser Name wird hier zur Kontrolle angezeigt und entspricht dem Namen, wie er im Gerät gespeichert wird. Wenn Sie mit anderen Tools arbeiten, die in der Lage sind, den Datenaustausch aufzuzeichnen und den tatsächlichen Gerätenamen auslesen, dann finden Sie den konvertierten Namen vor.

### Weitere Besonderheiten

Bei PROFINET-Geräten mit mehreren PROFINET-Schnittstellen wird der Name der Schnittstelle an den Namen der Baugruppen angehängt, durch einen Punkt getrennt.

Beispiel:

- Name der Baugruppe: myController
- Name der Schnittstelle: Interface\_1
- PROFINET-Gerätename: mycontroller.interface\_1

### Gerätenamen zuweisen über Memory Card

#### Einführung

Sie können den Gerätenamen von PROFINET IO-Devices offline projektieren. Dazu speichern Sie einen projektierten Gerätenamen auf ein Memory Card und stecken anschließend die Memory Card in das dafür vorgesehene IO-Device.

Wenn im Fall eines Gerätedefekts ein IO-Device komplett getauscht werden muss, führt der IO-Controller automatisch eine Parametrierung und Konfigurierung des neuen Gerätes durch. Mit der Memory Card ist ein Gerätetausch ohne PG möglich.

#### Voraussetzung

- Das PG verfügt über einen Kartenleser für Memory Cards.
- Das IO-Device muss das Zuweisen des Gerätenamens über Memory Card unterstützen.
- Die Station mit ihrem PROFINET IO-System ist projektiert.

#### Vorgehen

Um einen Gerätenamen auf einer Memory Card zu speichern, gehen Sie folgendermaßen vor:

1. Stecken Sie die Memory Card in den Kartenleser.
2. Selektieren Sie das IO-Device, dem der Gerätename über Memory Card zugewiesen werden soll.
3. Wählen Sie im Menü "Projekt" den Befehl "Card Reader > Gerätenamen auf Micro Memory Card speichern".

Wenn die Memory Card nicht leer ist, werden Sie durch eine Meldung auf diesen Umstand aufmerksam gemacht und können die Memory Card vor dem Speichern löschen.

## **Remanenz von IP-Adressparametern und Gerätenamen**

Die Remanenz von IP-Adressparametern (IP-Adresse, Subnetzmaske, Routereinstellung) und Geräteiname ist davon abhängig, wie die Adressvergabe erfolgt.

Die nicht remanente temporäre Vergabe bedeutet:

- IP-Adressparameter und Geräteiname bleiben über folgenden Zeitraum gültig:
  - bis zum nächsten NETZ-AUS
  - bis zum nächsten Urlöschen
  - bis zum Beenden der Online-Verbindung (z. B. nach Laden des Programms)  
Nach NETZ-AUS / NETZ-EIN bzw. Urlöschen ist die CPU nur noch über die MAC-Adresse erreichbar.

Wenn die IP-Adressparameter nicht remanent sind, kann nach den oben genannten Ereignissen (z. B. nach NETZ-AUS/NETZ-EIN) keine Kommunikation mehr stattfinden, die auf dem IP-Protokoll basiert.

Die Vergabe einer temporären IP-Adresse löscht auch remanent gespeicherte IP-Adressparameter.

## **IP-Adressparameter und Geräteiname nicht remanent vergeben**

IP-Adressparameter und Geräteiname sind in folgenden Fällen nicht remanent:

- Eine temporäre IP-Adresse, die nicht remanent ist, wird im Fall der Funktion "Erreichbare Teilnehmer" implizit vergeben, wenn das Gerät (z. B. CPU) noch keine IP-Adresse hat.
- Das Gerät ist ein "normaler" IO-Controller (d. h. kein I-Device) und im Anwenderprogramm (Anweisung "IP\_Conf") ist festgelegt, dass die IP-Adressparameter/Geräteiname nicht remanent sein sollen.

## **IP-Adressparameter und Geräteiname remanent vergeben**

IP-Adressparameter und Geräteiname sind in folgenden Fällen remanent:

- In den Eigenschaften der PROFINET-Schnittstelle ist eingestellt, dass die IP-Adresseparameter im Projekt eingestellt sind (Option "IP-Adresse im Projekt einstellen").
- In den Eigenschaften der PROFINET-Schnittstelle ist eingestellt, dass die IP-Adresse auf anderem Weg bezogen werden soll.
  - Nach dem Laden der Projektierung werden über STEP 7 oder ein Setup-Tool wie PST die IP-Adressparameter bzw. der Geräteiname vergeben (STEP 7: Online- und Diagnosefunktion "IP-Adresse zuweisen"). Die zugewiesenen IP-Adressparameter sind remanent.
  - Das Gerät ist ein "normaler" IO-Controller (d. h. kein I-Device) und im Anwenderprogramm (Anweisung "IP\_Conf") ist festgelegt, dass die IP-Adressparameter/Geräteiname remanent sein sollen.

### Besonderheiten beim I-Device

In den Eigenschaften der PROFINET-Schnittstelle des I-Device ist eingestellt, dass die IP-Adressparameter auf anderem Weg bezogen werden sollen. Die IP-Adressparameter für das I-Device werden durch den übergeordneten IO-Controller zugewiesen.

- Wenn priorisierter Hochlauf eingestellt ist, sind die IP-Adressparameter remanent.
- Wenn **kein** priorisierter Hochlauf eingestellt ist, sind die IP-Adressparameter nicht remanent.

### Empfehlung

Verwenden Sie, wenn möglich, die Option "IP-Adresse im Projekt einstellen" und legen Sie eine geeignete IP-Adresse fest. In diesem Fall ist die IP-Adresse remanent vergeben.

### Rücksetzen von remanenten IP-Adressparametern und Gerätenamen

Remanente IP-Adressparameter und Gerätenamen werden durch die Online- und Diagnosefunktion "Rücksetzen auf Werkseinstellungen" zurückgesetzt.

---

#### Hinweis

#### Folgen der erneuten Vergabe von IP-Adressparametern auf bestehende IP-Adressparameter

- Durch die temporäre Vergabe von IP-Adressparametern / Gerätenamen erfolgt ein Rücksetzen von ggf. remanent gespeicherten IP-Adressparametern / Gerätenamen.
- Bei einer festen Vergabe von IP-Adressparametern / Gerätenamen werden vorher remanent gespeicherte Parameter durch die neu vergebenen Parameter ersetzt.

---

#### Hinweis

#### Wiederverwendung von Geräten

Führen Sie "Rücksetzen auf Werkseinstellungen" durch, bevor Sie ein Gerät mit remanenten IP-Adressparametern / Gerätenamen in andere Subnetze / Anlagen einbauen oder ins Lager legen.

---

### PROFINET IO-System anlegen

Ein PROFINET IO-System besteht aus einem PROFINET IO-Controller und seinen zugeordneten PROFINET IO-Devices.

Zum Anlegen eines PROFINET IO-Systems benötigen Sie einen IO-Controller (z. B. CPU 1214C) und ein- oder mehrere IO-Devices (z. B. eine Kopfbaugruppe der dezentralen Peripheriefamilie ET 200S).

Sobald Sie einen IO-Controller mit einem IO-Device verbinden, erfolgt eine Controller-Device-Kopplung.

## Vorgehen

Um ein PROFINET IO-System anzulegen, gehen Sie folgendermaßen vor:

1. Ziehen Sie per Drag & Drop einen IO-Controller aus dem Hardware-Katalog (z. B. CPU 1214C) in die freie Fläche der Netzsicht.  
Der IO-Controller ist im Projekt angelegt.
2. Ziehen Sie per Drag & Drop ein IO-Device aus dem Hardware-Katalog (z. B. ET 200S) in die freie Fläche der Netzsicht.
3. Klicken Sie auf die PROFINET-Schnittstelle des IO-Controllers oder des IO-Devices.
4. Ziehen Sie mit gedrückter Maustaste eine Verbindung von der selektierten Schnittstelle auf die Schnittstelle des Partnergeräts.  
Es wird ein Subnetz mit einem IO-System zwischen IO-Controller und IO-Device erzeugt.
5. Passen Sie gegebenenfalls die Eigenschaften des Ethernet-Subnetzes oder des IO-Controllers (z. B. IP-Adresse) im Inspektorfenster unter "Eigenschaften" an.

## PROFINET IO-System hantieren

Über die Kontextmenübefehle haben Sie die Möglichkeit, PROFINET IO-Systeme in der Netzsicht zu löschen, neu anzulegen oder auch die Schnittstelle mit einem anderen Subnetz zu verbinden.

Auf diese Weise korrigieren Sie eine bestehende PROFINET-Konfiguration in der Netzsicht.

## Neues PROFINET IO-System für IO-Controller anlegen

Um für einen IO-Controller ein neues PROFINET IO-System anzulegen, gehen Sie folgendermaßen vor:

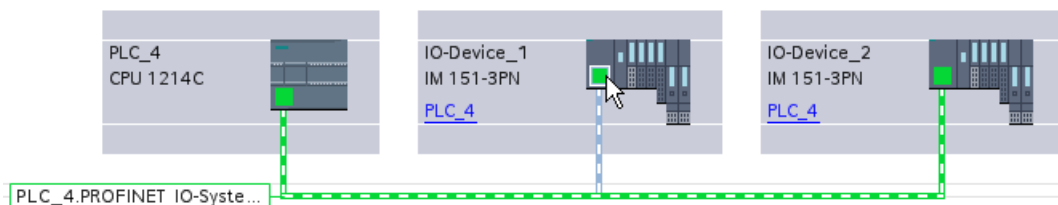
1. Vergewissern Sie sich, dass dem IO-Controller kein IO-System zugewiesen ist. Falls dem IO-Controller bereits ein IO-System zugewiesen ist, dann ist der Kontextmenübefehl "IO-System zuweisen" nicht aktiv.
2. Selektieren Sie die PROFINET-Schnittstelle und wählen Sie den Kontextmenübefehl "IO-System zuweisen".

Ein neues PROFINET IO-System wird am IO-Controller angelegt und sie können IO-Devices diesem IO-System zuweisen.

### PROFINET IO-Devices vom PROFINET IO-System trennen

Um bereits vernetzte PROFINET IO-Devices von ihrem PROFINET IO-System zu trennen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die PROFINET-Schnittstelle eines IO-Devices.



2. Wählen Sie den Kontextmenübefehl "Vom IO-System trennen".  
Das IO-Device, das diesem IO-System zugewiesen war, ist danach nicht mehr zugeordnet.



Sie haben die Möglichkeit, ein neues IO-System anzulegen und die nicht zugeordneten IO-Devices jeweils einem IO-Controller zuzuweisen.

### PROFINET IO-Devices anderen IO-Controllern zuweisen

Bestehende PROFINET IO-Systeme können Sie in der Netzsicht einfach umkonfigurieren:

1. Selektieren Sie die Schnittstelle eines IO-Devices und wählen Sie das Kontextmenü. Hier haben Sie folgende Möglichkeiten:
  - Das IO-Device einem neuen Subnetz zuweisen oder vom bestehenden Subnetz trennen
  - Das IO-Device einem neuen IO-Controller zuweisen
  - Das IO-Device einem neuen IO-System zuweisen oder vom bestehenden IO-System trennen.
2. Um das IO-Device einem andern IO-Controller zuzuweisen, wählen Sie den Kontextmenübefehl "Neuem IO-Controller zuweisen".  
Im Fall einer fehlenden Vernetzung wird automatisch ein Subnetz angelegt und das IO-Device dem IO-System des neuen IO-Controllers zugewiesen.

#### Tipp: Schnelles Konfigurieren von IO-Systemen

Wenn das IO-System viele IO-Devices hat, weisen Sie alle per Drag & Drop platzierten IO-Devices in einem Schritt einem IO-Controller zu.

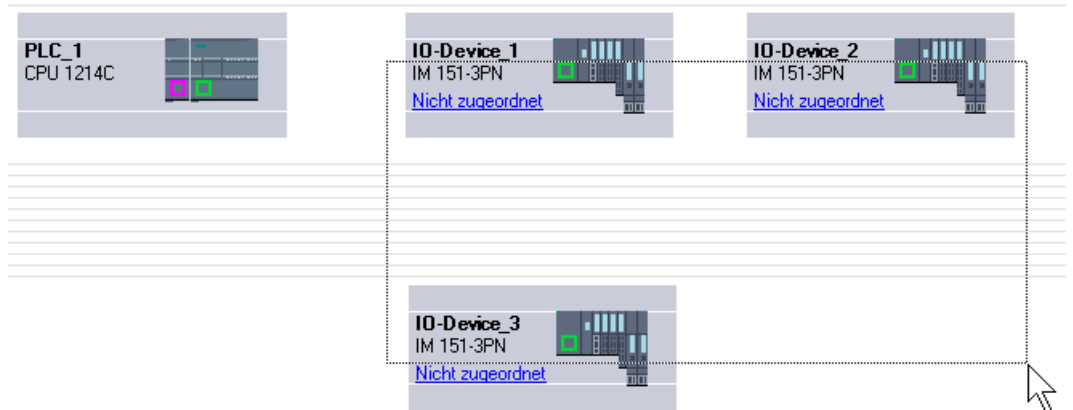
#### Voraussetzung

IO-Controller und IO-Devices sind in der Netzsicht platziert.

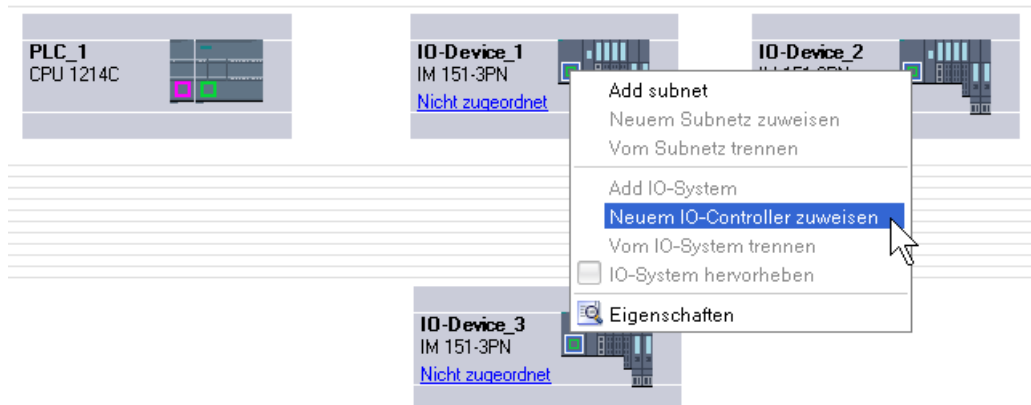
## IO-Devices einem IO-System zuweisen

Gehen Sie folgendermaßen vor:

1. Wählen Sie einen passenden Zoomfaktor, so dass Sie möglichst viele IO-Devices in der Netzsicht sehen.
2. Ordnen Sie die IO-Devices in maximal zwei Reihen an.
3. Wählen Sie mit dem Mauszeiger alle IO-Schnittstellen aus (nicht die Geräte!). Das gelingt nur, wenn Sie außerhalb des ersten IO-Devices mit dem Ziehen des Mauszeigers beginnen und beim letzten IO-Device die Maustaste loslassen (Auswahl mit dem Lasso).



4. Wählen Sie das Kontextmenü "Neuem IO-Controller zuweisen" und markieren Sie im Folgedialog die entsprechende IO-Schnittstelle des IO-Controllers.



5. Die IO-Devices werden automatisch mit dem IO-Controller vernetzt und bilden mit ihm zusammen ein IO-System.

### Hinweis

Bei einem hervorgehobenem IO-System können Sie im Hardware-Katalog auf ein IO-Device doppelklicken und damit schnell weitere IO-Devices hinzuzügeln. Das IO-Device wird dadurch automatisch in das hervorgehobene IO-System aufgenommen.

## Ports verschalten

Wenn ein IO-Device einem IO-Controller zugewiesen ist, dann ist damit noch nicht festgelegt, wie die Ports untereinander verschaltet sind.

Eine Portverschaltung ist nicht notwendig, um die PROFINET-Funktionalität nutzen zu können, bietet aber folgende Vorteile:

- Mit der Portverschaltung wird eine Solltopologie vorgegeben. Anhand eines Online-Offline-Vergleichs ist es möglich, einen Soll-Ist-Vergleich durchzuführen bei den Geräten, die diese Funktionalität unterstützen.
- Nur bei IRT-Kommunikation: Wenn eine Portverschaltung projektiert ist, kann STEP 7 präziser die notwendige Bandbreite ermitteln. Das führt in der Regel zu einer höheren Performance.

Beachten Sie, dass durch die Verschaltung von Ports keine unzulässigen Ringstrukturen entstehen.

Sinnvoll ist die Portverschaltung nur für Geräte, die die Topologieprojektierung unterstützen.

## Ports im Inspektorfenster verschalten

Um Ports miteinander zu verschalten, gehen Sie folgendermaßen vor:

1. Selektieren Sie das PROFINET-Gerät oder die PROFINET-Schnittstelle.
2. Navigieren Sie zur Port-Eigenschaft "Portverschaltung".  
Bei markierter PROFINET-Schnittstelle ist diese Einstellung wie folgt im Inspektorfenster zu finden: Eigenschaften > Allgemein > Erweiterte Optionen > Port [...] > Portverschaltung.
3. Im Abschnitt "Lokaler Port" sind die Einstellungen am lokalen Port zu finden. Für Lichtwellenleiter sind hier die z. B. Kabelbezeichnungen einstellbar.  
Im Abschnitt "Partnerport" klicken Sie auf das schwarze Dreieck im Feld "Partnerport", um sich die verfügbaren Partnerports anzeigen zu lassen und auszuwählen.
4. Wenn es sich um eine Portverschaltung mit dem Medium Kupfer handelt und die Geräte IRT-Kommunikation unterstützen, können Sie noch Leitungslänge bzw. Signallaufzeit einstellen.

Wenn die PROFINET-Schnittstelle nicht vernetzt war, wird sie durch diese Aktion automatisch vernetzt. In den Eigenschaften des Subnetzes können Sie einstellen, ob dieses Subnetz für die Vernetzung herangezogen werden soll oder nicht.

## Siehe auch

Übersicht (Seite 502)



## Sendetakt einstellen

### Voraussetzungen zum Ändern des Sendetakts am PROFINET-Gerät

Es darf kein IRT (Isochronous Realtime) projektiert sein, das bedeutet im Einzelnen:

- Es darf am IO-System kein Gerät als Sync-Slave oder Sync-Master projektiert sein.
- Alle Geräte am IO-System müssen unsynchronisiert sein.

Wenn IRT projektiert ist, d. h. wenn der IO-Controller als Sync-Master projektiert ist, läßt sich der Sendetakt nur in der Sync-Domain einstellen.

## Vorgehen

Um den Sendetakt am PROFINET-Gerät einzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den PROFINET IO-Controller in der Geräte- oder Netzsicht.
2. Ändern Sie in den Eigenschaften der PROFINET-Schnittstelle den Wert für das kleinst mögliche Aktualisierungsintervall unter "PROFINET-Schnittstelle > Erweiterte Optionen > Echtzeit-Einstellungen > IO-Kommunikation > Sendetakt".

Der Sendetakt ist für alle PROFINET-Geräte am IO-System gültig. Wenn die Synchronisationsrolle auf einen anderen Wert als "Unsynchronisiert" einstellen, dann können Sie den Sendetakt nur noch in der Sync-Domain, d. h. zentral am PROFINET IO-System einstellen.

## Aktualisierungszeit einstellen

### Aktualisierungszeit

Innerhalb dieses Zeitintervalls wird ein IO-Device/IO-Controller im PROFINET IO-System vom IO-Controller/IO-Device mit neuen Daten versorgt. Die Aktualisierungszeit kann für jedes IO-Device separat projektiert werden und bestimmt den Zeitabstand, in dem Daten vom IO-Controller zum IO-Device (Ausgänge) sowie Daten vom IO-Device zum IO-Controller (Eingänge) gesendet werden.

In der Voreinstellung berechnet STEP 7 die Aktualisierungszeit automatisch für jedes IO-Device des PROFINET IO-Systems unter Berücksichtigung des auszutauschenden Datenvolumens und des eingestellten Sendetakts.

### Aktualisierungszeit einstellen

Wenn Sie die Aktualisierungszeit nicht automatisch berechnen lassen möchten, können Sie die Einstellung ändern.

Um die Aktualisierungszeit zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht oder in der Gerätesicht die PROFINET-Schnittstelle des IO-Devices.
2. Ändern Sie in den Eigenschaften der Schnittstelle unter "Erweiterte Optionen > Echtzeit-Einstellungen > IO-Zyklus" die Aktualisierungszeit.
  - Um eine passende Aktualisierungszeit automatisch berechnen zu lassen, wählen Sie "Automatisch".
  - Um die Aktualisierungszeit selbst zu wählen, wählen Sie "Einstellbar" und geben Sie die gewünschte Aktualisierungszeit in ms an.
3. Um die Konsistenz zwischen Sendetakt und Aktualisierungszeit sicherzustellen, aktivieren Sie die Option "Aktualisierungszeit bei Änderung des Sendetakts anpassen".  
Mit dieser Option ist sichergestellt, dass die Aktualisierungszeit nicht kleiner eingestellt ist als der Sendetakt.

Die nicht-automatische Einstellung des Sendetakts kann zu Fehlern führen, wenn die zur Verfügung stehende Bandbreite nicht ausreichend ist oder wenn andere Grenzen überschritten wurden (z. B. zu viele Teilnehmer projektiert sind).

## Ansprechüberwachungszeit einstellen

### Ansprechüberwachungszeit

Für PROFINET IO-Devices können Sie eine Ansprechüberwachungszeit projektieren.

Wenn das IO-Device nicht innerhalb der Ansprechüberwachungszeit vom IO-Controller mit Ein- oder Ausgangsdaten (IO-Daten) versorgt wird, schaltet es in den sicheren Zustand.

Sie geben die Ansprechüberwachungszeit nicht direkt ein, sondern als "Anzahl akzeptierter Aktualisierungszyklen ohne IO-Daten". Das macht die Einstellung einfacher, weil die Aktualisierungszeit je nach Leistungsfähigkeit des IO-Devices bzw. je nach Einstellung kürzer oder länger sein kann.

Aus der "Anzahl akzeptierter Aktualisierungszyklen ohne IO-Daten" wird die resultierende Ansprechüberwachungszeit automatisch berechnet.

### Ansprechüberwachungszeit projektieren

Um die Ansprechüberwachungszeit festzulegen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht oder in der Gerätesicht die PROFINET-Schnittstelle des IO-Devices.
2. Navigieren Sie in den Eigenschaften der Schnittstelle zu "Erweiterte Optionen > Echtzeit-Einstellungen > IO-Zyklus".
3. Wählen Sie aus der Klappliste "Ansprechüberwachung nach # Zyklen mit fehlenden IO-Daten auslösen" die gewünschte Anzahl Zyklen aus.

Die Ansprechüberwachungszeit wird anschließend auf Grund des vorgegebenen Faktors automatisch berechnet. Sie darf nicht mehr als 1,92 Sekunden betragen.

---

**Hinweis**

Die Voreinstellung sollten Sie nur im Ausnahmefall ändern, z. B. in der Inbetriebnahmephase.

---

## Berechnete Bandbreite für zyklische IO-Daten

### Berechnete Bandbreite für zyklische IO-Daten

Das System überwacht die Einhaltung der maximal zur Verfügung stehenden Bandbreite für zyklische IO-Daten. Die maximale Bandbreite ist vom Sendetakt abhängig. Bei Sendetakten größer oder gleich 1 ms beträgt die maximale Bandbreite 0,5 ms. Bei kleineren Sendetakten verringert sich die maximal zur Verfügung stehende Bandbreite.

Die tatsächlich benötigte Bandbreite für zyklische IO-Daten wird vom System auf Basis der Anzahl der projektierten IO-Geräte und IO-Baugruppen bestimmt. Des Weiteren ist die benötigte Bandbreite von der verwendeten Aktualisierungszeit abhängig.

Im Allgemeinen erhöht sich die berechnete Bandbreite bei:

- einer größeren Anzahl von IO-Geräten
- einer größeren Anzahl von IO-Baugruppen
- kleineren Aktualisierungszeiten.

### Maximale Bandbreite für zyklische IO-Daten in Abhängigkeit vom Sendetakt

Die folgende Tabelle zeigt, wie sich die maximal zur Verfügung stehende Bandbreite für zyklische IO-Daten in Abhängigkeit zum Sendetakt verhält:

| Sendetakt          | Maximale Bandbreite für zyklische IO-Daten |
|--------------------|--|
| 250 µs – 468,75 µs | << 125 µs                                  |
| 500 µs – 968,75 µs | = Sendetakt / 2                            |
| 1 – 4 ms           | = 500 µs                                   |

## Port-Optionen festlegen

### Festlegen der Port-Optionen

### Verbindungseinstellungen für den PROFINET-IO-Port ändern

Bei Bedarf können Sie die Netzwerkeinstellungen für den PROFINET-IO-Port ändern. Standardmäßig werden die Einstellungen automatisch festgelegt, was im Normalfall eine problemlose Kommunikation gewährleistet.

## Einstellmöglichkeiten Übertragungsmedium/Duplex

Abhängig von dem gewählten Gerät können Sie folgende Einstellungen für "Übertragungsmedium/Duplex" vornehmen:

- Automatische Einstellung  
Empfohlene Default-Einstellung des Ports. Die Übertragungseinstellungen mit dem Partnerport werden automatisch "ausgehandelt". In der Voreinstellung ist automatisch auch die Option "Autonegotiation aktivieren" aktiviert.
- TP/ITP mit x Mbit/s Vollduplex (Halbduplex)  
Einstellung der Übertragungsgeschwindigkeit und des Vollduplex/Halbduplex-Betriebs. Die Wirkung ist abhängig von der eingestellten Option "Autonegotiation aktivieren":
  - Autonegotiation aktiviert  
Sie können sowohl Crosskabel als auch Patchkabel verwenden.
  - Autonegotiation deaktiviert  
Achten Sie darauf, dass Sie das korrekte Kabel verwenden (Patch- oder Crosskabel)! Auch bei dieser Einstellung wird der Port überwacht.
- Deaktiviert  
Abhängig vom Baugruppentyp kann in der Klappliste die Option "deaktiviert" angeboten werden. Sie haben damit beispielsweise die Möglichkeit, den Zugang zu einem unbenutzten Port aus Sicherheitsgründen zu unterbinden. Bei dieser Einstellung werden keine Diagnoseereignisse erzeugt.

### Option "Überwachen"

Mit dieser Option wird die Port-Diagnose aktiviert bzw. deaktiviert. Beispiele für Port-Diagnose: Der Link-Status wird überwacht, d. h. Diagnose bei Link-Down erzeugt und bei Fiber Optic Ports wird die Systemreserve überwacht.

### Option "Autonegotiation aktivieren"

Die Einstellung für Autonegotiation ist nur änderbar, wenn ein konkretes Medium (z. B. TP 100 Mbit/s Vollduplex) ausgewählt ist. Ob ein konkretes Medium eingestellt werden kann, hängt von den Eigenschaften des Moduls ab.

Ist die Autonegotiation deaktiviert, dann wird die feste Einstellung des Ports erzwungen, ähnlich wie das z. B. bei einem priorisierten Hochlaufs des IO-Devices erforderlich ist.

Sie müssen für gleiche Einstellungen beim Partnerport sorgen, da bei dieser Option die Betriebsparameter des angeschlossenen Netzes nicht erkannt werden und entsprechend die Datenübertragungsgeschwindigkeit und der Übertragungsmodus nicht optimal eingestellt werden können.

---

#### Hinweis

STEP 7 übernimmt bei einem verschalteten lokalen Port die Einstellung für den Partnerport, falls der Partnerport die Einstellung unterstützt. Falls der Partnerport die Einstellung nicht unterstützt, wird eine Fehlermeldung erzeugt.

---

## Verkabelungsregeln bei deaktivierter Autonegotiation

### Voraussetzung

Sie haben z. B. zur Beschleunigung der Hochlaufzeit des IO-Devices für den betreffenden Port folgende Einstellungen vorgenommen:

- Feste Übertragungsgeschwindigkeit
- Autonegotiation inkl. Autocrossing deaktiviert

Die Zeit für das Aushandeln der Übertragungsgeschwindigkeit beim Hochlauf wird somit eingespart.

Wenn Sie Autonegotiation deaktiviert haben, müssen Sie die Verkabelungsregeln beachten.

## Verkabelungsregeln bei deaktivierter Autonegotiation

PROFINET-Geräte besitzen folgende zwei Arten von Ports:

| Art des Ports                                  | PROFINET-Geräte   | Bemerkung  |
|--|---|--|
| Switchport mit gekreuzter Pinbelegung          | Bei IO-Devices: Port 2<br>Bei S7-CPU's mit 2 Ports: Port 1 und Port 2 | Gekreuzte Pinbelegung bedeutet, dass die Pinbelegung der Ports für Senden und Empfangen zwischen den betreffenden PROFINET-Geräten intern vertauscht werden. |
| Endgeräteport mit nicht gekreuzter Pinbelegung | Bei IO-Devices: Port 1<br>Bei S7-CPU's mit einem Port: Port 1         | -  |

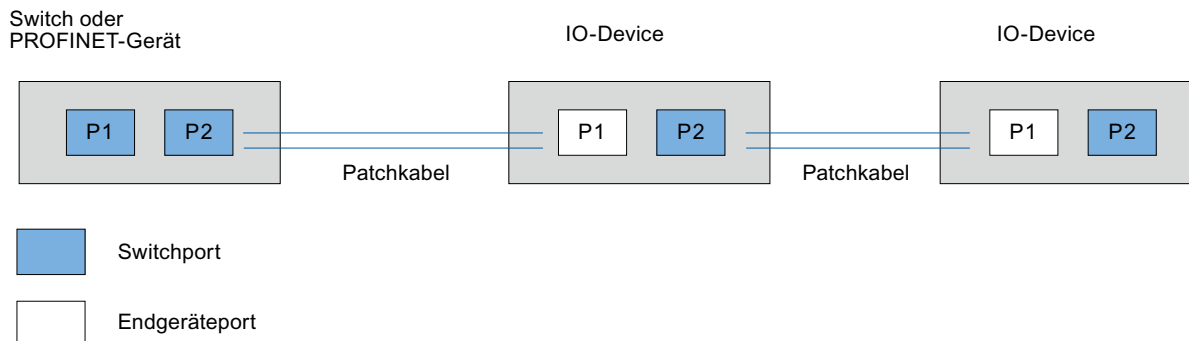
### Gültigkeit der Verkabelungsregeln

Die im folgenden Absatz beschriebenen Verkabelungsregeln gelten ausschließlich für den Fall, dass Sie eine feste Porteinstellung vorgenommen haben.

### Regeln für die Verkabelung

Mehrere IO-Devices können Sie mit einem Patchkabel (Eins-zu-eins-Verdrahtung beider Stecker) in Linie verbinden. Dazu verbinden Sie Port 2 (P2) des IO-Devices mit dem Port 1

(P1) des nächsten IO-Devices. In der folgenden Grafik ist ein Beispiel mit zwei IO-Devices dargestellt.



## Begrenzungen am Port

### Voraussetzung

Um mit Boundaries ("Begrenzungen") arbeiten zu können, muss das jeweilige Gerät mehr als einen Port besitzen. Wenn das PROFINET keine Boundaries-Einstellungen unterstützt, werden sie auch nicht angezeigt.

### Boundaries aktivieren

Unter "Boundaries" versteht man Grenzen für die Übertragung bestimmter Ethernet-Frames. Es können folgende Begrenzungen an einem Port gesetzt werden:

- "Ende der Erfassung erreichbarer Teilnehmer"  
DCP-Frames zur Erfassung erreichbarer Teilnehmer werden nicht weitergeleitet. Hinter diesem Port liegende Teilnehmer werden nicht mehr unter "erreichbare Teilnehmer" in der Projektnavigation angezeigt. Hinter diesem Port liegende Teilnehmer können von der CPU nicht mehr erreicht werden.
- "Ende der Topologieerkennung"  
LLDP-Frames (Link Layer Discovery Protocol) zur Topologieerkennung werden nicht weitergeleitet.
- "Ende der Sync-Domain"  
Sync-Frames, die zur Synchronisierung von Teilnehmern innerhalb einer Sync-Domain übertragen werden, werden nicht weitergeleitet.  
Wenn Sie z. B. ein PROFINET-Gerät mit mehr als zwei Ports in einem Ring betreiben, dann sollten Sie das Einspeisen von Sync-Frames in den Ring durch Setzen einer Sync-Boundary (an den Ports, die nicht im Ring sind) verhindern.  
Weiteres Beispiel: Möchten Sie mehrere Sync-Domains nutzen, dann projektieren Sie eine Sync-Domain-Boundary für den Port, der mit einem PROFINET-Gerät der jeweils anderen Sync-Domain verbunden ist.

## Einschränkungen

Folgende Einschränkungen sind zu beachten:

- Nur wenn der Port das jeweilige Feature unterstützt sind die einzelnen Optionskästchen bedienbar.
- Wenn für den Port ein Partnerport bestimmt wurde sind folgende Optionskästchen nicht bedienbar:
  - "Ende der Erfassung erreichbarer Teilnehmer"
  - "Ende der Topologieerkennung"
- Wenn Autonegotiation deaktiviert ist, dann sind alle Optionskästchen nicht bedienbar.

## Gerätetausch ohne Wechselmedium aktivieren

### Austausch eines IO-Devices ohne Wechselmedium

Der Austausch von IO-Devices ist in Automatisierungssystemen häufiger nötig. Üblicherweise wird den IO-Devices entweder über das Stecken eines Wechselmediums oder über das PG ein Geräte-Name zugewiesen. Die CPU identifiziert das IO-Device über diesen Geräte-Name.

Der Austausch eines IO-Devices kann unter bestimmten Voraussetzungen ohne das Stecken eines Wechselmediums (z. B. Memory Card) oder ohne PG durchgeführt werden. Hierzu werden die Nachbarschaftsbeziehungen zwischen den einzelnen IO-Devices und dem IO-Controller über Ethernet-Mechanismen analysiert. Aus diesen, im IO-Controller gespeicherten, Nachbarschaftsbeziehungen erkennt der IO-Controller, welches IO-Device ersetzt wurde, und definiert den Geräte-Name neu.

## Voraussetzungen

- Es ist bereits eine Portverschaltung projektiert.
- Die beteiligten IO-Devices des Automatisierungssystems müssen den Gerätetausch ohne Wechselmedium unterstützen.  
Wenn einzelne IO-Devices des Automatisierungssystems den Gerätetausch ohne Wechselmedium nicht unterstützen, wird für das IO-Device eine entsprechende Meldung ausgegeben.

---

### Hinweis

Verwenden Sie als Austauschgeräte nur neue IO-Devices oder versetzen Sie bereits parametrisierte IO-Devices zuvor in den Auslieferungszustand.

---

## Vorgehen

Um den Austausch eines IO-Devices ohne Wechselmedium zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht die PROFINET-Schnittstelle des entsprechenden IO-Controllers.
2. Aktivieren Sie in den Eigenschaften der Schnittstelle unter "Erweiterte Einstellungen > Schnittstellen-Optionen" das Optionskästchen "Gerätetausch ohne Wechselmedium ermöglichen".

## Siehe auch

Komponenten mit der Funktion Gerätetausch ohne Wechselmedium (<http://support.automation.siemens.com/WW/view/de/36752540>)

## GSD-Dateien verwenden

### GSD-Dateien für IO-Devices

#### Grundsätzliches zu GSD-Dateien von IO-Devices

Die Eigenschaften von PROFINET IO-Devices sind nicht in einer Schlüsselwort-basierten Textdatei hinterlegt (wie bei PROFIBUS DP-Slaves), sondern in einer XML-Datei, deren Struktur und Regeln von einem GSDML-Schema bestimmt wird.

Die Sprache zur Beschreibung der GSD Dateien ist GSDML (Generic Station Description Markup Language). Sie wird durch das GSDML-Schema definiert.

Ein GSDML-Schema enthält Gültigkeitsregeln, die es erlauben, z. B. die Syntax einer GSD-Datei zu überprüfen. GSDML-Schemata (in Form von Schema-Dateien) beziehen Hersteller von IO-Devices von PROFIBUS International.

Funktionserweiterungen im Bereich von PROFINET IO haben Auswirkungen auf die GSDML-Spezifikation und das zugehörige Schema. Durch eine Funktionserweiterung entsteht eine neue Version der Spezifikation und des Schemas.

#### Namen der GSD-Dateien für IO-Devices

Der Name einer GSD-Datei für IO-Devices kann beispielsweise folgendermaßen aussehen:

"GSDML-V1.0-Siemens-ET200S-20030616.xml"

| Namensbestandteil | Erläuterung   |
|-------------------|---|
| GSDML             | Zeichenfolge, mit der jede GSD-Datei für IO-Devices beginnt |
| V1.0              | Version des GSDML-Schemas                                   |
| Siemens           | Hersteller  |
| ET200S            | Name des Geräts   |



| Namensbestandteil | Erläuterung                             |
|-------------------|---|
| 20030616          | Kennzeichnung des Ausgabestands (Datum) |
| .xml              | Dateiendung                             |

## Versionierung der GSD-Dateien für IO-Geräte

Für GSD-Dateien gibt es eine zweigeteilte Versionsinformation:

Zum einen wird die Version des GSDML-Schemas angegeben. Dadurch ist festgelegt, welchen Sprachumfang eine GSD-Datei nutzt.

Zum anderen wird der Ausgabestand in Form eines Datums angegeben. Der Ausgabestand von GSD-Dateien erhöht sich, wenn z. B eine Fehlerbeseitigung durchgeführt wurde oder eine Funktionserweiterung eingebracht wurde.

Funktionserweiterungen können zu einer neuen Version des GSDML-Schemas führen. Eine neue Version eines GSDML-Schemas kann möglicherweise nur eingeschränkt unterstützt werden.

## GSD-Datei installieren

### Einführung

In einer GSD-Datei (Gerätebeschreibungsdatei) sind alle Eigenschaften eines IO-Devices hinterlegt. Falls Sie ein IO-Device projektieren möchten, das nicht im Hardware-Katalog vorhanden ist, müssen Sie die vom Hersteller mitgelieferte GSD-Datei installieren. Über GSD-Dateien installierte IO-Devices werden im Hardware-Katalog angezeigt und können dann selektiert und konfiguriert werden.

### Voraussetzung

- Der Hardware- und Netzwerkeditor ist geschlossen.
- Sie haben Zugriff auf die benötigten GSD-Dateien in einem Verzeichnis auf der Festplatte.

### Vorgehen

Um eine GSD-Datei zu installieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Gerätebeschreibungsdateien installieren".
2. Wählen Sie im Dialog "Gerätebeschreibungsdateien installieren" das Verzeichnis aus, in dem die GSD-Dateien liegen.
3. Wählen Sie aus der Liste der angezeigten GSD-Dateien eine oder mehrere Dateien aus.
4. Klicken Sie auf die Schaltfläche "Installieren".
5. Um eine Protokolldatei für die Installation zu erstellen, klicken Sie auf die Schaltfläche "Protokolldatei speichern".  
Eventuell auftretende Probleme bei der Installation lassen sich anhand der Protokolldatei nachvollziehen.

Sie finden die neuen, über GSD-Dateien installierten IO-Devices unter "Weitere Feldgeräte > PROFINET" im Hardware-Katalog.

---

#### **Hinweis**

Die Installation einer GSD-Datei kann nicht rückgängig gemacht werden.

---

### **Ausgabestand einer GSD-Datei ändern**

#### **Ausgabestand einer GSD-Datei ändern**

Sie können den Ausgabestand der GSD-Datei für IO-Device ändern:

- Nur für das aktuelle IO-Device
- Alle passenden IO-Devices des IO-Systems
- Alle passenden IO-Devices des gesamten Projekts

Zunächst werden alle verfügbaren GSD-Dateien für das aktuelle IO-Device angezeigt. Die angezeigten GSD-Dateien unterscheiden sich nur im Ausgabestand. Die aktuelle verwendete GSD-Datei ist markiert.

#### **Voraussetzung**

- Die E/A-Daten sind für alle IO-Devices, für die der Ausgabestand geändert werden soll, gleich.
- Die Bestellnummer hat sich nicht geändert.
- Die Anzahl der Submodule ist gleich.
- Die Parametrierdaten haben sich nicht verändert.
- Es darf sich kein Modul oder Submodul auf einem Steckplatz befinden, der nach der neuen GSD-Datei nicht mehr zulässig ist.

#### **Vorgehen**

Um den Ausgabestand von einem oder mehreren IO-Devices zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie das IO-Device, für dessen GSD-Datei Sie den Ausgabestand ändern möchten.
2. Klicken Sie in den Eigenschaften des IO-Device unter "Allgemein > Kataloginformation" auf die Schaltfläche "Ausgabestand ändern".  
Der Dialog "Ausgabestand ändern" wird geöffnet.
3. Selektieren Sie in der Tabelle "Verfügbare Ausgabestände" die GSD-Revision, die Sie verwenden möchten.

4. Wählen Sie unter "Markierte Ausgabestände verwenden für", für welche Geräte der Ausgabestand geändert werden soll:
  - Nur für das aktuelle IO-Device
  - Für alle passenden IO-Devices in des IO-Systems
  - Für alle passenden IO-Devices des Projekts
5. Klicken Sie auf die Schaltfläche "Übernehmen".

#### 8.1.4.9 Buskopplung mit PN/PN Coupler

##### Anwendungsgebiet und Funktion

##### Anwendungsgebiet

Der PN/PN Coupler dient dazu, zwei Ethernet-Subnetze miteinander zu verbinden und Daten auszutauschen. Dabei können Nutzdaten über Eingangs- oder Ausgangsadressbereiche oder Datensätze verwendet werden. Die maximale Größe der übertragbaren Eingangs- und Ausgangsdaten beträgt 1024 Byte. Die Aufteilung auf Eingangsdaten und Ausgangsdaten ist beliebig, so dass z. B. 800 Byte Eingangsdaten und 200 Byte Ausgangsdaten projektiert werden können.

Der PN/PN Coupler hat als ein Gerät zwei PROFINET-Schnittstellen, die jeweils mit einem Subnetz verbunden werden.

In der Projektierung werden aus diesem einen PN/PN Coupler zwei IO-Devices gemacht, und zwar für jede Station mit ihrem Subnetz jeweils ein IO-Device. Der jeweils andere Teil des PN/PN Couplers wird als Koppel-Partner bezeichnet. Mit Abschluss der Projektierung werden die beiden Teile zusammengeführt.

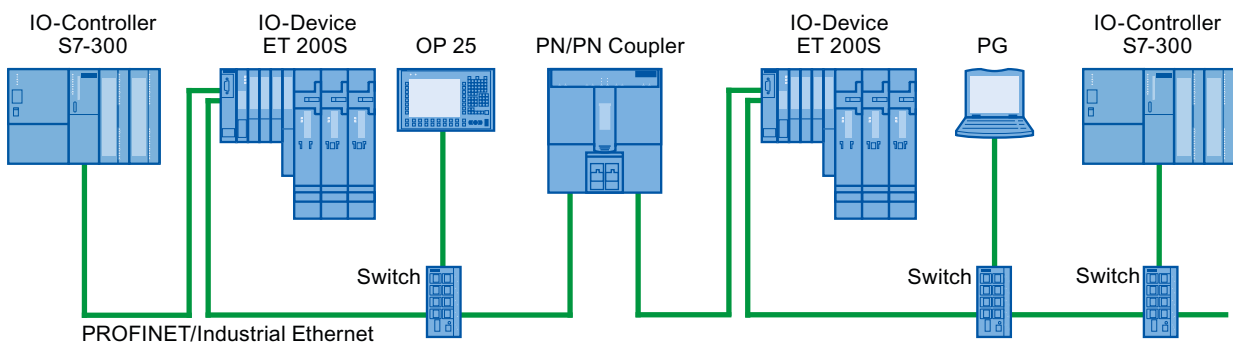


Bild 8-2 Kopplung zweier PROFINET IO-Subnetze mit einem PN/PN Coupler

##### Weitere Informationen

Beachten Sie für weitere Informationen zum Thema " PN/PN Coupler" den Service & Support im Internet (<http://support.automation.siemens.com/WW/view/de/44319532>).

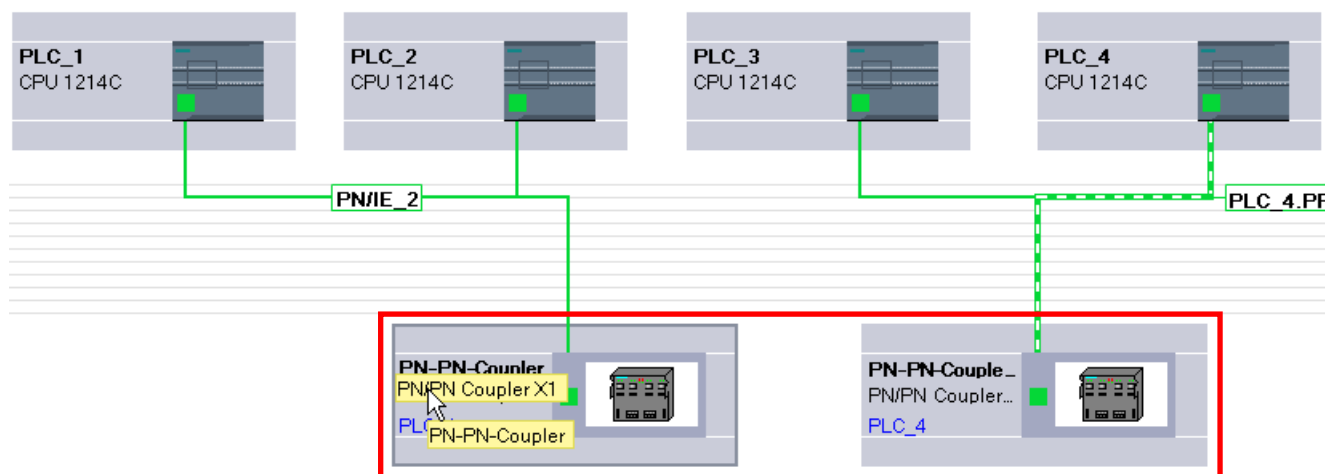
## Ethernet-Subnetze koppeln

### Ethernet-Subnetze mit einem PN/PN Coupler koppeln

Mit dem Normgerät PN/PN Coupler können Sie Ethernet-Subnetze miteinander koppeln.

Um Ethernet-Subnetze zu koppeln, gehen Sie folgendermaßen vor:

1. Legen Sie Ihre Ethernet-Subnetze an.
  2. Wählen Sie im Hardware-Katalog die Norm-Feldgeräte. Im Ordner "PROFINET IO" finden sie den PN/PN Coupler als Kopfmodul.
  3. Ziehen Sie beide Komponenten X1 und X2 der gewünschten Version des PN/PN Coupler per Drag & Drop in die Netzsicht. Die Komponenten bilden ein Gerät, werden aber für die bessere Handhabung getrennt dargestellt.
  4. Verbinden Sie die Ethernetschnittstelle des PN/PN Coupler X1 mit dem ersten Ethernet-Subnetz.
  5. Verbinden Sie die Ethernetschnittstelle des PN/PN Coupler X2 mit dem zweiten Ethernet-Subnetz.
- Die Ethernet-Subnetze sind nun durch die beiden Komponenten des PN/PN Coupler gekoppelt.



#### 8.1.4.10 Externe Tools einbinden

### Einbindung von S7-externen Tools

#### Einführung

Für das Konfigurieren dezentraler Geräte können STEP 7-externe Tools ("Device Tools") mit einer speziellen Aufrufschnittstelle (Tool Calling Interface) verwendet werden. Solche Geräte werden auch als "TCI-fähig" bezeichnet.

Diese Tools gehen in ihrem Leistungsumfang über die Möglichkeiten der GSD-Projektierung hinaus, z. B. können sie erweiterte grafische Eingabemöglichkeiten bieten.

Als dezentrale Geräte kommen in Frage:

- PROFIBUS DP-Slaves
- Baugruppen innerhalb eines DP-Slaves
- PROFINET IO-Devices
- Baugruppen innerhalb eines IO-Devices

---

#### **Hinweis**

#### **Gewährleistung und Haftung**

Siemens übernimmt keine Verantwortung für die per TCI (Tool Calling Interface) aufgerufene Drittsoftware (Device Tools) und für das ordnungsgemäße Zusammenspiel mit den dazugehörigen Geräten.

---

### **Voraussetzung**

Die Aufrufchnittstelle des Tools genügt der TCI-Spezifikation. Über diese Aufrufchnittstelle werden Parameter und Kommandos an das dezentrale Gerät übergeben.

Solche Tools müssen über ein vom Hersteller bereitgestelltes Setup installiert werden. Ausnahme ist das Device Tool "S7-PCT" (Port Configuration Tool) für IO-Link Master-Module und IO-Link Devices, das mit STEP 7 mitgeliefert wird. Besonderheit: Nach der Installation wird das Tool nicht in der Liste der installierten Software bzw. in der Liste der Software-Produkte im Projekt angezeigt.

Die GSD-Datei des dezentralen Gerätes, das mit dem Device Tool konfiguriert werden soll, muss installiert sein.

### **Starten des Device Tools**

Den Befehl für das Starten des Device Tools finden Sie im Kontextmenü des TCI-fähigen Gerätes im Kontextmenü der grafischen und tabellarischen Gerätesicht: "Device Tool starten".

### **Siehe auch**

SIMATIC S7-PCT starten (Seite 823)

### **SIMATIC S7-PCT starten**

#### **Einführung**

Mit STEP 7 wird das Device Tool "S7-PCT" (Port Configuration Tool) installiert.

Das Tool ermöglicht die Parametrierung der Ports von IO-Link Modulen wie 4SI IO-Link (S7-1200, ET 200S) oder 4IOL+8DI+4DO (ET 200eco PN).

### Voraussetzung

Sie haben die entsprechende CPU, den DP-Slave oder das IO-Device mit einem IO-Link Modul konfiguriert.

### Vorgehen

Für den Start über die grafische Gerätesicht gehen Sie folgendermaßen vor:

1. Markieren Sie das IO-Link Modul in der Gerätesicht.
2. Wählen Sie den Kontext-Menübefehl "Device Tool starten".

ODER gehen Sie für den Start über die tabellarische Geräteübersicht folgendermaßen vor:

1. Markieren Sie das IO-Link Modul in der Gerätesicht.
2. Ordnen Sie die Bereiche im Arbeitsbereich so an, dass die tabellarische Geräteübersicht sichtbar ist (befindet sich zwischen Gerätesicht und Inspektorfenster).
3. Markieren Sie die Zeile mit dem IO-Link Modul in der Geräteübersicht.
4. Wählen Sie den Kontext-Menübefehl "Device Tool starten".

### Ergebnis

Das Tool startet und Sie können die Ports konfigurieren.

### Siehe auch

Einbindung von S7-externen Tools (Seite 822)

## 8.1.4.11 Konfiguration laden

### Einführung zum Laden einer Konfiguration

Um ein Gerät in Betrieb zu nehmen, ist es notwendig, dass auf dem PG/PC wie auch auf den angeschlossenen Geräten identische Konfigurationen gespeichert sind. Den Abgleich zwischen PG/PC und den angeschlossenen Geräten nehmen Sie über das Laden einer Konfiguration vor. Das Laden von Konfigurationsdaten kann grundsätzlich in zwei Richtungen erfolgen:

- Konfiguration vom PG/PC in ein Gerät laden
- Konfiguration von einem Gerät in den PG/PC laden

## Siehe auch

- Projektdatei von einem Gerät laden (Seite 259)
- Allgemeine Informationen zum Laden (Seite 256)
- Laden einer Konfiguration in ein Gerät (Seite 825)
- Projektdatei in ein Gerät laden (Seite 257)
- Laden einer Konfiguration in den PG/PC (Seite 826)
- Besonderheiten beim Anlauf (Seite 838)

## Laden einer Konfiguration in ein Gerät

### Laden der Hardware-Konfiguration

Wenn Sie ein neues Gerät in das Projekt eingefügt und konfiguriert haben, oder wenn Sie eine bestehende Hardware-Konfiguration verändert haben, müssen Sie zunächst die aktuelle Konfiguration in das Gerät laden. Auf diese Weise ist sichergestellt, dass auf dem PG/PC und der physikalisch vorhandenen Baugruppe dieselbe Konfiguration eingestellt ist.

Beim erstmaligen Laden werden die Hardware-Projektdatei vollständig geladen. Bei weiteren Ladevorgängen werden nur noch Änderungen der Konfiguration geladen.

Zum Laden der Hardware-Konfiguration stehen Ihnen folgende Möglichkeiten zur Verfügung:

- Laden in der Geräte- oder Netzansicht
- Laden in der Projektnavigation
- Laden in einen erreichbaren Teilnehmer

#### **WARNUNG**

##### **Ladevorgang nur in STOP durchführen**

Nach dem Laden kann es bei einer Fehlparametrierung zu unerwartetem Verhalten der Maschine oder des Prozesses kommen. Die CPU muss für den Ladevorgang zwingend in den Betriebszustand STOP gesetzt werden, um möglichen Schaden an der Anlage oder Personenschäden auszuschließen.

## Besonderheiten beim Laden von taktsynchronen Applikationen

Taktsynchrone Applikationen haben einen Hardwarekonfigurations-Anteil und einen Software-Anteil.

Beispiel: Wenn Sie in der Hardware-Konfiguration die Nummer eines IO-Systems die Verzögerungszeit oder die Teilprozessabbild-Zuordnung der taktsynchronen Peripherie ändern, hat das Einfluss auf die Parameter des Taktsynchronalarm-OBs und damit auf den Software-Anteil.

Wir empfehlen bei takt synchronen Applikationen ein komplettes Laden (Hardware und Software). Beim partiellen Laden (Hardware und Software getrennt zu unterschiedlichen Zeitpunkten) können Inkonsistenzen entstehen, die z. B. für die CPU ein Anlaufhindernis darstellen oder bewirken, dass die Applikation nicht takt synchron läuft.

### Siehe auch

Allgemeine Informationen zum Laden (Seite 256)

Projektdateien in ein Gerät laden (Seite 257)

## Laden einer Konfiguration in den PG/PC

### Einführung

Wenn Sie ein neues Gerät an einen PG/PC angeschlossen haben, das Gerät aber noch nicht im Projekt eingefügt haben, können Sie die komplette Konfiguration von dem neu angeschlossenen Gerät auf den PG/PC übertragen. Das Gerät wird dadurch im Projekt neu angelegt.

Das Laden eines neuen Geräts erfolgt grundsätzlich über die Liste der erreichbaren Teilnehmer in der Projektnavigation. Sie können durch Multiselektion mehrere Geräte auf einmal in das Projekt laden. Das Laden einer Konfiguration ist mehrfach möglich. Bei jedem Ladevorgang wird ein neues Gerät angelegt, auch wenn das Gerät bereits zuvor geladen wurde.

### Voraussetzungen

- Die ursprüngliche Hardware-Konfiguration muss im TIA-Portal ab V12 erstellt sein. Ein Projekt mit einer älteren Version als der aktuellen Version muss unter Umständen hochgerüstet (Seite 235) werden.
- Das geöffnete Projekt befindet sich im Offline-Modus.

### Ladeumfang

Die folgende Liste zeigt eine genaue Übersicht über die übertragenen Konfigurationsbestandteile:

- Parameter des Geräts  
Alle eingestellten Parameter des Moduls werden übertragen.
- PROFIBUS Mastersysteme und alle PROFIBUS-relevanten Einstellungen  
Ein DP-Mastersystem und alle angeschlossenen Slaves werden in das Projekt eingefügt. Die jeweiligen Einstellungen bleiben erhalten. Ist bereits ein passendes PROFIBUS-Subnetz angelegt, werden die geladenen Module mit PROFIBUS-Schnittstelle an das vorhandene Subnetz angeschlossen.



- **PROFINET-IO-Systeme und alle PROFINET-relevanten Einstellungen**  
Die Geräte mit IO-Controller, alle IO-Systemen sowie alle IO-Devices werden in das Projekt übertragen. Einstellungen zur Topologie werden ebenfalls übernommen.  
Besteht bereits ein passendes Ethernet-Netzwerk im Projekt, werden die geladenen Geräte in das vorhandene Netzwerk integriert.  
Beziehungen zwischen IO-Controllern und I-Devices werden im Projekt nur dann hergestellt, wenn sowohl der IO-Controller als auch das I-Device in das PG geladen werden. Ob Sie zuerst den IO-Controller oder die I-Devices laden, ist unerheblich.
- **I-Devices und I-Slaves**  
Master-Slave-Beziehungen zwischen I-Slave und zugeordnetem DP-Master werden im Projekt nur dann hergestellt, wenn sowohl der Master als auch der I-Slave in das PG geladen werden. Ob Sie zuerst das Mastersystem oder die I-Devices und I-Slaves laden, ist unerheblich. Sobald beide Geräte geladen sind, werden auch die Verbindungen wieder hergestellt.
- **Direkter Datenaustausch**  
Die Konfiguration eines direkten Datenaustauschs zwischen zwei Geräten kann ebenfalls in das Projekt geladen werden. Dazu müssen Sie die beiden Partner nacheinander laden.
- **S7-Verbindungen**  
S7-Verbindungen werden automatisch als einseitig beim Laden einer Gerätekonfiguration übernommen, auch wenn die S7-Verbindung im Ursprungsprojekt zweiseitig projektiert war. Sind beide Verbindungspartner geladen, wird beim nächsten Übersetzen die Verbindung wieder zusammengeführt.
- **Busparameter**  
Geladene Busparameter weichen nach dem Laden eines einzelnen Geräts zunächst von den Einstellungen im Ursprungsprojekt ab. Erst wenn alle beteiligten Geräte geladen sind und sich keine zusätzlichen Geräte am selben Bus befinden, stimmen die Busparameter wieder mit dem Ursprungsprojekt überein.
- **Einer CPU zugehörige Peripheriemodule**  
Nach dem Laden einer CPU werden automatisch auch alle anderen Module, die sich im Adressraum der CPU befinden, mit geladen.

## Siehe auch

Projektdatei von einem Gerät laden (Seite 259)

Allgemeine Informationen zum Laden (Seite 256)

## 8.1.5 Meldungen anzeigen







### 8.1.5.1 Übersicht über die Meldungsanzeige

Mit der Funktion "Meldungsanzeige" können asynchrone Meldungen von Diagnoseereignissen und anwenderdefinierte Diagnosemeldungen sowie Meldungen von ALARM-Anweisungen ausgegeben werden.

Außerdem können Sie von der Meldungsanzeige aus über den Kontextmenübefehl "Meldung bearbeiten" den Meldungseditor starten und Anwender-Diagnosemeldungen anlegen.

## Symbole

Die folgende Tabelle zeigt die Symbole und ihre Funktionsweise:

| Symbol   | Funktion  |
|--|---|
| <br>Archivansicht       | Zeigt die im Archiv befindlichen Meldungen an.  |
| <br>Aktive Meldungen    | Zeigt die aktuell anstehenden Meldungen an. Quittierpflichtige Meldungen werden in blauer Schrift dargestellt.        |
| <br>Ignorieren          | Ignoriert das Eintreffen von Meldungen. Diese Meldungen werden weder im Fenster angezeigt noch im Archiv gespeichert. |
| <br>Quittieren          | Bestätigt die markierte Meldung als gelesen. Zu quittierende Meldungen werden in blauer Schrift angezeigt.            |
| <br>Archiv leeren       | Löscht alle Meldungen im Archiv.  |
| <br>Archiv exportieren | Exportiert das aktuelle Meldungsarchiv in eine Datei im xml-Format.   |

### 8.1.5.2 Archivansicht

In der Archivansicht werden Meldungen nach ihrem zeitlichen Erscheinen angezeigt und archiviert. Die Archivgröße (zwischen 200 und 3000 Meldungen) können Sie über den Menübefehl "Extras > Einstellungen > Online&Diagnose" festlegen. Ist die eingestellte Archivgröße überschritten, wird jeweils die älteste vorhandene Meldung gelöscht.

Quittierpflichtige Meldungen werden in blauer Schrift angezeigt und können mit dem Kontextmenübefehl "Meldung(en) quittieren" quittiert werden.

Das Archiv wird ständig aktualisiert und muss nicht explizit gespeichert werden.

### 8.1.5.3 Aufbau der Meldungen in der Archivansicht

In der Archivansicht werden alle Ereignisse, die auf den angewählten CPUs aufgetreten sind, protokolliert. Dabei wird für jedes einzelne Ereignis ein neuer Eintrag erstellt und in der Tabelle als eine weitere Zeile dargestellt.

#### Aufbau der Tabelle

Alle Attribute der Meldungen können als Spalten dargestellt werden. Sie können die einzelnen Spalten ein- oder ausblenden und die Breite und Reihenfolge der Spalten ändern. Diese Einstellungen werden beim Schließen des Projekts gespeichert.

Die Meldungen können ein- oder mehrzeilig dargestellt werden. Bei der einzeiligen Darstellung wird nur die erste Zeile der mehrzeiligen Meldungsdaten angezeigt.

Die Meldungen können quittierpflichtig (quittierpflichtige Meldungen) und nicht quittierpflichtig sein. Die quittierpflichtigen, aber noch nicht quittierten Alarme werden durch blaue Schrift hervorgehoben und können entweder kontextsensitiv über das Symbol in der Funktionsleiste oder über den Kontextmenübefehl "Meldung(en) quittieren" quittiert werden.

#### 8.1.5.4 Meldungen empfangen

Damit die Meldungen angezeigt werden, müssen Sie zunächst das Empfangen der Meldungen für jede CPU einstellen.

##### Vorgehen

Um Meldungen zu empfangen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Ordner "Online & Diagnose" der entsprechenden CPU in der Projektnavigation.
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Online-Zugänge".
3. Aktivieren Sie die Option "Meldungen empfangen".

---

**Hinweis**

**Wenn Sie diese Vorgehensweise wählen, werden Meldungen erst empfangen, wenn Sie erneut eine Online-Verbindung zum Gerät hergestellt haben.**

---

Oder:

1. Markieren Sie in der Geräte-, Netz- oder Topologiesicht die entsprechende CPU.
2. Wählen Sie im Menü "Online" oder im Kontextmenü den Befehl "Meldungen empfangen".

Oder:

1. Markieren Sie in der Projektnavigation die CPU.
2. Wählen Sie im Menü "Online" oder im Kontextmenü den Befehl "Meldungen empfangen".

---

**Hinweis**

**Wenn Sie eine der beiden oberhalb genannten Vorgehensweisen wählen, müssen Sie zuerst eine Online-Verbindung zum Gerät hergestellt haben.**

---

#### 8.1.5.5 Archiv exportieren

Zum Archivieren von Meldungen können Sie das Archiv exportieren. Gehen Sie dazu folgendermaßen vor:

1. Wechseln Sie in die Archivansicht.
2. Klicken Sie auf das Symbol "Archiv exportieren".
3. Wählen Sie im aufgeblendeten Dialog den Pfad aus, wohin das Archiv exportiert werden soll.

## Ergebnis

Das Archiv wird als xml-Datei an die von Ihnen angegebene Stelle gespeichert.

### 8.1.5.6 Archiv leeren

Das Archiv ist als Ringpuffer organisiert, d. h., wenn es gefüllt ist, werden für neue Meldungen die jeweils ältesten Meldungen aus dem Archiv gelöscht. Über das Symbol "Archiv leeren" löschen Sie das gesamte Archiv.

## Vorgehen

Um das Archiv zu leeren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste der Meldungsanzeige auf das Symbol "Archiv leeren".

### 8.1.5.7 Ansicht "Aktive Meldungen"

Die Ansicht "Aktive Meldungen" ist ein Abbild des Meldequittierspeichers der angewählten CPU(s).

### 8.1.5.8 Aufbau der Meldungen in der Ansicht "Aktive Meldungen"

Die Ansicht "Aktive Meldungen" stellt ein Abbild des Meldequittierspeichers der angewählten CPUs dar. Pro anstehender Meldung wird in der Tabelle ein Eintrag angezeigt. Ereignisse einer Meldung ("Kommen", "Gehen" und "Quittiert") werden in einer Zeile angezeigt.

#### Aufbau der Tabelle

Alle Attribute der Meldungen können als Spalten dargestellt werden. Sie können die einzelnen Spalten ein- oder ausblenden und die Breite und Reihenfolge der Spalten ändern. Diese Einstellungen werden beim Schließen des Projekts gespeichert.

Die Meldungen können ein- oder mehrzeilig dargestellt werden. Bei der einzeiligen Darstellung wird nur die erste Zeile der mehrzeiligen Meldungsdaten angezeigt.

Die Meldungen können quittierpflichtig und nicht quittierpflichtig sein. Die quittierpflichtigen, aber noch nicht quittierten Alarme werden durch blaue Schrift hervorgehoben und können entweder über kontextsensitiv über das Symbol in der Funktionsleiste oder über den Kontextmenübefehl "Meldung(en) quittieren" quittiert werden.

### 8.1.5.9 Status der Meldungen

Abhängig davon, ob Sie sich in der Ansicht "Aktive Meldungen" oder in der Archivansicht befinden, können die angezeigten Meldungen verschiedene Status haben.

### Status der Meldungen in der Ansicht "Aktive Meldungen"

- K: Meldung ist gekommen
- KQ: Meldung ist gekommen und quittiert
- KG: Meldung ist gegangen

Wenn mehr Signalwechsel kommen als gesendet werden können (Signal Overflow), wird beim Status ein O angezeigt und der Status wird rot dargestellt.

### Status der Meldungen in der Archivansicht

- keine Angabe: nur bei Meldungen, die vom PG/PC erzeugt und im Register "Archiv" angezeigt werden, z. B. Anmeldezustand, Verbindungsabbruch, Betriebszustandsänderungen
- K: Meldung ist gekommen
- Q: Meldung ist gekommen und quittiert
- G: Meldung ist gegangen
- L: Meldung wurde gelöscht

Wenn mehr Signalwechsel kommen als gesendet werden können (Signal Overflow), wird beim Status ein O angezeigt und der Status wird rot dargestellt.

#### 8.1.5.10 Meldungen quittieren

Quittierpflichtige Meldungen werden in blauer Schrift angezeigt.

### Vorgehen

Um eine Meldung zu quittieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gewünschte(n) Meldung(en) in der Tabelle.
2. Klicken Sie auf die Schaltfläche "Quittieren".

---

#### **Hinweis**

Sie können mehrere zu quittierende Meldungen gleichzeitig selektieren. Halten sie dazu die Taste <Strg> gedrückt und selektieren Sie dann die gewünschten Meldungen in der Tabelle.

---

### Ergebnis

Die ausgewählte Meldung wurde quittiert und wird in normaler Schrift angezeigt.

---

#### **Hinweis**

In der Ansicht "Aktive Meldungen" werden quittierte Meldungen, die bereits gegangen sind, nicht mehr angezeigt.

---

### 8.1.5.11 Meldungen ignorieren

#### Meldungen ignorieren

Um Meldungen zu ignorieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Ignorieren".  
Das Symbol wird grau hinterlegt.

#### Ergebnis

Alle Meldungen werden ab diesem Zeitpunkt ignoriert. In der Archivansicht wird eine Meldung erstellt, die besagt, dass das Anzeigen von Meldungen und Ereignissen deaktiviert ist.

#### Ignorieren von Meldungen deaktivieren

Um das Ignorieren von Meldungen zu deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Ignorieren".  
Das Symbol wird weiß hinterlegt.

#### Ergebnis

Alle Meldungen, d. h. auch die Meldungen, die auf der CPU aktuell anstehen, während die Funktion "Meldungen ignorieren" aktiviert war, werden ab diesem Zeitpunkt wieder angezeigt. In der Archivansicht wird eine Meldung erstellt, die besagt, dass das Anzeigen von Meldungen und Ereignissen aktiviert ist.

### 8.1.5.12 Tastaturbedienung in der Meldungsanzeige

#### Meldungsanzeige

| Funktion                               | Tastenkombination (Shortcut) |
|--|------------------------------|
| Alle Meldungen auswählen               | Ctrl+A                       |
| Alle ausgewählten Meldungen quittieren | Ctrl+Q                       |

## 8.1.6 Zusatzinformationen zu Konfigurationen

### 8.1.6.1 Arbeitsweise von S7-1200 CPUs

#### Betriebszustände

#### Grundlagen zu den Betriebszuständen von S7-CPU

##### Einführung

Betriebszustände beschreiben das Verhalten der CPU. Folgende Betriebszustände sind möglich:

- ANLAUF
- RUN
- STOP

In diesen Betriebszuständen ist die CPU kommunikationsfähig, z. B. über die PN/IE-Schnittstelle.

##### Weitere Betriebszustände

Wenn die CPU nicht betriebsbereit ist, befindet sie sich in einem der beiden folgenden Betriebszustände:

- Spannungslos, d. h. die Netzspannung ist ausgeschaltet.
- Defekt, d. h. es ist ein interner Fehler aufgetreten.  
Wenn der Zustand "Defekt" durch einen Firmware-Fehler hervorgerufen wurde, wird dieser Zustand durch die Status-LEDs der CPU signalisiert (siehe Beschreibung zur CPU). Gehen Sie folgendermaßen vor, um die Ursache zu ermitteln:
  - Schalten Sie den Netzschalter aus und wieder ein.
  - Lesen Sie den Diagnosepuffer aus, wenn die CPU anläuft und senden die Daten zur Analyse an den Customer Support.

Wenn die CPU nicht anläuft, tauschen Sie sie aus.

##### Siehe auch

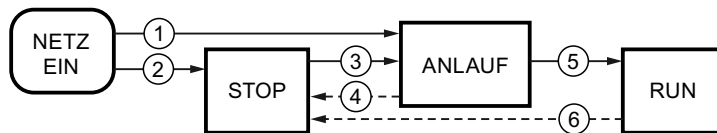
Betriebszustand "STOP" (Seite 839)

Betriebszustand "RUN" (Seite 838)

## Betriebszustandsübergänge

### Übersicht

Das folgende Bild zeigt die Betriebszustände und die Betriebszustandsübergänge von S7-1200-CPU:



Die folgende Tabelle zeigt die Bedingungen, unter welchen die Betriebszustände wechseln:

| Nr. | Betriebszustandsübergang | Bedingungen   |
|-----|--------------------------|---|
| ①   | NETZ EIN → ANLAUF        | Die CPU geht nach dem Einschalten in den Betriebszustand "ANLAUF", wenn: <ul style="list-style-type: none"> <li>• die Anlaufart "Warmstart" eingestellt ist und</li> <li>• die Hardware-Konfiguration und die Programmbausteine konsistent sind.</li> </ul> Nicht-remanenter Speicher wird gelöscht und der Inhalt nicht-remanenter DBs wird auf die Anfangswerte des Ladespeichers zurückgesetzt. Remanenter Speicher und remanenter DB-Inhalt bleiben erhalten. |
| ②   | NETZ EIN → STOP          | Bei Einstellung der Anlaufart "Kein Anlauf" geht die CPU nach dem Einschalten der Versorgungsspannung in den Betriebszustand "STOP".<br>Nicht-remanenter Speicher wird gelöscht und der Inhalt nicht-remanenter DBs wird auf die Anfangswerte des Ladespeichers zurückgesetzt. Remanenter Speicher und remanenter DB-Inhalt bleiben erhalten.   |
| ③   | STOP → ANLAUF            | Die CPU geht in den Betriebszustand "ANLAUF", wenn: <ul style="list-style-type: none"> <li>• die CPU vom Programmiergerät aus auf "RUN" gesetzt wird und</li> <li>• die Hardware-Konfiguration und die Programmbausteine konsistent sind.</li> </ul>  |
| ④   | ANLAUF → STOP            | Die CPU geht in folgenden Fällen wieder in den Betriebszustand "STOP": <ul style="list-style-type: none"> <li>• Während des Anlaufs wird ein Fehler erkannt.</li> <li>• Die CPU wird vom Programmiergerät aus auf "STOP" gesetzt.</li> <li>• Ein STOP-Befehl wird im Anlauf-OB bearbeitet.</li> </ul>   |
| ⑤   | ANLAUF → RUN             | Wenn der Anlauf erfolgreich ist, wechselt die CPU in "RUN".   |
| ⑥   | RUN → STOP               | Die CPU geht in folgenden Fällen wieder in den Betriebszustand "STOP": <ul style="list-style-type: none"> <li>• Ein Fehler wird erkannt, der die Weiterarbeit verhindert.</li> <li>• Die CPU wird vom Programmiergerät aus auf "STOP" gesetzt.</li> <li>• Ein STOP-Befehl wird im Anwenderprogramm bearbeitet.</li> </ul>   |



## Betriebszustand "ANLAUF"

### Grundlagen zum Betriebszustand "ANLAUF"

#### Funktion

Bevor die CPU nach dem Einschalten mit der Bearbeitung des zyklischen Anwenderprogramms beginnt, wird ein Anlaufprogramm bearbeitet.

Im Anlaufprogramm können Sie durch entsprechende Programmierung von Anlauf-OBs bestimmte Initialisierungs-Variablen für Ihr zyklisches Programm festlegen. Die Anzahl von Anlauf-OBs ist nicht vorgeschrieben. Sie haben die Möglichkeit, keinen Anlauf-OB, einen Anlauf-OB oder mehrere Anlauf-OBs zu programmieren.

#### Parametrieren des Anlaufverhaltens

Sie können festlegen, ob die CPU im Betriebszustand STOP verbleiben soll, oder ob ein Warmstart durchgeführt wird. Darüber hinaus können Sie das Verhalten im Anlauf (RUN oder vorheriger Betriebszustand) in der Gruppe "Anlauf" der CPU-Eigenschaften parametrieren.

#### Besonderheiten

Beachten Sie für den Betriebszustand "ANLAUF" folgende Punkte:

- Die Anlauf-OBs werden abgearbeitet. Unabhängig von der gewählten Anlaufart werden alle programmierten Anlauf-OBs bearbeitet.
- Keine zeitgesteuerte Programmbearbeitung möglich.
- Alarmgesteuerte Programmbearbeitung beschränkt auf:
  - OB 82 (Diagnosealarm)
- Die Ausgänge auf Baugruppen sind gesperrt.
- Das Prozessabbild wird nicht aktualisiert; direkter Peripheriezugriff auf Eingänge ist aber möglich.

#### Siehe auch

Eigenschaften und Parameter bearbeiten (Seite 408)

Grundlagen zu den Betriebszuständen von S7-CPU (Seite 833)

Organisationsbausteine für den Anlauf (Seite 885)

Neustart (Warmstart) (Seite 836)

## Neustart (Warmstart)

### Funktion

Beim Neustart (Warmstart) werden alle nicht remanenten Merker gelöscht und nicht remanente DB-Inhalte auf Startwerte aus dem Ladespeicher zurückgesetzt. Remanente Merker und remanente DB-Inhalte bleiben erhalten.

Die Programmbearbeitung beginnt mit dem ersten Anlauf-OB.

### Auslösen von Neustart (Warmstart)

In folgenden Fällen können Sie über einen Menübefehl vom PG aus einen "Neustart (Warmstart)" durchführen:

- Die CPU ist im Betriebszustand "STOP".
- Nach dem Urlöschen
- Nach dem Laden eines konsistenten Programms und einer konsistenten Hardware-Konfiguration im Betriebszustand "STOP" der CPU.

Bei "NETZ-EIN" wird ein "Neustart (Warmstart)" ausgelöst, wenn Sie das Anlaufverhalten folgendermaßen parametrieren:

- Anlaufart "Warmstart - RUN" (unabhängig vom Betriebszustand der CPU vor NETZ-AUS).
- "Warmstart - Betriebsart vor NETZ-AUS" (abhängig vom Betriebszustand der CPU vor NETZ-AUS. Die CPU muss vorher im RUN gewesen sein.)

### Siehe auch

Remanente Speicherbereiche (Seite 844)

### Anlauf Tätigkeiten

### Übersicht

Die folgende Tabelle zeigt, welche Tätigkeiten die CPU beim Anlauf durchführt:

| Tätigkeiten in der Bearbeitungsreihenfolge                | bei Neustart (Warmstart) |
|---|--------------------------|
| Nicht remanente Merker löschen                            | ja                       |
| Alle Merker löschen                                       | nein                     |
| Prozessabbild der Ausgänge löschen                        | ja                       |
| Bearbeitung der Anlauf-OBs                                | ja                       |
| Prozessabbild der Eingänge aktualisieren                  | ja                       |
| Ausgänge freigeben nach Betriebszustandsübergang in "RUN" | ja                       |

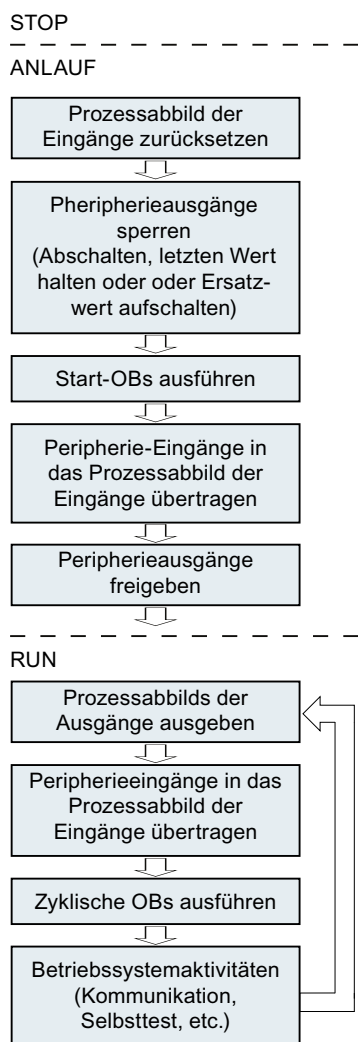
## Ablauf

Das folgende Bild zeigt die Tätigkeiten der CPU in den Betriebszuständen "STOP", "ANLAUF" und "RUN".

Den Zustand der Peripherieausgänge im ersten Zyklus des Anwenderprogramms können Sie durch folgende Maßnahmen festlegen:

- Parametrierbare Ausgabebaugruppen verwenden, um Ersatzwerte ausgeben zu können oder um den letzten Wert zu halten.
- Ausgänge in Anlauf-OBs vorbesetzen.

Während des Anlaufs werden alle Alarm-Ereignisse in eine Warteschlange gestellt, um sie dann im Betriebszustand RUN abzuarbeiten. Im Betriebszustand RUN können zu jeder Zeit Prozessalarme verarbeitet werden.



## Besonderheiten beim Anlauf

### Verhalten bei Sollausbau ungleich Istausbau

Der Sollausbau wird repräsentiert durch die in die CPU geladene projektierte Konfiguration. Der Istausbau ist der tatsächliche Ausbau des Automatisierungssystems.

Wenn Sollausbau und Istausbau voneinander abweichen, geht die CPU trotzdem zunächst in RUN.

### Abbrechen eines Anlaufs

Wenn während des Anlaufs Fehler auftreten, wird der Anlauf abgebrochen und die CPU bleibt in "STOP".

Ein Anlauf wird unter folgenden Bedingungen nicht durchgeführt oder abgebrochen:

- Wenn eine ungültige SD-Karte gesteckt ist.
- Wenn keine Hardware-Konfiguration geladen wurde.

### Siehe auch

Übersicht über die CPU-Eigenschaften (Seite 856)

## Betriebszustand "RUN"

### Funktion

Im Betriebszustand "RUN" erfolgt die zyklische, zeit- und alarmgesteuerte Programmbearbeitung:

- Das Prozessabbild der Ausgänge wird ausgegeben.
- Das Prozessabbild der Eingänge wird gelesen.
- Das Anwenderprogramm wird abgearbeitet.

Der aktive Austausch von Daten zwischen S7-1200 CPUs über Open User Communication ist nur im Betriebszustand "RUN" möglich.

### Abarbeiten des Anwenderprogramms

Nachdem die CPU die Eingänge gelesen hat, wird das zyklische Programm beginnend mit der ersten Anweisung bis zur letzten Anweisung abgearbeitet.

Wenn Sie eine Mindestzykluszeit parametrieren, dann beendet die CPU den Zyklus erst nach Ablauf der Mindestzykluszeit, auch wenn das Anwenderprogramm in kürzerer Zeit durchlaufen wurde.

Um sicherzustellen, dass das zyklische Programm in einer festgelegten Zeit durchlaufen wird, ist eine Zyklusüberwachungszeit eingestellt, die Sie nach Ihren Anforderungen anpassen können. Wenn das zyklische Programm nicht innerhalb dieser Zeit durchlaufen ist, reagiert das System mit einem Zeitfehler.

Weitere Ereignisse wie z. B. Prozessalarme oder Diagnosealarme können den zyklischen Programmfluss unterbrechen und die Zykluszeit verlängern.

### **Siehe auch**

Grundlagen zu den Betriebszuständen von S7-CPU (Seite 833)  
Ereignisse und OBs (Seite 847)

## **Betriebszustand "STOP"**

### **Funktion**

Im Betriebszustand "STOP" wird das Anwenderprogramm nicht bearbeitet. Alle Ausgänge werden deaktiviert bzw. reagieren wie parametrisiert: sie liefern einen parametrisierten Ersatzwert oder halten den letzten ausgegebenen Wert und bringen damit den gesteuerten Prozess in einen sicheren Betriebszustand.

Die CPU prüft folgende Punkte:

- Hardware, z. B. ob alle Baugruppen verfügbar sind
- Ob für die CPU die Standardeinstellungen gelten oder Parametersätze vorliegen
- Ob die Randbedingungen für das programmierte Anlaufverhalten stimmen

### **Siehe auch**

Grundlagen zu den Betriebszuständen von S7-CPU (Seite 833)

## **Grundlagen zum Urlöschen**

### **Funktion**

Das Urlöschen der CPU ist nur im Betriebszustand STOP möglich.

Beim Urlöschen wird die CPU in einen so genannten "Anfangszustand" versetzt. Das bedeutet:

- Der Arbeitsspeicher-Inhalt sowie remanente und nichtremanente Daten werden gelöscht.
- Anschließend wird der Ladespeicher (Code- und Datenbausteine) in den Arbeitsspeicher kopiert. Das hat zur Folge, dass die DBs keine Aktualwerte mehr haben, sondern nur ihre Anfangswerte.
- Eine bestehende Online-Verbindung zwischen Ihrem PG/PC und der CPU wird abgebaut.
- Der Diagnosepuffer, die Uhrzeit, die IP-Adresse, die Hardware-Projektierung und laufende Force-Aufträge bleiben erhalten.

## Speicherbereiche

### Wissenswertes zu Memory Cards

#### Funktion der Memory Card

Die SIMATIC Memory Card für eine S7-1200 ist eine von Siemens vorformatierte SD-Speicherkarte für das Anwenderprogramm der CPU.

Sie dürfen nur Dateien und Ordner löschen. Wenn Sie die Memory Card mit Windows-Mitteln formatieren, z. B. mit einem handelsüblichen Kartenleser, machen Sie die Memory Card unbrauchbar als Speichermedium für eine S7-CPU.



#### Einstellen des Kartentyps

Sie können die Memory Card als Übertragungskarte, als Programmkarte oder als Firmware-Update-Karte verwenden.

Zur Einstellung des Kartentyps stecken Sie die Memory Card in den Kartenleser des Programmiergeräts und wählen den Ordner "Card Reader/USB-Speicher" in der Projektnavigation. In den Eigenschaften der markierten Memory Card bestimmen Sie den Kartentyp:

- **Programm**  
Bei der Verwendung als Programmkarte können Sie das Anwenderprogramm auf die Memory Card laden. Dadurch wird der interne Ladespeicher des Geräts durch die Memory Card ersetzt und der interne Ladespeicher wird gelöscht. Das Anwenderprogramm ist in diesem Fall vollständig von der Memory Card lauffähig. Wird die Memory Card mit dem Anwenderprogramm entfernt, ist kein Programm mehr vorhanden.
- **Übertragung**  
Bei der Verwendung als Übertragungskarte können Sie das Anwenderprogramm von der Memory Card in den internen Ladespeicher der CPU übertragen. Die Memory Card können Sie danach entfernen.
- **Firmware Card**  
Auf einer Memory Card kann Firmware für S7-1200 Baugruppen gespeichert werden. Daher ist es möglich mit Hilfe einer speziell vorbereiteten Memory Card ein Firmware Update durchzuführen. Ebenso kann eine Sicherungskopie einer Baugruppen-Firmware auf der Memory Card gespeichert werden.

## Objekte aus dem Projekt auf Memory Card übertragen

Wenn die Memory Card im PG bzw. im externen Kartenleser steckt, können Sie folgende Objekte aus der Projektnavigation auf die Memory Card übertragen:

- Einzelne Bausteine (Mehrfachselektion möglich)  
In diesem Fall wird ein konsistentes Übertragen angeboten, d. h. die Abhängigkeiten der Bausteine untereinander durch Bausteinaufrufe wird berücksichtigt.
- PLC  
In diesem Fall werden alle ablaufrelevanten Objekte, u. a. Bausteine und die Hardware-Konfiguration auf die Memory Card übertragen - wie beim Laden.

Um die Übertragung durchzuführen, können Sie die Objekte per Drag&Drop übertragen oder Sie verwenden den Befehl "Card Reader/USB-Speicher > Auf Memory Card schreiben" im Menü "Projekt".

## Objekte von der Memory Card in das Projekt übertragen

Einzelne Bausteine (Mehrfachselektion möglich) übertragen Sie per Drag&Drop in das Projekt. Eine Hardware-Konfiguration ist nicht von der Memory Card ins Projekt übertragbar.

## Firmware aktualisieren mit einer Memory Card

Die neuesten Firmware-Dateien erhalten Sie aus dem Internet über die Seiten des Service & Support:

<http://support.automation.siemens.com> (<http://support.automation.siemens.com/WW/view/de/34143537>)

Speichern Sie die Firmware-Dateien auf der Festplatte und stecken Sie die SIMATIC Memory Card in den Card Reader Ihres Programmiergeräts.

Um die Datei auf die Memory Card zu speichern, markieren Sie in der Projektnavigation die Memory Card im Ordner "Card Reader/USB-Speicher". Wählen Sie das Kontextmenü "Card Reader/USB-Speicher > Firmware-Update Memory Card erstellen".

Folgen Sie anschließend den Anweisungen im Service & Support-Portal zur Durchführung des Firmware-Updates mit ihrer CPU.

Durch die Aktualisierung der Firmware erhält die CPU einen neuen Firmwarestand. Wenn Sie die CPU im Projekt verwendet haben, müssen Sie durch einen Gerätetausch die bereits projektierte CPU durch die CPU mit dem neuen Firmwarestand offline nachziehen und das Programm bzw. die Konfiguration anpassen und anschließend laden.

## Siehe auch

Hardware-Komponente tauschen (Seite 408)

Wissenswertes zu CPU-Firmware-Versionen und STEP 7-Versionen (Seite 908)

## Ladespeicher

### Funktion

Jede CPU besitzt einen internen Ladespeicher. Die Größe dieses internen Ladespeichers ist von der jeweiligen CPU abhängig.

Dieser interne Ladespeicher kann extern durch den Einsatz von Memory Cards ersetzt werden. Ohne gesteckte Memory Card verwendet die CPU den internen Ladespeicher, mit gesteckter Memory Card verwendet die CPU die Memory Card als Ladespeicher.

Die Größe des verwendbaren externen Ladespeichers kann jedoch nicht größer sein als der interne Ladespeicher, selbst wenn die gesteckte SD-Karte mehr freien Platz bietet.

### Siehe auch

Memory Cards verwenden (Seite 310)

## Arbeitsspeicher

### Funktion

Der Arbeitsspeicher ist ein nicht remanenter Speicherbereich und nimmt die für den Programmablauf relevanten Teile des Anwenderprogramms auf. Das Anwenderprogramm wird ausschließlich im Arbeits- und Systempeicher bearbeitet.

## Systemspeicher

### Systemspeicherbereiche

### Funktion

Der Systempeicher enthält die Speicherelemente, die jede CPU dem Anwenderprogramm zur Verfügung stellt, z. B. das Prozessabbild der Ein- und Ausgänge und Merker.

Durch Verwendung der entsprechenden Operationen adressieren Sie in Ihrem Programm die Daten direkt in den jeweiligen Operandenbereich.

Die folgende Tabelle zeigt die Operandenbereiche des Systemspeichers:

| Operandenbereich           | Beschreibung   | Zugriff über Einheiten der folgenden Größe: | S7- Notation |
|----------------------------|--|---|--------------|
| Prozessabbild der Ausgänge | Zu Beginn des Zyklus schreibt die CPU die Werte aus dem Prozessabbild der Ausgänge in die Ausgabebaugruppen. | Ausgang (Bit)                               | A            |
|                            |  | Ausgangsbyte                                | AB           |
|                            |  | Ausgangswort                                | AW           |
|                            |  | Ausgangsdoppelwort                          | AD           |
| Prozessabbild der Eingänge | Während des Zyklus liest die CPU die Eingänge aus den  | Eingang (Bit)                               | E            |



| Operandenbereich           | Beschreibung   | Zugriff über Einheiten der folgenden Größe: | S7- Notation |
|----------------------------|--|---|--------------|
|                            | Eingabebaugruppen und speichert die Werte in das Prozessabbild der Eingänge.   | Eingangsbyte                                | EB           |
|                            |  | Eingangswort                                | EW           |
|                            |  | Eingangsdoppelwort                          | ED           |
| Merker                     | Dieser Bereich stellt Speicherplatz für im Programm errechnete Zwischenergebnisse zur Verfügung.   | Merker (Bit)                                | M            |
|                            |  | Merkerbyte                                  | MB           |
|                            |  | Merkerwort                                  | MW           |
|                            |  | Merkerdoppelwort                            | MD           |
| Datenbaustein              | Datenbausteine speichern Informationen für das Programm. Sie können entweder so definiert sein, dass alle Codebausteine auf sie zugreifen können (globale DBs), oder sie sind einem bestimmten FB oder SFB zugeordnet (Instanz-DB).<br>Voraussetzung: Baustein-Attribut "Optimierter Bausteinzugriff" nicht aktiviert. | Datenbit                                    | DBX          |
|                            |  | Datenbyte                                   | DBB          |
|                            |  | Datenwort                                   | DBW          |
|                            |  | Datendoppelwort                             | DBD          |
| Lokaldaten                 | Dieser Speicherbereich nimmt die temporären Lokaldaten eines Bausteins für die Dauer der Bearbeitung dieses Bausteins auf.<br>Voraussetzung: Baustein-Attribut "Optimierter Bausteinzugriff" nicht aktiviert.<br>Empfehlung: symbolisch auf Lokaldaten (Temp) zugreifen.   | Lokaldatenbit                               | L            |
|                            |  | Lokaldatenbyte                              | LB           |
|                            |  | Lokaldatenwort                              | LW           |
|                            |  | Lokaldatendoppelwort                        | LD           |
| Peripheriebereich Eingänge | Die Peripheriebereiche der Ein- und Ausgänge erlauben direkten Zugriff auf zentrale und dezentrale Ein- und Ausgabebaugruppen.   | Peripherieeingangsbit                       | <Variable>:P |
|                            |  | Peripherieeingangsbyte                      |              |
|                            |  | Peripherieeingangswort                      |              |
|                            |  | Peripherieeingangsdoppelwort                |              |
| Peripheriebereich Ausgänge |  | Peripherieausgangsbit                       |              |
|                            |  | Peripherieausgangsbyte                      |              |
|                            |  | Peripherieausgangswort                      |              |
|                            |  | Peripherieausgangsdoppelwort                |              |

## Siehe auch

Diagnosepuffer (Seite 846)

Grundlagen zu den Prozessabbildern (Seite 844)

Zugriff auf die Peripherieadressen (Seite 847)

## Remanente Speicherbereiche

### Remanente Speicherbereiche

Um Datenverluste bei Spannungsausfall zu vermeiden, können Sie bestimmte Daten als remanent kennzeichnen. Diese werden in einem remanenten Speicherbereich abgelegt. Ein remanenter Speicherbereich ist ein Bereich, dessen Inhalt bei einem Neustart (Warmstart), d. h. nach Ausschalten der Versorgungsspannung und nach Einschalten bei einem Übergang von STOP nach RUN erhalten bleibt.

Als remanent können Sie die folgenden Daten definieren:

- Merker: Die genaue Breite des remanenten Speicherbereichs können Sie für Merker in der PLC-Variablen-tabelle oder im Belegungsplan festlegen.
- Variablen eines Funktionsbausteins (FB): In der Schnittstelle eines FBs können Sie einzelne Variablen als remanent definieren, wenn für diesen Baustein der optimierte Bausteinzugriff aktiviert ist. Wenn der optimierte Bausteinzugriff für einen FB nicht aktiviert ist, können Remanenzeinstellungen nur im zugeordneten Instanz-Datenbaustein vorgenommen werden.
- Variablen eines globalen Datenbausteins: In einem globalen Datenbaustein können Sie abhängig von der Einstellung für den Zugriff entweder einzelne oder alle Variablen des Bausteins als remanent definieren:
  - Baustein mit optimiertem Zugriff: Remanenz ist für jede einzelne Variable einstellbar.
  - Baustein mit Standardzugriff: Remanenzeinstellung gilt für alle Variablen des DBs; entweder sind alle Variablen remanent oder keine Variable ist remanent.

## Siehe auch

Neustart (Warmstart) (Seite 836)

## Prozessabbild der Ein- und Ausgänge

### Grundlagen zu den Prozessabbildern

#### Funktion

Werden im Anwenderprogramm die Operandenbereiche Eingänge (E) und Ausgänge (A) angesprochen, dann werden nicht die Signalzustände auf den digitalen Signalbaugruppen abgefragt oder geändert, sondern ein Speicherbereich im Systemspeicher der CPU. Dieser Speicherbereich wird als Prozessabbild bezeichnet.

## Vorteile des Prozessabbilds

Der Zugriff auf das Prozessabbild hat gegenüber dem direkten Zugriff auf die Ein- und Ausgabebaugruppen den Vorteil, dass der CPU für die Dauer der zyklischen Programmbearbeitung ein konsistentes Abbild der Prozesssignale zur Verfügung steht. Wenn sich während der Programmbearbeitung ein Signalzustand auf einer Eingabebaugruppe ändert, bleibt der Signalzustand im Prozessabbild bis zur Prozessabbildaktualisierung im nächsten Zyklus erhalten. Dadurch können Sie ein Eingangssignal in Ihrem Programm mehrmals abfragen und erhalten immer eine konsistente Eingangsinformation.

Außerdem benötigt der Zugriff auf das Prozessabbild wesentlich weniger Zeit als der direkte Zugriff auf die Signalbaugruppen, weil sich das Prozessabbild im internen Speicher der CPU befindet.

## Aktualisieren der Prozessabbilder

### Ablauf

Die Speicherbereiche der Prozessabbilder werden vom Betriebssystem zyklisch aktualisiert, wenn nicht durch Konfiguration etwas anderes festgelegt wurde. Die Aktualisierung der Prozessabbilder der Ein- und Ausgänge erfolgt in dieser Reihenfolge:

1. Die internen Aufgaben des Betriebssystems werden ausgeführt.
2. Das Prozessabbild aller Ausgänge (PAA) wird auf die Ausgänge der Baugruppen geschrieben.
3. Der Zustand aller Eingänge wird ins Prozessabbild der Eingänge (PAE) gelesen.
4. Das Anwenderprogramm mit allen darin aufgerufenen Bausteinen wird abgearbeitet.

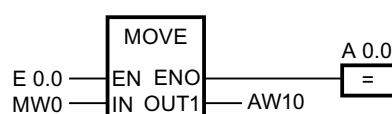
Das Betriebssystem steuert selbstständig das Schreiben des Prozessabbilds der Ausgänge auf die Ausgänge der Baugruppen und das Einlesen des Prozessabbilds der Eingänge.

### Besonderheiten

Sie haben die Möglichkeit, über einen direkten Peripheriezugriff direkt auf Eingänge bzw. Ausgänge zuzugreifen.

- Wenn eine Anweisung direkt auf einen Ausgang zugreift und die Ausgangsadresse liegt im Prozessabbild der Ausgänge, dann wird das Prozessabbild des betreffenden Ausgangs nachgeführt.
- Wenn eine Anweisung direkt auf einen Ausgang zugreift und die Ausgangsadresse liegt **nicht** im Prozessabbild der Ausgänge, dann wird das Prozessabbild des betreffenden Ausgangs **nicht** nachgeführt.

### Beispiel für einen normalen Zugriff auf die Peripherie über das Prozessabbild



Aktualisiere AW10 im Peripheriebereich Ausgänge mit dem Wert von MW0.

## Peripheriezugriffsfehler (PZF) bei Prozessabbild-Aktualisierung

Bei einem Fehler während der Prozessabbild-Aktualisierung (Peripheriezugriffsfehler) reagiert die CPU mit der voreingestellten Systemreaktion "STOP".

### Siehe auch

Anfangsadresse einer Baugruppe (Seite 846)

Zugriff auf die Peripherieadressen (Seite 847)

Anlauffähigkeiten (Seite 836)

## Diagnosepuffer

### Funktion

Der Diagnosepuffer ist Teil des Systemspeichers der CPU. Er enthält die von der CPU und diagnosefähigen Baugruppen erkannten Fehler. Dazu gehören u. a. die folgenden Ereignisse:

- Jeder Betriebszustandswechsel der CPU (z. B. NETZEIN, Übergang in den Betriebszustand STOP, Übergang in den Betriebszustand RUN)
- Jeder Diagnosealarm

Für die S7-1200-CPU hat der Diagnosepuffer eine Kapazität von 50 Einträgen, wovon bei einem NETZ-AUS-NETZ-EIN-Übergang die letzten (jüngsten) 10 Einträge erhalten bleiben.

Die Einträge können nur durch Rücksetzen der CPU auf Werkseinstellungen gelöscht werden.

Mittels der Online- und Diagnosesicht können Sie den Inhalt des Diagnosepuffers auslesen.

### Siehe auch

Grundlagen zum Diagnosepuffer (Seite 1009)

## Peripheriedatenbereich

### Anfangsadresse einer Baugruppe

### Definition

Die Anfangsadresse ist die niedrigste Byteadresse einer Baugruppe. Sie stellt die Anfangsadresse des Nutzdatenbereichs der Baugruppe dar.

### Baugruppen-Anfangsadressen konfigurieren

Die Zuordnung zwischen Adressen, die im Anwenderprogramm verwendet werden, und den Baugruppen erfolgt durch die Konfiguration der Baugruppen.

Die beim Platzieren der Baugruppen automatisch vergebenen Anfangsadressen können Sie in den Eigenschaften der Baugruppe ändern (Gruppe "E/A-Adressen...").

Sie haben außerdem Möglichkeit einzustellen, ob die Adressen im Prozessabbild liegen oder nicht.

## Zugriff auf die Peripherieadressen

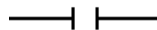
### Peripherieadressen

Wenn Sie eine Baugruppe in der Gerätesicht stecken, liegen deren Nutzdaten im Prozessabbild der S7-1200 CPU (Voreinstellung). Die CPU übernimmt automatisch den Datenaustausch zwischen Baugruppe und Prozessabbild-Bereich beim Aktualisieren der Prozessabbilder.

Wenn das Programm direkt auf die Baugruppe und nicht über das Prozessabbild zugreifen soll, dann müssen Sie die Peripherieadresse durch ":P" ergänzen.

`%E0.0:P`

`"TAG_1":P`



Das ist der Fall z. B. bei einem zeitkritischen Programm, in dem die Ausgänge noch im selben Zyklus beeinflusst werden sollen.

## Grundlagen zur Programmbearbeitung

### Ereignisse und OBs

### Ereignisse und OBs

Das Betriebssystem der S7-1200-CPU basiert auf Ereignissen. Es sind zwei Arten von Ereignissen zu unterscheiden:

- Ereignisse, die einen OB starten können
- Ereignisse, die keinen OB starten können

Ein Ereignis, das einen OB starten kann, hat nach seinem Auftreten diese Reaktion zur Folge:

- Falls Sie dem Ereignis einen OB zugeordnet haben, wird dieser OB aufgerufen. Wenn momentan ein Aufruf dieses OB nicht möglich ist, wird das Ereignis in eine seiner Priorität entsprechende Warteschlange eingereiht.
- Falls Sie dem Ereignis keinen OB zugeordnet haben, wird die voreingestellte Systemreaktion durchgeführt.

Ein Ereignis, das keinen OB starten kann, hat nach seinem Auftreten die für die zugehörige Ereignisklasse voreingestellte Systemreaktion zur Folge.

8.1 Geräte und Netze konfigurieren

Der Ablauf des Anwenderprogramms basiert damit auf Ereignissen, der Zuordnung von OBs zu den Ereignissen und auf dem Code, der sich innerhalb der OBs befindet oder von dort aus aufgerufen wird.

Die folgende Tabelle gibt einen Überblick über diejenigen Ereignisse, die einen OB starten können, samt zugehörigen Ereignisklassen und OBs. Sie ist nach der voreingestellten OB-Priorität sortiert. Dabei entspricht 1 der niedrigsten Priorität.

| Ereignisklasse                            | OB-Nr.      | OB-Anzahl   | Startereignis   | OB-Priorität (Voreinstellung) |
|---|-------------|---|---|-------------------------------|
| Zyklisches Programm                       | 1, >= 123   | >= 1  | Anlaufende oder Ende des letzten Zyklus-OB  | 1                             |
| Anlauf                                    | 100, >= 123 | >=0   | STOP-RUN-Übergang   | 1                             |
| Uhrzeitalarm                              | >= 123      | Max. 2  | Startzeitpunkt wurde erreicht   | 2                             |
| Verzögerungsalarm                         | >= 123      | Max. 4  | Verzögerungszeit abgelaufen   | 3                             |
| Weckalarm                                 | >= 123      |   | Äquidistanter Zeitabschnitt abgelaufen  | 8                             |
| Prozessalarm                              | >= 123      | Max. 50 (mehr verwendbar durch DETACH und ATTACH) | <ul style="list-style-type: none"> <li>• Steigende Flanke (max. 16)</li> <li>• Fallende Flanke (max. 16)</li> </ul>   | 18                            |
|   |             |   | <ul style="list-style-type: none"> <li>• HSC: Zählwert= Referenzwert (max. 6)</li> <li>• HSC: Zählrichtung geändert (max. 6)</li> <li>• HSC: Externes Reset (max. 6)</li> </ul>   | 18                            |
| Statusalarm                               | 55          | 0 oder 1  | CPU hat Statusalarm empfangen   | 4                             |
| Update-Alarm                              | 56          | 0 oder 1  | CPU hat Update-Alarm empfangen  | 4                             |
| Hersteller- bzw. profilspezifischer Alarm | 57          | 0 oder 1  | CPU hat hersteller- bzw. profilspezifischen Alarm empfangen   | 4                             |
| Diagnosealarm                             | 82          | 0 oder 1  | Baugruppe hat Fehler erkannt  | 5                             |
| Ziehen/Stecken-Alarm                      | 83          | 0 oder 1  | Ziehen/Stecken von Baugruppen der dezentralen Peripherie  | 6                             |
| Baugruppenträgerfehler                    | 86          | 0 oder 1  | Fehler im Ein-/Ausgabesystem der dezentralen Peripherie   | 6                             |
| Zeitfehler                                | 80          | 0 oder 1  | <ul style="list-style-type: none"> <li>• Zyklusüberwachungszeit überschritten</li> <li>• Aufgerufener OB wird noch ausgeführt</li> <li>• Uhrzeitalarm verpasst</li> <li>• Uhrzeitalarm verpasst während STOP</li> <li>• Warteschlange übergelaufen</li> <li>• Alarmverlust durch zu hohe Alarmlast</li> </ul> | 22                            |

Die folgende Tabelle beschreibt diejenigen Ereignisse, die keinen OB-Start zur Folge haben, und die jeweilige Reaktion des Betriebssystems. Sie ist nach der Priorität der Ereignisse sortiert.

| Ereignisklasse   | Ereignis   | Ereignis-Priorität | Systemreaktion |
|--|--|--------------------|----------------|
| Ziehen/Stecken von zentralen Baugruppen                  | Ziehen/Stecken einer Baugruppe                           | 21                 | STOP           |
| Peripheriezugriffsfehler bei Prozessabbildaktualisierung | Peripheriezugriffsfehler bei Prozessabbildaktualisierung | 22                 | Ignorieren     |

| Ereignisklasse                             | Ereignis  | Ereignis-Priorität | Systemreaktion |
|--|---|--------------------|----------------|
| Programmierfehler                          | Programmierfehler in einem Baustein, für den Sie die vom Betriebssystem bereitgestellten Systemreaktionen verwenden (Hinweis: Falls Sie die lokale Fehlerbehandlung aktiviert haben, wird die im Baustein programmierte Fehlerbehandlung wirksam.)        | 23                 | RUN            |
| Peripheriezugriffsfehler                   | Peripheriezugriffsfehler in einem Baustein, für den Sie die vom Betriebssystem bereitgestellten Systemreaktionen verwenden (Hinweis: Falls Sie die lokale Fehlerbehandlung aktiviert haben, wird die im Baustein programmierte Fehlerbehandlung wirksam.) | 24                 | RUN            |
| Zyklusüberwachungszeit 2 mal überschritten | Zyklusüberwachungszeit 2 mal überschritten  | 27                 | STOP           |

### Zuordnung zwischen OBs und Ereignissen

Mit Ausnahme des Zyklischen Programms und des Anlaufprogramms kann einem Ereignis nur genau ein OB zugeordnet werden. In manchen Ereignisklassen wie z. B. bei den Prozessalarmen kann ein und derselbe OB aber mehreren Ereignissen zugeordnet sein.

Die Zuordnung zwischen OB und Ereignis erfolgt bei der Konfigurierung der Hardware. Eine einmal getroffene Zuordnung können Sie zur Laufzeit mit den Anweisungen ATTACH und DETACH wieder ändern.

### OB-Priorität und Ablaufverhalten

Die S7-1200-CPU unterstützen die Prioritäten 1 (niedrigste Priorität) bis 27 (höchste Priorität). Ein OB erhält die Priorität seines Starterereignisses.

Grundsätzlich gilt, dass OBs prioritätsgesteuert bearbeitet werden: Die OBs mit höchster Priorität werden zuerst bearbeitet. Ereignisse gleicher Priorität werden in der Reihenfolge ihres Auftretens bearbeitet. Das bedeutet:

- Jeder OB mit einer Priorität  $\geq 2$  unterbricht das zyklische Programm.
- Ein OB der Priorität 2 bis 25 kann von keinem Ereignis der Priorität 2 bis 25 unterbrochen werden. Das gilt auch dann, wenn ein Ereignis mit höherer Priorität als der des momentan aktiven OB auftritt. Diese Ereignisse werden zu einem späteren Zeitpunkt bearbeitet.
- Ein aufgetretener Zeitfehler (Priorität 26) unterbricht jeden anderen OB.

### OB-Startinformation

Manche OBs haben eine Startinformation, manche nicht. Darauf wird bei der Beschreibung des jeweiligen OB näher eingegangen.

## Siehe auch

Ereignisbasierte Programmbearbeitung (Seite 850)

## Ereignisbasierte Programmbearbeitung

### OB-Priorität und Ablaufverhalten

Die S7-1200-CPU's unterstützen die Prioritäten 1 (niedrigste Priorität) bis 27 (höchste Priorität). Ein OB erhält die Priorität seines Starterereignisses.

Ein Alarm-OB kann ausschließlich von einem Zeitfehler-OB unterbrochen werden. Dies gilt auch dann, wenn ein Ereignis mit höherer Priorität als der des momentan aktiven OB auftritt. Mit Ausnahme des Zeitfehler-OB kann also nur ein einziger Alarm-OB aktiv sein.

Tritt während der Bearbeitung eines Alarm-OB ein weiteres Ereignis auf, wird dieses Ereignis in eine seiner Priorität entsprechende Warteschlange eingereiht. Die Starterereignisse einer Warteschlange werden zu einem späteren Zeitpunkt in der Reihenfolge ihres Auftretens abgearbeitet.

### Programmbearbeitung in der CPU

Die zyklischen OBs werden durch Alarm-OBs unterbrochen.

Alarm-OBs selbst können nur durch Zeitfehler-OBs unterbrochen werden.

Das folgende Bild zeigt den grundsätzlichen Ablauf:





- ④ Während der Alarmbearbeitung tritt ein weiteres alarmauslösendes Ereignis auf. Dieses neue Ereignis wird in eine Warteschlange eingereiht. Erst wenn die Bearbeitung des aktuellen Alarm-OBs abgeschlossen ist, rufen die Ereignisse in der Warteschlange hintereinander die jeweils zugeordneten OBs auf, und zwar nach folgenden Regeln:
  - Ereignisse werden nach ihrer Priorität bearbeitet (mit der höchsten Priorität beginnend)
  - Ereignisse der gleichen Priorität werden in chronologischer Reihenfolge bearbeitet
- ⑤ Die zyklischen OBs werden hintereinander abgearbeitet.

### Hinweise zur Warteschlange

- Für jede Prioritätsklasse (aufzurufende OBs gleicher Priorität) existiert eine eigene Warteschlange. Jede dieser Warteschlangen hat eine vorgegebene Größe.
- Wenn eine Warteschlange gefüllt ist und ein weiteres Ereignis eintritt, dann wird dieses neue Ereignis verworfen, es ist damit verloren. Gleichzeitig wird ein "Zeitfehler-Ereignis" erzeugt. In der Startinfo des Zeitfehler-OBs (OB 80) wird die Information zum fehlerverursachenden OB mitgegeben. Im Zeitfehler-OB kann eine entsprechende Reaktion programmiert werden, z. B. eine Meldung erzeugt werden.

### Beispiel für ein Prozessalarm-Ereignis

Am Beispiel eines Prozessalarm-auslösenden Moduls wird die Funktionsweise der ereignisorientierten Bearbeitung des Programms in der S7-1200-CPU beschrieben.

### Prozess-Ereignisse und ihre Priorität

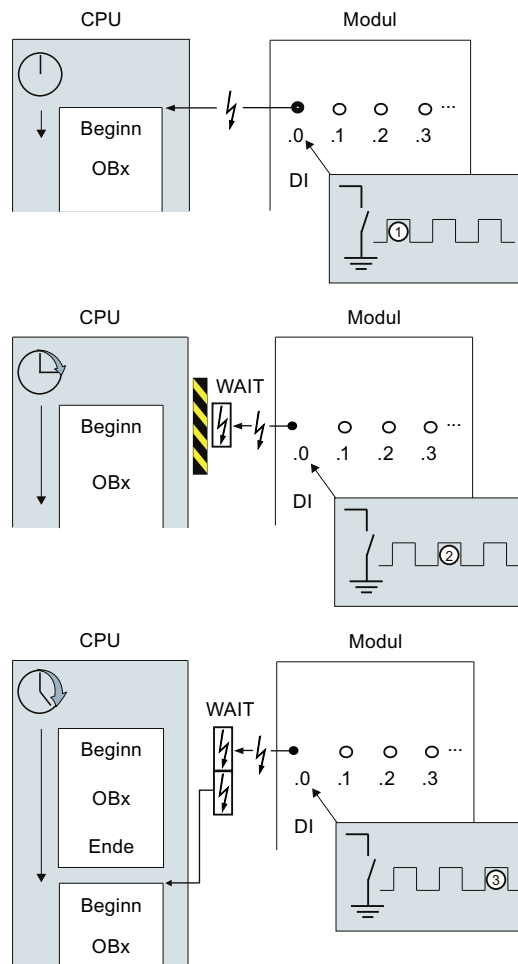
Prozess-Ereignisse sind Ereignisse, die von der Peripherie (z. B. einem digitalen Eingang) ausgelöst werden und in der Folge in der S7-1200-CPU den Aufruf eines zugeordneten OBs bewirken. OBs, die einem Prozess-Ereignis zugeordnet sind, werden daher Prozessalarm-OBs genannt.

Beispiele für Prozess-Ereignisse und deren Priorität:

- Prozess-Ereignis "steigende Flanke" bzw. "fallende Flanke" einer alarmauslösenden Baugruppe: Der daraufhin gestartete Prozessalarm-OB hat immer die Priorität 5.
- Prozess-Ereignisse vom schnellen Zähler
  - Zählwert entspricht dem Referenzwert
  - Zählrichtung ändern
  - Externer Reset des schnellen ZählersDer daraufhin gestartete Prozessalarm-OB hat immer die Priorität 6.

Das folgende Bild zeigt den zeitlichen Ablauf der Prozessalarm-Bearbeitung: Bei unmittelbar aufeinanderfolgenden Prozessalarm-Ereignissen wird das zweite Prozessalarm-auslösende Ereignis so lange in der Warteschlange zurückgehalten, bis der erste OBx abgearbeitet ist. Erst wenn der OBx abgearbeitet ist, kann das nächste Prozessalarm-auslösende Ereignis

den zugehörigen OBx starten. Weitere Prozessalarm-auslösende Ereignisse werden nach diesem Prinzip ebenfalls in die Warteschlange eingereiht.



## Prozessalarm-Bearbeitung

- ① Ein Prozessalarm-auslösendes Ereignis, z. B. steigende Flanke am Eingang, ruft einen OB auf, der diesem Ereignis zugeordnet ist.
- ② Wenn ein neues Prozessalarm-auslösendes Ereignis eintritt, während der OB bearbeitet wird, dann wird dieses Ereignis in eine Warteschlange eingereiht.
- ③ Das neue Prozessalarm-auslösende Ereignis startet den Prozessalarm-OB, der diesem Ereignis zugeordnet ist.

## Zuordnen des alarmauslösenden Ereignisses

Das alarmauslösende Ereignis ordnen Sie in der Gerätesicht, Eigenschaften der Eingänge, einem OB zu.

- Ein alarmauslösendes Ereignis kann genau nur einem OB zugeordnet werden.
- Ein OB kann mehreren alarmauslösenden Ereignissen zugeordnet sein. Sie können z. B. sowohl steigende als auch fallende Flanke ein und demselben Alarm-OB zuordnen, um bei jeder Änderung des Eingangssignals unmittelbar auf die selbe Art und Weise reagieren zu können.
- Der gestartete OB kann einen Zyklus-OB bei jeder Anweisung unterbrechen. Konsistenz bei Datenzugriffen ist sichergestellt bis zur Doppelwort-Größe.
- Je nach Baugruppe können verschiedene alarmauslösende Ereignisse parametrisiert werden, z. B. steigende Flanke sowie fallende Flanke am Eingang.
- Die Zuordnung zwischen alarmauslösendem Ereignis und gestartetem OB legen Sie beim Projektieren des alarmauslösenden Moduls fest. Mit der Anweisung DETACH können Sie aber im gestarteten Prozessalarm-OB diese Zuordnung lösen und mit der Anweisung ATTACH demselben Ereignis einen anderen OB zuweisen. Auf diese Weise können Sie flexibel auf externe Prozesssignale reagieren.

## Einstellen des Betriebsverhaltens

### Eigenschaften der Baugruppen ändern

### Standardeinstellungen

Alle parametrierbaren Hardware-Komponenten sind bei Lieferung auf Werte eingestellt, die für Standardanwendungen geeignet sind. Mit diesen Standardwerten können Sie die Hardware-Komponenten ohne weitere Einstellungen einsetzen.

Sie können jedoch das Verhalten und die Eigenschaften der Hardware-Komponenten parametrieren und damit auf Ihre Erfordernisse und die Gegebenheiten Ihrer Anlage einstellen. Parametrierbare Hardware-Komponenten sind z. B. CPUs, Kommunikationsmodule sowie einige analoge und digitale Signalmodule.

### Einstellen und Laden der Parameter

Wenn Sie in der Geräte- oder Netzsicht eine Hardware-Komponente selektiert haben, können Sie die Eigenschaften im Inspektorfenster einstellen. Beim Speichern einer Gerätekonfiguration mit ihren Parametern werden Daten erzeugt, die in die CPU geladen werden müssen. Von dort werden diese Daten beim Anlauf in die zugehörigen Baugruppen übertragen.

## Eigenschaften der CPUs

Von besonderer Bedeutung für das Systemverhalten sind die Eigenschaften der CPUs. Bei einer CPU können Sie z. B. einstellen:

- Schnittstellen
- Eingänge und Ausgänge
- Schnelle Zähler
- Impulsgeneratoren
- Anlaufverhalten
- Uhrzeit
- Schutzstufe
- Merker für System- und Takt
- Zykluszeit
- Kommunikationslast

Was einstellbar ist und in welchen Wertebereichen, wird durch die Eingabemöglichkeiten vorgegeben. Nicht editierbare Felder sind ausgegraut, bzw. erscheinen nicht im Eigenschaftsfenster.

## Voraussetzung

Sie haben die Hardware-Komponenten, deren Eigenschaften Sie ändern wollen, bereits auf einem Baugruppenträger angeordnet.

## Vorgehen

Um die Eigenschaften und Parameter der Hardware-Komponenten zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht die Hardware-Komponente oder Schnittstelle, die Sie bearbeiten möchten.
2. Bearbeiten Sie die Einstellungen für das gewählte Objekt:
  - In der Geräteübersicht können Sie z. B. Adressen und Namen bearbeiten.
  - Im Inspektorfenster stehen Ihnen zusätzliche Einstellmöglichkeiten zur Verfügung.

Sie müssen Ihre Eingabe nicht bestätigen, die geänderten Werte werden sofort übernommen.

## Siehe auch

Eigenschaften und Parameter bearbeiten (Seite 408)

Einführung zum Laden einer Konfiguration (Seite 824)

## CPU-Eigenschaften

### Übersicht über die CPU-Eigenschaften

#### Übersicht

Die folgende Tabelle zeigt die Eigenschaften der CPU im Überblick:

| Gruppe                 | Eigenschaften          | Beschreibung   |
|------------------------|------------------------|--|
| Allgemein              | Projektinformationen   | Allgemeine Angabe zur Beschreibung der gesteckten CPU. Bis auf die Steckplatznummer können Sie die Angaben ändern.   |
|                        | Kataloginformationen   | Nur lesbare Informationen aus dem Hardware-Katalog für diese CPU.  |
| PROFINET-Schnittstelle | Allgemein              | Name und Kommentar für diese PROFINET-Schnittstelle. Der Name ist auf 110 Zeichen begrenzt.  |
|                        | Ethernet-Adressen      | Auswahl, ob die PROFINET-Schnittstelle vernetzt ist. Wenn bereits Subnetze im Projekt angelegt wurden, stehen Sie zur Auswahl in der Klappliste. Ansonsten kann mit der Schaltfläche "Neues Subnetz hinzufügen" ein neues Subnetz angelegt werden.<br><br>Unter IP-Protokoll stehen Angaben zur IP-Adresse, Subnetzmaske und IP-Router-Verwendung im Subnetz. Wenn ein IP-Router verwendet wird, ist die Angabe der IP-Adresse des IP-Routers notwendig.   |
|                        | Erweiterte Optionen    | Name, Kommentar und weitere Einstellmöglichkeiten des Ports der Ethernet-Schnittstelle.  |
|                        | Uhrzeitsynchronisation | Einstellungen zur Uhrzeitsynchronisation im NTP-Uhrzeitformat. Das NTP (network time protocol) ist ein allgemeines Verfahren zur Synchronisation von Systemuhren in lokalen und globalen Netzwerken.<br><br>Beim NTP-Verfahren sendet die Schnittstelle der CPU in regelmäßigen Zeitabständen Uhrzeitanfragen (im Client-Modus) an NTP-Server im Subnetz (LAN), deren Adressen hier parametrisiert werden müssen. Anhand der Antworten der Server wird die zuverlässigste und genaueste Uhrzeit ermittelt und synchronisiert. Der Vorteil dieses Verfahrens liegt in der möglichen Uhrzeitsynchronisation über Subnetzgrenzen hinweg. Die Genauigkeit ist abhängig von der Qualität des verwendeten NTP-Servers. |
| DI#/DO#                | Allgemein              | Name und Kommentar für die integrierten Digitaleingänge der CPU.   |

| Gruppe                | Eigenschaften          | Beschreibung   |
|-----------------------|------------------------|--|
|                       | Digitaleingänge        | <p>Für die Digitaleingänge sind Eingangsverzögerungen einstellbar. Die Eingangsverzögerungen lassen sich gruppenweise einstellen (jeweils für 4 Eingänge).</p> <p>Für jeden Digitaleingang ist die Erkennung einer steigenden und einer fallenden Flanke aktivierbar. Diesem Ereignis kann ein Name und ein Prozessalarm zugeordnet werden.</p> <p>Abhängig von der CPU können an verschiedenen Eingängen Impulsabgriffe aktiviert werden. Wenn der Impulsabgriff aktiviert ist, werden auch die Flanken von Impulsen erkannt, die kürzer sind als die Zykluszeit des Programms.</p>   |
|                       | Digitalausgänge        | <p>Für alle Digitalausgänge ist die Reaktion bei Betriebszustandswechsel von RUN nach STOP einstellbar: Der Zustand kann entweder eingefroren werden (entspricht einem letzten Wert halten) oder Sie stellen einen Ersatzwert ein ("0" oder "1")</p>   |
|                       | E/A-Adressen           | Der Adressraum der Eingabe- und Ausgabeadressen sowie das Prozessabbild werden festgelegt.   |
|                       | HW-Kennung             | Die Hardware-Kennung des Geräts wird angezeigt.  |
| AI#                   | Allgemein              | Name und Kommentar für die integrierten Analogeingänge der CPU.  |
|                       | Analogeingänge         | <p>In der Rauschminderung werden durch die eingestellte Integrationszeit die Störfrequenzen der angegebenen Frequenz (in Hz) unterdrückt.</p> <p>Die Kanaladresse, Messart, Spannungsbereich, Glättung und Überlaufdiagnose sind in der Gruppe "Kanal #" einzustellen. Messart und Spannungsbereich sind fest eingestellt auf Spannung, 0 bis 10 V.</p> <p>Durch die Glättung von Analogwerten wird ein stabiles Analogsignal für die Weiterverarbeitung zur Verfügung gestellt. Die Glättung der Analogwerte ist sinnvoll bei langsamen Messwertänderungen, z. B. bei Temperaturmessungen. Die Messwerte werden mittels digitaler Filterung geglättet. Die Glättung wird erreicht, indem die Baugruppe Mittelwerte aus einer festgelegten Anzahl von gewandelten (digitalisierten) Analogwerten bildet. Die parametrisierte Stufe (gering, mittel, stark) bestimmt die Anzahl der Analogsignale, die zur Mittelwertbildung herangezogen wird.</p> <p>Wenn die Überlaufdiagnose aktiviert ist, wird bei Überlauf ein Diagnoseereignis erzeugt.</p> |
|                       | E/A-Adressen           | Der Adressraum der Eingabeadressen sowie das Prozessabbild werden festgelegt.  |
|                       | HW-Kennung             | Die Hardware-Kennung des Geräts wird angezeigt.  |
| Schnelle Zähler (HSC) | Schnelle Zähler (HSC)# | <p>Schnelle Zähler werden typischerweise als Antrieb für Zählwerke eingesetzt.</p> <p>Siehe: Schnelle Zähler konfigurieren (Seite 866)</p>   |

| Gruppe                      | Eigenschaften                                 | Beschreibung  |
|-----------------------------|---|---|
| Impulsgeneratoren (PTO/PWM) | PTO#/PWM#                                     | <p>Ein Impulsgenerator wird aktiviert und kann mit Projektinformationen versorgt werden.</p> <p>Für die Parametrierung eines aktivierten Impulsgenerators wird die Verwendung als PWM (Pulse Width Modulation; "Impulsdauermodulation") oder als PTO (Puls Train Output; "Impulsfolge") eingestellt.</p> <p>Für den PWM müssen Ausgangsquelle, Zeitbasis, Impulsdauerformat, Zykluszeit und anfängliche Impulsdauer festgelegt werden. Als Hardwareausgang ist ein Impulsausgang vorgegeben. Der PWM-Ausgang wird von der Anweisung CTRL_PWM gesteuert, siehe CTRL_PWM.</p> <p>Für den PTO muss die Ausgangsquelle angegeben werden. Als Hardwareausgänge sind ein Impulsausgang und ein Richtungsausgang vorgegeben. Ein PTO wird zusammen mit einem HSC in der Zählart "Bewegungsachse" betrieben und vom Technologieobjekt Motion Control gesteuert (siehe Stichwort "Motion Control S7-1200")</p> <p>In den E/A-/Diagnoseadressen wird die Hardware-Kennung angezeigt und bei Auswahl der Funktion PWM kann der Adressraum der Ausgabeadressen und das Prozessabbild ausgewählt werden.</p> |
| Anlauf                      | Anlauf nach NETZ-EIN                          | <p>Einstellung des Anlaufverhaltens nach einem NETZ-AUS/NETZ-EIN- Übergang.</p> <p>Siehe: Grundlagen zum Betriebszustand "ANLAUF" (Seite 835)</p>   |
|                             | Vergleich Sollausbau zu Istausbau             | <p>Legt fest, ob Module (SM, SB, CM, CP oder auch die CPU) ersetzt werden können:</p> <ul style="list-style-type: none"> <li>Anlauf der CPU nur bei Kompatibilität</li> <li>Anlauf der CPU auch bei Unterschieden</li> </ul> <p>Beispiel: Ein Signalmodul mit 16 Digitaleingängen und 16 Digitalausgängen (DI16/DQ16) kann ein kompatibler Ersatz für ein Signalmodul mit 8 Digitalausgängen (DQ8) oder 4 Digitaleingängen (DI4) sein.</p>  |
|                             | Parametrierungszeit für dezentrale Peripherie | <p>Legt einen maximalen Zeitraum (Standard: 60000 ms) fest, in dem die dezentrale Peripherie hochlaufen muss. (Die CMs und CPs werden während des Anlaufs von der CPU mit Spannung und Kommunikationsparametern versorgt. Diese Parametrierungszeit gestattet einen Zeitraum, während dessen die an CM oder CP angeschlossenen E/A-Module hochlaufen müssen.)</p> <p>Die CPU geht in RUN, sobald die dezentrale Peripherie hochgelaufen und betriebsbereit ist, unabhängig vom Parameter "Parametrierungszeit für dezentrale Peripherie". Ist die dezentrale Peripherie während dieses Zeitraums nicht hochgelaufen, geht die CPU ohne die dezentrale Peripherie in RUN.</p>  |
| Zyklus                      | Maximale Zykluszeit und Mindestzykluszeit.    | <p>Festlegen einer maximalen Zykluszeit oder einer festen Mindestzykluszeit.</p> <p>Wenn die Zykluszeit die Zyklusüberwachungszeit überschreitet, dann geht die CPU in den Betriebszustand STOP.</p> <p>Siehe: Zykluszeit und Zyklusüberwachungszeit (Seite 860)</p>  |



| Gruppe                 | Eigenschaften   | Beschreibung   |
|------------------------|---|--|
| Kommunikationslast     | Maximaler Anteil des Zyklus für die Kommunikation (in%) | Steuert in einem gewissen Rahmen die Dauer von Kommunikationsprozessen, die immer auch die Zykluszeit verlängern. Kommunikationsprozesse können z. B. sein: Datenübertragung zu einer anderen CPU oder Laden von Bausteinen (über PC angestoßen).<br>Siehe: Zyklusbelastung durch Kommunikation (Seite 861)  |
| System- und Taktmerker | Systemmerkerbits und Taktmerkerbits                     | Systemmerker verwenden Sie für folgende Abfragen: <ul style="list-style-type: none"> <li>• Ist der aktuelle Zyklus der erste Zyklus nach NETZ-AUS/ NETZ-EIN?</li> <li>• Haben sich Diagnosezustandsänderungen gegenüber dem vorangegangenen Zyklus ergeben?</li> <li>• Abfrage auf "1" (high)</li> <li>• Abfrage auf "0" (low)</li> </ul> Taktmerker wechseln periodisch in festgelegten Zeitabständen ihren Wert.<br>Siehe: Verwenden von Systemmerkern (Seite 879)<br>Siehe: Verwenden von Taktmerkern (Seite 879) |
| Webserver              | Automatische Aktualisierung                             | Schickt die angeforderte Webseite mit aktuellen CPU-Daten periodisch zum Webbrowser. Die Periodendauer geben Sie unter "Aktualisierungsintervall" ein. Die automatische Aktualisierung ist nur aktivierbar, wenn der Webserver aktiviert ist.<br>Siehe: Auto-Hotspot   |
|                        | Anwenderdefinierte Webseiten                            | Ermöglicht den Zugriff auf frei gestaltete Webseiten der CPU mit einem Webbrowser.<br>Siehe: Auto-Hotspot  |
| Uhrzeit                | Ortszeit und Sommerzeit                                 | Einstellung der Zeitzone, in der die CPU betrieben wird und Einstellung der Sommerzeit/Winterzeit-Umstellung.  |
| Schutz                 | Schutz und Passwort für Lese-/Schreibzugriff            | Einstellen des Lese-/Schreibschutzes und des Passworts für den Zugriff auf die CPU.<br>Siehe: Einstellmöglichkeiten für die Schutzstufe (FW V1 bis V3) (Seite 880)<br>Siehe: Einstellmöglichkeiten für den Schutz (FW ab V4) (Seite 881)   |
| Verbindungsressourcen  | -   | Anzeige der verfügbaren, reservierten und bereits konfigurierten Verbindungsressourcen der CPU.  |
| Adressübersicht        | -   | Tabellarische Darstellung aller von der CPU verwendeten Adressen für die integrierten Ein/Ausgänge sowie für die gesteckten Module. Adressen, die von keinem Modul verwendet werden, sind als Lücke dargestellt.<br>Die Ansicht lässt sich filtern nach <ul style="list-style-type: none"> <li>• Eingabeadressen</li> <li>• Ausgabeadressen</li> <li>• Adresslücken</li> </ul>   |

**Siehe auch**

- Ein- und Ausgangsadressen festlegen (Seite 629)
- Prozessalarm-OBs parametrieren (Seite 903)
- Zugriff auf die Peripherieadressen (Seite 847)
- Adressierung von Baugruppen (Seite 628)
- Besonderheiten beim Anlauf (Seite 838)

**Zykluszeit und Zyklusüberwachungszeit**

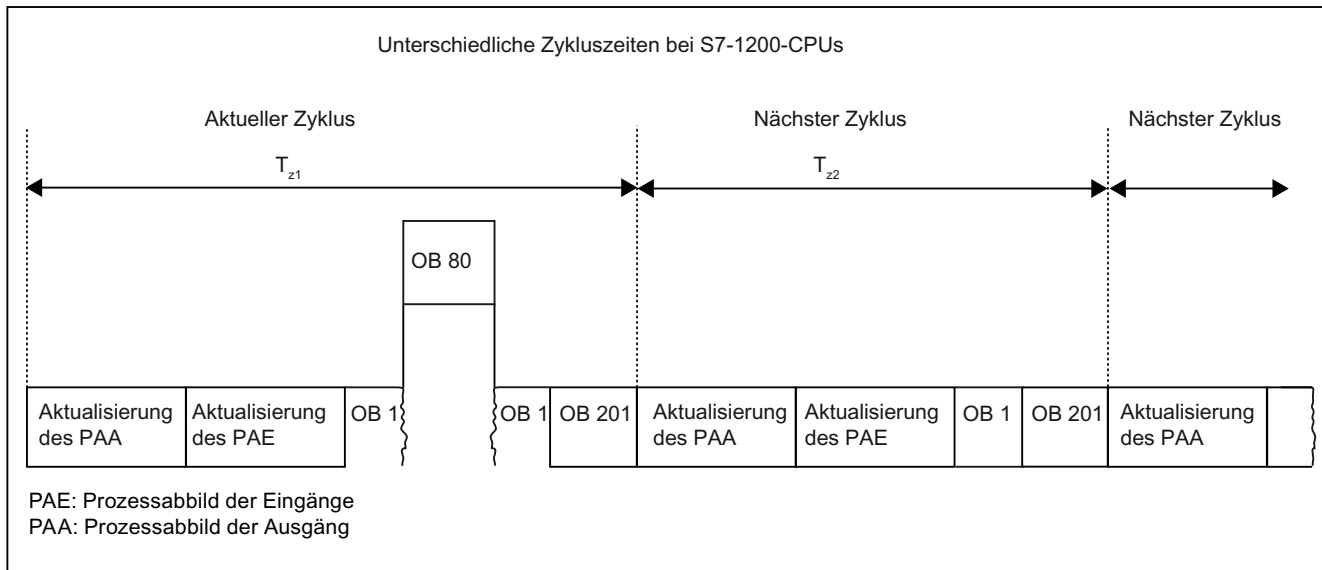
**Funktion**

Die Zykluszeit ist die Zeit, die das Betriebssystem für die zyklische Programmbearbeitung sowie für alle diesen Zyklus unterbrechende Programmteile benötigt. Die Programmbearbeitung kann unterbrochen werden durch:

- Zeitfehler und 2xMaxCycleTime-Fehler
- Systemtätigkeiten, z. B. Prozessabbildaktualisierung

Die Zykluszeit (TZ) ist daher nicht für jeden Zyklus gleich lang.

Das folgende Bild zeigt ein Beispiel für unterschiedliche Zykluszeiten ( $TZ1 \neq TZ2$ ) für S7-1200-CPU:



Im aktuellen Zyklus wird der hier verwendete zyklische OB (z. B. OB 1) durch einen Zeitfehler (z. B. OB 80) unterbrochen. Im Anschluss des zyklischen OBs wird der nächste Zyklus-OB 201 abgearbeitet.

## Zyklusüberwachungszeit

Das Betriebssystem überwacht die Laufzeit des zyklischen Programms auf eine projektierbare Obergrenze, die maximale Zykluszeit. Sie können durch Aufruf der Anweisung RE\_TRIGR an jeder Stelle Ihres Programms diese Zeitüberwachung neu starten.

Wenn das zyklische Programm die Zyklusüberwachungszeit überschreitet, versucht das Betriebssystem den Zeitfehler-OB (OB 80) zu starten. Ist der OB nicht vorhanden, ignoriert die CPU die Überschreitung der Zyklusüberwachungszeit.

Neben der Überwachung der Laufzeit auf Überschreitung der Zyklusüberwachungszeit wird die Einhaltung einer minimalen Zykluszeit garantiert. Das Betriebssystem verzögert dabei den Beginn eines neuen Zyklus so lange, bis die minimale Zykluszeit erreicht ist. Während dieser Wartezeit werden neue Ereignisse und Betriebssystemdienste bearbeitet.

Wenn die Zyklusüberwachungszeit ein zweites Mal überschritten wird, z. B. während der Zeitfehler-OB bearbeitet wird (2xMaxCycleTime-Fehler), geht die CPU in den Betriebszustand STOP.

## Zyklusbelastung durch Kommunikation

### Funktion

Die Zykluszeit der CPU kann sich durch Kommunikationsprozesse verlängern. Zu diesen Kommunikationsprozessen gehören z. B.:

- Datenübertragung zu einer anderen CPU
- Über ein Programmiergerät (PG) angestoßenes Laden von Bausteinen

Die Dauer dieser Kommunikationsprozesse können Sie mit dem CPU-Parameter "Zyklusbelastung durch Kommunikation" in einem gewissen Rahmen steuern.

Neben den Kommunikationsprozessen verlängern auch Testfunktionen die Zykluszeit. Dabei kann über den Parameter "Zyklusbelastung durch Kommunikation" Einfluss auf die Dauer genommen werden.

### Wirkungsweise des Parameters

Über den Parameter "Zyklusbelastung durch Kommunikation" geben Sie den Prozentsatz der gesamten CPU-Verarbeitungsleistung an, der den Kommunikationsprozessen zur Verfügung stehen soll. Diese Verarbeitungsleistung wird der Kommunikation ständig von der CPU zugewiesen. Wenn die Kommunikation diese Verarbeitungsleistung nicht benötigt, steht sie der Programmbearbeitung zur Verfügung.

### Auswirkung auf die tatsächliche Zykluszeit

Durch die Verwendung des Parameters "Zyklusbelastung durch Kommunikation" verlängert sich die Zykluszeit des zyklischen Organisationsbausteins (z. B. OB 1) um einen Faktor, der sich nach der folgenden Formel berechnen lässt:

$$\frac{100}{100 - \text{"Zyklusbelastung durch Kommunikation"}}$$

Die Formel berücksichtigt dabei den Einfluss von asynchronen Ereignissen wie Prozess- oder Weckalarmen auf die Zykluszeit nicht.

Wenn sich die Zykluszeit durch Kommunikationsprozesse verlängert, können auch mehr asynchrone Ereignisse innerhalb der Zykluszeit des zyklischen Organisationsbausteins auftreten. Dies verlängert den Zyklus zusätzlich. Die Verlängerung ist davon abhängig, wie viele Ereignisse auftreten und wie lange ihre Bearbeitung dauert.

### Beispiel 1 – keine zusätzlichen asynchronen Ereignisse:

Bei Einstellung des Parameters "Zyklusbelastung durch Kommunikation" auf 50 % kann die Zykluszeit des zyklischen Organisationsbausteins auf das Doppelte ansteigen.

### Beispiel 2 – zusätzliche asynchrone Ereignisse:

Bei einer reinen Zykluszeit von 500 ms kann sich durch eine Kommunikationslast von 50 % eine tatsächliche Zykluszeit von bis zu 1000 ms ergeben, unter der Voraussetzung, dass die CPU immer genügend Kommunikationsaufträge zum Bearbeiten hat. Wenn nun parallel dazu alle 100 ms ein Weckalarm mit 20 ms Bearbeitungszeit abläuft, dann würde sich dieser ohne Kommunikationslast mit insgesamt  $5 \cdot 20 \text{ ms} = 100 \text{ ms}$  auf den Zyklus verlängernd auswirken, d. h. die tatsächliche Zykluszeit wäre 600 ms. Da ein Weckalarm auch die Kommunikation unterbricht, wirkt er sich bei 50 % Kommunikationslast mit  $10 \cdot 20 \text{ ms}$  auf die Zykluszeit aus, d. h. in diesem Fall beträgt die tatsächliche Zykluszeit nicht 1000 ms, sondern 1200 ms.

---

#### Hinweis

Beachten Sie folgende Hinweise:

- Überprüfen Sie die Auswirkungen einer Wertänderung des Parameters "Zyklusbelastung durch Kommunikation" im Anlagenbetrieb.
  - Die Kommunikationslast muss beim Einstellen der minimalen Zykluszeit berücksichtigt werden, da es sonst zu Zeitfehlern kommen kann.
- 

## Empfehlungen

- Vergrößern Sie den Wert nur dann, wenn die CPU hauptsächlich zu Kommunikationszwecken eingesetzt wird und das Anwenderprogramm zeitunkritisch ist.
- In allen anderen Fällen sollten Sie den Wert nur verringern.

## Uhrzeitfunktionen

### Grundlagen zu den Uhrzeitfunktionen

Alle S7-1200 CPUs sind mit einer internen Uhr ausgestattet. Die Pufferung unterstützt die Anzeige der korrekten Uhrzeit für eine Unterbrechung der Spannungsversorgung für bis zu 10 Stunden.

## Uhrzeitformat

Die Uhr zeigt immer die Uhrzeit mit einer Auflösung von 1 Millisekunde und das Datum mit Wochentag an. Die durch die Sommerzeit bedingte Zeitumstellung wird berücksichtigt.

## Stellen und Lesen der Uhrzeit

### Uhrzeit stellen und lesen mit Anweisungen

Sie können Uhrzeit und Datum der CPU-Uhr über folgende Anweisungen im Anwenderprogramm stellen, starten und lesen:

- Uhrzeit stellen: "WR\_SYS\_T"
- Uhrzeit lesen "RD\_SYS\_T"
- Lokalzeit lesen "RD\_LOC\_T"
- Zeitzone setzen "SET\_TIMEZONE"

### Manuelle Einstellung

Das Lesen und Einstellen der Uhrzeit können Sie auch manuell in der Online- und Diagnosesicht unter "Funktionen > Uhrzeit einstellen" durchführen.

## Parametrieren der Uhr

### Parameter der Uhr

Über die Parameter der Uhr nehmen Sie folgende Einstellungen vor:

- Uhrzeitsynchronisation über NTP-Server aktivieren  
Aktivieren Sie das Optionskästchen, wenn die interne Uhr mit dem NTP-Synchronisationsverfahren synchronisiert werden soll.
- Netzwerk-Uhrzeitserver  
Es sind die IP-Adressen von bis zu vier NTP-Servern zu projektieren.
- Aktualisierungsintervall  
Das Aktualisierungsintervall definiert den Zeitabstand der Uhrzeitanfragen.

## Schnelle Zähler

### Allgemeines zu schnellen Zählern

### Einführung

Schnelle Zähler werden typischerweise als Antrieb für Zählwerke eingesetzt, bei denen eine Welle, die mit einer konstanten Drehzahl läuft, mit einem Winkelschrittgeber versehen ist. Der Winkelschrittgeber sorgt für eine bestimmte Anzahl von Zählwerten pro Umdrehung sowie für

einen Rücksetzimpuls einmal pro Umdrehung. Der bzw. die Taktgeber und der Rücksetzimpuls des Winkelschrittgebers liefern die Eingänge für den schnellen Zähler.

Den verschiedenen S7-1200 CPUs stehen eine unterschiedliche Anzahl von schnellen Zählern zu Verfügung:

| S7-1200 CPU                         | Anzahl HSCs                      | Bezeichnung HSC      |
|-------------------------------------|----------------------------------|----------------------|
| CPU 1211C                           | 3 (mit digitalem Signalboard 4)* | HSC1...3 (und HSC5)* |
| CPU 1212C                           | 4 (mit digitalem Signalboard 5)* | HSC1...4 (und HSC5)* |
| CPU 1214C<br>CPU 1215C<br>CPU 1217C | 6                                | HSC1...6             |

\* mit DI2/DO2 Signalboard

### Funktionsweise

Der erste von mehreren voreingestellten Werten wird in den schnellen Zähler geladen. Die gewünschten Ausgänge werden für die Zeitspanne aktiviert, während der aktuelle Wert des Zählers kleiner als der voreingestellte Wert ist. Der Zähler wird so eingerichtet, dass ein Interrupt auftritt, wenn der aktuelle Wert des Zählers gleich dem voreingestellten Wert ist oder wenn der Zähler zurückgesetzt wird.

Wenn der aktuelle Wert gleich dem voreingestellten Wert ist und es zu einem Interrupt-Ereignis kommt, dann wird ein neuer voreingestellter Wert geladen und der nächste Signalzustand für die Ausgänge gesetzt. Tritt ein Interrupt-Ereignis auf, weil der Zähler zurückgesetzt wird, dann werden der erste voreingestellte Wert und die ersten Signalzustände der Ausgänge gesetzt und der Zyklus wiederholt.

Da die Interrupts in einer sehr viel geringeren Geschwindigkeit auftreten als der schnelle Zähler zählt, kann eine präzise Steuerung der schnellen Operationen mit relativ geringem Einfluss auf den gesamten Zyklus des Automatisierungssystems implementiert werden. Da Sie Interrupts bestimmten Interrupt-Programmen zuordnen können, kann jede neue Voreinstellung in einem getrennten Interrupt-Programm geladen werden, damit so der Zustand einfach gesteuert werden kann.

---

#### Hinweis

Sie können auch alle Interrupt-Ereignisse in einem einzigen Interrupt-Programm bearbeiten.

---

### Zählalgorithmen der unterschiedlichen Zähler

Alle Zähler arbeiten auf die gleiche Weise, aber nicht jeder schnelle Zähler unterstützt alle Zählalgorithmen. Es gibt vier grundlegende Zählalgorithmen:

- Einphasenzähler mit interner Richtungssteuerung
- Einphasenzähler mit externer Richtungssteuerung
- Zweiphasenzähler mit 2 Takteingängen
- A/B-Zähler

Jeder schnelle Zähler kann mit oder ohne Rücksetzeingang verwendet werden. Wird der Rücksetzeingang aktiviert, setzt dieser den aktuellen Wert zurück. Der aktuelle Wert bleibt solange zurückgesetzt, bis der Rücksetzeingang deaktiviert wird.

## Siehe auch

Schnelle Zähler konfigurieren (Seite 866)

Abhängigkeiten der Zählerart und Zählereingänge (Seite 865)

## Abhängigkeiten der Zählerart und Zählereingänge

### Allgemeines zu Zählerart und Zählereingängen

Den schnellen Zählern können Sie Zählerarten und Zählereingänge und darüber die Funktionen wie Taktgeber, Richtungssteuerung und Zurücksetzen zuweisen. Dabei gelten folgende Regeln:

- Ein Eingang kann nicht für zwei verschiedene Funktionen verwendet werden.
- Wird ein Eingang nicht von der aktuellen Zählerart des definierten schnellen Zählers benötigt, kann er für andere Zwecke genutzt werden.

Setzen Sie beispielsweise HSC1 in Zählerart 1 ein, in der die Eingänge E0.0 und E0.3 benötigt werden, können Sie E0.1 für Flankeninterrupts oder für HSC2 verwenden.

Setzen Sie beispielsweise HSC1 und HSC5 ein, werden bei den Zählerarten Zählen und Frequenz immer die Eingänge E0.0 (HSC1) und E1.0 (HSC5) verwendet. Daher stehen dann diese beiden Eingänge beim Betrieb der Zähler für keine anderen Funktionen zur Verfügung.

Zusätzliche Eingänge stehen Ihnen bei Verwendung eines digitalen Signalboards zur Verfügung.

### Übersicht über die Abhängigkeiten von Zählerart und Zählereingängen

| Zählerart         | Beschreibung                                    | Eingänge                         |                                  |                                  |
|-------------------|---|----------------------------------|----------------------------------|----------------------------------|
|                   | HSC1  | E0.0 (CPU)<br>E4.0 (Signalboard) | E0.1 (CPU)<br>E4.1 (Signalboard) | E0.3 (CPU)<br>E4.3 (Signalboard) |
|                   | HSC2  | E0.2 (CPU)<br>E4.2 (Signalboard) | E0.3 (CPU)<br>E4.3 (Signalboard) | E0.1 (CPU)<br>E4.1 (Signalboard) |
|                   | HSC3*   | E0.4 (CPU)                       | E0.5 (CPU)                       | E0.7 (CPU)                       |
|                   | HSC4 (nur CPU 1212/14/15/17C)                   | E0.6 (CPU)                       | E0.7 (CPU)                       | E0.5 (CPU)                       |
|                   | HSC5 (nur CPU 1214/15/17C)**                    | E1.0 (CPU)<br>E4.0 (Signalboard) | E1.1 (CPU)<br>E4.1 (Signalboard) | E1.2 (CPU)<br>E4.3 (Signalboard) |
|                   | HSC6 (nur CPU 1214/15/17C)**                    | E1.3 (CPU)                       | E1.4 (CPU)                       | E1.5 (CPU)                       |
| Zählen / Frequenz | Einphasenzähler mit interner Richtungssteuerung | Taktgeber                        | -                                | -                                |
| Zählen            |   |                                  |                                  | Rücksetzen                       |
| Zählen / Frequenz | Einphasenzähler mit externer Richtungssteuerung | Taktgeber                        | Richtung                         | -                                |

| Zählerart         | Beschreibung                         | Eingänge   |                     |            |
|-------------------|--------------------------------------|--|---------------------|------------|
| Zählen            |                                      |  |                     | Rücksetzen |
| Zählen / Frequenz | Zweiphasenzähler mit 2 Takteingängen | Taktgeber vorwärts   | Taktgeber rückwärts | -          |
| Zählen            |                                      |  |                     | Rücksetzen |
| Zählen / Frequenz | A/B-Zähler                           | Taktgeber A  | Taktgeber B         | -          |
| Zählen            |                                      |  |                     | Rücksetzen |
| Bewegungsachse    | Impulsgeneratoren PWM/PTO            | HSC1 und HSC2 unterstützen die Zählart Bewegungsachse für die Impulsgeneratoren PTO1 und PTO2: <ul style="list-style-type: none"> <li>• HSC1 wertet für PTO1 den Ausgang A0.0 für die Anzahl der Impulse aus</li> <li>• HSC2 wertet für PTO2 den Ausgang A0.2 für die Anzahl der Impulse aus</li> </ul> Als Ausgang für die Bewegungsrichtung wird A0.1 verwendet. |                     |            |

\* HSC3 ist für CPU 1211 nur ohne Rücksetz-Eingang verwendbar

\*\* HSC5 ist bei Verwendung eines Signalboards DI2/DO2 auch für CPU 1211/12 nutzbar

**Siehe auch**

Allgemeines zu schnellen Zählern (Seite 863)

Schnelle Zähler konfigurieren (Seite 866)

**Schnelle Zähler konfigurieren**

**Voraussetzung**

Eine S7-1200 CPU ist in die Hardware-Konfiguration eingefügt.

**Vorgehen**

Um einen schnellen Zähler zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Geräte- oder Netzsicht eine S7-1200 CPU.
2. Klicken Sie im Inspektorfenster unter "Eigenschaften > Schnelle Zähler" auf den gewünschten schnellen Zähler:
  - CPU 1211C: HSC1 bis HSC3 (mit Signalboard DI2/DO2 auch HSC5)
  - CPU 1212C: HSC1 bis HSC4 (mit Signalboard DI2/DO2 auch HSC5)
  - CPU 1214C / 1215C / 1217C: HSC1 bis HSC6



3. Aktivieren Sie den schnellen Zähler in der Parametergruppe "Allgemein" mit dem entsprechenden Kontrollkästchen.

---

**Hinweis**

Wenn Sie eine CPU 1211C oder CPU 1212C mit Signalboard DI2/DO2 verwenden, können Sie auch den schnellen Zähler HSC5 aktivieren.

---

**Hinweis**

Wenn Sie die Impulsgeneratoren aktivieren und als PTO1 oder PTO2 betreiben, werden von diesen die zugehörigen schnellen Zähler HSC1 bzw. HSC2 mit der Zählart "Bewegungsachse" für die Auswertung der Hardware-Ausgänge verwendet. Wenn Sie die schnellen Zähler HSC1 oder HSC2 für andere Zählaufgaben konfigurieren, können diese nicht von den Impulsgeneratoren PTO1 bzw. PTO2 verwendet werden.

---

Bei Bedarf können Sie hier einen Namen und einen Kommentar zu dem schnellen Zähler eingeben.

4. Definieren Sie die Funktionsweise des schnellen Zählers in der Parametergruppe "Funktion":
  - Zählart: Wählen Sie über die Klappliste aus, was gezählt werden soll.
  - Betriebsphase: Wählen Sie über die Klappliste den Zählalgorithmus aus.
  - Eingangsquelle: Wählen Sie über die Klappliste als Eingangsquelle für die Zählimpulse die On-board CPU-Eingänge oder die Eingänge eines optionalen digitalen Signalboards aus.
  - Zählrichtung wird angegeben von: Wenn eine einphasige Betriebsphase ausgewählt wurde, wählen Sie über die Klappliste aus, ob die Zählrichtung intern durch einen SFB-Parameter des Anwenderprogramms oder extern durch einen Digitaleingang vorgegeben werden soll.
  - Anfängliche Zählrichtung: Wenn bei der Zählrichtung das Anwenderprogramm als interne Richtungssteuerung eingestellt ist, können Sie über die Klappliste die Zählrichtung bei Zählbeginn auswählen.
  - Frequenzmessperiode: Wenn als Zählart die Frequenz eingestellt ist, können Sie über die Klappliste die Dauer der Frequenzmessperioden auswählen.
5. Legen Sie die Anfangswerte und Rücksetzbedingung des schnellen Zählers in der Parametergruppe "Auf Anfangswerte zurücksetzen" fest:
  - Anfänglicher Zählerwert: Geben Sie einen Startwert für den schnellen Zähler vor.
  - Anfänglicher Referenzwert: Geben Sie einen Höchstwert für den schnellen Zähler vor.

Sie können hier außerdem festlegen, ob der schnelle Zähler einen Rücksetzeingang verwenden soll und den entsprechenden Signalpegel für den Rücksetzeingang über die Klappliste einstellen.

6. Konfigurieren Sie die Reaktion des schnellen Zählers auf bestimmte Ereignisse in der Parametergruppe "Ereigniskonfiguration". Die folgenden Ereignisse können einen Alarm auslösen:
  - Der Zählerwert entspricht dem Referenzwert.
  - Ein externes Rücksetzereignis wurde generiert.
  - Ein Richtungswechsel wurde angestoßen.

Aktivieren Sie eine Alarmreaktion über das Kontrollkästchen und geben Sie einen Namen und über die Klappliste einen Hardware-Interrupt für den Alarm ein.

7. Vergeben Sie die Anfangsadresse für den schnellen Zähler in der Parametergruppe "E/A-Diagnoseadressen".
- 

#### **Hinweis**

In der Parametergruppe "Hardware-Eingänge" können Sie sehen, welche Hardware-Eingänge und Werte für den Takt, die Richtungsbestimmung, den Rücksetzimpuls und die maximale Zählgeschwindigkeit verwendet werden.

---

## **Ergebnis**

Sie haben nun die Parameter des schnellen Zählers an Ihre Projekterfordernisse angepasst.

## **Siehe auch**

Allgemeines zu schnellen Zählern (Seite 863)

Abhängigkeiten der Zählerart und Zählereingänge (Seite 865)

## **Punkt-zu-Punkt-Kommunikation**

### **Überblick über die Punkt-zu-Punkt-Kommunikation**

Die PtP-Kommunikation ist eine Kommunikation über eine serielle Schnittstelle, welche standardisierte UART-Datenübertragung (Universal Asynchronous Receiver Transmitter) verwendet. Die S7-1200 verwendet Kommunikationsmodule mit einer RS232- oder RS485-Schnittstelle, um die PtP-Kommunikation aufzubauen.

## **Funktionen der Punkt-zu-Punkt-Kommunikation**

Die Punkt-zu-Punkt-Kommunikation (PtP) bietet eine große Menge an Anwendungsmöglichkeiten:

- Direktes Senden von Informationen zu einem externen Gerät, wie z. B. einem Drucker oder einem Barcode-Lesegerät
- Empfangen von Informationen von externen Geräten, wie z. B. Barcode-Lesern, RFID-Lesern, Kameras und optische Systeme von Drittherstellern sowie von vielen anderen Geräten.
- Austausch von Informationen mit Drittherstellergeräten, wie z. B. GPS-Geräten, Radio-Modems und vielen mehr

## **Das Freeport-Protokoll**

Die S7-1200 unterstützt das Freeport-Protokoll für zeichenbasierte serielle Kommunikation. Mit Hilfe der Freeport-Kommunikation kann das Datenübertragungsprotokoll vollständig über das Anwenderprogramm konfiguriert werden.

Siemens stellt Bibliotheken mit Freeport-Kommunikations-Funktionen zur Verfügung, die Sie im Anwenderprogramm verwenden können:

- USS Drive Protocol
- Modbus RTU Master Protocol
- Modbus RTU Slave Protocol

## **Siehe auch**

Kommunikations-Port konfigurieren (Seite 870)

## **Verwenden von RS232- und RS485-Kommunikationsmodulen**

### **Kommunikationsmodule mit RS232- und RS485-Schnittstellen**

In einer S7-1200 CPU können zwei unterschiedliche Kommunikationsmodule verwendet werden:

- RS232-Kommunikationsmodul
- RS485-Kommunikationsmodul

Die Kommunikationsmodule können mit der S7-1200 CPU über den linksseitigen I/O-Kanal verbunden werden. Das Stecken von bis zu drei Modulen ist möglich.

## **Eigenschaften der Kommunikationsmodule**

Die Kommunikationsmodule zeichnen sich durch folgende Eigenschaften aus:

- Unterstützung des Freeport-Protokolls
- Konfiguration über das Anwenderprogramm mit Hilfe von erweiterten Anweisungen und Bibliotheksfunktionen

## Kommunikations-Port konfigurieren

### Kommunikationsport konfigurieren

Nachdem Sie ein Kommunikationsmodul mit einer RS232- oder RS485-Schnittstelle gesteckt haben, müssen Sie zunächst die Schnittstellenparameter einstellen. Die Schnittstelle parametrieren Sie entweder in den Eigenschaften der Schnittstelle oder Sie steuern die Schnittstellenparameter über das Anwenderprogramm mit Hilfe der PORT\_CFG Anweisung. Die folgende Beschreibung bezieht sich auf die grafische Projektierung.

---

#### Hinweis

Wenn Sie die Port-Einstellung über das Anwenderprogramm ändern, werden die Einstellungen der grafischen Projektierung überschrieben.

Beachten Sie weiterhin, dass über das Anwenderprogramm vorgenommene Einstellungen im Falle eines Spannungsverlusts nicht erhalten bleiben.

---

### Voraussetzung

- Ein Kommunikationsmodul ist bereits gesteckt.
- Sie befinden sich in der Gerätesicht.

### Vorgehen

Um den Kommunikations-Port zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der grafischen Darstellung innerhalb der Gerätesicht die Schnittstelle. Im Inspektorfenster werden die Eigenschaften der Schnittstelle angezeigt.
2. Wählen Sie in der Bereichsnavigation des Inspektorfensters die Gruppe "Port-Konfiguration". Die Einstellungen des Ports werden angezeigt.
3. Wählen Sie aus der Klappliste "Übertragungsgeschwindigkeit" die Geschwindigkeit für die Datenübertragung. Beachten Sie bei frei programmierter Kommunikation den Einfluss der Übertragungsgeschwindigkeit auf die Umschaltzeit.
4. Wählen Sie aus der Klappliste "Parität" die Art der Erkennung fehlerhaft übertragener Informationswörter aus.
5. Wählen Sie aus der Klappliste "Datenbits" aus, ob ein Zeichen aus acht Bit oder sieben Bit besteht.
6. Wählen Sie aus der Klappliste "Stoppbit" aus, mit wie vielen Bits das Ende eines übertragenen Worts gekennzeichnet werden soll.

7. Wählen Sie aus der Klappliste "Flusskontrolle" aus, durch welches Verfahren ein reibungsloser Datenstrom zwischen Sender und Empfänger sichergestellt werden soll. Dieser Parameter ist nur für die RS232-Schnittstelle einstellbar.
  - Geben Sie in das Feld "XON-Zeichen" einen HEX-Wert ein, bei dessen Auftreten die Nachrichtenübertragung fortgesetzt wird. Dieser Parameter ist nur für die softwaregesteuerte Datenflusskontrolle einstellbar.
  - Geben Sie in das Feld "XOFF-Zeichen" einen HEX-Wert ein, bei dessen Auftreten die Nachrichtenübertragung für die parametrisierte Wartezeit ausgesetzt werden soll. Dieser Parameter ist nur für die softwaregesteuerte Datenflusskontrolle einstellbar.
8. Geben Sie in das Feld "Wartezeit" eine Zeit in ms ein, die nach dem Ende einer Nachricht gewartet werden muss, bevor die nächste Übertragung starten kann.

---

#### Hinweis

Sie können die Schnittstelle auch in der Netzsicht projektieren. Hierzu müssen Sie aber zuerst über die tabellarische Netzsicht das Kommunikationsmodul auswählen und anschließend im Inspektorfenster die Schnittstelle auswählen. Danach können Sie wie oben beschrieben fortfahren.

---

#### Siehe auch

Einstellen der Datenflusskontrolle (Seite 871)

### Einstellen der Datenflusskontrolle

#### Datenflusskontrolle

Datenflusskontrolle ist ein Verfahren, mit dem für ein ausgeglichenes Send- und Empfangsverhalten gesorgt wird. Optimalerweise gehen durch die intelligente Steuerung keine Daten verloren. Sie stellt sicher, dass ein Gerät nicht mehr Informationen versendet als der empfangende Partner verarbeiten kann.

Für die Datenflusskontrolle gibt es zwei Verfahren:

- Hardwaregesteuerte Datenflusskontrolle
- Softwaregesteuerte Datenflusskontrolle

Bei beiden Methoden müssen die DSR-Signale der Kommunikationspartner beim Beginn der Übertragung aktiv sein. Sind die DSR-Signale inaktiv, wird nicht mit der Übertragung begonnen.

Das RS232-Kommunikationsmodul kann mit beiden Verfahren umgehen. Das RS485-Kommunikationsmodul unterstützt keine Datenflusskontrolle.

#### Hardwaregesteuerte Datenflusskontrolle

Hardwaregesteuerte Datenflusskontrolle erfolgt über die Signale Request-to-send (RTS) und Clear-to-send (CTS). Mit dem RS232-Kommunikationsmodul wird das RTS-Signal über den Ausgang von Pin 7 übertragen. Das CTS-Signal wird über Pin 8 empfangen.

Ist die hardwaregesteuerte Datenflusskontrolle aktiviert, wird das RTS-Signal immer dann aktiv gesetzt, wenn Daten gesendet werden. Gleichzeitig wird das CTS-Signal überwacht, um festzustellen, ob das empfangende Gerät Daten annehmen kann. Ist das CTS-Signal aktiv, kann das Modul Daten solange übertragen bis das CTS-Signal inaktiv wird. Wird das CTS-Signal inaktiv, muss die Datenübertragung für die parametrisierte Wartezeit ausgesetzt werden. Ist das CTS-Signal auch nach der parametrisierten Wartezeit noch inaktiv, wird die Datenübertragung abgebrochen und ein Fehler an das Anwenderprogramm zurückgemeldet.

### Datenflusskontrolle durch Hardware-Handshaking

Wird die Datenflusskontrolle durch Hardware-Handshaking geregelt, setzt das sendende Gerät das RTS-Signal standardmäßig aktiv. Ein Gerät, wie z. B. ein Modem, kann in diesem Fall zu jeder Zeit Daten übertragen. Es wartet nicht auf das CTS-Signal des Empfängers. Das sendende Gerät überwacht selbst seine eigene Übertragung indem es nur eine begrenzte Anzahl an Nachrichten-Frames (Zeichen) sendet, beispielsweise um einen Overflow im Empfangspuffer zu verhindern. Kommt es dennoch im Empfangspuffer zu einem Overflow, muss das übertragende Gerät die Nachricht zurückhalten und einen Fehler an das Anwenderprogramm zurückmelden.

### Softwaregesteuerte Datenflusskontrolle

Softwaregesteuerte Datenflusskontrolle verwendet bestimmte Zeichen innerhalb der Nachrichten, welche die Übertragung steuern. Diese Zeichen sind ASCII-Zeichen, die unter XON und XOFF parametrisiert werden.

XOFF gibt an, wann eine Übertragung ausgesetzt werden muss. XON gibt an, wann eine Übertragung fortgesetzt werden kann.

Empfängt das sendende Gerät das XOFF-Zeichen, muss es für die parametrisierte Wartezeit mit dem Senden aussetzen. Wird nach der parametrisierten Wartezeit das XON-Zeichen gesendet, wird die Übertragung fortgeführt. Kommt nach der Wartezeit kein XON-Zeichen, wird ein Fehler an das Anwenderprogramm zurückgegeben.

Software-Datenflusskontrolle erfordert Full-Duplex-Kommunikation, da der empfangende Partner das XON-Zeichen während der laufenden Übertragung senden muss.

### Siehe auch

Kommunikations-Port konfigurieren (Seite 870)

### Konfiguration der Nachrichtenübertragung

#### Frei programmierbare Kommunikation

Sie können den Datenverkehr zwischen einem Kommunikationsmodul und einem extern über die serielle Schnittstelle angeschlossenen Gerät frei steuern. Dafür müssen Sie selbst ein Kommunikationsprotokoll definieren. In der frei programmierbaren Kommunikation werden ASCII- und Binärprotokolle zur Nachrichtenübertragung unterstützt.

Innerhalb des Kommunikationsprotokolls müssen Sie festlegen, anhand welcher Kriterien Anfang und Ende einer übertragenen Nachricht im laufenden Datenstrom erkannt werden sollen.

Die frei programmierbare Kommunikation kann nur im Betriebszustand RUN aktiviert werden. Beim Versetzen in den Betriebszustand STOP wird die frei programmierbare Kommunikation angehalten.

### **Festlegen des Kommunikationsprotokolls**

Das Kommunikationsprotokoll können Sie folgendermaßen festlegen:

- Mit Hilfe des Anwenderprogramms
  - Das Verhalten beim Senden von Daten wird über die Anweisung SEND\_CFG gesteuert.
  - Das Verhalten beim Empfangen von Daten wird über die Anweisung RCV\_CFG gesteuert.
- Mit Hilfe der grafischen Parametrierung im Inspektorfenster

---

#### **Hinweis**

Wenn Sie das Kommunikationsprotokoll über das Anwenderprogramm ändern, werden die Einstellungen der grafischen Projektierung überschrieben.

Beachten Sie, dass über das Anwenderprogramm vorgenommene Einstellungen im Falle eines Spannungsverlusts nicht erhalten bleiben.

---

### **Siehe auch**

Frei programmierbare Kommunikation mit RS232-Geräten (Seite 873)

Festlegen der Einstellungen für das Senden (Seite 875)

Festlegen des Nachrichtenanzugs (Seite 876)

Festlegen des Nachrichtenendes (Seite 877)

### **Frei programmierbare Kommunikation mit RS232-Geräten**

#### **RS232/PPI-Multi-Master-Kabel und frei programmierbare Kommunikation mit RS232-Geräten**

Mit Hilfe des RS232/PPI-Multi-Master-Kabels und der frei programmierbaren Kommunikation können Sie an die Kommunikationsmodule der S7-1200 viele Geräte anschließen, die mit dem RS232-Standard kompatibel sind. Das Kabel muss jedoch zunächst für den Modus "PPI/frei programmierbare Kommunikation" eingestellt werden.

### Einstellungen am Kabel

Die Schalter am Kabel müssen folgendermaßen eingestellt werden:

- Schalter 5 muss auf 0 stehen
- Schalter 6 stellt entweder den lokalen Modus (DCE) oder den entfernten Modus (DTE) ein:
  - Schalter auf 0 für den lokalen Modus
  - Schalter auf 1 für den entfernten Modus

### Umschalten zwischen Sende- und Empfangsmodus

Das RS232/PPI-Multi-Master-Kabel ist im Sendemodus, wenn Daten von der RS232-Schnittstelle an die RS485-Schnittstelle gesendet werden. Das Kabel ist im Empfangsmodus, wenn es sich im Ruhezustand befindet oder wenn Daten von der RS485-Schnittstelle an die RS232-Schnittstelle gesendet werden. Das Kabel schaltet sofort vom Empfangs- in den Sendemodus, wenn es Zeichen in der RS232-Sendeleitung erkennt.

### Unterstützte Übertragungsgeschwindigkeiten

Das RS232/PPI-Multi-Master-Kabel unterstützt Übertragungsgeschwindigkeiten zwischen 1200 Baud und 115,2 kBaud. Mit Hilfe der DIP-Schalter am PC/PPI-Kabel kann das RS232/PPI-Multi-Master-Kabel auf die gewünschte Übertragungsgeschwindigkeit eingestellt werden.

Die folgende Tabelle zeigt die Schalterstellungen für die entsprechenden Übertragungsgeschwindigkeiten:

| Übertragungsgeschwindigkeit | Umschaltzeit | Einstellungen (1 = oben) |
|-----------------------------|--------------|--------------------------|
| 115200 bit/s                | 0,15 ms      | 110                      |
| 57600 bit/s                 | 0,3 ms       | 111                      |
| 38400 bit/s                 | 0,5 ms       | 000                      |
| 19200 bit/s                 | 1,0 ms       | 001                      |
| 9600 bit/s                  | 2,0 ms       | 010                      |
| 4800 bit/s                  | 4,0 ms       | 011                      |
| 2400 bit/s                  | 7,0 ms       | 100                      |
| 1200 bit/s                  | 14,0 ms      | 101                      |

Das Kabel kehrt zum Empfangsmodus zurück, wenn sich die RS232-Sendeleitung für eine bestimmte Zeit, die als Umschaltzeit des Kabels definiert ist, im Ruhezustand befindet. Die eingestellte Übertragungsgeschwindigkeit beeinflusst die Umschaltzeit, wie in der Tabelle dargestellt.



## Einfluss der Umschaltzeit

Bei der Arbeit mit einem RS232/PPI-Multi-Master-Kabel in einem System, in dem auch die frei programmierte Kommunikation eingesetzt wird, muss das Programm die Umschaltzeit aus folgenden Gründen beachten:

- Die Kommunikationsbaugruppe reagiert auf Meldungen, die von dem RS232-Gerät gesendet werden.  
Nachdem die Kommunikationsbaugruppe vom RS232-Gerät eine Anforderung empfangen hat, muss sie das Senden der Reaktionsmeldung für einen Zeitraum verzögern, der größer oder gleich der Umschaltzeit des Kabels ist.
- Das RS232-Gerät reagiert auf Meldungen, die von der Kommunikationsbaugruppe gesendet werden.  
Nachdem die Kommunikationsbaugruppe vom RS232-Gerät eine Reaktionsmeldung empfangen hat, muss sie das Senden der nächsten Anforderungsmeldung für einen Zeitraum verzögern, der größer oder gleich der Umschaltzeit des Kabels ist.

In beiden Situationen enthält das RS232-PPI-Multi-Master-Kabel durch die Verzögerung genügend Zeit, um vom Sendemodus in den Empfangsmodus zu wechseln, damit die Daten von der RS485-Schnittstelle an die RS232-Schnittstelle gesendet werden können.

## Siehe auch

- Konfiguration der Nachrichtenübertragung (Seite 872)
- Festlegen der Einstellungen für das Senden (Seite 875)
- Festlegen des Nachrichtenanzangs (Seite 876)
- Festlegen des Nachrichtenendes (Seite 877)

## Festlegen der Einstellungen für das Senden

### Senden von Nachrichten

Sie können zwischen einzelnen Nachrichten Pausen programmieren.

Die folgende Tabelle zeigt, welche Pausen vorgegeben werden können:

| Parameter           | Definition   |
|---------------------|--|
| RTS ON Verzögerung  | Sie können die Zeit parametrieren, die nach der Sendeanforderung RTS (Request to send) vergehen muss, ehe die tatsächliche Datenübertragung beginnt. |
| RTS OFF Verzögerung | Sie können die Zeit parametrieren, die nach der vollständigen Übertragung verstreichen soll, ehe das RTS-Signal deaktiviert wird.                    |

| Parameter                        | Definition   |
|----------------------------------|--|
| Sende Pause zu Meldungsbeginn    | Sie können festlegen, dass nach dem Beginn jeder Nachrichtenübertragung zusätzlich eine Pause gesendet wird, wenn die RTS ON-Verzögerungszeit verstrichen ist.<br>Die Pausenzeit wird in Bitzeiten angegeben.  |
| Sende Idle Line nach einer Pause | Sie können festlegen, dass nach einer parametrisierten Pause zu Meldungsbeginn, das Signal "Idle Line" ausgegeben wird und die Leitung somit als "unbeschäftigt" signalisiert wird. Um den Parameter zu aktivieren, muss "Sende Pause zu Meldungsbeginn" aktiviert sein.<br>Die Dauer des "Idle Line"-Signals wird in Bitzeiten angegeben. |

**Siehe auch**

Festlegen des Nachrichtenanzangs (Seite 876)

Festlegen des Nachrichtenendes (Seite 877)

Frei programmierbare Kommunikation mit RS232-Geräten (Seite 873)

**Festlegen des Nachrichtenanzangs**

**Erkennen des Nachrichtenanzangs**

Um dem Empfänger zu signalisieren, wann die Übertragung einer Nachricht beendet ist und wann die nächste Nachrichtenübertragung beginnt, müssen im Übertragungsprotokoll Kriterien festgelegt werden, die das Ende und den Anfang einer Nachricht kennzeichnen.

Ist ein Kriterium erfüllt, welches den Nachrichtenbeginn anzeigt, wird mit dem Durchsuchen des Datenstroms nach Kriterien für das Nachrichtenende begonnen.

Zum Erkennen des Nachrichtenbeginns gibt es zwei unterschiedliche Verfahren:

- Mit beliebigem Zeichen beginnen:  
Jedes beliebige Zeichen kann den Beginn einer Nachricht definieren. Dieses Verhalten ist voreingestellt.
- Mit spezieller Bedingung beginnen:  
Der Nachrichtenbeginn wird anhand von festgelegten Bedingungen erkannt.

## Bedingungen für das Erkennen des Nachrichtenbeginns

Die folgende Tabelle zeigt die verschiedenen Möglichkeiten, wie Sie den Beginn einer Nachricht definieren können:

| Parameter  | Definition   |
|--|--|
| Erkenne Nachrichtenbeginn bei Line Break         | Der Empfänger erkennt einen Line Break, wenn der Empfangsdatenstrom für mehr als ein Zeichen Länge unterbrochen wird. Ist dies der Fall, wird durch den Line Break der Nachrichtenbeginn festgelegt.   |
| Erkenne Nachrichtenbeginn bei Idle Line          | Der Beginn einer Nachricht wird erkannt, wenn die Übertragungsleitung zum Senden für eine bestimmte Zeit (angegeben in Bitzeiten) im Zustand "Idle" ist und anschließend von einem Ereignis, wie z. B. dem Empfang eines Zeichens, gefolgt wird. |
| Erkenne Nachrichtenbeginn mit einzelner Zeichen  | Der Beginn einer Nachricht wird beim Auftreten eines bestimmten Zeichens erkannt. Das Zeichen können Sie als HEX-Wert eingeben..   |
| Erkenne Nachrichtenbeginn mit einer Zeichenkette | Der Beginn einer Nachricht wird erkannt, wenn eine der festgelegten Zeichenfolgen im Datenstrom auftritt. Sie können bis zu vier Zeichensequenzen mit jeweils bis zu fünf Zeichen festlegen.   |

Die einzelnen Bedingungen lassen sich beliebig verknüpfen.

## Siehe auch

Festlegen der Einstellungen für das Senden (Seite 875)

Frei programmierbare Kommunikation mit RS232-Geräten (Seite 873)

## Festlegen des Nachrichtendes

### Erkennen des Nachrichtendes

Um dem Empfänger zu signalisieren, wann die Übertragung einer Nachricht beendet ist und wann die nächste Nachrichtenübertragung beginnt, müssen im Übertragungsprotokoll Kriterien festgelegt werden, die das Ende und den Anfang einer Nachricht kennzeichnen.

Zum Erkennen des Nachrichtendes gibt es insgesamt sechs unterschiedliche Verfahren, die sich beliebig verknüpfen lassen. Die folgende Tabelle zeigt die verschiedenen Einstellmöglichkeiten:

| Parameter  | Definition  |
|--|---|
| Erkenne Nachrichtende durch Nachrichten-Zeitüberschreitung | Das Nachrichtende wird automatisch erkannt, wenn eine vorgegebene Maximaldauer für eine Nachricht überschritten wird. Es sind Werte von 0 bis 65535 ms einstellbar. |
| Erkenne Nachrichtende durch Antwort-Zeitüberschreitung     | Das Nachrichtende wird erkannt, wenn nach dem Übertragen von Daten in der vorgegebenen Zeit keine Antwort erfolgt. Es sind Werte von 0 bis 65535 ms einstellbar.    |

| Parameter  | Definition  |
|--|---|
| Erkenne Nachrichtenende durch Zeitüberschreitung innerhalb der Zeichen | Das Nachrichtenende wird erkannt, wenn eine vorgegebene Zeit in Bitzeiten zwischen zwei Zeichen überschritten wird. Es sind Werte von 0 bis 2500 Bitzeiten einstellbar.<br>Die S7-1200 CPU gibt eine maximale Zeit von acht Sekunden vor, auch wenn der eingestellte Wert mehr als acht Sekunden Dauer ergibt.  |
| Erkenne Nachrichtenende durch maximale Länge                           | Das Nachrichtenende wird erkannt, wenn eine maximale Länge der Nachricht überschritten wird. Es sind Werte von 1 bis 1023 Zeichen einstellbar.  |
| Lese Nachrichtenlänge aus Nachricht                                    | Die Nachricht selbst enthält Informationen über die Länge der Nachricht. Das Nachrichtenende wird erreicht, wenn der Wert, der aus der Nachricht entnommen wurde, erreicht wird. Welche Zeichen für die Auswertung der Nachrichtenlänge herangezogen werden sollen, wird mit den folgenden Parametern festgelegt: <ul style="list-style-type: none"> <li>• Offset des Längenfeldes in der Nachricht<br/>Der Wert bestimmt die Position des Zeichens in der Nachricht, welches für die Bestimmung der Nachrichtenlänge herangezogen werden soll.<br/>Es sind Werte von 1 bis 1022 Zeichen einstellbar.</li> <li>• Größe des Längenfeldes<br/>Der Wert gibt an, wie viele Zeichen ab der ersten Auswertungsposition zur Bestimmung der Nachrichtenlänge herangezogen werden sollen.<br/>Es sind Werte von 0, 1, 2 und 4 Zeichen einstellbar.</li> <li>• Den Daten folgendes Längengebiet (gehört nicht zur Nachrichtenlänge)<br/>Der Wert gibt die Anzahl der Bytes nach dem Längengebiet an, die bei der Auswertung der Nachrichtenlänge nicht berücksichtigt werden sollen.<br/>Es sind Werte von 0 bis 255 Zeichen einstellbar.</li> </ul> |
| Erkenne Nachrichtenende mit einer Zeichenkette                         | Das Ende einer Nachricht wird erkannt, wenn die festgelegte Zeichenfolgen im Datenstrom auftritt. Sie können bis zu fünf Zeichen innerhalb der Zeichenkette festlegen, die überprüft werden sollen. Wenn die festgelegten Zeichen an der richtigen Stelle innerhalb der Nachricht auftreten, wird das Nachrichtenende erkannt. Um das Nachrichtenende beispielsweise dann zu erkennen, wenn Zeichen 1 und Zeichen 3 einen bestimmten Wert besitzen, müssen die die Optionskästchen für Zeichen 1 und Zeichen 3 aktiviert und ein Zeichenwert eingetragen sein.  |

**Siehe auch**

Festlegen der Einstellungen für das Senden (Seite 875)

Frei programmierbare Kommunikation mit RS232-Geräten (Seite 873)

## Verwenden von Systemmerkern

### Systemmerker

Ein Systemmerker ist ein Merker mit definierten Werten.

Welches Merkerbyte der CPU zum Systemmerkerbyte wird, bestimmen Sie bei der Parametrierung des Systemmerkers.

### Nutzen

Systemmerker können Sie im Anwenderprogramm verwenden, um z. B. Programmteile nur im ersten Zyklus nach dem Anlauf auszuführen oder die Diagnose bei Änderung des Diagnosezustands auszuwerten. Zwei Systemmerker sind konstant 1 bzw. konstant 0.

### Bits des Systemmerkerbytes

Die folgende Tabelle zeigt die Bedeutung der Systemmerker:

| Bit des Systemmerkerbytes | 7               | 6               | 5               | 4               | 3  | 2  | 1                                  | 0  |
|---------------------------|-----------------|-----------------|-----------------|-----------------|----|----|------------------------------------|--|
| Bedeutung                 | Reserviert (=0) | Reserviert (=0) | Reserviert (=0) | Reserviert (=0) | =0 | =1 | =1 bei Änderung des Diagnosestatus | =1 im ersten Zyklus nach dem Anlauf, sonst 0 |

---

#### Hinweis

Das gewählte Merkerbyte kann nicht für die Zwischenspeicherung von Daten genutzt werden.

---

## Verwenden von Taktmerkern

### Taktmerker

Ein Taktmerker ist ein Merker, der seinen Binärzustand periodisch im Puls-Pausen-Verhältnis 1:1 ändert.

Welches Merkerbyte der CPU zum Taktmerkerbyte wird, bestimmen Sie bei der Parametrierung des Taktmerkers.

### Nutzen

Taktmerker können Sie im Anwenderprogramm verwenden, um z. B. Leuchtmelder mit Blinklicht anzusteuern oder periodisch wiederkehrende Vorgänge anzustoßen, etwa das Erfassen eines Istwerts.

## Mögliche Frequenzen

Jedem Bit des Taktmerkerbytes ist eine Frequenz zugeordnet. Die folgende Tabelle zeigt die Zuordnung:

|                         |     |       |     |      |     |     |     |     |
|-------------------------|-----|-------|-----|------|-----|-----|-----|-----|
| Bit des Taktmerkerbytes | 7   | 6     | 5   | 4    | 3   | 2   | 1   | 0   |
| Periodendauer (s)       | 2,0 | 1,6   | 1,0 | 0,8  | 0,5 | 0,4 | 0,2 | 0,1 |
| Frequenz (Hz)           | 0,5 | 0,625 | 1   | 1,25 | 2   | 2,5 | 5   | 10  |

### Hinweis

Taktmerker laufen asynchron zum CPU-Zyklus, d. h. in langen Zyklen kann sich der Zustand des Taktmerkers mehrfach ändern.

Das gewählte Merkerbyte kann nicht für die Zwischenspeicherung von Daten genutzt werden.

## Schutz

### Einstellmöglichkeiten für die Schutzstufe (FW V1 bis V3)

#### Schutzstufe

Im Folgenden erfahren Sie, wie Sie die einzelnen Schutzstufen der S7-1200 CPUs V1 bis V3 nutzen.

#### Auswirkungen der eingestellten Schutzstufe

Sie können zwischen folgenden Schutzstufen wählen:

- **Kein Schutz:** Dies entspricht dem voreingestellten Verhalten. Sie können kein Passwort eingeben. Lesender und schreibender Zugriff ist immer erlaubt.
- **Schreibschutz:** Es ist nur lesender Zugriff möglich. Sie können keine Daten in der CPU ändern und keine Bausteine und keine Konfiguration laden. Vom Schreibschutz ausgenommen sind HMI-Zugriffe und Kommunikation zwischen CPUs. Für die Auswahl dieser Schutzstufe ist die Vergabe eines Passwortes erforderlich.
- **Schreib-/Leseschutz:** Es ist im Bereich "Erreichbare Teilnehmer" und im Projekt für Teilnehmer, die online geschaltet sind, kein schreibender und kein lesender Zugriff möglich. Lediglich der CPU-Typ und die Identifikationsdaten können in der Projektnavigation unter "Erreichbare Teilnehmer" angezeigt werden. Die Anzeige von Online-Informationen oder von Bausteinen unter "Erreichbare Teilnehmer" bzw. im Projekt für Online geschaltete Teilnehmer ist möglich.  
Vom Schreibschutz ausgenommen sind HMI-Zugriffe und Kommunikation zwischen CPUs. Für die Auswahl dieser Schutzstufe ist die Vergabe eines Passwortes erforderlich.

## Verhalten einer Passwort-geschützten CPU im Betrieb

Der Schutz der CPU ist wirksam, nachdem die Einstellungen in die CPU geladen wurden.

Vor der Ausführung einer Online-Funktion wird die Zulässigkeit geprüft. Im Falle eines Passwortschutzes werden Sie zur Eingabe eines Passworts aufgefordert.

Beispiel: Die Baugruppe wurde mit Schreibschutz parametrieren und Sie wollen die Funktion "Variable steuern" ausführen. Da es sich um einen schreibenden Zugriff handelt, muss zur Ausführung der Funktion das parametrierte Passwort eingegeben werden.

Die durch Passwort geschützten Funktionen können zu einem Zeitpunkt nur von einem PG/PC ausgeführt werden. Ein weiteres PG/PC kann sich nicht mit Passwort anmelden.

Die Zugangsberechtigung zu den geschützten Daten gilt für die Dauer der Online-Verbindung oder bis die Zugangsberechtigung manuell über "Online > Zugriffsrechte löschen" wieder aufgehoben wird. Wenn das Projekt geschlossen wird, erlischt die Zugangsberechtigung ebenfalls.

---

### Hinweis

Funktionen zur Prozessführung, -beobachtung und -kommunikation können Sie nicht einschränken.

Einige Funktionen werden durch ihre Verwendung als Onlinedaten dennoch geschützt. RUN/STOP in der Taskcard "Online Tools" oder "Uhrzeit stellen" im Diagnose- und Onlineeditor sind daher schreibgeschützt.

---

## Einstellmöglichkeiten für den Schutz (FW ab V4)

### Schutzstufe

Im Folgenden erfahren Sie, wie Sie die einzelnen Zugriffsstufen der S7-1200 CPUs ab V4 nutzen.

S7-1200 CPUs bieten verschiedene Zugriffsstufen, um den Zugang zu bestimmten Funktionen einzuschränken.

Die Parametrierung der Zugriffsstufen erfolgt in einer Tabelle. Die grünen Haken in den Spalten rechts der jeweiligen Zugriffsstufe geben an, welche Operationen maximal möglich sind, ohne das Passwort dieser Zugriffsstufe zu kennen. Möchten Sie die Funktionen nicht markierter Optionskästchen nutzen, ist die Eingabe eines Passwortes notwendig.

|   |
|---|
| <b>ACHTUNG</b>  |
| <b>Projektierung einer Zugriffsstufe ersetzt nicht den Know-how-Schutz</b>  |
| Die Parametrierung von Zugriffsstufen verhindert unrechtmäßige Änderungen an der CPU, indem die Rechte zum Download eingeschränkt werden. Bausteine auf der Memory Card sind jedoch nicht schreib- oder lesegeschützt. Um den Code von Bausteinen auf der Memory Card zu schützen, verwenden Sie den Know-how-Schutz. |

## Voreingestelltes Verhalten

Voreingestellt ist die Zugriffsstufe "Vollzugriff (kein Schutz)". Jeder Nutzer kann die Hardware-Konfiguration und die Bausteine lesen und verändern. Ein Passwort ist nicht parametrierbar und wird auch für den Online-Zugriff nicht benötigt.

## Die Zugriffsstufen im Einzelnen

Bei einer S7-1200 CPU können Sie folgende Zugriffsstufen parametrieren:

- **Vollzugriff (kein Schutz):** Die Hardware-Konfiguration und die Bausteine können von jedem gelesen und verändert werden.
- **Lesezugriff:** Mit dieser Zugriffsstufe ist ohne Angabe des Passwortes nur lesender Zugriff auf die Hardware-Konfiguration und die Bausteine möglich, d. h. Sie können Hardware-Konfiguration und Bausteine ins Programmiergerät laden. Möglich ist außerdem der HMI-Zugang und Zugriff auf Diagnosedaten.  
Sie können ohne Eingabe des Passwortes keine Bausteine und keine Hardware-Konfiguration in die CPU laden. Außerdem sind ohne Passwort schreibende Testfunktionen und Firmware-Updates **nicht** möglich.
- **HMI-Zugriff:** Mit dieser Zugriffsstufe ist ohne Angabe des Passwortes nur der HMI-Zugang und der Zugriff auf Diagnosedaten möglich.  
Sie können ohne Angabe des Passwortes weder Bausteine und Hardware-Konfiguration in die CPU laden noch von der CPU Bausteine und Hardware-Konfiguration ins Programmiergerät laden. Außerdem ist ohne Passwort Folgendes **nicht** möglich: schreibende Testfunktionen, Wechsel des Betriebszustands (RUN/STOP) und Firmware-Update.
- **Kein Zugriff (kompletter Schutz):** Wenn die CPU komplett geschützt ist, dann ist weder lesender noch schreibender Zugriff auf die Hardware-Konfiguration und die Bausteine möglich. Auch der HMI-Zugriff ist nicht möglich. Die Server-Funktion für PUT/GET-Kommunikation ist in dieser Zugriffsstufe deaktiviert (nicht änderbar).  
Durch die Legitimation mit dem Passwort erhalten Sie wieder Vollzugriff auf die CPU.

## Verhalten einer Passwort-geschützten Baugruppe im Betrieb

Der Schutz der CPU ist wirksam, nachdem die Einstellungen in die CPU geladen wurden.

Vor der Ausführung einer Online-Funktion wird die Zulässigkeit geprüft. Im Falle eines Passwortschutzes werden Sie zur Passworteingabe aufgefordert.

Beispiel: Die Baugruppe wurde mit Lesezugriff parametrierbar und Sie wollen die Funktion "Variable steuern" ausführen. Da es sich um einen schreibenden Zugriff handelt, muss zur Ausführung der Funktion das parametrierbare Passwort eingegeben werden.

Die durch Passwort geschützten Funktionen können zu einem Zeitpunkt nur von einem PG/PC ausgeführt werden. Ein weiteres PG/PC kann sich nicht anmelden.

Die Zugangsberechtigung zu den geschützten Daten gilt für die Dauer der Online-Verbindung oder bis die Zugangsberechtigung manuell über "Online > Zugriffsrechte löschen" wieder aufgehoben wird.

Jede Zugriffsstufe lässt auch ohne Eingabe eines Passwortes den uneingeschränkten Zugriff auf bestimmte Funktionen zu, z. B. Identifikation über die Funktion "Erreichbare Teilnehmer".



## Zugriffsstufen parametrieren

Im Folgenden erfahren Sie, wie Sie für eine S7-1200 CPU ab V4 eine Zugriffsstufe parametrieren und Passwörter eingeben.

Für eine S7-1200 CPU können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in einer Tabelle eingegeben, so dass jedem Passwort genau eine Zugriffsstufe zugeordnet ist.

Wie das Passwort wirkt, ist in der Spalte "Zugriff" visualisiert und steht im erläuternden Text unter der Tabelle.

## Beispiel

Sie wählen die Zugriffsstufe "Kein Zugriff (kompletter Schutz)" für eine Standard-CPU (d. h. keine F-CPU) und geben für jede der in der Tabelle darüber liegenden Zugriffsstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist die CPU komplett geschützt. Nicht einmal HMI-Zugriffe sind möglich.

Für Nutzer, die eines der parametrierten Passwörter kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile 1 (Vollzugriff (kein Schutz) ) wirkt, als wäre die CPU ungeschützt. Nutzer, die dieses Passwort kennen, haben uneingeschränkten Zugriff auf die CPU.
- Das Passwort in Zeile 2 (Lesezugriff) wirkt, als wäre die CPU schreibgeschützt. Nutzer, die dieses Passwort kennen, haben nur lesenden Zugriff auf die CPU.
- Das Passwort in Zeile 3 (HMI-Zugriff) wirkt, als wäre die CPU schreib- und lesegeschützt, so dass für die Nutzer, die dieses Passwort kennen, nur HMI-Zugriffe möglich sind.

## Vorgehen

Um die Zugriffsstufen für eine S7-1200 CPU zu parametrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Eigenschaften der Baugruppe im Inspektorfenster.
2. Öffnen Sie in der Bereichsnavigation den Eintrag "Schutz".  
Eine Tabelle mit den möglichen Zugriffsstufen wird im Inspektorfenster angezeigt.

| Zugriffsstufe  | Zugriff |       |           | Zugriffserlaubnis |             |
|--|---------|-------|-----------|-------------------|-------------|
|  | HMI     | Lesen | Schreiben | Passwort          | Bestätigung |
| <input checked="" type="radio"/> Vollzugriff (kein Schutz) | ✓       | ✓     | ✓         |                   |             |
| <input type="radio"/> Lesezugriff                          | ✓       | ✓     |           |                   |             |
| <input type="radio"/> HMI-Zugriff                          | ✓       |       |           |                   |             |
| <input type="radio"/> Kein Zugriff (kompletter Schutz)     |         |       |           |                   |             |

Bild 8-4 Zugriffschutz

3. Aktivieren Sie die gewünschte Zugriffsstufe in der ersten Spalte der Tabelle. Die grünen Haken in den Spalten rechts der jeweiligen Schutzstufe zeigen Ihnen, welche Operationen noch möglich sind, ohne das Passwort einzugeben.

4. Vergeben Sie in der Spalte "Passwort" in der ersten Zeile ein Passwort für den Vollzugriff. Wiederholen Sie zum Schutz vor Fehleingaben das gewählte Passwort in der Spalte "Bestätigung".  
Achten Sie darauf, dass das Passwort ausreichend sicher ist, d. h. dass es kein durch eine Maschine erkennbares Muster besitzt!  
Die Eingabe eines Passwortes in der ersten Zeile "Vollzugriff (kein Schutz)" ist obligatorisch und ermöglicht dem Passwort-Kenner uneingeschränkten Zugriff auf die CPU, unabhängig von der gewählten Schutzstufe.
5. Weisen Sie weiteren Zugriffsstufen nach Bedarf weitere Passwörter zu, falls die gewählte Schutzstufe das erlaubt.
6. Laden Sie die Hardware-Konfiguration, damit die Zugriffsstufe wirksam wird.

## Ergebnis

Die Hardware-Konfiguration und die Bausteine sind entsprechend der eingestellten Zugriffsstufe vor unberechtigtem Zugriff geschützt. Wenn eine Operation aufgrund der parametrisierten Zugriffsstufe nicht ohne Passwort ausgeführt werden kann, wird ein Dialog zur Eingabe eines Passwortes aufgeblendet.

## Einschränken von Kommunikationsdiensten

### Einführung

Die CPU kann Server für eine Reihe von Kommunikationsdiensten sein, d. h. ohne dass Sie für die CPU Verbindungen projektiert und programmiert haben, können andere Kommunikationsteilnehmer auf CPU-Daten zugreifen.

Damit entfällt für die lokale CPU als Server die Möglichkeit, die Kommunikation zu den Clients kontrollieren.

Ob diese Art der Kommunikation im Betrieb für die lokale CPU zulässig ist oder nicht, wird durch den Parameter "Verbindungsmechanismen" im Bereich "Schutz" der CPU-Parameter bestimmt.

### Zugriff über PUT/GET-Kommunikation durch entfernte Partner erlauben

In der Voreinstellung ist die Option "Zugriff über PUT/GET-Kommunikation durch entfernte Partner (...) erlauben" deaktiviert. In diesem Fall ist lesender und schreibender Zugriff auf CPU-Daten nur möglich bei Kommunikationsverbindungen, die eine Projektierung bzw. Programmierung sowohl für die lokale CPU als auch für den Kommunikationspartner voraussetzen. Zugriffe über BSEND/BRCV-Anweisungen sind z. B. möglich.

Verbindungen, für die die lokale CPU nur Server ist (d. h. für die lokale CPU ist keine Projektierung/Programmierung der Kommunikation zum Kommunikationspartner vorhanden), sind damit im Betrieb der CPU nicht möglich; z. B.

- bei PUT/GET-, FETCH/WRITE- oder FTP-Zugriffen über Kommunikationsmodule
- bei PUT/GET-Zugriffen von anderen S7-CPU's
- bei HMI-Zugriffen, die über PUT/GET-Kommunikation realisiert sind

Wenn Sie den Client-seitigen Zugriff auf CPU-Daten erlauben wollen, d. h. wenn Sie die Kommunikationsdienste der CPU nicht einschränken wollen, dann aktivieren Sie die Option "Zugriff über PUT/GET-Kommunikation durch entfernte Partner erlauben".

## Organisationsbausteine

### Organisationsbausteine für den Anlauf

#### Beschreibung

Sie können die Randbedingungen für das Anlaufverhalten Ihrer CPU bestimmen, z. B. die Initialisierungswerte für "RUN". Dazu schreiben Sie ein Anlaufprogramm. Das Anlaufprogramm besteht aus einem oder mehreren Anlauf-OBs (OB-Nummern 100 oder  $\geq 123$ ).

Das Anlaufprogramm wird einmalig beim Übergang vom Betriebszustand "STOP" in den Betriebszustand "RUN" bearbeitet. Für das Anlaufprogramm stehen weder aktuelle Werte aus dem Prozessabbild der Eingänge zur Verfügung noch können Ausgänge gesetzt werden.

Nach der vollständigen Bearbeitung der Anlauf-OBs wird das Prozessabbild der Eingänge eingelesen und das zyklische Programm wird gestartet.

Für die Ausführung des Anlaufprogramms besteht keine zeitliche Begrenzung. Daher ist die Zyklusüberwachungszeit nicht aktiv. Die Verwendung von zeit- oder alarmgesteuerten Organisationsbausteinen ist nicht möglich.

#### Startinformation

Ein Anlauf-OB hat die folgende Startinformation:

| Variable      | Datentyp | Beschreibung  |
|---------------|----------|---|
| LostRetentive | BOOL     | = 1, falls remanente Datenbereiche verloren gegangen sind |
| LostRTC       | BOOL     | = 1, falls die Echtzeituhr verloren gegangen ist          |

#### Siehe auch

Ereignisse und OBs (Seite 847)

### Organisationsbausteine für die zyklische Programmbearbeitung

#### Einführung

Damit die Programmbearbeitung starten kann, muss im Projekt mindestens ein Zyklus-OB vorhanden sein. Das Betriebssystem ruft diesen Zyklus-OB einmal in jedem Zyklus auf und startet damit die Bearbeitung des Anwenderprogramms. Sie können mehrere Zyklus-OBs (OB-Nummern  $\geq 123$ ) verwenden. Bei Verwendung mehrerer Zyklus-OBs werden diese nacheinander aufgerufen, und zwar in der Reihenfolge ihrer OB-Nummern. Der Zyklus-OB mit der kleinsten OB-Nummer wird zuerst aufgerufen.

Die Zyklus-OBs haben die Prioritätsklasse 1. Das entspricht der niedrigsten Priorität aller OBs. Das zyklische Programm kann also von Ereignissen jeder anderen Ereignisklasse unterbrochen werden.

### Programmieren zyklischer Programmbearbeitung

Sie programmieren die zyklische Programmbearbeitung, indem Sie Ihr Anwenderprogramm in die Zyklus-OBs und die darin aufgerufenen Bausteine schreiben.

Die erste zyklische Programmbearbeitung beginnt, sobald das Anlaufprogramm fehlerfrei beendet wurde. Nach dem Ende jeder zyklischen Programmbearbeitung beginnt der Zyklus erneut.

### Ablauf der zyklischen Programmbearbeitung

Ein Zyklus der Programmbearbeitung umfasst folgende Schritte:

1. Das Betriebssystem startet die Zyklusüberwachungszeit.
2. Das Betriebssystem schreibt die Werte aus dem Prozessabbild der Ausgänge in die Ausgabebaugruppen.
3. Das Betriebssystem liest den Zustand der Eingänge an den Eingabebaugruppen aus und aktualisiert das Prozessabbild der Eingänge.
4. Das Betriebssystem bearbeitet das Anwenderprogramm und führt die im Programm angegebenen Operationen aus.
5. Am Ende eines Zyklus kann das Betriebssystem weitere anstehende Aufgaben ausführen, z. B. Laden und Löschen von Bausteinen oder Aufruf anderer Zyklus-OBs.
6. Anschließend kehrt die CPU zum Zyklusbeginn zurück und startet die Zykluszeitüberwachung erneut.

Siehe auch: Auto-Hotspot

### Unterbrechungsmöglichkeiten

Die zyklische Programmbearbeitung kann durch folgende Ereignisse unterbrochen werden:

- Alarm
- STOP-Befehl, ausgelöst durch
  - Bedienung vom Programmiergerät
  - Anweisung "STP"
- Netzspannungsausfall
- Auftreten eines Geräte- oder Programmfehlers

### Startinformation

- keine
- Optimierte Startinformation:

| Name       | Datentyp | Bedeutung  |
|------------|----------|--|
| first_scan | BOOL     | = TRUE im ersten Aufruf dieses OB bei: <ul style="list-style-type: none"> <li>• Übergang von STOP bzw. HALT nach RUN</li> <li>• Nach Neuladen</li> </ul> |
| remanence  | BOOL     | = TRUE, falls remanente Daten verfügbar sind   |

**Siehe auch**

Ereignisse und OBs (Seite 847)

**Organisationsbausteine für die alarmgesteuerte Programmbearbeitung**

**Organisationsbausteine für Uhrzeitalarme**

**Funktion**

Organisationsbausteine für Uhrzeitalarme (OB-Nummern  $\geq 123$ ) können wie folgt bearbeitet werden:

- Einmalig zu einem vorgegebenen Zeitpunkt (Datum mit Uhrzeit)
- Periodisch mit vorgegebenem Startzeitpunkt und den folgenden Zeitabständen:
  - Minütlich
  - Stündlich
  - Täglich
  - Wöchentlich
  - Monatlich
  - Jährlich
  - Am Monatsende

Uhrzeitalarm-OBs werden daher verwendet, um Teile des Anwenderprogramms zeitgesteuert ablaufen zu lassen.

**Zustände für Uhrzeitalarme**

Die folgende Tabelle enthält die möglichen Zustände eines Uhrzeitalarms sowie deren Bedeutung.

| Zustand   | Bedeutung  |
|-----------|--|
| Storniert | Die einmalige Bearbeitung ist bereits erfolgt, oder das Startereignis eines noch nicht bearbeiteten Uhrzeitalarms wurde mit der erweiterten Anweisung CAN_TINT gelöscht. |
| Gestellt  | Sie haben den Zeitpunkt bzw. Startzeitpunkt seiner Bearbeitung festgelegt.   |
| Aktiviert | Sie haben festgelegt, ob er einmalig oder periodisch bearbeitet werden soll, und haben im Falle periodischer Bearbeitung den Zeitabstand festgelegt.                     |

### Regeln für Uhrzeitalarme

Für die Verwendung von Uhrzeitalarmen gelten folgende Regeln:

- Ein Uhrzeitalarm kann nur bearbeitet werden, wenn er gestellt und aktiviert wurde und ein zugehöriger Organisationsbaustein im Anwenderprogramm vorhanden ist.
- Die Startzeitpunkte periodischer Uhrzeitalarme müssen einem realen Datum entsprechen. Zum Beispiel ist die monatliche Wiederholung eines Organisationsbausteins, dessen erstmalige Bearbeitung am 31. Januar erfolgt, nicht möglich. In diesem Fall würde der OB nur in den Monaten gestartet, die 31 Tage haben.
- Ein Uhrzeit-Alarm, der während des Anlaufs durch Aufruf der erweiterten Anweisung ACT\_TINT aktiviert wird, wird erst nach der Beendigung des Anlaufs bearbeitet.
- Nach jedem Anlauf der CPU müssen Sie gestellte Uhrzeitalarme neu aktivieren.

### Stellen und Aktivieren eines Uhrzeitalarms-OB

Bevor ein Uhrzeitalarm ausgelöst und dadurch der zugeordnete Uhrzeitalarm-OB vom Betriebssystem bearbeitet werden kann, müssen Sie den Alarm stellen und aktivieren. Dabei gibt es folgende Möglichkeiten:

| Uhrzeitalarm stellen                             | Uhrzeitalarm aktivieren                         |
|--|---|
| Mittels Projektierung                            | Mittels Projektierung                           |
| Mittels Projektierung                            | Durch Aufruf der erweiterten Anweisung ACT_TINT |
| Durch Aufruf der erweiterten Anweisung SET_TINTL | Durch Aufruf der erweiterten Anweisung ACT_TINT |

---

#### Hinweis

Wenn Sie einen Uhrzeitalarm so konfigurieren, dass der zugehörige OB einmalig bearbeitet werden soll, darf der Startzeitpunkt nicht in der Vergangenheit liegen (bezogen auf die Echtzeituhr der CPU).

Wenn Sie einen Uhrzeitalarm so konfigurieren, dass der zugehörige OB periodisch bearbeitet werden soll, der Startzeitpunkt aber in der Vergangenheit liegt, dann wird der Uhrzeitalarm-OB bei dem nächsten fälligen Zeitpunkt nach der aktuellen Uhrzeit bearbeitet.

---

### Abfragen des Status eines Uhrzeitalarms

Um den Status eines Uhrzeitalarms abzufragen, rufen Sie die erweiterte Anweisung QRY\_TINT auf.

### Stornieren eines Uhrzeitalarms

Sie können Uhrzeitalarme, die noch nicht bearbeitet wurden, mit der erweiterten Anweisung CAN\_TINT stornieren.

Stornierte Uhrzeitalarme können Sie mit der erweiterten Anweisung SET\_TINTL wieder stellen und mit der erweiterten Anweisung ACT\_TINT aktivieren.

### Bedingungen, die Uhrzeitalarm-OBs beeinträchtigen

Da ein Uhrzeitalarm nur in bestimmten Intervallen auftritt, können bestimmte Bedingungen die Funktion der zugehörigen OBs während der Bearbeitung Ihres Programms beeinträchtigen. Die folgende Tabelle zeigt einige dieser Bedingungen und beschreibt die Wirkung auf die Bearbeitung eines Uhrzeitalarm-OB.

| Bedingung  | Ergebnis   |
|--|--|
| Im Anwenderprogramm wird die erweiterte Anweisung CAN_TINT aufgerufen.   | Das Betriebssystem löscht das Startereignis (Datum und Uhrzeit) des Uhrzeitalarms. Sie müssen den Uhrzeitalarm erneut stellen und aktivieren, wenn der zugehörige Uhrzeitalarm-OB wieder aufgerufen werden soll.   |
| Beim Synchronisieren oder Korrigieren der Systemuhr der CPU wurde die Uhrzeit vorgestellt. Dabei wurde der Startzeitpunkt für einen Uhrzeitalarm-OB übersprungen.                              | Das Betriebssystem ruft den Zeitfehler-OB (OB 80) auf und trägt das Startereignis, die Nummer und die Priorität des ersten übersprungenen Uhrzeitalarm-OB in dessen Startinformation ein. Nach Beendigung des OB 80 bearbeitet das Betriebssystem den übersprungenen Uhrzeitalarm-OB genau einmal. |
| Beim Synchronisieren oder Korrigieren der Systemuhr der CPU wurde die Uhrzeit zurückgestellt. Die korrigierte Uhrzeit liegt vor dem Startzeitpunkt eines bereits ausgeführten Uhrzeitalarm-OB. | Der Uhrzeitalarm-OB wird wiederholt.   |
| Ein Uhrzeitalarm-OB wird noch bearbeitet, als das Startereignis für seine nächste Bearbeitung auftritt.  | Das Betriebssystem ruft den Zeitfehler-OB (OB 80) auf. Nach dessen Bearbeitung und der Weiterbearbeitung des aktuellen Uhrzeitalarm-OB wird die angeforderte OB-Bearbeitung nachgeholt.  |

### Startinformation

Ein Uhrzeitalarm-OB hat die folgende Startinformation:

| Variable   | Datentyp | Beschreibung  |
|------------|----------|---|
| CaughtUp   | BOOL     | = 1, falls der OB-Aufruf wegen vorgestellter Uhrzeit nachgeholt wird  |
| SecondTime | BOOL     | = 1, falls der OB wegen zurückgestellter Uhrzeit ein zweites Mal aufgerufen wird (genauer: falls der geplante Zeitpunkt der aktuellen OB-Bearbeitung früher oder gleich dem geplanten Zeitpunkt der vorhergehenden OB-Bearbeitung liegt).<br>Hinweis: SecondTime wird nur einmal gesetzt. |

## Organisationsbaustein für Statusalarme

### Beschreibung

Das Betriebssystem der S7-1200-CPU ruft den Statusalarm-OB auf, wenn es einen Statusalarm von einem DP-Master oder IO-Controller empfangen hat. Dies kann der Fall sein, wenn ein Modul eines Slaves seinen Betriebszustand wechselt, z. B. von "RUN" nach "STOP". Genauere Informationen zu den Ereignissen, die einen Statusalarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

### Struktur der Startinformation

Der Statusalarm-OB hat die folgende Startinformation:

| Name      | Datentyp | Bedeutung   |
|-----------|----------|---|
| LADDR     | HW_IO    | Hardware-Adresse der Alarm auslösenden Komponente |
| Slot      | UINT     | Steckplatz-Nr. der Alarm auslösenden Komponente   |
| Specifier | WORD     | Alarm Specifier aus dem Alarm-Telegramm           |

### Siehe auch

Ereignisse und OBs (Seite 847)

## Organisationsbaustein für Update-Alarme

### Beschreibung

Das Betriebssystem der S7-1200-CPU ruft den Update-Alarm-OB auf, wenn es einen Update-Alarm von einem DP-Master oder IO-Controller empfangen hat. Dies kann der Fall sein, wenn Sie an einem Steckplatz eines Slaves oder Devices eine Parameteränderung durchgeführt haben. Genauere Informationen zu den Ereignissen, die einen Update-Alarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

### Struktur der Startinformation

Der Update-Alarm-OB hat die folgende Startinformation:

| Name      | Datentyp | Bedeutung   |
|-----------|----------|---|
| LADDR     | HW_IO    | Hardware-Adresse der Alarm auslösenden Komponente |
| Slot      | UINT     | Steckplatz-Nr. der Alarm auslösenden Komponente   |
| Specifier | WORD     | Alarm Specifier aus dem Alarm-Telegramm           |

### Siehe auch

Ereignisse und OBs (Seite 847)



## Organisationsbaustein für hersteller- bzw. profilspezifische Alarmer

### Beschreibung

Das Betriebssystem der S7-1200-CPU ruft den OB57 auf, wenn es einen hersteller- oder profilspezifischen Alarm von einem DP-Master oder IO-Controller empfangen hat. Genauere Informationen zu den Ereignissen, die einen solchen Alarm auslösen, entnehmen Sie der Dokumentation des Slave- bzw. Device-Herstellers.

### Struktur der Startinformation

Der OB für hersteller- bzw. profilspezifische Alarmer hat die folgende Startinformation:

| Name      | Datentyp | Bedeutung   |
|-----------|----------|---|
| LADDR     | HW_IO    | Hardware-Adresse der Alarm auslösenden Komponente |
| Slot      | UINT     | Steckplatz-Nr. der Alarm auslösenden Komponente   |
| Specifier | WORD     | Alarm Specifier aus dem Alarm-Telegramm           |

### Siehe auch

Ereignisse und OBs (Seite 847)

## Organisationsbausteine für Verzögerungsalarmer

### Beschreibung

Ein Verzögerungsalarm-OB wird nach einer parametrierbaren Verzögerungszeit vom Betriebssystem gestartet. Die Verzögerungszeit beginnt nach Aufruf der Anweisung SRT\_DINT abzulaufen.

Sie können bis zu vier Verzögerungsalarm-OBs oder Weckalarm-OBs (OB-Nummern  $\geq 123$ ) in Ihrem Programm verwenden. Verwenden Sie z. B. bereits zwei Weckalarm-OBs, können Sie noch maximal zwei Verzögerungsalarm-OBs in Ihr Programm einfügen.

Sie können die Bearbeitung eines Verzögerungsalarms, der noch nicht gestartet wurde, mit der Anweisung CAN\_DINT verhindern.

### Funktionsweise der Verzögerungsalarm-OBs

Nach Ablauf der Verzögerungszeit, die Sie zusammen mit einer OB-Nummer und einer Kennung an die Anweisung SRT\_DINT übergeben haben, startet das Betriebssystem den zugehörigen OB.

Wenn Sie einen Verzögerungsalarm-OB verwenden möchten, dann müssen Sie die folgenden Aufgaben ausführen:

- Sie müssen die Anweisung SRT\_DINT aufrufen.
- Sie müssen den Verzögerungsalarm-OB als Teil Ihres Programms in die CPU laden.



Startzeitpunkt zu verhindern, falls die Zeitraster der Weckalarm-OBs ein gemeinsames Vielfaches aufweisen.

Als Zeitraster können Sie eine Zeitdauer zwischen 1 ms und 60000 ms vorgeben.

Sie können bis zu vier Weckalarm-OBs oder Verzögerungsalarm-OBs (OB-Nummern  $\geq 123$ ) in Ihrem Programm verwenden. Verwenden Sie z. B. bereits zwei Verzögerungsalarm-OBs, können Sie noch maximal zwei Weckalarm-OBs in Ihr Programm einfügen.

---

#### Hinweis

Die Laufzeit jedes Weckalarm-OB muss deutlich kleiner sein als sein Zeitraster. Falls ein Weckalarm-OB noch nicht beendet ist, aber wegen des abgelaufenen Zeittakts erneut zur Bearbeitung ansteht, wird der Zeitfehler-OB gestartet. Anschließend wird der Fehler verursachende Weckalarm nachgeholt oder verworfen.

---

### Beispiel für die Verwendung der Phasenverschiebung

Sie haben in Ihr Programm zwei Weckalarm-OBs eingefügt:

- Weckalarm-OB1
- Weckalarm-OB2

Für Weckalarm-OB1 haben Sie ein Zeitraster von 20 ms und für Weckalarm-OB2 ein Zeitraster von 100 ms eingestellt. Nach Ablauf des Zeitrasters von 100 ms erreicht Weckalarm-OB1 zum fünften Mal seinen Startzeitpunkt, Weckalarm-OB2 zum ersten Mal. Um die Weckalarm-OBs dennoch zeitversetzt zu bearbeiten, geben Sie für einen der beiden Weckalarm-OBs eine Phasenverschiebung ein.

### Startinformation

- keine
- Optimierte Startinformation:

| Name        | Datentyp | Bedeutung   |
|-------------|----------|---|
| first_scan  | BOOL     | = TRUE im ersten Aufruf dieses OB <ul style="list-style-type: none"><li>• Beim Übergang von STOP bzw. HALT nach RUN</li><li>• Nach Neuladen</li></ul> |
| event_count | INT      | Anzahl der verlorenen Startereignisse seit dem letzten Start dieses OB  |

### Siehe auch

Weckalarm-OBs parametrieren (Seite 902)

Ereignisse und OBs (Seite 847)

## Organisationsbausteine für Prozessalarme

### Beschreibung

Sie können Prozessalarm-OBs einsetzen, um auf bestimmte Ereignisse zu reagieren. Dabei können Sie einem Ereignis, das einen Alarm auslöst, nur genau einen Prozessalarm-OB zuordnen. Einem Prozessalarm-OB können hingegen mehrere Ereignisse zugeordnet sein.

Prozessalarme können durch Schnelle Zähler und Eingangskanäle ausgelöst werden. Für jeden Schnellen Zähler und Eingangskanal, der einen Prozessalarm auslösen soll, ist die Parametrierung der folgenden Eigenschaften erforderlich:

- Das Prozessereignis, das den Prozessalarm auslösen soll (z. B. die Änderung der Zählrichtung eines Schnellen Zählers)
- Die Nummer des Prozessalarm-OB, der diesem Prozessereignis zugeordnet wird

Sie können bis zu 50 voneinander unabhängige Prozessalarm-OBs (OB-Nummern  $\geq 123$ ) in Ihrem Programm verwenden.

### Funktionsweise eines Prozessalarm-OB

Nach Auslösen eines Prozessalarms identifiziert das Betriebssystem den Kanal des Eingangs oder den Schnellen Zähler und ermittelt den zugeordneten Prozessalarm-OB.

Wenn kein anderer Alarm-OB aktiv ist, wird der ermittelte Prozessalarm-OB aufgerufen. Wird bereits ein anderer Alarm-OB ausgeführt, wird der Prozessalarm in die Warteschlange seiner Prioritätsklasse eingereiht. Die Quittierung des Prozessalarms erfolgt nach Beendigung des zugeordneten Prozessalarm-OB.

Wenn in der Zeit zwischen der Identifikation und der Quittierung eines Prozessalarms auf demselben Modul erneut ein Prozessereignis auftritt, dann gilt:

- Tritt das Ereignis bei dem Kanal auf, der den aktuellen Prozessalarm ausgelöst hat, wird kein weiterer Prozessalarm ausgelöst. Ein weiterer Prozessalarm kann erst dann ausgelöst werden, wenn der aktuelle Prozessalarm quittiert ist.
- Tritt das Ereignis bei einem anderen Kanal auf, wird ein Prozessalarm ausgelöst.

Prozessalarm-OBs werden nur im Betriebszustand "RUN" der CPU aufgerufen.

### Startinformation

- keine
- Optimierte Startinformation:

| Name  | Datentyp | Bedeutung   |
|-------|----------|---|
| Laddr | HW_IO    | HW-Kennung des Moduls, das den Prozessalarm auslöst           |
| USI   | WORD     | Kennung für zukünftige Erweiterungen (nicht anwenderrelevant) |

| Name      | Datentyp | Bedeutung   |
|-----------|----------|---|
| IChannel  | USINT    | Nummer des Kanals, der den Prozessalarm auslöst   |
| EventType | BYTE     | Kennung für den Ereignistyp, zu dem das Alarm auslösende Ereignis gehört (z. B. steigende Flanke)<br>Diese Kennung ist der Beschreibung des jeweiligen Moduls zu entnehmen. |

## Siehe auch

Prozessalarm-OBs parametrieren (Seite 903)

Ereignisse und OBs (Seite 847)

## Organisationsbaustein für Zeitfehler

### Beschreibung

Das Betriebssystem ruft den Zeitfehler-OB (OB 80) auf, wenn eines der folgenden Ereignisse auftritt:

- Das zyklische Programm überschreitet die Zyklusüberwachungszeit.
- Der aufgerufene OB wird derzeit noch ausgeführt (möglich bei Verzögerungsalarm-OBs und Weckalarm-OBs).
- Ein Uhrzeitalarm wurde verpasst wegen Vorstellen der Uhrzeit um mehr als 20 Sekunden.
- Ein Uhrzeitalarm wurde verpasst während STOP.
- Eine Warteschlange für Alarm-OBs ist übergelaufen.
- Ein Alarm geht durch zu hohe Alarmlast verloren.

Wenn Sie keinen Zeitfehler-OB programmiert haben, verhält sich die S7-1200-CPU wie folgt:

- CPUs mit Firmware-Stand V1.0: Die CPU bleibt im RUN.
- CPUs ab Firmware-Stand V2.0:
  - Die CPU geht beim Überschreiten der Zyklusüberwachungszeit in STOP.
  - Bei allen anderen Startereignissen des Zeitfehler-OB bleibt die CPU im RUN.

Bei CPUs mit Firmware-Stand V1.0 führt das zweimalige Überschreiten der Zyklusüberwachungszeit nicht zum Aufruf eines OB, sondern zum STOP der CPU. Sie können die zweite Überschreitung vermeiden, indem Sie mit der Anweisung RE\_TRIGR die Zyklusüberwachung der CPU neu starten.

Sie können nur einen Zeitfehler-OB in Ihrem Programm verwenden.

### Startinformation

Der Zeitfehler-OB hat die folgende Startinformation:

| Variable | Datentyp | Beschreibung   |
|----------|----------|--|
| fault_id | BYTE     | <ul style="list-style-type: none"> <li>• 0x01: Zyklusüberwachungszeit überschritten</li> <li>• 0x02: Aufgerufener OB wird noch ausgeführt</li> <li>• 0x05: Abgelaufener Uhrzeitalarm durch Uhrzeitsprung</li> <li>• 0x06: Abgelaufener Uhrzeitalarm bei Wiedereintritt in RUN</li> <li>• 0x07: Warteschlange übergelaufen</li> <li>• 0x09: Alarmverlust durch zu hohe Alarmlast</li> </ul> |
| csg_OBnr | OB_ANY   | Nummer des zum Fehlerzeitpunkt bearbeiteten OB   |
| csg_prio | UINT     | Priorität des zum Fehlerzeitpunkt bearbeiteten OB  |

### Siehe auch

Ereignisse und OBs (Seite 847)

### Organisationsbaustein für Diagnosealarme

#### Beschreibung

Sie können für diagnosefähige Module den Diagnosealarm freigeben, damit das Modul Änderungen am Peripheriezustand erkennt. Dadurch löst das Modul in folgenden Fällen einen Diagnosealarm aus:

- Es liegt eine Störung vor (kommendes Ereignis)
- Es liegt keine Störung mehr vor (gehendes Ereignis)

Wenn kein anderer Alarm-OB aktiv ist, dann wird der Diagnosealarm-OB (OB 82) aufgerufen. Wird bereits ein anderer Alarm-OB ausgeführt, wird der Diagnosealarm in die Warteschlange seiner Prioritätsgruppe eingereiht.

Sie können nur einen Diagnosealarm-OB in Ihrem Programm verwenden.

### Startinformation

Der Diagnosealarm-OB hat die folgende Startinformation:

| Variable    | Datentyp | Beschreibung  |
|-------------|----------|---|
| IO_state    | WORD     | Enthält den Peripheriezustand des diagnosefähigen Moduls. |
| laddr       | HW_ANY   | HW-Identifikation   |
| channel     | UINT     | Kanalnummer   |
| multi_error | BOOL     | = 1, falls mehr als ein Fehler vorhanden ist              |

## Variable IO\_state

Die folgende Tabelle zeigt die möglichen Peripheriezustände, die in der Variable IO\_state enthalten sein können:

| IO_state | Beschreibung  |
|----------|---|
| Bit 0    | Konfiguration korrekt:<br>= 1, falls die Konfiguration korrekt ist<br>= 0, falls die Konfiguration nicht mehr korrekt ist   |
| Bit 4    | Fehler:<br>= 1, falls eine Störung vorliegt, z. B. ein Drahtbruch<br>= 0, falls die Störung nicht mehr vorliegt   |
| Bit 5    | Konfiguration nicht korrekt:<br>= 1, falls die Konfiguration nicht korrekt ist<br>= 0, falls die Konfiguration wieder korrekt ist   |
| Bit 6    | Peripherie nicht erreichbar:<br>= 1, falls ein Peripheriezugriffsfehler aufgetreten ist<br>In diesem Fall enthält laddr die Hardwarekennung der Peripherie mit dem Zugriffsfehler.<br>= 0, falls die Peripherie wieder erreichbar ist |

## Siehe auch

Ereignisse und OBs (Seite 847)

## Organisationsbaustein für Ziehen und Stecken

### Beschreibung

Das Betriebssystem der S7 1200-CPU ruft den Ziehen/Stecken-OB (OB 83) auf, wenn ein projektiertes und nicht deaktiviertes Modul oder Submodul der dezentralen Peripherie (PROFIBUS, PROFINET, AS-i) gezogen oder gesteckt wird.

---

#### Hinweis

Das Ziehen bzw. Stecken eines zentralen Moduls führt zum STOP der CPU.

---

### Startinformation

Der Ziehen/Stecken-OB hat die folgende Startinformation:

| Name        | Datentyp | Bedeutung  |
|-------------|----------|--|
| LADDR       | HW_IO    | Hardware-Kennung des betroffenen Moduls bzw. Submoduls   |
| Event_Class | BYTE     | <ul style="list-style-type: none"> <li>• B#16#38: (Sub-) Modul gesteckt</li> <li>• B#16#39: (Sub-) Modul gezogen bzw. nicht ansprechbar</li> </ul> |
| Fault_ID    | BYTE     | Fehlercode (mögliche Werte: B#16#51, B#16#54, B#16#55, B#16#56, B#16#57, B#16#58)  |

Die folgende Tabelle zeigt, welches Ereignis den Start des Ziehen/Stecken-OB verursacht hat.

| ev_class (B#16# ...) | fault_id (B#16# ...) | Bedeutung   |
|----------------------|----------------------|---|
| 39                   | 51                   | Modul gezogen   |
| 39                   | 54                   | Submodul gezogen  |
| 38                   | 54                   | Submodul gesteckt, das dem parametriertem Submodul entspricht       |
| 38                   | 55                   | Submodul gesteckt, das dem parametrierten Submodul nicht entspricht |
| 38                   | 56                   | Submodul gesteckt, jedoch Fehler bei Parametrierung                 |
| 38                   | 57                   | Submodul oder Modul gesteckt, jedoch mit Störung oder Wartung       |
| 38                   | 58                   | Submodul-Zugriffsfehler beseitigt                                   |

### Siehe auch

Ereignisse und OBs (Seite 847)

## Organisationsbaustein für Baugruppenträgerfehler

### Beschreibung

Das Betriebssystem der S7 1200-CPU ruft den OB86 in den folgenden Fällen auf:

- Der Ausfall eines DP-Mastersystems bzw. eines PROFINET IO-Systems wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Der Ausfall eines DP-Slaves bzw. eines IO-Devices wird erkannt (sowohl bei kommendem als auch bei gehendem Ereignis).
- Der Ausfall eines Teils der Submodule eines PROFINET I-Device wird erkannt.



## Struktur der Startinformation

Der Baugruppenträgerfehler-OB hat die folgende Startinformation:

| Name        | Datentyp | Bedeutung   |
|-------------|----------|---|
| LADDR       | HW_IO    | Hardware-Kennung des fehlerhaften Hardware-Objekts  |
| Event_Class | BYTE     | <ul style="list-style-type: none"> <li>• B#16#32: Aktivierung eines DP-Slaves bzw. eines IO-Devices</li> <li>• B#16#33: Deaktivierung eines DP-Slaves bzw. eines IO-Devices</li> <li>• B#16#38: gehendes Ereignis</li> <li>• B#16#39: kommendes Ereignis</li> </ul> |
| Fault_ID    | BYTE     | Fehlercode (mögliche Werte: B#16#C3, B#16#C4, B#16#C5, B#16#C6, B#16#C7, B#16#C8, B#16#C9, B#16#CA, B#16#CB, B#16#CC, B#16#CD, B#16#CE, B#16#CF, B#16#F8, B#16#F9)  |

Die folgende Tabelle zeigt, welches Ereignis den Start des OB86 verursacht hat.

| Ev_class<br>B#16# ... | Fault_id<br>B#16# ... | Bedeutung  |
|-----------------------|-----------------------|--|
| 39                    | C3                    | Ausfall eines DP-Mastersystems   |
| 39/38                 | C4                    | Ausfall/Wiederkehr eines DP-Slaves   |
| 38                    | C5                    | Wiederkehr eines DP-Slaves, Slave jedoch gestört   |
| 38                    | C6                    | Erweiterungsgerätewiederkehr, jedoch Fehler bei Baugruppenparametrierung                           |
| 38                    | C7                    | Wiederkehr eines DP-Slaves, jedoch Fehler bei Baugruppenparametrierung                             |
| 38                    | C8                    | Wiederkehr einer DP-Slaves, jedoch Abweichung Soll-/Istausbau                                      |
| 32/33                 | C9                    | Aktivierung/Deaktivierung eines DP-Slaves mit der Anweisung "D_ACT_DP"                             |
| 39                    | CA                    | Ausfall eines PROFINET IO-Systems  |
| 39/38                 | CB                    | Ausfall/Wiederkehr eines PROFINET IO-Devices   |
| 38                    | CC                    | Wiederkehr eines PROFINET IO-Devices mit Störung oder Wartung                                      |
| 38                    | CD                    | Wiederkehr eines PROFINET IO-Devices, Sollausbau weicht von Istausbau ab                           |
| 38                    | CE                    | Wiederkehr eines PROFINET IO-Devices, Fehler bei der Baugruppenparametrierung                      |
| 32/33                 | CF                    | Aktivierung/Deaktivierung eines IO-Devices mit der Anweisung "D_ACT_DP"                            |
| 39/38                 | F8                    | Ausfall/Wiederkehr eines Teils der Submodule eines PROFINET I-Devices                              |
| 38                    | F9                    | Wiederkehr eines Teils der Submodule eines PROFINET I-Devices mit Device-Konfigurationsunterschied |

## Siehe auch

Ereignisse und OBs (Seite 847)

## Bausteinparameter von Organisationsbausteinen

### Grundlagen zu Bausteinparametern

#### Einführung

Einige Organisationsbausteine (OBs) besitzen Eigenschaften, über die ihr Verhalten oder ihre Zuordnung zu bestimmten Ereignissen gesteuert werden kann. Diese Eigenschaften können Sie durch Parametrierung beeinflussen.

#### Übersicht

Für folgende Organisationsbausteine können Sie die Eigenschaften parametrieren:

- Uhrzeitalarm-OBs
- Weckalarm-OBs
- Prozessalarm-OBs

#### Siehe auch

Prozessalarm-OBs parametrieren (Seite 903)

Weckalarm-OBs parametrieren (Seite 902)

### Uhrzeitalarm-OBs parametrieren

#### Vorgehen zur Einstellung der Parameter

Um die Parameter eines Uhrzeitalarm-OB einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den zum betrachteten Uhrzeitalarm-OB gehörenden Dialog "Eigenschaften".
2. Klicken Sie in der Bereichsnavigation auf die Gruppe "Uhrzeitalarm".

#### Überblick über die einstellbaren Parameter

Sie können folgende Parameter einstellen:

- Ausführung
- Startdatum und Uhrzeit
- Optionsfelder "Lokalzeit" und "Systemzeit"

## Parameter "Ausführung"

Über die Klappliste "Ausführung" geben Sie vor, mit welcher Periode der Uhrzeitalarm -OB ausgeführt werden soll. Die Zeitintervalle beziehen sich auf die Einstellungen bei "Startdatum" und "Uhrzeit".

Folgende Werte von "Ausführung" sind möglich:

- nie
- einmalig
- minütlich
- stündlich
- täglich
- wöchentlich
- monatlich
- jährlich
- am Monatsende

---

### **Hinweis**

Beim Wert "am Monatsende" ist der unter "Startdatum" angegebene Wert irrelevant.

---

## Parameter "Startdatum" und "Uhrzeit"

Hier stellen Sie ein, zu welchem Zeitpunkt der Uhrzeitalarm-OB zum ersten Mal ausgeführt werden soll.

Beispiel: Startdatum = 05.07.2013, Uhrzeit = 11:16

Abhängig vom Wert des Parameters "Ausführung" erzeugt die CPU periodisch weitere Uhrzeitalarme. Die Startzeit bezieht sich je nach Einstellung entweder auf die Lokalzeit oder auf die koordinierte Weltzeit UTC.

---

### **Hinweis**

Wenn Sie für den Parameter "Ausführung" "monatlich" einstellen, sind der 29., der 30. und der 31. als Startdatum nicht möglich. Wenn der Uhrzeitalarm-OB am Monatsende gestartet werden soll, wählen Sie stattdessen "am Monatsende" beim Parameter "Ausführung".

---

## "Lokalzeit" oder "Systemzeit"

Hier stellen Sie ein, auf welche Zeit sich die Startzeit des Uhrzeitalarm-OB bezieht:

- "Lokalzeit": Die Startzeit bezieht sich auf die eingestellte Zeitzone der CPU.
- "Systemzeit": Die Startzeit bezieht sich auf die koordinierte Weltzeit UTC (Universal Time Coordinated).

## Weckalarm-OBs parametrieren

### Einführung

Mit Weckalarm-OBs können Sie Programme in gleichmäßigen Zeitabständen starten. Dazu müssen Sie für jeden verwendeten Weckalarm-OB ein Zeitraster und eine Phasenverschiebung eingeben.

Sie können bis zu vier Weckalarm-OBs oder Verzögerungsalarm-OBs (OB-Nummern  $\geq 200$ ) in Ihrem Programm verwenden. Verwenden Sie z. B. bereits zwei Verzögerungsalarm-OBs, können Sie noch maximal zwei Weckalarm-OBs in Ihr Programm einfügen.

---

#### Hinweis

Wenn Sie mehrere Weckalarm-OBs parametrieren, müssen Sie jedem Weckalarm-OB eine andere Zykluszeit oder Phasenverschiebung geben, um einer gleichzeitigen Ausführung bzw. einer Warteschlange vorzubeugen. Beim Erstellen eines Weckalarm-OB wird die Zykluszeit 100 und die Phasenverschiebung 0 als Startwert eingetragen.

---

### Vorgehen

Um für einen Weckalarm-OB ein Zeitraster und eine Phasenverschiebung einzugeben, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Weckalarm-OB.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Der Dialog "<Name des Weckalarm-OB>" wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf die Gruppe "Weckalarm".  
Die Eingabefelder für das Zeitraster und die Phasenverschiebung werden angezeigt.
5. Geben Sie das Zeitraster und die Phasenverschiebung ein.
6. Bestätigen Sie Ihre Eingaben mit "OK".

### Siehe auch

Grundlagen zu Bausteinparametern (Seite 900)

Organisationsbausteine für Weckalarme (Seite 892)

## Prozessalarm-OBs parametrieren

### Einführung

Für jeden Eingangskanal und Schnellen Zähler, der einen Prozessalarm auslösen soll, müssen Sie das entsprechende Ereignis aktivieren und folgende Parameter vergeben:

- Ereignisnamen
- Nummer des Prozessalarm-OB, der diesem Prozessereignis zugeordnet wird

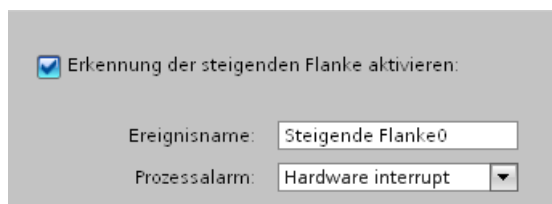
Die Parametrierung von Prozessalarmen erfolgt in den Eigenschaften des entsprechenden Geräts. Sie können bis zu 50 Prozessalarm-OBs parametrieren.

Den zu parametrierenden Prozessalarm-OB können Sie entweder vor oder während der Aktivierung eines Ereignisses erstellen.

### Vorgehen

Um einen Prozessalarm-Ereignis zu projektieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Befehl "Geräte & Netze". Der Hardware- und Netzwerkeditor wird in der Netzsicht geöffnet.
2. Wechseln Sie zur Gerätesicht.
3. Wenn das Inspektorfenster in der Gerätesicht geschlossen ist, aktivieren Sie im Menü "Ansicht" das Optionskästchen "Inspektorfenster". Das Inspektorfenster wird geöffnet.
4. Klicken Sie auf das Register "Eigenschaften".
5. Selektieren Sie in der Gerätesicht die Baugruppe, für die Sie einen Prozessalarm parametrieren möchten.
6. Aktivieren Sie das entsprechende Ereignis, das einen Prozessalarm auslösen soll, z. B. eine steigende Flanke.



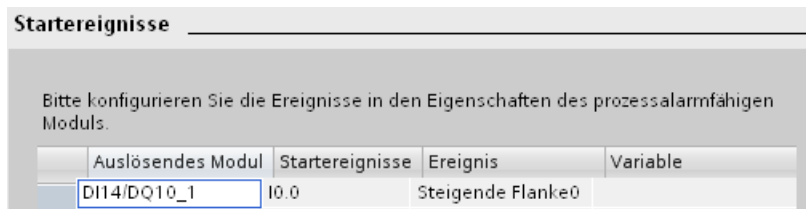
Erkennung der steigenden Flanke aktivieren:

Ereignisname:

Prozessalarm:  ▼

7. Geben Sie einen Ereignisnamen ein.

- Wählen Sie aus der Klappliste "Prozessalarm" einen bestehenden Prozessalarm-OB aus oder legen Sie einen neuen Prozessalarm-OB an. Falls Sie zuvor noch keinen Prozessalarm-OB angelegt haben, klicken Sie in der Klappliste auf die Schaltfläche "Neuen Baustein hinzufügen".  
Die Startinformation des entsprechenden Prozessalarm-OBs mit allen Angaben zum alarmanlösenden Ereignis wird aktualisiert.



- Wenn Sie weitere Prozessalarmlen parametrieren möchten, wiederholen Sie die Schritte 5 bis 8.

Für das Ereignis, das durch den eindeutigen Ereignisnamen gekennzeichnet ist, wird automatisch eine Systemkonstante vom Datentyp Event\_Hwlnt angelegt. Die Systemkonstanten werden in der Standard-Variablen-tabelle angezeigt.

**Siehe auch**

- Grundlagen zu Bausteinparametern (Seite 900)
- Organisationsbausteine für Prozessalarmlen (Seite 894)
- Ereignisse und OBs (Seite 847)

**Symbolische und numerische Bezeichnung von Anweisungen**

**Beschreibung**

Die Anweisungen aus der Task Card setzen sich aus Funktionen (FCs), Funktionsbausteinen (FBs), Systemfunktionen (SFCs) und Systemfunktionsbausteinen (SFBs) zusammen, die intern durch Nummern bezeichnet werden.

Die folgenden Tabellen zeigen die Zuordnungen der numerischen und symbolischen Bezeichnung.

**Funktionsbausteine (FBs)**

| Numerische Bezeichnung | Symbolische Bezeichnung |
|------------------------|-------------------------|
| FB 105                 | TC_CONFIG               |
| FB 110                 | Port_Config             |
| FB 111                 | Send_Config             |
| FB 112                 | Receive_Config          |
| FB 113                 | Send_P2P                |

| Numerische Bezeichnung | Symbolische Bezeichnung |
|------------------------|-------------------------|
| FB 114                 | Receive_P2P             |
| FB 115                 | Receive_Reset           |
| FB 116                 | Signal_Get              |
| FB 117                 | Get_Features            |
| FB 118                 | Set_Features            |
| FB 163                 | TC_SEND                 |
| FB 164                 | TC_RECV                 |
| FB 165                 | TC_CON                  |
| FB 166                 | TC_DISCON               |
| FB 804                 | SET_TIMEZONE            |
| FB 1030                | TSEND_C                 |
| FB 1031                | TRCV_S                  |
| FB 1071                | USS_DRIVE               |
| FB 1080                | MB_COMM_LOAD            |
| FB 1081                | MB_MASTER               |
| FB 1082                | MB_SLAVE                |
| FB 1084                | MB_CLIENT               |
| FB 1085                | MB_SERVER               |
| FB 1100                | MB_Halt                 |
| FB 1101                | MC_Home                 |
| FB 1102                | MC_MoveAbsolute         |
| FB 1103                | MC_MoveJog              |
| FB 1104                | MC_MoveRelative         |
| FB 1105                | MC_MoveVelocity         |
| FB 1107                | MC_Power                |
| FB 1108                | MC_Reset                |
| FB 1110                | MC_MoveInterrupt        |
| FB 1111                | MC_ChangeDynamik        |
| FB 1112                | MC_CommandTable         |
| FB 1113                | MC_MoveLinearAbs_2D     |
| FB 1114                | MC_MoveLinearRel_2D     |
| FB 1115                | MC_MoveCircular_2D      |
| FB 1130                | PID_Compact             |
| FB 1134                | PID_3Step               |
| FB 1140                | HSC                     |
| FB 2040                | RecipeCreate            |
| FB 2041                | RecipeOpen              |
| FB 2042                | RecipeRead              |
| FB 2043                | RecipeWrite             |
| FB 2044                | RecipeAppend            |
| FB 2045                | RecipeClose             |

### Funktionen (FCs)

| Numerische Bezeichnung        | Symbolische Bezeichnung |
|-------------------------------|-------------------------|
| FC 2 <sup>(1)</sup>           | CONCAT                  |
| FC 4 <sup>(1)</sup>           | DELETE                  |
| FC 11 <sup>(1)</sup>          | FIND                    |
| FC 17 <sup>(1)</sup>          | INSERT                  |
| FC 20 <sup>(1)</sup>          | LEFT                    |
| FC 21 <sup>(1)</sup>          | LEN                     |
| FC 22 <sup>(1)</sup>          | LIMIT                   |
| FC 25 <sup>(1)</sup>          | MAX                     |
| FC 26 <sup>(1)</sup>          | MID                     |
| FC 27 <sup>(1)</sup>          | MIN                     |
| FC 31 <sup>(1)</sup>          | REPLACE                 |
| FC 32 <sup>(1)</sup>          | RIGHT                   |
| FC 36 <sup>(1)</sup>          | ENCO                    |
| FC 36 <sup>(1)</sup>          | SEL                     |
| FC 37                         | DECO                    |
| FC 800                        | LED                     |
| FC 801                        | IM_DATA                 |
| FC 802                        | DeviceStates            |
| FC 803                        | ModuleStates            |
| FC 1070                       | USS_PORT                |
| FC 1072                       | USS_RPM                 |
| FC 1073                       | USS_WPM                 |
| <sup>(1)</sup> MC7+ Anweisung |                         |

### Systemdatentypen (SDTs)

| Numerische Bezeichnung | Symbolische Bezeichnung |
|------------------------|-------------------------|
| SDT 99                 | WWW_CDB                 |
| SDT 513                | CONDITIONS              |
| SDT 581                | Send_Conditions         |
| SDT 582                | Receive_Conditions      |

### Systemfunktionsbausteine (SFBs)

| Numerische Bezeichnung | Symbolische Bezeichnung |
|------------------------|-------------------------|
| SFB 0 <sup>(1)</sup>   | CTU                     |
| SFB 1 <sup>(1)</sup>   | CTD                     |
| SFB 2 <sup>(1)</sup>   | CTUD                    |
| SFB 3 <sup>(1)</sup>   | TP                      |



| Numerische Bezeichnung | Symbolische Bezeichnung |
|------------------------|-------------------------|
| SFB 4 <sup>(1)</sup>   | TON                     |
| SFB 5 <sup>(1)</sup>   | TOF                     |
| SFB 27                 | START_OB                |
| SFB 52                 | RDREC                   |
| SFB 53                 | WRREC                   |
| SFB 54                 | RALRM                   |
| SFB 105                | T_CONFIG                |
| SFB 106                | TDIAG                   |
| SFB 107                | TRESET                  |
| SFB 110                | PORT_CFG                |
| SFB 111                | SEND_CFG                |
| SFB 112                | RCV_CFG                 |
| SFB 113                | SEND_PTP                |
| SFB 114                | RCV_PTP                 |
| SFB 115                | SGN_GET                 |
| SFB 116                | SGN_SET                 |
| SFB 117                | RCV_RST                 |
| SFB 120                | CTRL_HSC                |
| SFB 122                | CTRL_PWM                |
| SFB 140                | DataLogCreate           |
| SFB 141                | DataLogOpen             |
| SFB 142                | DateLogWrite            |
| SFB 143                | DataLogClear            |
| SFB 144                | DataLogClose            |
| SFB 145                | DataLogDelete           |
| SFB 146                | DataLogNewFile          |

### Systemfunktionen (SFCs)

| Numerische Bezeichnung | Symbolische Bezeichnung |
|------------------------|-------------------------|
| SFC 7                  | DP_PRAL                 |
| SFC 11                 | DPSYC_FR                |
| SFC 13                 | DPNRM_DG                |
| SFC 14                 | DPRD_DAT                |
| SFC 16                 | RD_OBINF                |
| SFC 23                 | DEL_DB                  |
| SFC 28                 | SET_TINT                |
| SFC 29                 | CAN_TINT                |
| SFC 30                 | ACT_TINT                |
| SFC 31                 | QRY_TINT                |
| SFC 32                 | SRT_DINT                |
| SFC 33                 | CAN_DINT                |

| Numerische Bezeichnung | Symbolische Bezeichnung |
|------------------------|-------------------------|
| SFC 34                 | QRY_DINT                |
| SFC 41                 | DIS_AIRT                |
| SFC 42                 | EN_AIRT                 |
| SFC 43                 | RE_TRIGR                |
| SFC 45                 | DE_ACT                  |
| SFC 46                 | STP                     |
| SFC 82                 | CREA_DBL                |
| SFC 83                 | READ_DBL                |
| SFC 84                 | WRIT_DBL                |
| SFC 86                 | CREATE_DB               |
| SFC 89                 | RST_EVOV                |
| SFC 99                 | WWW                     |
| SFC 101                | RTM                     |
| SFC 117                | GET_DIAG                |
| SFC 124                | ATTR_DB                 |
| SFC 140                | IO2MOD                  |
| SFC 143                | RD_ADDR                 |
| SFC 154                | RD_LOC_T                |
| SFC 154                | DPWR_DAT                |
| SFC 161                | WR_LOC_T                |
| SFC 180                | ID2LOG                  |
| SFC 181                | LOG2ID                  |
| SFC 182                | ID2GEO                  |
| SFC 190                | SET_CINT                |
| SFC 191                | QRY_CINT                |
| SFC 192                | ATTACH                  |
| SFC 193                | DETACH                  |
| MC7+ Anweisung         | GET_ERROR               |
| MC7+ Anweisung         | GET_ERR_ID              |

### Wissenswertes zu CPU-Firmware-Versionen und STEP 7-Versionen

CPUs und Engineering Software zum Projektieren der CPUs werden ständig weiterentwickelt, um die Leistungsfähigkeit und Sicherheit zu verbessern. Auf diese Weise entstehen neue Versionen, die im Zusammenspiel der Komponenten einige Besonderheiten aufweisen. Die folgenden Abschnitte beschreiben Besonderheiten der S7-1200-CPU mit Firmware-Version V4 im Vergleich zu den Firmware-Versionen V1 bis V3.

### Erforderliche Engineering Software

S7-1200-CPU V4 sind mit STEP 7 ab V12, Servicepack 1 projektierbar.

### **Kompatibilität zwischen Memory-Card-Inhalten und Firmware-Version der CPU**

Memory Cards mit Konfiguration und Programm für eine S7-1200-CPU V1, V2 oder V3 funktionieren nicht in einer S7-1200-CPU V4.

Memory Cards mit Konfiguration und Programm für eine S7-1200-CPU V4 funktionieren nicht in einer S7-1200-CPU V1, V2 oder V3.

Wenn Sie die Memory Card in eine CPU mit einer inkompatiblen Firmware-Version stecken, dann läuft die CPU nicht an. Wenn Sie eine Memory Card für eine CPU V1, V2 oder V3 in eine S7-1200-CPU V4 stecken, gibt diese CPU einen Versionsfehler aus.

### **Online-Gehen und Laden**

Wenn Sie eine S7-1200-CPU mit der Firmware-Version V1, V2 oder V3 mit STEP 7 projektiert haben, dann muss die CPU, zu der Sie online gehen wollen bzw. die Sie laden wollen, ebenfalls eine dieser Firmware-Versionen haben. Sie können nicht mit einer projektierten S7-1200-CPU V1, V2 oder V3 zu einer S7-1200-CPU V4 online gehen.

Umgekehrt können Sie nicht mit einer projektierten S7-1200-CPU V4 zu einer S7-1200-CPU V1, V2 oder V3 online gehen oder diese CPU laden.

### **Bestehende CPU ersetzen**

Wenn Sie in Ihrer Anlage eine bestehende S7-1200-CPU V1, V2 oder V3 durch eine neue CPU mit einer Firmware-Version größer oder gleich 4 ersetzen, müssen Sie die Hardware für diese CPU neu in STEP 7 konfigurieren und parametrieren. Aus der bestehenden CPU bzw. aus dem bestehenden Projekt übertragen Sie die Bausteine des Anwenderprogramms, die Variablen und die Datentypen.

Eine bestehende Konfiguration können Sie nicht übernehmen.

### **Kommunikation mit HMI-Geräten**

Wenn Sie an eine S7-1200-CPU V4 ein HMI-Gerät anschließen, achten Sie auf die geeignete Version der Runtime-Software des HMI-Gerätes.

Gegebenenfalls müssen Sie die neueste HMI-Runtime-Version über die Engineering Software WinCC übertragen.

### **Siehe auch**

Wissenswertes zu Memory Cards (Seite 840)

### 8.1.6.2 Dezentrale Peripherie

#### Dezentrale Peripheriesysteme

#### SIMATIC ET 200 - Für jede Anwendung die richtige Lösung

Mit SIMATIC ET 200 stehen unterschiedlichste dezentrale Peripheriesysteme zur Auswahl.

- Lösungen zum Einsatz im Schaltschrank
- Lösungen ohne Schaltschrank direkt an der Maschine

Weiterhin gibt es Komponenten für den Einsatz im explosionsgefährdeten Bereich. SIMATIC ET 200-Systeme für den schaltschranklosen Aufbau sind in einem robusten, glasfaserverstärkten Kunststoffgehäuse untergebracht und daher schlagfest, schmutzresistent und wasserdicht.

Der modulare Aufbau erlaubt es, die ET 200-Systeme einfach und in kleinen Schritten zu skalieren und zu erweitern. Fertig integrierte Zusatzmodule senken die Kosten und bieten gleichzeitig breit gefächerte Anwendungsmöglichkeiten. Dabei stehen unterschiedlichste Kombinationsmöglichkeiten zur Auswahl:

- digitale und analoge Ein-/Ausgänge
- intelligente Module mit CPU-Funktionalität,
- Sicherheitstechnik,
- Pneumatik,
- Frequenzumrichter
- sowie verschiedene Technologiemodule.

Die Kommunikation über PROFIBUS und PROFINET, das einheitliche Engineering, transparente Diagnosemöglichkeiten sowie die optimale Anbindung an SIMATIC Controller und HMI-Geräte belegen die einzigartige Durchgängigkeit von Totally Integrated Automation.

Die folgende Tabelle zeigt eine Übersicht der Peripheriegeräte für den Einsatz im Schaltschrank:

| Peripheriegerät | Eigenschaften   |
|-----------------|---|
| ET 200S         | <ul style="list-style-type: none"> <li>• Feinmodularer Aufbau mit Mehrleiteranschluss</li> <li>• Multifunktional durch breites Modulspektrum</li> <li>• Einsatz im explosionsgefährdeten Bereich (Zone 2)</li> </ul>                              |
| ET 200S COMPACT | <ul style="list-style-type: none"> <li>• Feinmodularer Aufbau mit Mehrleiteranschluss</li> <li>• Multifunktional durch breites Modulspektrum</li> <li>• Einsatz im explosionsgefährdeten Bereich (Zone 2)</li> <li>• Integrierte DE/DA</li> </ul> |
| ET 200L         | <ul style="list-style-type: none"> <li>• Kostengünstige digitale Blockperipherie</li> <li>• Digitale Elektronikblöcke bis 32 Kanäle</li> </ul>  |

| Peripheriegerät | Eigenschaften  |
|-----------------|--|
| ET 200M         | <ul style="list-style-type: none"> <li>• Modularer Aufbau mit Standard-Baugruppen der SIMATIC-S7-300</li> <li>• Fehlersichere E/A-Baugruppen</li> <li>• Einsatz in explosionsgefährdeten Bereich bis in Zone 2, Sensoren und Aktoren bis in Zone 1</li> <li>• Hohe Anlagenverfügbarkeit, z. B. durch Ziehen und Stecken im laufenden Betrieb</li> </ul>        |
| ET 200iSP       | <ul style="list-style-type: none"> <li>• Modularer Aufbau, auch redundant möglich</li> <li>• Robust und eigensichere Bauweise</li> <li>• Einsatz im explosionsgefährdeten Bereich bis in Zone 1/21, Sensoren und Aktoren dürfen sogar in Zone 0/20 liegen</li> <li>• Hohe Anlagenverfügbarkeit, z. B. durch Ziehen und Stecken im laufenden Betrieb</li> </ul> |

Die folgende Tabelle zeigt eine Übersicht der Peripheriegeräte für den Einsatz ohne Schaltschrank:

| Peripheriegerät | Eigenschaften  |
|-----------------|--|
| ET 200pro       | <ul style="list-style-type: none"> <li>• Modularer Aufbau mit kompakten Gehäuse</li> <li>• Einfache Montage</li> <li>• Multifunktional durch breites Modulspektrum</li> <li>• Hohe Anlagenverfügbarkeit durch Ziehen und Stecken im laufenden Betrieb und stehende Verdrahtung</li> <li>• Umfangreiche Diagnose</li> </ul>   |
| ET 200eco PN    | <ul style="list-style-type: none"> <li>• Kostengünstige, Platz sparende Blockperipherie</li> <li>• Digitale Module mit bis 16 Kanälen (auch parametrierbar)</li> <li>• Analoge Module, IO-Link Master und Lastspannungsverteiler</li> <li>• PROFINET-Anschluss mit 2-Port-Switch in jedem Modul</li> <li>• Über PROFINET flexibel in Linien- und/oder Sternstruktur direkt in der Anlage verteilbar</li> </ul> |
| ET 200eco       | <ul style="list-style-type: none"> <li>• Kostengünstige digitale Blockperipherie</li> <li>• Flexible Anschlussmöglichkeiten</li> <li>• Fehlersichere Module</li> <li>• Hohe Anlagenverfügbarkeit</li> </ul>  |
| ET 200R         | <ul style="list-style-type: none"> <li>• Speziell für den Einsatz an Robotern</li> <li>• Montage direkt am Chassis</li> <li>• Resistent gegen Schweißspritzer durch widerstandsfähiges Metallgehäuse</li> </ul>  |

### Siehe auch

Dokumentation zu ET 200L (<http://support.automation.siemens.com/WW/view/de/1142908>)

Dokumentation zu ET 200S (<http://support.automation.siemens.com/WW/view/de/1144348>)

Dokumentation zu ET 200M (<http://support.automation.siemens.com/WW/view/de/1142798>)

Dokumentation zu ET 200pro (<http://support.automation.siemens.com/WW/view/de/21210852>)

Dokumentation zu ET 200iSP (<http://support.automation.siemens.com/WW/view/de/28930789>)

Dokumentation zu ET 200R (<http://support.automation.siemens.com/WW/view/de/11966255>)

Dokumentation zu ET 200eco PN (<http://support.automation.siemens.com/WW/view/de/29999018>)

Dokumentation zu ET 200eco (<http://support.automation.siemens.com/WW/view/de/12403834>)

## ET 200iSP

### Dezentrales Peripheriegerät ET 200iSP

#### Definition

Das Dezentrale Peripheriegerät ET 200iSP ist ein feinmodularer und eigensicherer DP-Slave mit Schutzart IP 30.

#### Einsatzgebiet

Das Dezentrale Peripheriegerät ET 200iSP ist einsetzbar in explosionsgefährdeten Bereichen mit einer Gas- und Staubatmosphäre:

| Zulassung | ET 200iSP Station* | Ein- und Ausgänge         |
|-----------|--------------------|---------------------------|
| ATEX      | Zone 1, Zone 21    | bis in Zone 0, Zone 20 ** |
| IECEX     | Zone 2, Zone 22    | bis in Zone 0, Zone 20 ** |

\* in Verbindung mit einem entsprechenden Gehäuse  
\*\* beim Elektronikmodul 2 DO Relay UC60V/2A: bis in Zone 1; Zone 21

Selbstverständlich ist das Dezentrale Peripheriegerät ET 200iSP auch im sicheren Bereich einsetzbar.

Direkt neben dem Interfacemodul, das die Daten an den DP-Master überträgt, können Sie Peripheriemodule der ET 200iSP in nahezu beliebiger Kombination stecken. Somit können Sie den Ausbau auf den jeweiligen Bedarf vor Ort ausrichten.

Jede ET 200iSP besteht aus einem Power Supply, einem Interfacemodul und maximal 32 Elektronikmodulen (z.B. Digitale Elektronikmodule). Beachten Sie die maximale Stromaufnahme.

## Terminal- und Elektronikmodule

Das Peripheriegerät ET 200iSP besteht prinzipiell aus verschiedenen passiven Terminalmodulen, auf denen Sie die das Power Supply und die Elektronikmodule stecken.

Die ET 200iSP wird über einen Anschlussstecker am Terminalmodul TM-IM/EM an den PROFIBUS RS 485-IS angeschlossen. Jede ET 200iSP ist ein DP-Slave am PROFIBUS RS 485-IS.

## DP-Master

Alle Module der ET 200iSP können mit allen DP-Mastern kommunizieren, die sich nach der Norm *IEC 61784-1:2002 Ed1 CP 3/1* mit dem Übertragungsprotokoll "DP" (DP steht für Dezentrale Peripherie) verhalten.

## Siehe auch

Dokumentation zur ET 200iSP (<http://support.automation.siemens.com/WW/view/de/28930789>)

## Kanal und IEEE-Variable zuordnen

### Eigenschaften

Die Analogen Elektronikmodule 4 AI I 2WIRE/HART, 4 AI I 4WIRE/ HART und 4 AO I HART unterstützen bis zu 4 IEEE-Variablen.

Im Prozessabbild der Eingänge (PAE) stehen pro Modul maximal 20 Byte für die IEEE-Variablen zur Verfügung. Für die 4 IEEE-Variablen sind somit 4 Blöcke mit jeweils 5 Byte innerhalb des PAE vorhanden.

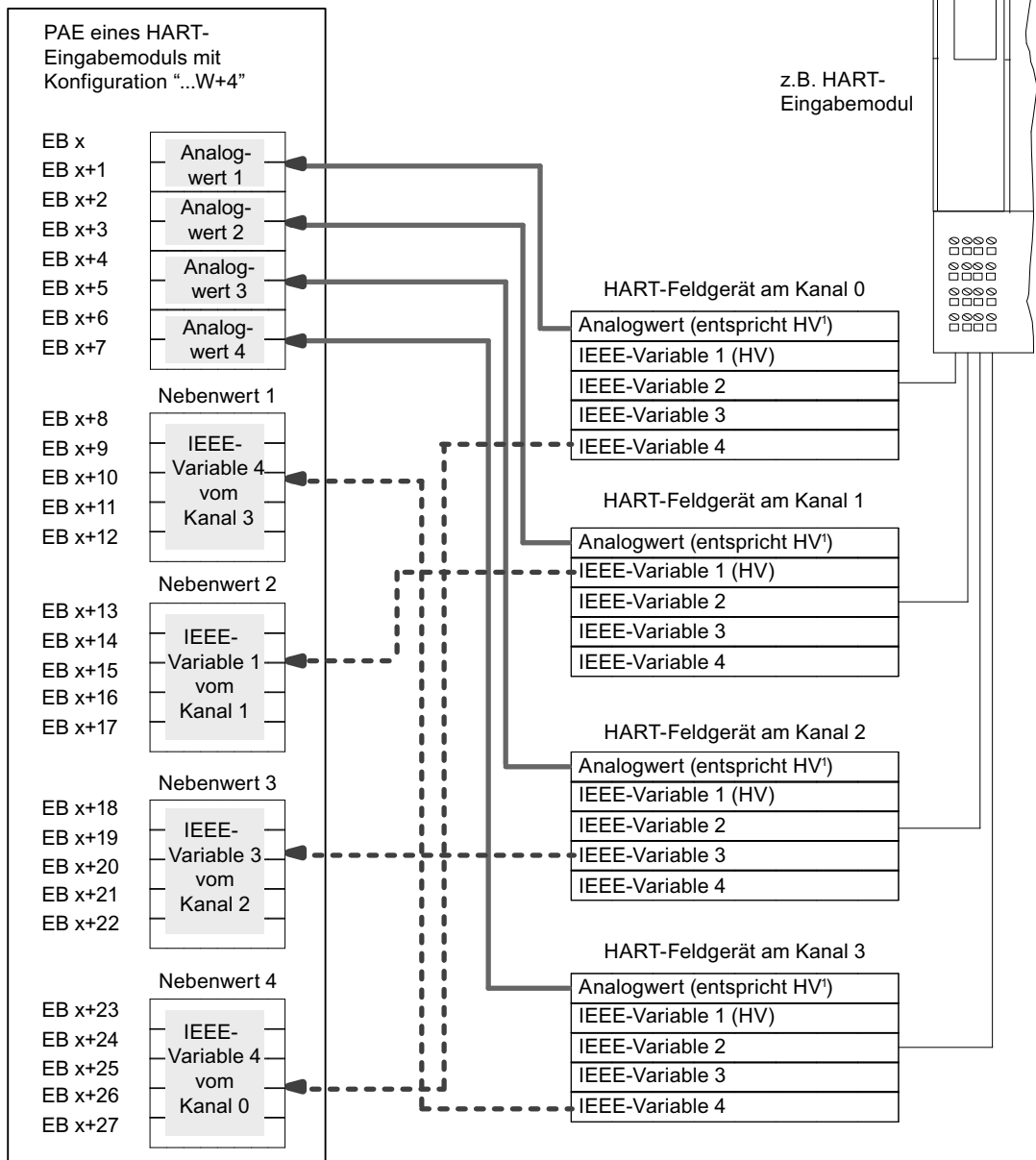
### Voraussetzungen

Das HART-Feldgerät muss die parametrisierte Anzahl der IEEE-Variablen unterstützen.

**IEEE-Variablen zuordnen**

Sie ordnen die IEEE-Variablen der Feldgeräte beliebig den 4 Blöcken im PAE zu.

- feste Zuordnung der Analogwerte im PAE
- - - - - beliebiges zuordnen der IEEE-Variablen im PAE



<sup>1</sup> HV=Hauptvariable

Siehe auch

Dokumentation zur ET 200iSP (<http://support.automation.siemens.com/WW/view/de/28930789>)



## Vergleichsstellen parametrieren für Thermoelemente

### Kompensation der Vergleichsstellentemperatur

Sie haben verschiedene Möglichkeiten, die Vergleichsstellentemperatur zu erfassen, um aus der Temperaturdifferenz zwischen Vergleichsstelle und Messstelle einen absoluten Temperaturwert zu erhalten.

Tabelle 8-49 Kompensation der Vergleichsstellentemperatur

| Möglichkeit   | Erläuterung  | Parameter Vergleichsstelle  |
|---|--|---|
| Keine Kompensation  | Sie erfassen nicht nur die Temperatur der Messstelle. Die Temperatur der Vergleichsstelle (Übergang von Cu-Leitung auf Thermoelementeleitung) beeinflusst zusätzlich die Thermospannung. Der Messwert ist somit fehlerbehaftet.  | keine   |
| Verwendung eines Widerstandsthermometers Pt100 Klimabereich zum Erfassen der Vergleichsstellentemperatur (günstigste Methode) | Sie können die Vergleichsstellentemperatur mittels eines Widerstandsthermometers (Pt100 Klimabereich) erfassen. Dieser Temperaturwert wird in der ET 200iSP bei entsprechender Parametrierung an die 4 AI TC-Module verteilt und in den Modulen mit den ermittelten Temperaturwert der Messstelle verrechnet.<br>Anzahl der Vergleichsstellen: 2 | Die Parametrierung der IM152 und des 4 AI TC muss abgestimmt sein: <ul style="list-style-type: none"> <li>• 4 AI RTD parametriert auf Pt100 Klimabereich am richtigen Steckplatz;</li> <li>• 4 AI TC: Vergleichsstelle : "ja"; Vergleichsstellennummer "1" oder "2" auswählen</li> <li>• IM 152-1: Zuweisung der Vergleichsstelle auf einen Steckplatz mit 4 AI RTD; Auswahl eines Kanals;</li> </ul> |
| Interne Kompensation 4 AI TC  | Auf die Klemmen des Terminalmoduls des EM 4 AI TC wird das TC-Sensormodul (Temperatursensor) montiert. Der Temperatursensor meldet die Klemmentemperatur an das 4 AI TC. Dieser Wert wird mit dem gemessenen Wert aus dem Kanal des Elektronikmoduls verrechnet.   | <ul style="list-style-type: none"> <li>• 4 AI TC: Vergleichsstellennummer "Intern"</li> </ul>   |

### Kompensation durch Widerstandsthermometer am 4 AI RTD

Wenn Thermoelemente, die an die Eingänge der 4 AI TC angeschlossen sind, dieselbe Vergleichsstelle haben, kompensieren Sie durch ein 4 AI RTD.

Für die Kanäle des Moduls 4 AI TC können Sie als Vergleichsstellennummer "1", "2" oder "Intern" wählen. Wenn Sie "1" oder "2" wählen, dann wird für die 4 Kanäle immer dieselbe Vergleichsstelle (RTD-Kanal) verwendet.

### Parametrierung der Vergleichsstelle

Die Vergleichsstellen für die Elektronikmodule 4 AI TC stellen Sie über folgende Parameter ein:

Tabelle 8-50 Parameter der Vergleichsstelle

| Parameter  | Modul   | Wertebereich   | Erläuterung  |
|--|---------|--|--|
| Steckplatz Vergleichsstelle 1 bis Steckplatz 2                             | IM 152  | keinen, 4 bis 35   | Mit diesem Parameter können Sie bis zu 2 Steckplätze (keinen, 4 bis 35) zuordnen, auf denen sich die Kanäle zur Vergleichstemperaturmessung (Ermittlung des Kompensationswertes) befinden. |
| Eingang Vergleichsstelle Vergleichsstelle 1 bis 4 Eingang Vergleichsstelle | IM 152  | RTD an Kanal 0<br>RTD an Kanal 1<br>RTD an Kanal 2<br>RTD an Kanal 3 | Mit diesem Parameter legen Sie den Kanal (0/1/2/3) zur Vergleichstemperaturmessung (Ermittlung des Kompensationswertes) für den zugeordneten Steckplatz fest.                              |
| Vergleichsstelle E0 bis Vergleichsstelle E3                                | 4 AI TC | keine<br>ja  | Über diesen Parameter können Sie die Verwendung der Vergleichsstelle freigeben.  |
| Vergleichsstellenummer   | 4 AI TC | 1<br>2<br>Intern   | Mit diesem Parameter weisen Sie die Vergleichsstelle (1, 2) zu, die die Vergleichstemperatur (Kompensationswert) enthält.  |

### Siehe auch

Dokumentation zur ET 200iSP (<http://support.automation.siemens.com/WW/view/de/28930789>)

### Grundlagen der Zeitstempelung

#### Eigenschaften

Zeitstempelung mit der IM 152 ist möglich in Kundenapplikationen mit Nutzung des FB 62 (FB TIMESTMP).

#### Funktionsweise

Ein geändertes Eingangssignal wird mit einem Zeitstempel versehen und in einen Puffer (Datensatz) abgelegt. Wenn zeitgestempelte Signale vorliegen oder ein Datensatz voll ist, wird ein Prozessalarm zum DP-Master generiert. Der Puffer wird mit "Datensatz lesen" ausgewertet. Bei Ereignissen, die die Zeitstempelung beeinflussen (Kommunikation mit DP-Master unterbrochen, Telegrammausfall vom Uhrzeitmaster, ...) werden Sondermeldungen generiert.

## Parametrieren

Mit der Parametrierung legen Sie fest, welche Nutzdaten der IM 152 überwacht werden. Für die Zeitstempelung sind das Digitaleingänge, die auf Signaländerungen überwacht werden.

| Parameter                            | Einstellung  | Beschreibung  |
|--------------------------------------|--|---|
| Zeitstempelung                       | <ul style="list-style-type: none"><li>• gesperrt</li><li>• freigegeben</li></ul>             | Aktivieren Sie die Zeitstempelung für die Kanäle des Elektronikmoduls 8 DI NAMUR. |
| Flankenauswertung kommendes Ereignis | <ul style="list-style-type: none"><li>• steigende Flanke</li><li>• fallende Flanke</li></ul> | Legen Sie die Art der Signaländerung fest, die zeitgestempelt wird.               |

## Zählen

### Eigenschaften Zählen

### Zählfunktionen

Das Elektronikmodul 8 DI NAMUR ist mit parametrierbaren Zählfunktionen ausgestattet:

- 2 16-Bit Vorwärtszähler (Normale Zählfunktion) oder
- 2 16-Bit Rückwärtszähler (Periodische Zählfunktion) oder
- 1 32-Bit Rückwärtszähler (Kaskadierzählfunktion)
- Vorgabe eines Sollwertes durch das PAA
- TOR-Funktion
- Die Funktion der Steuersignale der Zähler können Sie konfigurieren:
  - Konfiguration Kanal 0..1: "Zähler", Kanal 2..7: "DI": Es werden 2 Zähler konfiguriert. Die Steuersignale der Zähler sind im PAA (Prozessabbild der Ausgänge) hinterlegt.
  - Konfiguration Kanal 0..1: "Zähler", Kanal 2..7: "Control": Es werden 2 Zähler konfiguriert. Die Steuersignale der Zähler sind im PAA hinterlegt. Zusätzlich werden Sie von den Digitaleingängen des 8 DI NAMUR gesteuert.

### Siehe auch

Funktionsweise (Seite 918)

Zähler konfigurieren (Seite 920)

Zähler parametrieren (Seite 923)

Funktionsweise

16-Bit Vorwärtszähler (Normale Zählfunktion)

Der Zählbereich beträgt 0 bis 65535.

Mit jedem Zählimpuls am Digitaleingang erhöht sich der Zählerstand um den Wert 1. Mit Erreichen der Zählgrenze wird der Zählerstand auf den Wert 0 gesetzt und von diesem Wert wieder hochgezählt.

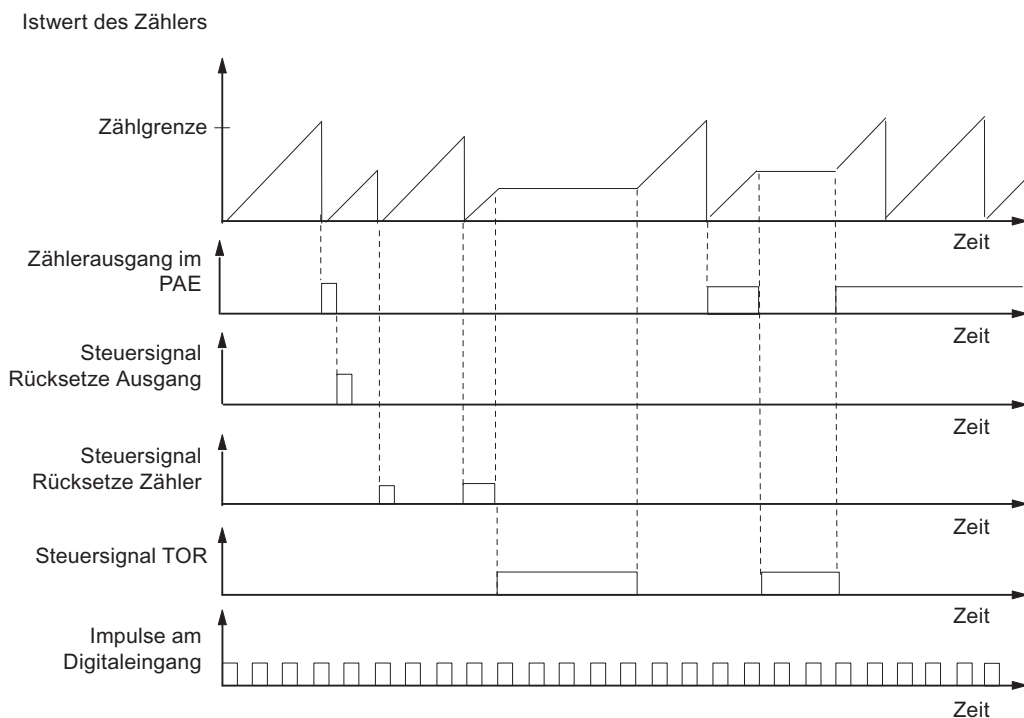
Bei einem Überlauf des Zählers wird der zugehörige Ausgang im PAE gesetzt.

Durch eine positive Flanke des Steuersignals *Rücksetze Ausgang* wird der Ausgang im PAE zurückgesetzt. Der aktuelle Zählwert wird dadurch nicht beeinflusst.

Beim 16-Bit Vorwärtszählen werden keine Ausgänge im PAA gesetzt. Diese sind generell zurückgesetzt.

Die positive Flanke des Steuersignals *Rücksetze Zähler* setzt den Zählerstand auf den Wert 0 und den gesetzten Zählerausgang zurück.

Das Steuersignal *TOR* hält den Zählvorgang bei einer positiven Flanke an. Erst bei einer negativen Flanke werden wieder Zählimpulse am Digitaleingang verarbeitet. Das Steuersignal *Rücksetze Zähler* ist auch bei aktiviertem *TOR* wirksam.



16-Bit Rückwärtszähler (Periodische Zählfunktion)

Der maximale Zählbereich beträgt 65535 bis 0.

Zum Startzeitpunkt des Zählers wird der Istwert auf den vorgegebenen Sollwert gesetzt. Mit jedem Zählimpuls reduziert sich der Istwert um den Wert 1. Erreicht der Istwert den Wert 0,

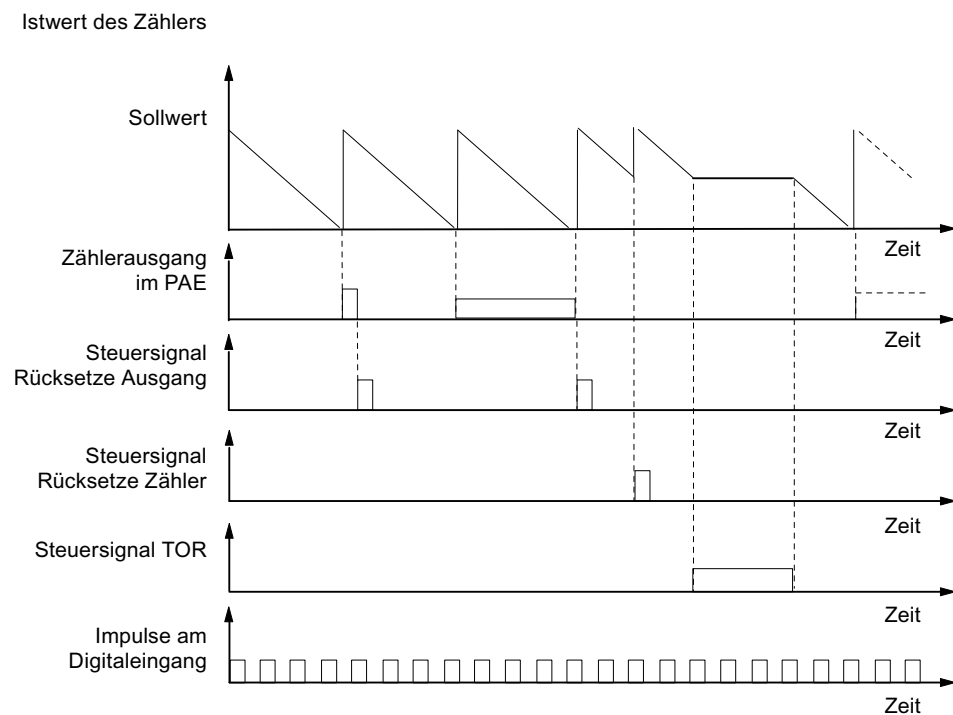
dann wird der zugehörige Ausgang im PAE eingeschaltet und der Istwert wieder auf den vorgegebenen Sollwert gesetzt. Von diesem Wert wird wieder zurückgezählt.

Die positive Flanke des Steuersignals *Rücksetze Zähler* setzt den Istwert auf den vorgegebenen Sollwert und den zugehörigen Ausgang im PAE zurück.

Durch eine positive Flanke des Steuersignals *Rücksetze Ausgang* wird der Ausgang im PAE zurückgesetzt. Der aktuelle Zählwert wird dadurch nicht beeinflusst.

Das Steuersignal *TOR* hält den Zählvorgang bei einer positiven Flanke an. Gleichzeitig wird der zugeordnete Ausgang im PAE zurückgesetzt. Erst bei einer negativen Flanke werden wieder Zählimpulse am Digitaleingang verarbeitet. Die Steuersignale *Rücksetze Ausgang* und *Rücksetze Zähler* sind auch bei aktiviertem *TOR* wirksam.

Der Sollwert des Zählers wird über das PAA vorgegeben und verändert. Durch die positive Flanke des Steuersignals *Rücksetze Zähler* oder einen Nulldurchgang des Zählers wird der Sollwert übernommen.



### 32-Bit Rückwärtszähler (Kaskadierzählfunktion)

Der maximale Zählbereich beträgt 4294967295 bis 0.

Die Funktionsweise ist identisch zum 16-Bit Rückwärtszähler. Der Kanal 1 hat keine Funktion.

### Siehe auch

Eigenschaften Zählen (Seite 917)

## Zähler konfigurieren

### Vorgehensweise

1. Ziehen Sie mit der Maus das Modul 8 DI Namur aus dem Hardware-Katalog in das dezentral Peripheriegerät ET 200iSP.
2. Wählen Sie die gewünschte Konfiguration (Kanal 0..1: "Zähler", Kanal 2..7: "DI" bzw. "Control"). In den Eigenschaften des Moduls (Inspektorfenster) finden Sie diese Einstellung unter "Parameter > Eingänge > Konfiguration".

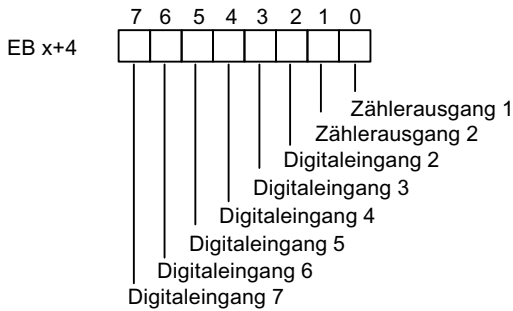
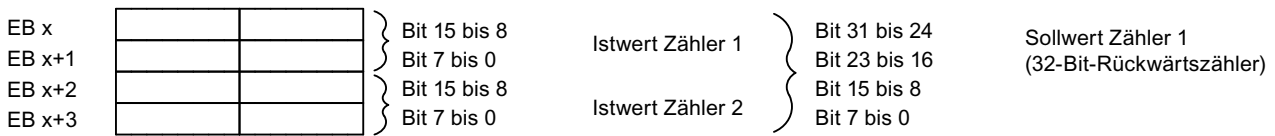
### Konfiguration Kanal 0..1: "Zähler", Kanal 2..7: "DI"

- Belegung der Digitaleingänge am Elektronikmodul 8 DI NAMUR

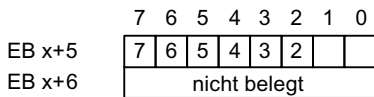
Tabelle 8-51 Belegung der Digitaleingänge bei Kanal 0..1: "Zähler", Kanal 2..7: "DI":

| Digitaleingang | Klemme | Belegung   |
|----------------|--------|--|
| Kanal 0        | 1, 2   | Zähler 1   |
| Kanal 1        | 5, 6   | Zähler 2 (bei 32-Bit Rückwärtszähler nicht relevant) |
| Kanal 2        | 9, 10  | Digitaleingang 2                                     |
| Kanal 3        | 13, 14 | Digitaleingang 3                                     |
| Kanal 4        | 3, 4   | Digitaleingang 4                                     |
| Kanal 5        | 7, 8   | Digitaleingang 5                                     |
| Kanal 6        | 11, 12 | Digitaleingang 6                                     |
| Kanal 7        | 15, 16 | Digitaleingang 7                                     |

• Belegung des Prozessabbildes der Eingänge (PAE)

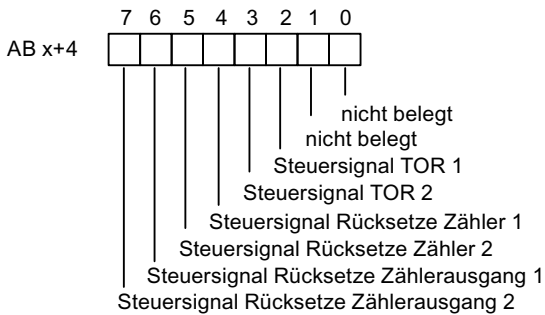
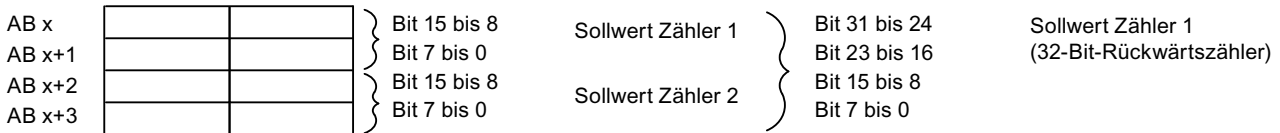


S7-Format



Wertstatus für Kanal 2 bis 7:  
 1<sub>B</sub>: Eingangssignal ist gültig  
 0<sub>B</sub>: Eingangssignal ist ungültig

• Belegung des Prozessabbildes der Ausgänge (PAA)



### Konfiguration Kanal 0..1: "Zähler", Kanal 2..7: "CONTROL"

Bei dieser Konfiguration können Sie die Zähler zusätzlich über die Digitaleingänge steuern.

- Belegung der Digitaleingänge am Elektronikmodul 8 DI NAMUR  
 Weitere Hinweise zur Anschlussbelegung finden Sie in den Technischen Daten des Elektronikmoduls 8 DI NAMUR.

Tabelle 8-52 Belegung der Digitaleingänge bei 2 Count/ 6 Control

| Digitaleingang | Klemme | Belegung   |
|----------------|--------|--|
| Kanal 0        | 1, 2   | Zähler 1   |
| Kanal 1        | 5, 6   | Zähler 2 (bei 32-Bit Rückwärtszähler nicht relevant) |
| Kanal 2        | 9, 10  | Steuersignal <i>TOR 1</i>                            |
| Kanal 3        | 13, 14 | Steuersignal <i>TOR 2</i>                            |
| Kanal 4        | 3, 4   | Steuersignal <i>Rücksetze Zähler 1</i>               |
| Kanal 5        | 7, 8   | Steuersignal <i>Rücksetze Zähler 2</i>               |
| Kanal 6        | 11, 12 | Steuersignal <i>Rücksetze Zählerausgang 1</i>        |
| Kanal 7        | 15, 16 | Steuersignal <i>Rücksetze Zählerausgang 2</i>        |

- Belegung des Prozessabbildes der Eingänge (PAE)  
 Die Belegung ist identisch zur Konfiguration 0..1: "Zähler", Kanal 2..7: "DI".
- Belegung des Prozessabbildes der Ausgänge (PAA)  
 Die Belegung ist identisch zur Konfiguration 0..1: "Zähler", Kanal 2..7: "DI".

#### Siehe auch

Eigenschaften Zählen (Seite 917)



## Zähler parametrieren

### Parameter für die Zählfunktion

Nachfolgend werden nur die Parameter erläutert, die für die Zähler relevant sind. Diese sind Bestandteil der Parameter des Elektronikmoduls 8 DI NAMUR und abhängig von der gewählten Konfiguration:

Tabelle 8-53 Parameter für die Zähler

| Parameter                  | Einstellung   | Beschreibung   |
|----------------------------|---|--|
| Gebertyp<br>Zählereingänge | <ul style="list-style-type: none"><li>• Kanal gesperrt</li><li>• NAMUR-Geber</li><li>• Einzelkontakt unbeschaltet</li></ul>               | Wählen Sie für den Kanal 0 oder 1 den Geber für den jeweiligen Zähler aus.   |
| Betriebsart Zähler 1       | <ul style="list-style-type: none"><li>• Normale Zählfunktion</li><li>• Periodische Zählfunktion</li><li>• Kaskadierzählfunktion</li></ul> | Wählen Sie die Betriebsart des Zählers 1.  |
| Betriebsart Zähler 2       | <ul style="list-style-type: none"><li>• Normale Zählfunktion</li><li>• Periodische Zählfunktion</li><li>• Kaskadierzählfunktion</li></ul> | Wählen Sie die Betriebsart des Zählers 2. Dieser Parameter ist nicht relevant, wenn Sie den Parameter "Betriebsart Zähler 1" auf "Kaskadierzählfunktion" gestellt haben. |

### Siehe auch

Eigenschaften Zählen (Seite 917)

## Frequenzmessen

### Eigenschaften Frequenzmessen

#### Eigenschaften

Das Elektronikmodul 8 DI NAMUR ermöglicht auf den Kanälen 0 und 1 das Messen von Frequenzen:

- 2 Frequenzmesser von 1 Hz bis 5 kHz
- Parametrierbares Messfenster (TOR)
- Die Signale der Frequenzmesser werden über die Digitaleingänge des Elektronikmoduls eingelesen.

## Siehe auch

- Funktionsweise (Seite 924)
- Frequenzmesser konfigurieren (Seite 924)
- Frequenzmesser parametrieren (Seite 926)

## Funktionsweise

### Frequenzmessung

Die Signalfrequenzen werden aus den Eingangssignalen des Kanal 0 bzw. 1 des Elektronikmoduls ermittelt. Zur Berechnung der Frequenz werden die Signale innerhalb eines parametrierbaren Messfensters gemessen.

Die Frequenz wird als 16-Bit Wert im Festpunktformat dargestellt und in das PAE übertragen.

Die Frequenzmesser berechnen die Frequenz nach folgender Formel:

$$\text{Frequenz [Hz]} = \frac{\text{Anzahl der steigenden Flanken am Digitaleingang}}{\text{Messfenster [s]}}$$

### Überschreitung der Eingangsfrequenz

Wenn die Eingangsfrequenz 5kHz übersteigt, dann wird als Istwert 7FFF<sub>H</sub> gemeldet. Bei einer Eingangsfrequenz über ca. 8kHz können keine korrekten Istwerte mehr ausgegeben werden.

## Siehe auch

- Eigenschaften Frequenzmessen (Seite 923)

## Frequenzmesser konfigurieren

### Vorgehensweise

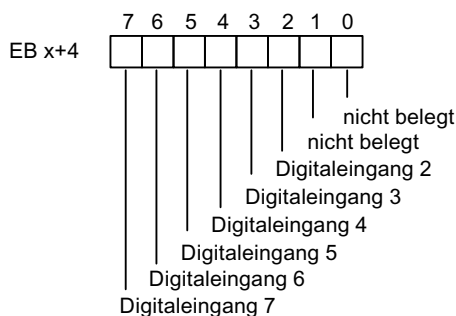
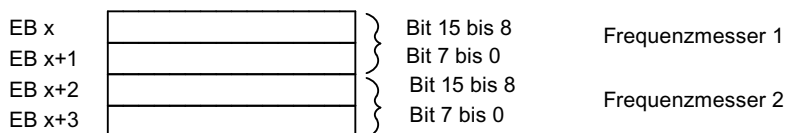
1. Ziehen Sie mit der Maus das Modul 8 DI Namur aus dem Hardware-Katalog in das dezentral Peripheriegerät ET 200iSP.
2. Wählen Sie die gewünschte Konfiguration (Kanal 0..1: "Trace", Kanal 2..7: "DI"). In den Eigenschaften des Moduls (Inspektorfenster) finden Sie diese Einstellung unter "Parameter > Eingänge > Konfiguration".

### Konfiguration 0..1: "Trace", Kanal 2..7: "DI"

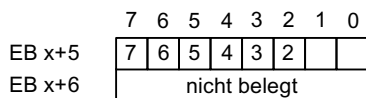
Belegung der Digitaleingänge am Elektronikmodul 8 DI NAMUR

| Digitaleingang | Klemme | Belegung         |
|----------------|--------|------------------|
| Kanal 0        | 1, 2   | Frequenzmesser 1 |
| Kanal 1        | 5, 6   | Frequenzmesser 2 |
| Kanal 2        | 9, 10  | Digitaleingang 2 |
| Kanal 3        | 13, 14 | Digitaleingang 3 |
| Kanal 4        | 3, 4   | Digitaleingang 4 |
| Kanal 5        | 7, 8   | Digitaleingang 5 |
| Kanal 6        | 11, 12 | Digitaleingang 6 |
| Kanal 7        | 15, 16 | Digitaleingang 7 |

Belegung des Prozessabbildes der Eingänge (PAE) bei Konfiguration Kanal 0..1: "Trace", Kanal 2..7: "DI"



S7-Format



Wertstatus für Kanal 2 bis 7:  
1<sub>B</sub>: Eingangssignal ist gültig  
0<sub>B</sub>: Eingangssignal ist ungültig

Belegung des Prozessabbildes der Ausgänge (PAA): Das PAA ist nicht belegt.

### Siehe auch

Eigenschaften Frequenzmessen (Seite 923)

## Frequenzmesser parametrieren

### Parameter für Frequenzmesser

Nachfolgend werden nur die Parameter erläutert, die für die Frequenzmesser relevant sind. Diese sind Bestandteil der Parameter des Elektronikmoduls 8 DI NAMUR:

Tabelle 8-54 Parameter für die Frequenzmesser

| Parameter                    | Einstellung   | Beschreibung  |
|------------------------------|---|---|
| Gebertyp<br>Frequenzeingänge | <ul style="list-style-type: none"> <li>• Kanal gesperrt</li> <li>• NAMUR-Geber</li> <li>• Einzelkontakt unbeschaltet</li> </ul> | Wählen Sie für den Kanal 0 oder 1 den Geber für den jeweiligen Frequenzmesser aus.  |
| Messfenster (TOR)            | <ul style="list-style-type: none"> <li>• 50 ms</li> <li>• 200 ms</li> <li>• 1 s</li> </ul>                                      | <p>Wählen Sie für den Kanal 0 bzw. 1 das erforderliche Messfenster aus.</p> <p>Damit Sie eine möglichst hohe Genauigkeit bei der Frequenzmessung erreichen, beachten Sie folgende Regeln:</p> <ul style="list-style-type: none"> <li>• hohe Frequenzen (&gt;4kHz): niedriges Messfenster (50ms) einstellen</li> <li>• variable/ mittlere Frequenzen: mittleres Messfenster (200ms) einstellen</li> <li>• niedrige Frequenzen (&lt;1kHz): hohes Messfenster (1s) einstellen</li> </ul> |

### Siehe auch

Eigenschaften Frequenzmessen (Seite 923)

## ET 200eco PN

### Dezentrales Peripheriegerät ET 200eco PN

#### Definition

Das Dezentrale Peripheriegerät ET 200eco PN ist ein kompaktes PROFINET IO-Device in der Schutzart IP 65/66 bzw. IP 67 und UL Enclosure Type 4x, Indoor use only.

## Einsatzgebiet

Die Einsatzgebiete der ET 200eco PN ergeben sich aus deren besonderen Eigenschaften.

- Durch die robuste Bauweise und die Schutzart IP 65/66 bzw. IP 67 ist das Dezentrale Peripheriegerät ET 200eco PN vor allem für den Einsatz in rauher Industrieumgebung geeignet.
- Der kompakte Aufbau der ET 200eco PN ermöglicht den Einsatz in platzsensitiven Bereichen.
- Das einfache Handling der ET 200eco PN gewährleistet eine schnelle Inbetriebnahme und Wartung.

## Eigenschaften

Die ET 200eco PN verfügt über die folgenden Eigenschaften:

- Integrierter Switch mit 2 Ports
- Unterstützte Ethernet-Dienste:
  - ping
  - arp
  - Netzdiagnose (SNMP)
  - LLDP
- Alarme
  - Diagnosealarme
  - Maintenance Alarme
- Port-Diagnose
- Isochronous Real-Time-Kommunikation
- Priorisierter Hochlauf
- Gerätetausch ohne PG
- Medienredundanz
- Über IO-Link Master Interfacemodul Anbindung an intelligente Sensoren/Aktoren.

## IO-Controller

Die ET 200eco PN kann mit allen IO-Controllern kommunizieren, die sich nach der Norm IEC 61158 verhalten.

Die ET 200eco PN ist projektierbar an einer CPU mit erweiterter Diagnose.

## Siehe auch

Dokumentation zu ET 200eco PN (<http://support.automation.siemens.com/WW/view/de/29999018>)

## Parameterbeschreibung Analogeingabe

### Sammeldiagnose

Mit diesem Parameter können Sie generell die Diagnose des Gerätes freigeben und sperren. Die Diagnosen "Fehler" und "Parametrierfehler" sind immer unabhängig von der Sammeldiagnose freigegeben.

### Diagnose fehlende 1L+

Wenn Sie diesen Parameter freigeben, dann wird die Prüfung auf fehlende Versorgungsspannung freigegeben.

### Diagnose Kurzschluss Gebersversorgung

Wenn Sie diesen Parameter freigeben, dann wird bei einem Kurzschluss der Gebersversorgung nach Masse bei aktiviertem Kanal eine Diagnose generiert. Die Gebersversorgung wird für die Stecker X1, X3, X5 und X7 überwacht. Es kann nicht differenziert werden, bei welchem Stecker der Geberkurzschluss auftritt.

### Störfrequenzunterdrückung

Mit diesem Parameter stellen Sie abhängig von der Auswahl der Störfrequenz die Integrationszeit des Gerätes ein. Wählen Sie hier die Frequenz der verwendeten Netzspannung. Störfrequenzunterdrückung **Aus** bedeutet 500 Hz, was bei einem Messkanal einer Integrationszeit von 2 ms entspricht.

### Temperatureinheit

Stellen Sie hier die Einheit ein, in der Sie die Temperatur erfassen möchten.

### Messart (kanalweise)

Mit diesem Parameter stellen Sie die Messart ein, z. B. Spannung. Wenn Sie einen Kanal nicht verwenden, dann müssen Sie als Einstellung **Deaktiviert** wählen. Bei einem deaktivierten Kanal ist die Wandlungs- und Integrationszeit des Kanals = 0 s und die Zykluszeit optimiert.

### Messbereich

Mit diesem Parameter stellen Sie den Messbereich der gewählten Messart ein.

### Temperaturkoeffizient (bei RTD, Thermowiderstand)

Der Korrekturfaktor für den Temperaturkoeffizienten ( $\alpha$ -Wert) gibt an, um wieviel sich der Widerstand eines bestimmten Materials relativ ändert, wenn sich die Temperatur um 1 °C erhöht.

Die  $\alpha$ -Werte entsprechen den Normen EN 60751, GOST 6651, JIS C 1604 und ASTM E-1137.

Der Temperaturkoeffizient ist abhängig von der chemischen Zusammensetzung des Materials.

## Glättung

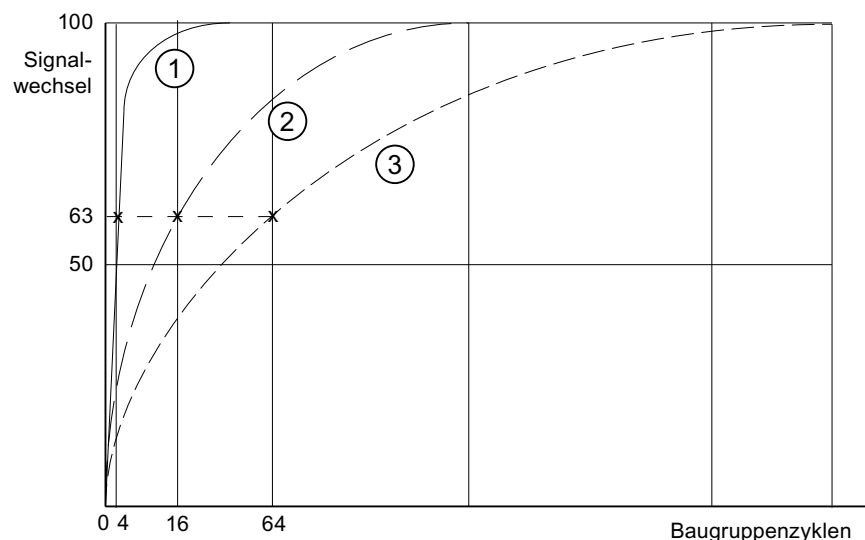
Durch die Glättung von Analogwerten wird ein stabiles Analogsignal für die Weiterverarbeitung erzeugt. Die Glättung der Analogwerte ist sinnvoll bei langsamen Nutzsignaländerungen (Messwertänderungen), z. B. Temperaturmessungen.

Die Messwerte werden mittels digitaler Filterung geglättet. Die Glättung wird erreicht, indem das Gerät Mittelwerte aus einer festgelegten Anzahl von gewandelten (digitalisierten) Analogwerten bildet.

Sie parametrieren die Glättung in maximal 4 Stufen (keine, schwach, mittel, stark). Die Stufe bestimmt die Anzahl der Baugruppenzyklen, die zur Mittelwertbildung herangezogen werden.

Je stärker die Glättung durchgeführt wird, umso stabiler ist der geglättete Analogwert und umso länger dauert es, bis der geglättete Analogwert nach einem Signalwechsel anliegt (siehe folgendes Beispiel).

Das folgende Bild zeigt, nach wie vielen Baugruppenzyklen bei einer Sprungantwort der geglättete Analogwert zu annähernd 100 % anliegt, in Abhängigkeit von der eingestellten Glättung. Das Bild gilt für jeden Signalwechsel am Analogeingang. Der Glättungswert gibt die Anzahl der Zyklen an, die die Baugruppe braucht, bis 63 % vom Endwert der Signalwechsel erreicht ist.



- ① Glättung schwach
- ② Glättung mittel
- ③ Glättung stark

## Diagnose Drahtbruch

Wenn Sie diesen Parameter freigeben, dann wird bei einem Drahtbruch die Diagnose **Leitungsbruch** generiert.

Beachten Sie bei einem Drahtbruch in den Messbereichen 1 bis 5 V und 4 bis 20 mA folgende Regeln:

| Parameter   | Ereignis   | Messwert          | Erläuterung   |
|---|------------|-------------------|---|
| Drahtbruch freigegeben <sup>1</sup>   | Drahtbruch | 7FFF <sub>H</sub> | Diagnose <b>Leitungsbruch</b>   |
| Drahtbruch gesperrt <sup>1</sup><br>Unterlauf freigegeben   | Drahtbruch | 8000 <sub>H</sub> | Messwert nach Verlassen des Untersteuerungsbereiches<br><br>Diagnosemeldung <b>Unterer Grenzwert unterschritten</b> |
| Drahtbruch gesperrt <sup>1</sup><br>Überlauf gesperrt   | Drahtbruch | 8000 <sub>H</sub> | Messwert nach Verlassen des Untersteuerungsbereiches  |
| <sup>1</sup> Messbereichsgrenzen für die Erkennung des Drahtbruchs und der Messbereichsunterschreitung: <ul style="list-style-type: none"> <li>• 1 bis 5 V: bei 0,296 V</li> <li>• 4 bis 20 mA: bei 1,185 mA</li> </ul> |            |                   |   |

### Diagnose Unterlauf

Wenn Sie diesen Parameter freigeben, dann wird die Diagnose **Unterlauf** generiert, wenn der Messwert den Bereich Unterlauf erreicht.

### Diagnose Überlauf

Wenn Sie diesen Parameter freigeben, dann wird die Diagnose **Überlauf** generiert, wenn der Messwert den Bereich Überlauf erreicht.

### Vergleichsstelle für Thermowiderstand (TC)

Wird die Messstelle einer anderen Temperatur ausgesetzt als die freien Enden des Thermopaars (Anschlussstelle), entsteht zwischen den freien Enden eine Spannung, die Thermospannung. Die Höhe der Thermospannung hängt von der Differenz zwischen der Temperatur der Messstelle und der Temperatur an den freien Enden ab, sowie von der Art der Werkstoffkombination des Thermopaars. Da mit einem Thermopaar immer eine Temperaturdifferenz erfasst wird, müssen die freien Enden an der Vergleichsstelle auf bekannter Temperatur gehalten werden, um die Temperatur der Messstelle bestimmen zu können.

Wenn Sie **Interne Kompensation** einstellen, wird die Temperatur der Messstelle im Gehäuse des Peripheriegerätes gemessen. Bei Einstellung **Externe Kompensation** können Sie eine Kompensationsbox vorschalten, um eine größere Genauigkeit bei der Temperaturmessung zu erzielen.

### Parameterbeschreibung Analogausgabe

#### Sammeldiagnose

Mit diesem Parameter können Sie generell die Diagnose des Gerätes freigeben und sperren.

Die Diagnosen "Fehler" und "Parametrierfehler" sind immer unabhängig von der Sammeldiagnose freigegeben.



### Diagnose fehlende 1L+

Wenn Sie diesen Parameter freigeben, dann wird die Prüfung auf fehlende Versorgungsspannung freigegeben.

### Diagnose Kurzschluss Gebersversorgung

Wenn Sie diesen Parameter freigeben, dann wird bei einem Kurzschluss der Gebersversorgung nach Masse eine Diagnose generiert. Diese Diagnose wird bei Freigabe der Sammeldiagnose freigeschaltet.

### Verhalten bei CPU/Master-STOP

Wählen Sie aus, wie die Ausgänge der Baugruppe im Fall eines CPU-STOPs reagieren sollen:

- Abschalten  
Das Peripheriegerät geht in den sicheren Zustand. Das Prozessabbild der Ausgänge wird gelöscht (=0).
- Letzten Wert halten  
Das Peripheriegerät behält den zuletzt vor STOP ausgegebenen Wert bei.
- Ersatzwert  
Das Peripheriegerät gibt den zuvor für den Kanal eingestellten Wert aus.

---

#### **Hinweis**

Vergewissern Sie sich, dass sich die Anlage im Fall "letzten Wert halten" immer in einem sicheren Zustand befindet!

---

### Ausgabeart

Mit diesem Parameter stellen Sie die Ausgabeart ein, z. B. Spannung. Wenn Sie einen Kanal nicht verwenden, dann sollten Sie als Einstellung **Deaktiviert** wählen. Bei einem deaktivierten Kanal ist die Wandlungs- und Integrationszeit des Kanals = 0 s und die Zykluszeit optimiert.

### Ausgabebereich

Mit diesem Parameter stellen Sie den Ausgabebereich der gewählten Ausgabeart ein.

### Diagnose Drahtbruch (im Strommodus)

Wenn Sie diesen Parameter freigeben, dann wird bei einem Drahtbruch die Diagnose **Leitungsbruch** generiert. Im Nullbereich ist diese Diagnose nicht erkennbar.

### Diagnose Kurzschluss (im Spannungsmodus)

Wenn Sie diesen Parameter freigeben, dann wird bei einem Kurzschluss der Ausgangsleitung eine Diagnose generiert. Im Nullbereich ist diese Diagnose nicht erkennbar.

## Diagnose Überlast

Wenn Sie diesen Parameter freigeben, dann wird bei einer Überlast die Diagnose generiert.

## Ersatzwerte

Bei diesem Parameter geben Sie einen Ersatzwert ein, den die Baugruppe bei CPU-STOP ausgeben soll. Der Ersatzwert muss im Nenn-, Übersteuerungs- oder Untersteuerungsbereich liegen.

## ET 200SP

### Dezentrales Peripheriesystem ET 200SP

#### Definition

Das Dezentrale Peripheriesystem ET 200SP ist ein skalierbares und hochflexibles, dezentrales Peripheriesystem zur Anbindung der Prozesssignale an eine Zentralsteuerung über einen Feldbus.

#### Einsatzgebiet

Das ET 200SP ist ein multifunktionales dezentrales Peripheriesystem für unterschiedliche Einsatzbereiche. Durch den skalierbaren Aufbau können Sie den Ausbau exakt auf den jeweiligen Bedarf vor Ort ausrichten.

Das ET 200SP ist für die Schutzart IP 20 zugelassen und für den Einbau in einem Schaltschrank vorgesehen.

#### Aufbau

Das ET 200SP wird auf eine Profilschiene montiert und setzt sich zusammen aus:

- einem Interfacemodul, das mit allen IO-Controllern kommuniziert, die sich nach der PROFINET-Norm IEC 61158 verhalten,
- bis zu 32 Peripheriemodulen, die in beliebiger Kombination auf passive BaseUnits gesteckt werden,
- einem Servermodul, das den Aufbau des ET 200SP abschließt.

## Parameter Interfacemodul

### Statusbytes

### Statusbytes

Wenn Sie die Option "Statusbytes" aktivieren, werden 4 Bytes Eingangsdaten reserviert für den Status der Versorgungsspannung jedes Peripheriemoduls.

|        | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |  |
|--------|----|----|----|----|----|----|----|----|--|
| Byte 0 | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | Steckplätze der Peripheriemodule                                 |
| Byte 1 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9  | Bit = 0: Lastspannung fehlt oder Peripheriemodul nicht vorhanden |
| Byte 2 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | Bit = 1: Lastspannung und Peripheriemodul vorhanden              |
| Byte 3 | 32 | 31 | 30 | 29 | 28 | 27 | 26 | 25 |  |

---

#### Hinweis

Ein gestecktes oder fehlendes Servermodul meldet immer für den Steckplatz Bit = 0.

---

## Sammeldiagnose fehlende Versorgungsspannung L+

### Sammeldiagnose fehlende Versorgungsspannung L+

Diese Diagnose ist eine Sammeldiagnose über den Versorgungsspannungsstatus aller Peripheriemodule einer Potenzialgruppe, die durch BaseUnits mit Spannungseinspeisung (helle BaseUnit BU...D) festgelegt sind.

Die Sammeldiagnose wird aus den Zuständen der Versorgungsspannung der gesteckten Peripheriemodule innerhalb der Potenzialgruppe gebildet.

Die Sammeldiagnose ist unabhängig von der Freigabe des Parameters "Fehlende Versorgungsspannung L+" der Peripheriemodule.

Das Servermodul beeinflusst nicht die Sammeldiagnose fehlende Versorgungsspannung L+.

### Voraussetzungen für die korrekte Funktionsweise der Sammeldiagnose fehlende Versorgungsspannung L+:

- Auf den hellen und dunklen BaseUnits müssen Peripheriemodule bzw. BU Cover stecken. Wenn auf einer hellen BaseUnit kein Peripheriemodul steckt, wird der Beginn dieser Potenzialgruppe vom Interfacemodul nicht erkannt; die Peripheriemodule dieser Potenzialgruppe gehören somit zur vorherigen Potenzialgruppe. Ein Sammelfehler fehlende Versorgungsspannung L+ wird dann der falschen Potenzialgruppe zugeordnet. Wenn ein Peripheriemodul auf die helle BaseUnit gesteckt wird, erkennt das Interfacemodul die neue Potenzialgruppe, bewertet den Status neu und meldet im Fehlerfall eine neue Sammeldiagnose.
- Das Servermodul muss gesteckt sein.  
Das Servermodul selbst beeinflusst aber nicht die Sammeldiagnose fehlende Versorgungsspannung L+.

### Konfigurationssteuerung

#### Funktionsprinzip

Durch die Konfigurationssteuerung betreiben Sie mit einer einzigen Projektierung des Dezentralen Peripheriegeräts ET 200SP unterschiedliche reale Konfigurationen (Optionen).

Durch die Konfigurationssteuerung haben Sie die Möglichkeit, das Dezentrale Peripheriegerät ET 200SP mit seinem Vollausbau zu projektieren und trotzdem mit fehlenden Modulen zu betreiben. Wenn fehlende Module später nachgerüstet werden, ist keine neue Projektierung erforderlich und damit auch kein erneutes Laden der Hardware-Konfiguration.

Mit Hilfe von Steuerdatensätzen, die im Anwenderprogramm an das Interfacemodul übermittelt werden, legen Sie eine aktuelle Sollkonfiguration fest.

- Auf einem Steckplatz ist das projektierte Modul nicht vorhanden.
  - Statt des projektierten Peripheriemoduls darf ein BU Cover auf diesem Steckplatz gesteckt sein. Da das projektierte Modul auf dem Steckplatz fehlt, wird auch von einer "Konfigurationssteuerung mit Leerplätzen" gesprochen.
  - Statt des projektierten Moduls darf auf diesem Steckplatz das Modul stecken, das rechts neben dem fehlenden Modul projektiert ist. Die Istkonfiguration erscheint durch das fehlende Modul zusammengeschoben. Da das projektierte Modul fehlt, aber keine Lücke in der Konfiguration entsteht, spricht man auch von einer "Konfigurationssteuerung ohne Leerplätze".
- Die Konfiguration wird durch ein bereits projektiertes Modul erweitert.
  - Bei der Konfigurationssteuerung mit Leerplätzen erweitern Sie die Konfiguration, indem Sie das projektierte Modul auf den entsprechenden Leerplatz stecken.
  - Bei der Konfigurationssteuerung ohne Leerplätze stecken Sie das projektierte Modul rechts neben dem letzten Modul der ET 200SP.

### Weiterführende Information und Beispiele

Informationen zu den Regeln zur Konfigurationssteuerung, zum Aufbau des Steuerdatensatzes, zum Verhalten im Betrieb und Beispiele für den Aufbau des

Steuerdatensatzes für verschiedene Konfigurationen finden Sie im Gerätehandbuch zum Interfacemodul IM 155-6 PN (<http://support.automation.siemens.com/WW/view/de/59768173>).

## Parameter Ausgabemodule

### Ersatzwertverhalten

#### Ersatzwertverhalten

Im ET 200SP wird das Ersatzwertverhalten durch den IO-Controller je Steckplatz durchgeführt.

Der jeweilige Ausgang verhält sich nach seinem parametrisierten Ersatzwertverhalten:

- "Abschalten"
- "Ersatzwert ausgeben"
- "Letzten Wert halten"

Das Ersatzwertverhalten wird in folgenden Fällen ausgelöst:

- IO-Controller in STOP
- Controller-Ausfall (Verbindungsunterbrechung)
- Firmware-Update
- Zurücksetzen auf Werkseinstellungen
- Ziehen von mehr als einem Peripheriemodul gleichzeitig
- Deaktivieren des IO-Devices
- Stationsstopp
  - fehlendes Servermodul
  - Ziehen von mehr als einem Peripheriemodul gleichzeitig
  - mindestens ein Peripheriemodul steckt auf einem falschen BaseUnit

---

#### **Hinweis**

##### **Verkleinern einer Konfiguration**

Wenn Sie die Konfiguration des ET 200SP verkleinern und die Projektierung in die CPU laden, dann behalten die nicht mehr projektierten, aber noch vorhandenen Module ihr ursprüngliches Ersatzwertverhalten bei. Dies gilt bis zum Abschalten der Versorgungsspannung am BaseUnit BU...D oder am Interfacemodul.

---

### Parameter Eingabemodule

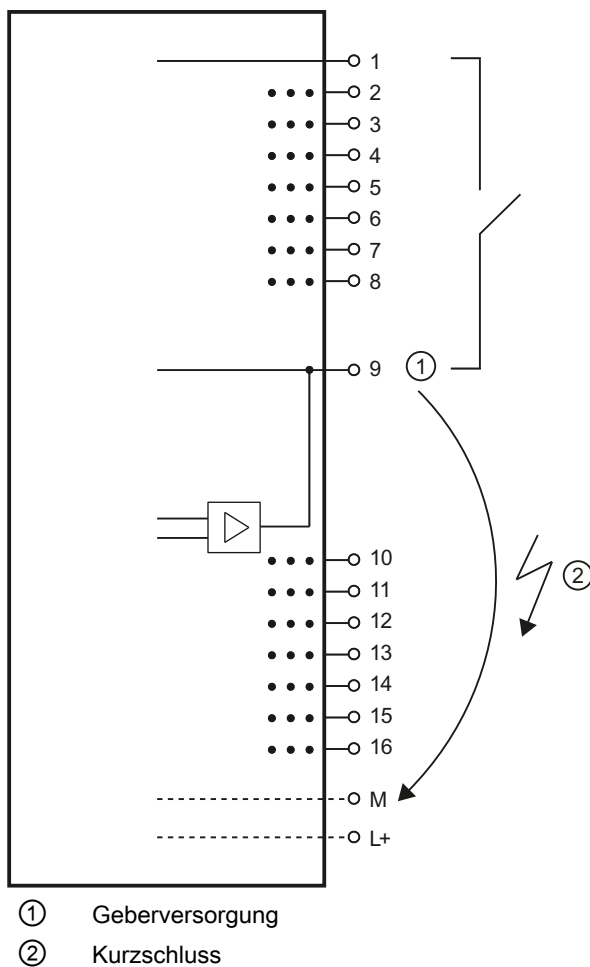
### Parameter der Digitaleingabemodule

### Diagnose fehlende Versorgungsspannung L+

Freigabe der Diagnosemeldung bei fehlender oder zu geringer Versorgungsspannung L+.

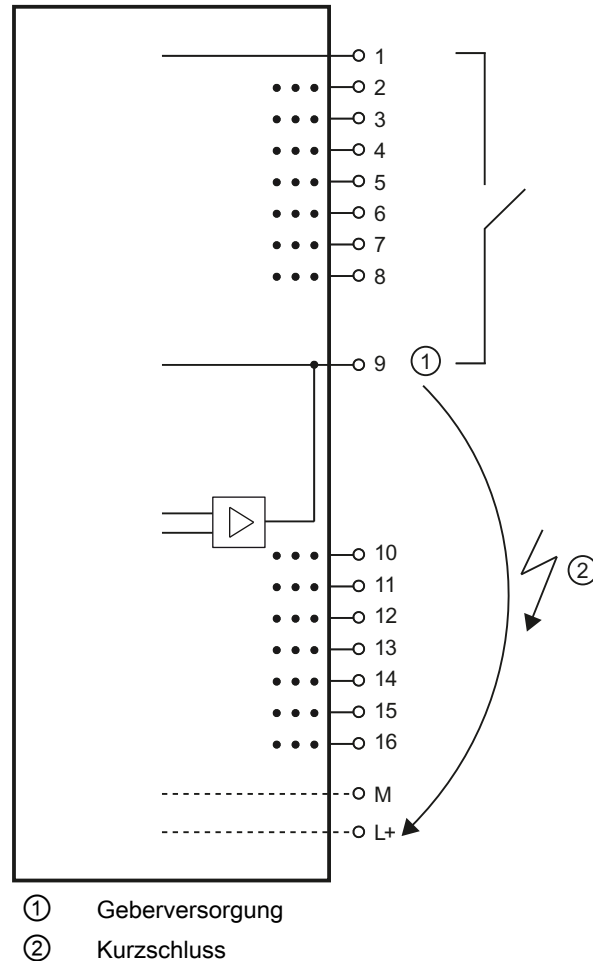
### Diagnose Kurzschluss nach M

Freigabe der Diagnose, wenn ein Kurzschluss der Geberversorgung nach M auftritt.



### Diagnose Kurzschluss nach L+

Freigabe der Diagnose, wenn ein Kurzschluss der Geberversorgung nach L+ auftritt.



### Diagnose Drahtbruch

Freigabe der Diagnose, wenn die Leitung zum Geber unterbrochen ist.

### Betriebsart

Legt fest, ob ein Kanal aktiviert oder deaktiviert ist.

### Impulsverlängerung (nur High Feature Module)

Die Impulsverlängerung ist eine Funktion zur Veränderung eines digitalen Eingangssignals. Ein Impuls an einem Digitaleingang wird mindestens auf die parametrisierte Länge verlängert. Ist der Eingangsimpuls bereits länger als die parametrisierte Länge, dann wird der Impuls nicht verändert.

Die Impulsverlängerung wird immer dann gestartet, wenn sich der Zustand des Eingangssignals ändert und keine Impulsverlängerung für diesen Kanal aktiv ist.

### Potentialgruppe des linken Moduls/Neue Potentialgruppe

Legt fest, ob sich das Peripheriemodul auf einem BaseUnit mit Einspeisung der Versorgungsspannung befindet (Neue Potentialgruppe) oder ob es sich auf einem BaseUnit ohne Einspeisung der Versorgungsspannung befindet (es gehört dann zur Potentialgruppe des linken Moduls).

### Parameter der Analogeingabemodule

#### Fehlende Versorgungsspannung L+

Freigabe der Diagnose, bei fehlender oder zu geringer Versorgungsspannung L+.

#### Vergleichsstelle (AI 4xRTD/TC 2-/3-/4-wire HF)

Als Vergleichsstelle kann ein BaseUnit mit internen Temperatursensor (BU..T) bzw. der Kanal 0 des Peripheriemoduls verwendet werden, wenn dieser als "Thermowiderstand Pt100 Klimabereich Celsius" parametrierung wurde.

Nachfolgend ist eine mögliche Parametrierung dargestellt:

Tabelle 8-55 RTD-Kanal

| Einstellung                | Beschreibung   |
|----------------------------|--|
| Kein Referenzkanalbetrieb  | Anliegender Temperaturwert am Kanal 0 kann bei entsprechender Parametrierung der anderen Kanäle als modulweiter Referenzwert verwendet werden. |
| Referenzkanal der Gruppe 0 | Der Kanal wirkt als Sender für die Vergleichsstellentemperatur von Gruppe 0. Die Verteilung erfolgt über das Interfacemodul.                   |

Tabelle 8-56 TC-Kanal

| Einstellung                | Beschreibung   |
|----------------------------|--|
| Referenzkanal des Moduls   | Der entsprechende TC-Kanal nutzt den Kanal 0 desselben Moduls als Vergleichsstellentemperatur. Dieser muss als "Thermowiderstand Pt 100 Klimabereich Celsius" und "Kein Referenzkanalbetrieb" parametrierung sein, sonst wird Diagnose Vergleichsstelle ausgelöst. |
| Interne Vergleichsstelle   | Die Vergleichsstellentemperatur wird von einem internen Temperatursensor auf der BaseUnit gelesen. Bei falschem BaseUnit-Typ wird Diagnose Vergleichsstelle ausgelöst.   |
| Referenzkanal der Gruppe 0 | Bei der Einstellung "TC" (Thermoelement...) wirkt der Kanal als Empfänger für die Vergleichsstellentemperatur von Gruppe 0   |
| Feste Referenztemperatur   | Es erfolgt keine Temperaturkompensation. Die Linearisierung wird mit einer angenommenen Vergleichsstellentemperatur von 0 °C durchgeführt.   |

### Überlauf

Freigabe der Diagnose, wenn der Messwert den Überlaufbereich überschreitet.



## Unterlauf

Freigabe der Diagnose, wenn der Messwert den Unterlaufbereich unterschreitet.

## Drahtbruch

Freigabe der Diagnose, wenn das Modul am entsprechend parametrieren Eingang keinen Stromfluss bzw. zu geringen Strom für die Messung hat.

## Glättung

Die einzelnen Messwerte werden mittels digitaler Filterung geglättet. Die Glättung ist in 4 Stufen einstellbar, wobei der Glättungsfaktor  $k$  multipliziert mit der Zykluszeit des Peripheriemoduls der Zeitkonstante des Glättungsfilters entspricht. Je größer die Glättung, umso größer ist die Zeitkonstante des Filters.

Nachfolgendes Bild zeigt die Sprungantwort bei den verschiedenen Glättungsfaktoren in Abhängigkeit von der Anzahl der Baugruppenzyklen.

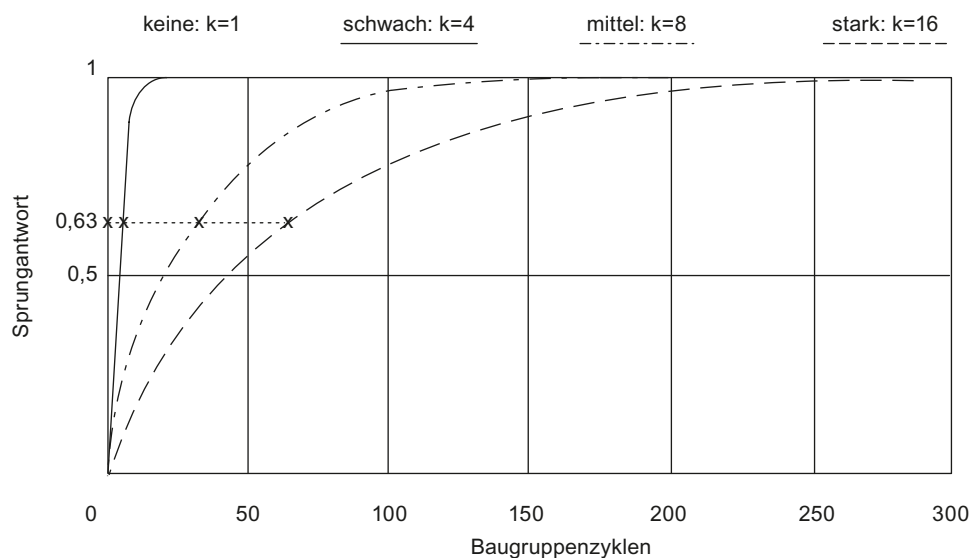


Bild 8-5 Glättung bei AI 4×RTD/TC 2-/3-/4-wire HF

## Störfrequenzunterdrückung

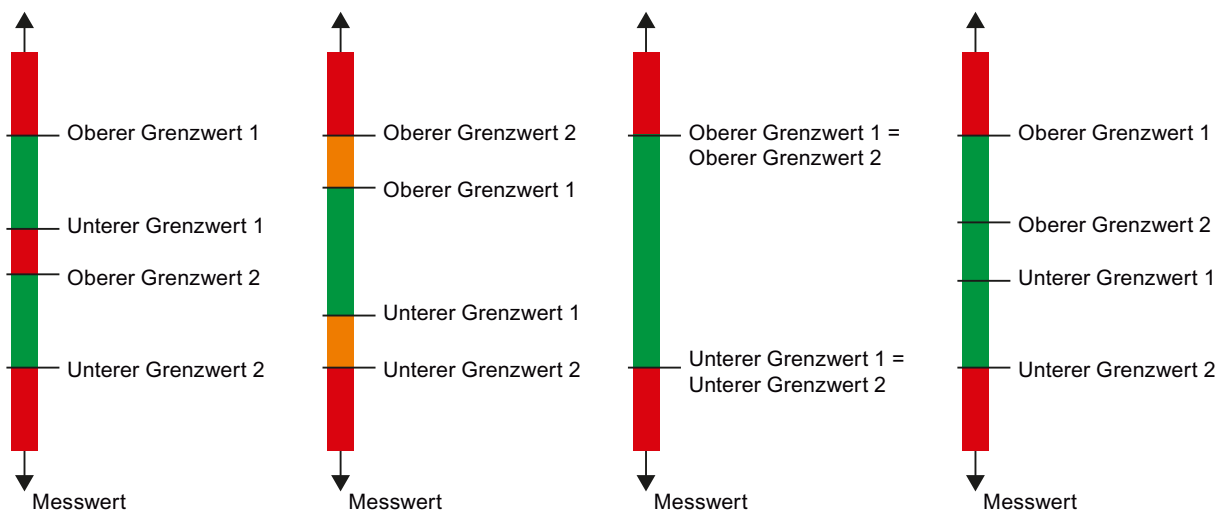
Unterdrückt bei Analogeingabebaugruppen die Störungen, die durch die Frequenz des verwendeten Wechselspannungsnetzes hervorgerufen werden.

Die Frequenz des Wechselspannungsnetzes kann sich besonders bei der Messung in kleinen Spannungsbereichen und bei Thermoelementen störend auf den Messwert auswirken. Mit diesem Parameter gibt der Anwender die Netzfrequenz an, die in seiner Anlage vorherrscht.

## Grenzwerte Prozessalarm

Wenn die obere Grenze  $1/2$  überschritten oder die untere Grenze  $1/2$  unterschritten wird, löst das Modul einen Prozessalarm aus.

Nachfolgend einige Beispiele für die Wahl der Grenzwerte 1 und 2.



### Unterer Grenzwert 1/2

Legen Sie eine Schwelle fest, bei deren Unterschreitung ein Prozessalarm ausgelöst wird.

### Oberer Grenzwert 1/2

Legen Sie eine Schwelle fest, bei deren Überschreitung ein Prozessalarm ausgelöst wird.

### Potentialgruppe des linken Moduls/Neue Potentialgruppe

Legt fest, ob sich das Peripheriemodul auf einem BaseUnit mit Einspeisung der Versorgungsspannung befindet (Neue Potentialgruppe) oder ob es sich auf einem BaseUnit ohne Einspeisung der Versorgungsspannung befindet (es gehört dann zur Potentialgruppe des linken Moduls).

### Temperaturkoeffizient (Messart Thermowiderstand)

Der Korrekturfaktor für den Temperaturkoeffizienten ( $\alpha$ -Wert) gibt an, um wieviel sich der Widerstand eines bestimmten Materials relativ ändert, wenn sich die Temperatur um 1 °C erhöht.

Der Temperaturkoeffizient ist abhängig von der chemischen Zusammensetzung des Materials. In Europa wird pro Sensorart nur ein Wert verwendet (voreingestellter Wert).

Die weiteren Werte ermöglichen eine sensorspezifische Einstellung des Temperaturkoeffizienten und somit eine noch höhere Genauigkeit.

### Siehe auch

Besonderheiten bei AI 4xRTD/TC 2-/3-/4-wire HF (Seite 941)

## Besonderheiten bei AI 4xRTD/TC 2-/3-/4-wire HF

### Einsatz von Cu10-Sensoren

- Wählen Sie in der Parametrierung "Thermowiderstand 3-Leiter" und "Cu10".
- Verdrahten Sie den Cu10-Sensor in 3-Leiter-Anschlusstechnik.
- Während des Betriebes findet eine automatische, interne Kompensation des Leitungswiderstandes der fehlenden Messleitung statt.

#### Hinweis

Zur Gewährleistung einer optimalen Leitungskompensation bei Cu10 beachten Sie bitte Folgendes:

- Die Summe aus Kabelwiderstand und Messwiderstand darf 31  $\Omega$  nicht überschreiten.
- Das Kabel darf maximal einen Widerstand von 8  $\Omega$  aufweisen, wenn Sie den Temperaturbereich bis über 312 °C nutzen möchten.  
Beispiel: Ein 200 m langes Cu-Kabel mit 0,5 mm<sup>2</sup> Adernquerschnitt hat etwa 7  $\Omega$ . Ein geringerer Querschnitt verkürzt die zulässige Kabellänge entsprechend.

### Einsatz von PTC-Widerständen

PTCs eignen sich für die Temperaturüberwachung bzw. als thermische Schutzeinrichtung von komplexen Antrieben oder Transformatorwicklungen.

- Wählen Sie in der Parametrierung "Widerstand 2-Leiter" und "PTC".
- Schließen Sie den PTC in 2-Leiter-Anschlusstechnik an.
- Verwenden Sie PTC-Widerstände vom Typ A (Kaltleiter) nach DIN/VDE 0660, Teil 302.
- Wenn die Diagnose "Über-/Unterlauf" freigegeben ist, wird bei Widerstandswerten < 18  $\Omega$  eine Diagnose "unterer Grenzwert unterschritten" erzeugt, die einen Kurzschluss anzeigt.
- Sensordaten zum PTC-Widerstand:

Tabelle 8-57 Einsatz von PTC-Widerständen

| Eigenschaft                   | Technische Daten  | Bemerkung  |
|-------------------------------|---|--|
| Schaltpunkte                  | <b>Verhalten bei steigender Temperatur</b>                                |  |
|                               | < 550 $\Omega$  | <b>Normalbereich:</b><br>• SIMATIC S7: Bit 0 = "0", Bit 2 = "0" (im PAE)   |
|                               | 550 $\Omega$ bis 1650 $\Omega$  | <b>Vorwarnbereich:</b><br>• SIMATIC S7: Bit 0 = "0", Bit 2 = "1" (im PAE)  |
|                               | > 1650 $\Omega$   | <b>Ansprechbereich:</b><br>• SIMATIC S7: Bit 0 = "1", Bit 2 = "0" (im PAE) |
|                               | <b>Verhalten bei fallender Temperatur</b>                                 |  |
|                               | > 750 $\Omega$  | <b>Ansprechbereich:</b><br>• SIMATIC S7: Bit 0 = "1", Bit 2 = "0" (im PAE) |
| 750 $\Omega$ bis 540 $\Omega$ | <b>Vorwarnbereich:</b><br>• SIMATIC S7: Bit 0 = "0", Bit 2 = "1" (im PAE) |  |

| Eigenschaft  | Technische Daten                                      | Bemerkung  |
|--|---|--|
|  | < 540 Ω   | <b>Normalbereich:</b><br>• SIMATIC S7: Bit 0 = "0", Bit 2 = "0" (im PAE) |
| (TNF-5) °C<br>(TNF+5) °C<br>(TNF+15) °C<br>Messspannung<br>Spannung am PTC | max. 550 Ω<br>min. 1330 Ω<br>min. 4000 Ω<br>max. 7,5V | TNF= Nennansprechtemperatur  |

- Belegung im Prozessabbild der Eingänge (PAE) bei SIMATIC S7

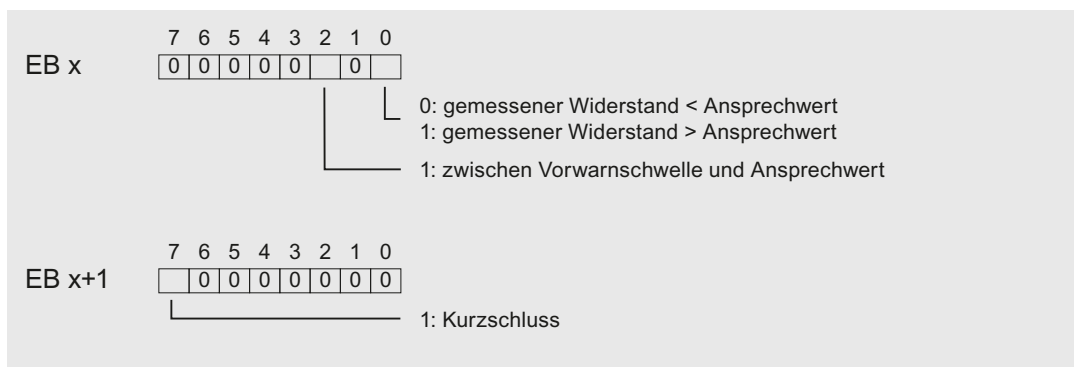


Bild 8-6 Belegung im Prozessabbild der Eingänge (PAE)

- Hinweise zur Programmierung

**Hinweis**

Im Prozessabbild der Eingänge sind lediglich die Bits 0+2 für die Auswertung relevant. Über die Bits 0+2 können Sie die Temperatur z. B. eines Motors überwachen.

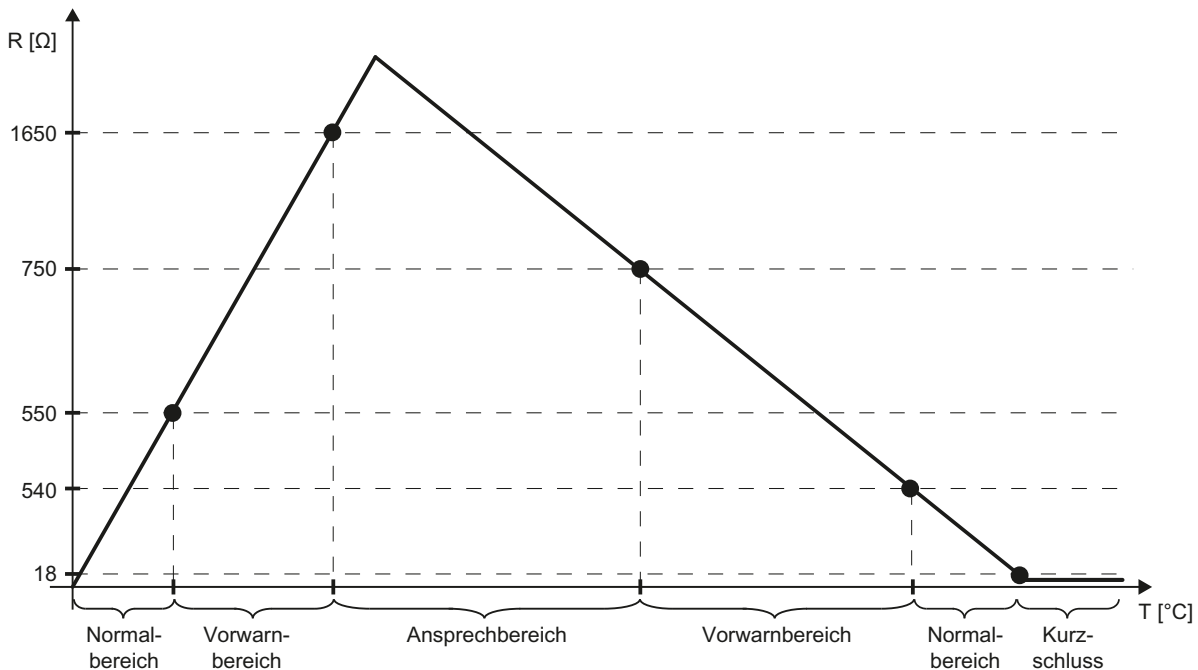
Die Bits 0+2 im Prozessabbild der Eingänge haben kein speicherndes Verhalten. Berücksichtigen Sie bei der Parametrierung, dass z. B. ein Motor kontrolliert (über eine Quittierung) anläuft.

Die Bits 0+2 können niemals gleichzeitig gesetzt sein, sondern werden nacheinander gesetzt.

Werten Sie aus Sicherheitsgründen immer die Diagnoseeinträge des AI 4×RTD/TC 2-/3-/4-wire HF aus, da bei gezogenen Peripheriemodulen, ausgefallener Versorgungsspannung des Peripheriemoduls, Drahtbruch oder Kurzschluss der Messleitungen keine Messung möglich ist.

## Beispiel

Das unten stehende Diagramm zeigt den Temperaturverlauf und die dazugehörigen Schaltpunkte.



## Siehe auch

Parameter der Analogeingabemodule (Seite 938)

## ET 200MP

### Dezentrales Peripheriesystem ET 200MP

## Definition

Das Dezentrale Peripheriesystem ET 200MP ist ein skalierbares und flexibles, dezentrales Peripheriesystem zur Anbindung der Prozesssignale an eine Zentralsteuerung über einen Feldbus.

## Einsatzgebiet

Das ET 200MP ist ein multifunktionales dezentrales Peripheriesystem für unterschiedliche Einsatzbereiche. Durch den skalierbaren Aufbau können Sie den Ausbau exakt auf den jeweiligen Bedarf vor Ort ausrichten.

Das ET 200MP ist für die Schutzart IP 20 zugelassen und für den Einbau in einem Schaltschrank vorgesehen.

## Aufbau

Das ET 200MP wird auf eine Profilschiene montiert und setzt sich zusammen aus:

- Einem Interfacemodul, das mit allen IO-Controllern kommuniziert, die sich nach der PROFINET-Norm IEC 61158 verhalten,
- Bis zu 30 Module (Stromversorgungsmodule und Peripheriemodule aus dem Peripheriespektrum der S7-1500) rechts neben dem Interfacemodul steckbar.
- Wenn Sie ein Stromversorgungsmodul links vom Interfacemodul einsetzen, ergibt sich ein möglicher Maximalausbau von insgesamt 32 Modulen.
- die Anzahl der steckbarer Peripheriemodule durch deren Strombedarf eingeschränkt sein.

## Steckplatzregeln

- Steckplatz 0: Stromversorgungsmodul (optional)
- Steckplatz 1: Interfacemodul
- Steckplatz 2 bis 31: Peripheriemodule bzw. Stromversorgungsmodule

## Parameter Interfacemodul

### Versorgungsspannung L+ angeschlossen

#### Parameter "Versorgungsspannung L+ angeschlossen"

Dieser Parameter hat Einfluss auf die Diagnose und auf die Prüfung der Leistungsbilanz.

- Diagnose des ET 200MP:  
Wenn die Ist-Konfiguration nicht der Soll-Konfiguration hinsichtlich der Versorgungsspannung des Interfacemoduls entspricht, erzeugt das Interfacemodul eine Diagnosemeldung. Beispiel: Sie haben die Option "Versorgungsspannung L+ angeschlossen" deaktiviert, im tatsächlichen Aufbau aber die Versorgungsspannung angeschlossen.
- Prüfung der Leistungsbilanz bei der Projektierung:  
Entsprechend der Einstellung des Parameters verändert sich die Leistungsbilanz:  
Entweder speist das Interfacemodul Leistung in den Rückwandbus ein oder es entnimmt Leistung aus dem Rückwandbus.

Die Voreinstellung (Option "Versorgungsspannung L+ angeschlossen" ist **aktiviert**) bedeutet, dass das Interfacemodul frontseitig mit DC 24 V versorgt wird und Leistung in den Rückwandbus einspeist.

Wenn die Option "Versorgungsspannung L+ angeschlossen" **deaktiviert** ist, dann darf das Interfacemodul nicht frontseitig mit DC 24 V versorgt werden.

In diesem Fall muss links neben dem Interfacemodul eine Systemstromversorgung (PS) stecken, das das Interfacemodul und die Module rechts neben dem Interfacemodul versorgt.

---

#### **Hinweis**

Wir empfehlen, das Interfacemodul immer frontseitig mit DC 24 V zu versorgen. Wenn zusätzlich eine Systemstromversorgung (PS) **vor** bzw. links neben dem Interfacemodul gesteckt und angeschlossen wird, dann steht dem Aufbau sowohl die Leistung der Systemstromversorgung (PS) als auch die Leistung der integrierten Stromversorgung des Interfacemoduls zur Verfügung.

Sie brauchen dann die Voreinstellung des Parameters nicht verändern.

---

### **Parameter Eingabemodule**

#### **Parameter der Analogeingabemodule**

#### **Fehlende Versorgungsspannung L+**

Freigabe der Diagnose bei fehlender oder zu geringer Versorgungsspannung L+.

#### **Drahtbruch**

Freigabe der Diagnose, wenn das Modul am entsprechend parametrisierten Eingang keinen Stromfluss bzw. zu geringen Strom für die Messung hat oder eine zu niedrige Spannung anliegt.

#### **Stromgrenze für Diagnose Drahtbruch**

Schwellwert bei dem Drahtbruch gemeldet wird. Je nach verwendetem Sensor kann der Wert auf 1,185 mA oder 3,6 mA eingestellt werden.

#### **Überlauf**

Freigabe der Diagnose, wenn der Messwert den Übersteuerungsbereich überschreitet.

#### **Unterlauf**

Freigabe der Diagnose, wenn der Messwert den Untersteuerungsbereich unterschreitet.

#### **Gleichtaktfehler**

Freigabe der Diagnose, wenn die zulässige Common-Mode-Spannung überschritten wird.

### Referenzkanalfehler (nur bei AI 8xU/I/RTD/TC ST)

- Freigabe der Diagnose, wenn ein Fehler am Kanal für die Temperaturkompensation vorliegt, z. B. Drahtbruch.
- Kompensationsart Dynamische Referenztemperatur ist parametrierbar und an das Modul wurde noch keine Referenztemperatur übertragen.

### Temperaturkoeffizient

Der Temperaturkoeffizient ist abhängig von der chemischen Zusammensetzung des Materials. In Europa wird pro Sensorart nur ein Wert verwendet (voreingestellter Wert).

Der Korrekturfaktor für den Temperaturkoeffizienten ( $\alpha$ -Wert) gibt an, um wie viel sich der Widerstand eines bestimmten Materials relativ ändert, wenn sich die Temperatur um 1 °C erhöht.

Die weiteren Werte ermöglichen eine sensorspezifische Einstellung des Temperaturkoeffizienten und somit eine noch höhere Genauigkeit.

### Störfrequenzunterdrückung

Unterdrückt bei Analogeingabemodulen die Störungen, die durch die Frequenz des verwendeten Wechselspannungsnetzes hervorgerufen werden.

Die Frequenz des Wechselspannungsnetzes kann sich besonders bei der Messung in kleinen Spannungsbereichen und bei Thermoelementen störend auf den Messwert auswirken. Mit diesem Parameter geben Sie die Netzfrequenz an, die in Ihrer Anlage vorherrscht.

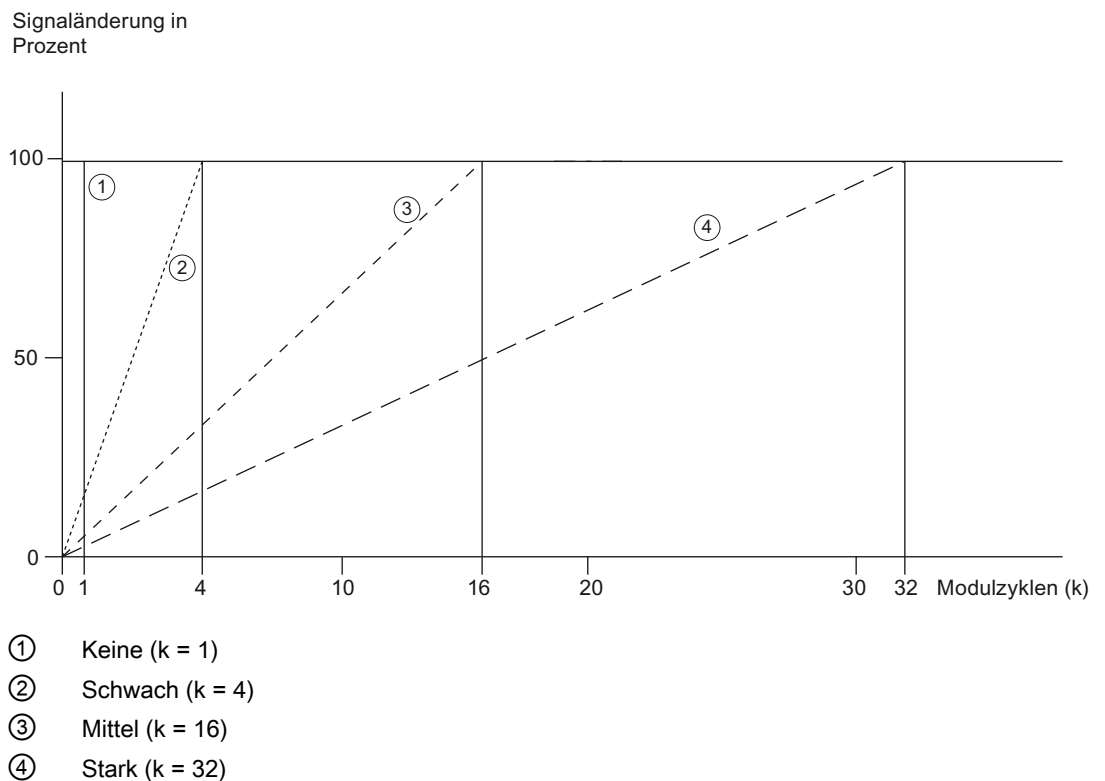
### Glättung

Die einzelnen Messwerte werden mittels Filterung geglättet. Die Glättung ist bei den Analogeingabemodulen AI 8xU/I/RTD/TC ST und AI 8xU/I HS in 4 Stufen einstellbar.

Glättungszeit = Anzahl der Modulzyklen (k) x Zykluszeit des Moduls.

Das folgende Bild zeigt, nach wie vielen Modulzyklen der geglättete Analogwert zu annähernd 100 % anliegt in Abhängigkeit der eingestellten Glättung. Gilt für jeden Signalwechsel am Analogeingang.





### Vergleichsstelle (nur bei AI 8xU//RTD/TC ST)

Für den Parameter Vergleichsstelle können folgende Einstellungen parametrierbar werden:

Tabelle 8-58 Mögliche Parametrierungen für den Parameter Vergleichsstelle

| Einstellung                   | Beschreibung   |
|-------------------------------|--|
| Feste Referenztemperatur      | Die Temperatur der Vergleichsstelle wird parametrierbar und als fester Wert im Modul hinterlegt.   |
| Dynamische Referenztemperatur | Die Temperatur der Vergleichsstelle wird im Anwenderprogramm mit der Anweisung WRREC (SFB 53) über die Datensätze 192 bis 199 von der CPU an das Modul übertragen. |
| Interne Vergleichsstelle      | Die Temperatur der Vergleichsstelle wird mit einem Sensor ermittelt, der im Modul integriert ist.  |
| Referenzkanal des Moduls      | Die Temperatur der Vergleichsstelle wird mit einem externen Widerstandsthermometer (RTD) am Referenzkanal (COMP) des Moduls ermittelt.                             |

#### Hinweis

##### Feste Referenztemperatur

Bei Parametrierung eines Thermoelements Typ B ist nur die Einstellung "Feste Referenztemperatur" mit einer Temperatur 0 °C möglich.

### Freigabe Prozessalarm 1 bzw. 2

Freigabe eines Prozessalarms, wenn die obere Grenze 1 bzw. 2 überschritten oder die untere Grenze 1 bzw. 2 unterschritten wird.

### Untere Grenze 1 bzw. 2

Legen Sie die Schwelle fest, bei deren Unterschreitung der Prozessalarm 1 bzw. 2 ausgelöst wird.

### Obere Grenze 1 bzw. 2

Legen Sie eine Schwelle fest, bei deren Überschreitung der Prozessalarm 1 bzw. 2 ausgelöst wird.

## Temperaturkompensation bei Thermoelementen

### Einleitung

Sie haben verschiedene Möglichkeiten, die Vergleichsstellentemperatur zu erfassen, um aus der Temperaturdifferenz zwischen Vergleichsstelle und Messstelle einen absoluten Temperaturwert zu erhalten.

Je nachdem wo (örtlich) Sie die Vergleichsstelle benötigen, können Sie die verschiedenen Kompensationsmöglichkeiten nutzen.

---

#### Hinweis

Bei Parametrierung eines Thermoelements Typ B ist nur die Einstellung "Feste Referenztemperatur" mit einer Temperatur 0 °C möglich.

---

## Möglichkeiten zur Kompensation der Vergleichsstellentemperatur

| Kompensationsmöglichkeiten    | Erläuterung  | Anwendungsfall  |
|-------------------------------|--|---|
| Interne Vergleichsstelle      | <p>Bei dieser Kompensation wird die Temperatur der Vergleichsstelle mit einem Sensor ermittelt, der im Modul integriert ist.</p> <p><b>Vorgehensweise</b></p> <p>Schließen Sie das Thermoelement direkt bzw. mit Ausgleichsleitungen an das Peripheriemodul an.</p>  | <ul style="list-style-type: none"> <li>• Sie verwenden für den Anschluss Ausgleichsleitungen, die dem Material des Thermoelements entsprechen.</li> <li>• Wenn in Ihrer Anlage die Temperatur der Vergleichsstelle und die Temperatur des Moduls identisch sind, dann dürfen Sie auch Leitungen aus unterschiedlichem Material verwenden.</li> </ul>  |
| Referenzkanal des Moduls      | <p>Die Temperatur der Vergleichsstelle wird mit einem externen Widerstandsthermometer (RTD) ermittelt.</p> <p><b>Vorgehensweise</b></p> <p>Verbinden Sie das Thermoelement direkt bzw. mit Ausgleichsleitungen an der Vergleichsstelle mit den Zuleitungen. Die Zuleitungen schließen Sie an die entsprechenden Klemmen des Moduls an.</p> <p>Schließen Sie das Widerstandsthermometer (RTD) am Referenzkanal des Moduls an. Das Widerstandsthermometer (RTD) muss im Bereich der Vergleichsstelle platziert werden.</p> | <ul style="list-style-type: none"> <li>• Sie möchten die Temperatur direkt an der Vergleichsstelle erfassen.</li> <li>• Die gemessenen Temperaturen aller Kanäle, die Sie für diese Kompensationsart projektiert haben, werden um den Temperaturwert der Vergleichsstelle automatisch korrigiert.</li> <li>• Von der Vergleichsstelle bis zum Modul können Sie preiswerte z. B. Kupferleitungen verwenden.</li> </ul> |
| Dynamische Referenztemperatur | <p>Die Temperatur der Vergleichsstelle wird über ein Modul ermittelt. Diesen Temperaturwert übergeben Sie im Anwenderprogramm über einen Datensatz an weitere Module.</p> <p><b>Vorgehensweise</b></p> <p>Schließen Sie das Widerstandsthermometer (RTD) für die Vergleichsstelle an einem beliebigen Kanal an.</p> <p>Die Temperatur der Vergleichsstelle wird mit der Anweisung WRREC durch den Datensatz 192 bis 199 von der CPU an das Modul übermittelt.</p>  | <ul style="list-style-type: none"> <li>• Sie setzen mehrere Module an der Vergleichsstelle ein und können deshalb alle Kanäle über einen gemeinsamen Temperaturwert kompensieren.</li> <li>• Für das Erfassen des Temperaturwerts benötigen Sie nur ein Widerstandsthermometer (RTD).</li> <li>• Von der Vergleichsstelle bis zum Modul können Sie preiswerte Leitungen, z. B. Kupferleitungen verwenden.</li> </ul>  |
| Feste Referenztemperatur      | <p>Die Temperatur der Vergleichsstelle wird als fester Wert im parametriert.</p> <p><b>Vorgehensweise</b></p> <p>Verbinden Sie das Thermoelement direkt bzw. mit Ausgleichsleitungen an der Vergleichsstelle mit den Zuleitungen. Die Zuleitungen schließen Sie an die entsprechenden Klemmen des Moduls an.</p> <p>Geben Sie bei der Projektierung des Moduls einen festen Temperaturwert für die Vergleichsstelle vor (z. B. 20°C).</p>  | <ul style="list-style-type: none"> <li>• Sie halten die Temperatur der Vergleichsstelle auf einer konstanten Temperatur und kennen den Temperaturwert.</li> <li>• Von der Vergleichsstelle bis zum Modul können Sie preiswerte Leitungen, z. B. Kupferleitungen verwenden.</li> </ul>   |

## Parameter Ausgabemodule

### Parameter der Analogausgabemodule

#### Fehlende Versorgungsspannung L+

Freigabe der Diagnose, bei fehlender oder zu geringer Versorgungsspannung L+.

#### Kurzschluss nach M

Freigabe der Diagnose, wenn ein Kurzschluss der Aktorversorgung nach M auftritt.

#### Drahtbruch

Freigabe der Diagnose, wenn die Leitung zum Geber unterbrochen ist.

#### Überlauf

Freigabe der Diagnose, wenn der Messwert den Überlaufbereich überschreitet.

#### Unterlauf

Freigabe der Diagnose, wenn der Messwert den Unterlaufbereich unterschreitet.

#### Verhalten bei CPU-STOP

Bestimmt das Verhalten des Ausgangs, wenn die CPU in den Betriebszustand STOP geht.

#### Ersatzwert

Die Ersatzwerte sind Werte, die die Ausgänge (der Ausgang) im Fall eines CPU-STOPS ausgeben.

## ET 200M

### ET 200M konfigurieren

#### Einführung

Für die Familie ET 200M finden Sie im Hardware-Katalog unter "Dezentrale Peripherie" ein breites Modulspektrum.

## Konfigurieren und Parametrieren

Informationen zum Konfigurieren und Parametrieren finden Sie in den folgenden Kapiteln.

## Aufbau ET 200M

### Definition

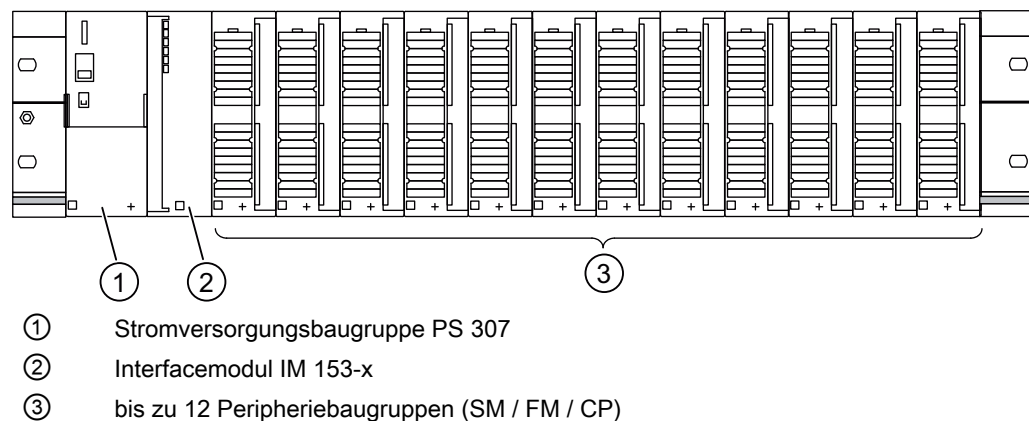
Das Dezentrale Peripheriegerät ET 200M ist ein modulares Peripheriegerät in der Schutzart IP 20.

ET 200M hat die Aufbautechnik des Automatisierungssystems S7-300 und besteht aus IM 153-x und Peripheriebaugruppen der S7-300.

ET 200M kann kommunizieren mit:

- allen DP-Mastern, die sich nach Norm IEC 61784-1:2002 Ed1 CP 3/1 verhalten
- allen IO-Controllern, die sich nach der Norm IEC 61158 verhalten

### Aufbau von ET 200M (Beispiel)



## Projektieren der Funktion 'Baugruppenwechsel im Betrieb'

### Einleitung

Die ET 200M unterstützt die Funktion "Baugruppenwechsel im Betrieb" und den damit verbundenen Ziehen/Stecken-Alarm.

Die Funktion "Baugruppenwechsel im Betrieb" ermöglicht es Ihnen, Module während des Betriebes aus dem Baugruppenträger der ET 200M zu ziehen oder zu stecken.

## Voraussetzung

Sie haben ein Interfacemodul projektiert, das den Baugruppenwechsel im Betrieb unterstützt. (ab IM 153-1, Bestellnr. 153-1AA02-0XB0).

Außerdem muss die projektierte CPU die Funktion ebenfalls unterstützen, z. B. für PROFIBUS eine S7-400 mit DP-Schnittstelle.

Für den Hardware-Aufbau müssen Sie den aktiven Rückwandbus (Busschiene mit Steckplätzen) verwenden. Die herkömmliche Profilschiene mit Busverbindern zwischen den Baugruppen unterstützt diese Funktion nicht.

## Projektierung

Wenn die Projektierungsvoraussetzungen erfüllt sind, wird im Inspektorfenster, Bereich "Baugruppenparameter" der Parameter "Baugruppenwechsel im Betrieb" angeboten, den Sie aktivieren können. Für die projektierten Baugruppen wird unterhalb dieses Parameters eine Tabelle angezeigt, die benötigte aktive Busmodule für den Hardware-Aufbau anzeigt.

Für eine PROFIBUS-Konfiguration wird außerdem die Option "Anlauf bei Sollausbau ungleich Istausbau" angezeigt. Diese Option wird automatisch aktiviert, wenn "Baugruppenwechsel im Betrieb" aktiviert ist.

## Projektieren von HART-Variablen

### Einleitung

Zahlreiche HART-Feldgeräte stellen zusätzliche Messgrößen (z. B. Sensortemperatur) zur Verfügung. Diese können ausgelesen werden, wenn sie in der Feldgeräte-Projektierung entsprechend eingestellt sind. Über die HART-Variablen ist es möglich, die eingestellten Messwerte direkt vom Feldgerät in den E/A-Bereich ihres Automatisierungssystems zu übernehmen.

Bei den HART-Baugruppen können unabhängig von der Anzahl der projektierten Kanäle max. 8 HART-Variablen parametrisiert werden, pro Kanal max. 4 HART-Variablen. Die HART-Variablen ordnen Sie in den Eigenschaften der Baugruppe einem Kanal zu (Bereich "HART-Variableneinstellung").

### Voraussetzung

Die HART-Baugruppe steckt in einer ET 200M (ab IM 153-2, 6ES7 153-2BA02-0AB0).

### Adressbelegung

Die HART-Baugruppe belegt 16 Ein-/Ausgangsbyte. Wenn Sie HART-Variablen projektieren, belegt die Baugruppe für jede HART-Variable zusätzlich 5 Byte.

Wenn Sie alle 8 HART-Variablen nutzen belegt die HART-Eingabebaugruppe insgesamt 56 Ein-/Ausgangsbytes (16 Byte + 8 x 5 Byte = 56 Byte.).

Die Projektierung "keine" belegt keine zusätzlichen Eingangsbyte.

## Projektierung von HART-Variablen

Für einen Kanal können Sie bis zu 4 HART-Variablen projektieren

- PV (Primary Variable)
- SV (Secondary Variable)
- TV (Tertiary Variable)
- QV (Quaternary)

CiR ist ein Platzhalter und reserviert den Adressraum für eine HART-Variable. HART-Variablen die Sie nicht nutzen, müssen Sie mit dem Parameter "keine" projektieren.

## Aufbau der HART-Variablen

Die HART-Variablen sind wie folgt aufgebaut:

|                   |           |
|-------------------|-----------|
| 4 Byte HART-Daten | 1 Byte QC |
|-------------------|-----------|

## Aufbau des Bytes "Quality-Code"

Der Quality-Code (QC) kann folgende Werte annehmen:

| Quality-Code (QC) | Bedeutung   |
|-------------------|---|
| 0x4C oder 0       | Initialisierung: 0 Wert von IM und 4C von Baugruppe |
| 0x18              | Kommunikationsabbruch / keine Kommunikation         |
| 0x0C              | Fehler im HART-Gerät                                |
| 0x47              | HART-Gerät ist beschäftigt                          |
| 0x84              | OK "Konfiguration geändert"                         |
| 0x80              | OK  |

## Siehe auch

Dokumentation zu HART-Analogbaugruppen (<http://support.automation.siemens.com/WWW/view/de/22063748>)

## Signalbaugruppen für Prozessautomatisierung

### Grundlagen

### Einführung

Signalbaugruppen für die Prozessautomatisierung sind Baugruppen der S7-300 wie z. B. die SM 321; DI 16xNAMUR oder SM 322; DO 16xDC24V/0,5A.

Sie werden in einem DP-Slave (IM 153-2) betrieben.

Die bieten gegenüber Standardbaugruppen folgende zusätzliche technologische Funktionen wie z. B. Impulsverlängerung und Flatterüberwachung.

### Siehe auch

Wechsler (Seite 954)

Technologische Parameter (Seite 955)

### Wechsler

#### Gebertyp "Wechsler"

Wenn die Digitaleingänge einer Kanalgruppe als "Wechsler" parametrier sind, führt die Baugruppe für diese Kanalgruppe eine Diagnose auf den Gebertyp Wechsler durch.

### Wechsler

Ein Wechsler ist ein Hilfsschalter mit nur einem beweglichen Schaltstück, der je eine Schließstellung bei geschlossenem und geöffnetem Schaltgerät hat.

Beachten Sie dabei folgende Vorschrift:

- Schließkontakt immer am "geraden" Kanal anschließen,
- Öffnerkontakt immer am "ungeraden" Kanal anschließen.

Die tolerierte Umschaltzeit zwischen den beiden Kanälen ist fest 300 ms lang.

Ist die Überprüfung negativ, dann

- kennzeichnet die Baugruppe den Wertstatus des Schließkanals als "ungültig"
- erzeugt die Baugruppe einen Diagnoseeintrag für den Schließkanal
- löst einen Diagnosealarm aus (sofern Diagnosealarm freigegeben wurde)

Nur für den Schließkanal werden das Digitaleingangssignal und der Wertstatus aktualisiert. Für den Öffnerkanal ist das Digitaleingangssignal fest auf "Null" und der Wertstatus "ungültig", da dieser Kanal nur zur Überprüfung des Sensors dient.

Die Diagnose ist abhängig vom Parameter "Auswahl" (des Gebers). Beachten Sie auch die Besonderheiten bei der Diagnose mit dem Gebertyp Wechsler im Handbuch "Signalbaugruppen für die Prozessautomatisierung".

### Siehe auch

Dokumentation zu Baugruppen für die Prozessautomatisierung (<http://support.automation.siemens.com/WW/view/de/7215812>)



## Technologische Parameter

### Impulsverlängerung und Flatterüberwachung

Die Impulsverlängerung ist eine Funktion zur Veränderung eines digitalen Eingangssignals. Ein Impuls an einem Digitaleingang wird mindestens auf die parametrisierte Länge verlängert. Ist der Eingangsimpuls bereits länger als die parametrisierte Länge, dann wird er nicht verändert.

Falls die Impulse verlängert werden sollen, klicken Sie in das Feld, um die Zeit auszuwählen. Wenn die Impulse nicht verlängert werden sollen, wählen Sie den Eintrag "---".

Die Flatterüberwachung ist eine leitentechnische Funktion für digitale Eingangssignale. Sie erkennt und meldet prozesstechnisch ungewöhnliche Signalverläufe, z. B. ein zu häufiges Schwanken des Eingangssignals zwischen "0" und "1".

Voraussetzung für das Aktivieren der Flatterüberwachung ist, dass die Sammeldiagnose für diesen Eingang ebenfalls freigegeben ist.

### Überwachungsfenster und Anzahl Signalwechsel

Die Funktionsweise der Flatterüberwachung wird mit Hilfe der beiden Parameter Überwachungsfenster und Anzahl Signalwechsel festgelegt.

Mit dem ersten Signalwechsel des Eingangssignals wird die Zeit gestartet, die als Überwachungsfenster parametrisiert wurde. Ändert sich das Eingangssignal innerhalb dieser Zeit öfter als die parametrisierte Anzahl Signalwechsel, so wird das als Flatterfehler erkannt. Wird innerhalb des Überwachungsfensters kein Flatterfehler erkannt, dann wird beim nächsten Signalwechsel das Überwachungsfenster erneut gestartet.

---

#### Hinweis

Wenn Sie für einen Eingangskanal eine Impulsverlängerung parametrisieren, hat das auch Auswirkungen auf die für diesen Kanal freigegebene Flatterüberwachung. Das "impulsverlängerte" Signal ist das Eingangssignal für die Flatterüberwachung. Stimmen Sie deshalb die Parametrisierungen von Impulsverlängerung und Flatterüberwachung aufeinander ab.

---

### Siehe auch

Dokumentation zu Baugruppen für die Prozessautomatisierung (<http://support.automation.siemens.com/WW/view/de/7215812>)

## IQ-Sense Modul

### Eigenschaften von 8 IQ-SENSE

#### Eigenschaften

Das 8 IQ-SENSE Modul zeichnet sich durch folgende Eigenschaften aus:

- Anschluss von Sensoren mit IQ-SENSE®, Photoelektrische Näherungsschalter: z. B. Reflexionslichtschranken, Reflexionslicht- und Lasertaster.
- Zentral in einer S7-300 oder dezentral in einer ET 200M einsetzbar.
- An jedes Modul können bis zu 8 Sensoren angeschlossen werden. Je Sensor ist eine Zweidrahtleitung erforderlich.
- Funktionsreserve parametrierbar.
- Parametrierbare Zeitfunktionen, Schalthysterese, Synchronbetrieb
- Vorgabe von Empfindlichkeits- und Abstandswerten (*IntelliTeach* über FB "IQ-SENSE Opto")
- Teach in
- Ziehen und Stecken der Sensoren im laufenden Betrieb (Automatische Nachparametrierung)

#### Antiinterferenzgruppe

Nur für optische IQ-Sense-Geräte (IQ-Profil-ID 1).

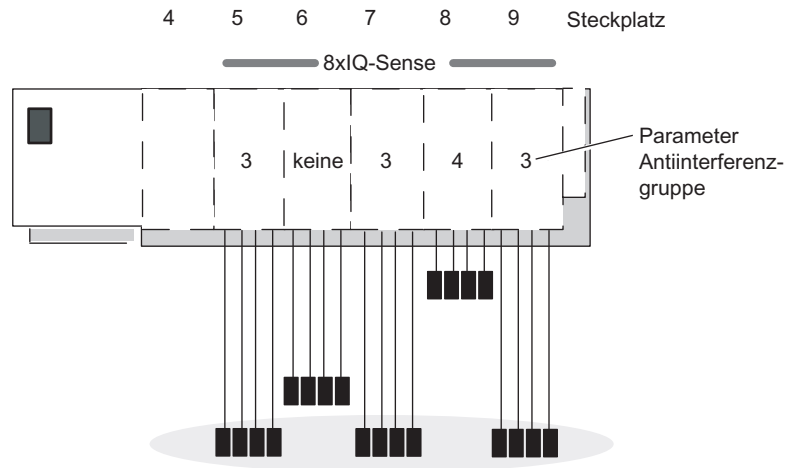
Für IQ-Sense-Geräte mit IQ-Profil-ID 128 (Ultraschall) siehe bei den kanalgranularen Parametern "Multiplex-/Synchronbetrieb".

Verhinderung einer Störbeeinflussung (Interferenz, z. B. Streulicht) durch Zuweisen einer Antiinterferenzgruppe. Das bedeutet:

- Antiinterferenzgruppe: Keine (= Voreinstellung)  
Optische Sensoren an einer oder mehreren Baugruppen können sich bei ungünstiger Anordnung gegenseitig beeinflussen.
- Antiinterferenzgruppe: 3 oder 4  
Optische Sensoren an der selben Baugruppe mit Antiinterferenzgruppe 3 oder 4 können sich gegenseitig nicht beeinflussen. Optische Sensoren an unterschiedlichen Baugruppe mit Antiinterferenzgruppe 3 oder 4 können sich ebenfalls gegenseitig nicht beeinflussen. Sie müssen keinen Mindestabstand zwischen den optischen IQ-Sense-Geräten einhalten und können z. B. zwei Reflexionsschranken auf einen gemeinsamen Reflektor ausrichten.

## Funktionsprinzip

Nachfolgendes Bild verdeutlicht die Funktionsweise des Parameters Antiinterferenzgruppe:



Eine gegenseitige Störbeeinflussung ist nur zwischen den optischen Sensoren der Baugruppen am Steckplatz 5, 6, 7 und 9 möglich, da sich diese in der gleichen Antiinterferenzgruppe 3 befinden bzw. "Keine" eingestellt ist.

### Hinweis

Sensoren in gleichen Antiinterferenzgruppen müssen so montiert werden, dass der Mindestabstand (siehe Beipackzettel Sensor) eingehalten wird und somit eine gegenseitige Beeinflussung ausgeschlossen ist.


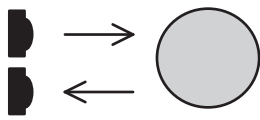
## Geberart

Mit diesem Parameter stellen Sie die Sensorart je Kanal ein:

- Reflexionslichtschranke oder
- Reflexionslichttaster oder
- deaktiviert

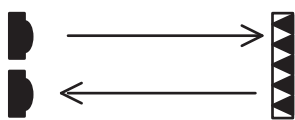
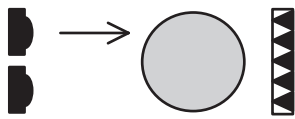
### Reflexionslichttaster

Tabelle 8-59 Reflexionslichttaster

| Reflexionslichttaster | Objekt  |  |
|-----------------------|---|--|
| Sender<br>Empfänger   |  | Schaltzustand 0: Kein Objekt erkannt, d. h. der Gegenstand befindet sich nicht im Strahlengang. Der Empfänger sieht kein Licht |
| Sender<br>Empfänger   |  | Schaltzustand 1: Objekt erkannt, d. h. der Gegenstand befindet sich im Strahlengang. Der Empfänger sieht kein Licht.           |

### Reflexionslichtschranke

Tabelle 8-60 Reflexionslichtschranke

| Reflexionslichtschranke | Objekt  |  |
|-------------------------|---|--|
| Sender<br>Empfänger     |   | Schaltzustand 0: Kein Objekt erkannt, d. h. der Gegenstand befindet sich nicht im Strahlengang. Der Empfänger sieht Licht. |
| Sender<br>Empfänger     |  | Schaltzustand 1: Objekt erkannt, d. h. der Gegenstand befindet sich im Strahlengang. Der Empfänger sieht kein Licht.       |

### Schalthyserese

Durch Störungen am Reflexionslichttaster oder im Fertigungsprozess kann ein sogenanntes "Signalflattern" auftreten. Dabei pendelt der Messwert um die Schaltschwelle von 100 % (Objekt erkannt – Objekt nicht erkannt). Dieses Flattern der Schaltschwelle können Sie durch den Parameter Schalthyserese verhindern. Dadurch erhalten Sie ein stabiles Ausgangssignal am Sensor.

Sie können die Bereiche 5 / 10 / 20 / 50 % als Schalthyserese parametrieren.

### Voraussetzungen

Der Parameter Schalthyserese ist nur einstellbar bei Reflexionslichttastern mit Hintergrundausblendung.

## Funktionsprinzip

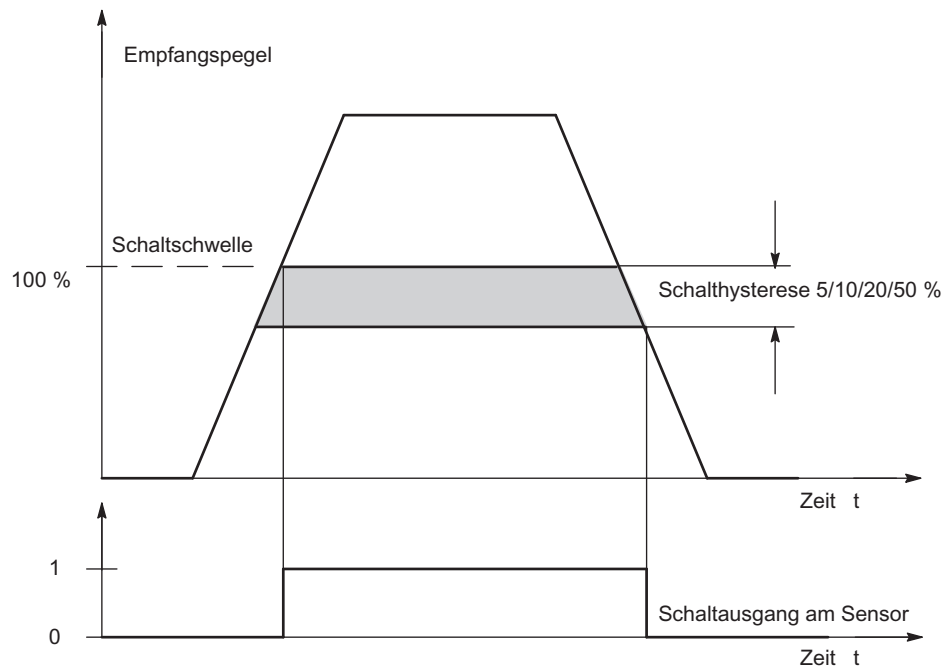


Bild 8-7 Parameter Schalthysterese

## Zeitfunktionen, Zeitwert

Mit diesen Parametern können Sie das Elektronikmodul an Ihre jeweilige Applikation anpassen.

**Funktionsprinzip**

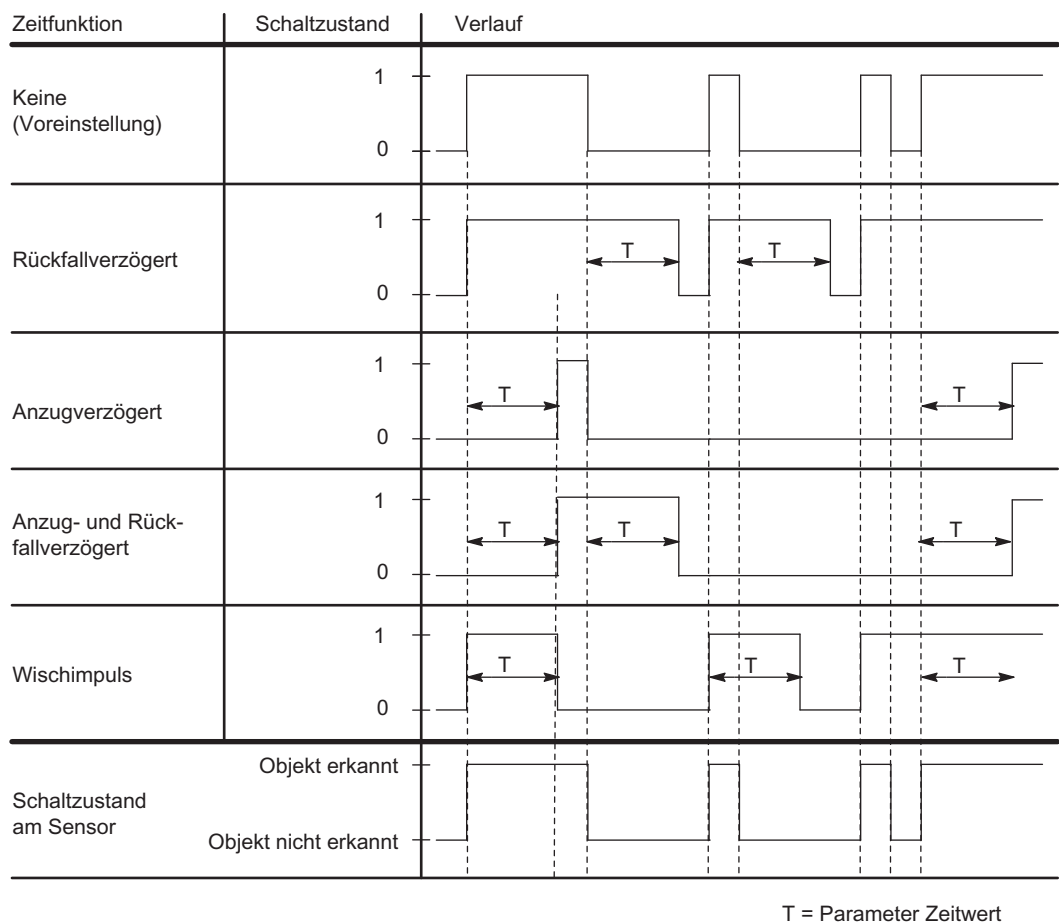


Bild 8-8 Parameter Zeitwert, Zeitfunktionen

**Multiplex-/Synchronbetrieb**

Zur Vermeidung gegenseitiger Beeinflussung räumlich benachbarter IQ-Sense-Ultraschallgeräte (Geräte mit IQ-Profil-ID 128) verwenden Sie den Parameter "Multiplex-/Synchronbetrieb".

**Einstellungen für den Parameter Multiplex-/Synchronbetrieb**

**Deaktiviert:** Eine gegenseitige Beeinflussung räumlich benachbarter IQ-Sense-Ultraschallsensoren ist möglich (Voreinstellung). Die Zykluszeit wird durch den IQ-Sense-Ultraschallsensor bestimmt.

**Multiplex:** Die IQ-Sense-Ultraschallsensoren ermitteln nacheinander den Prozesswert (Abstand), sodass eine gegenseitige Beeinflussung ausgeschlossen ist. Die Zykluszeit ist hierbei die Summe der parametrisierten Synchronzykluszeiten der zu multiplexenden IQ-Sense-Ultraschallsensoren.

**Synchronisation:** Die IQ-Sense-Ultraschallsensoren ermitteln zum exakt gleichen Zeitpunkt

den Prozesswert (Abstand), sodass eine gegenseitige Beeinflussung ausgeschlossen ist. Die Zykluszeit entspricht hierbei der größten parametrisierten Synchronzykluszeit der zu synchronisierenden IQ-Sense-Ultraschallsensoren.

Den Synchronbetrieb können Sie z. B. für eine Vorhangfunktion verwenden, bei der sich mehrere parallel angeordnete IQ-Sense-Ultraschallsensoren einen ausgedehnten Erfassungsbereich teilen. Die Sensoren senden gleichzeitig einen Ultraschallimpuls aus. Wenn ein Objekt in den Erfassungsbereich eintritt, so empfängt derjenige Sensor das Echo am schnellsten, der dem Objekt am nächsten liegt. So lässt sich das Objekt nicht nur detektieren, sondern auch lokalisieren.

## AFI-Wert

Mit Hilfe des AFI-Wertes (Application Family Identifier, definiert in der internationalen Norm ISO 15693-3) können Transponder für unterschiedliche Applikationen ausgewählt werden. Es werden nur Transponder bearbeitet, deren AFI-Wert mit dem am Sensor eingestellten Wert übereinstimmt. Hat ein Transponder den AFI-Wert "0", so kann er unabhängig vom AFI-Wert des Sensors identifiziert und bearbeitet werden.

Dieser Parameter hat nur Bedeutung, wenn er vom Identifikationssystem unterstützt wird, ansonsten kann er einen beliebigen Wert (im Normalfall "0") erhalten.

## Transpondertyp

Je nach Typ des Transponders müssen Sie einstellen, ob es sich um einen ISO-Transponder oder um einen herstellerspezifischen Typ handelt.

Bei Transpondern gemäß internationalem Standard nach ISO 15693 ist der Wert "1" auszuwählen, bei allen anderen Typen wird "0" eingestellt. Aufgrund dieser Einstellung wird im Sensor einer von zwei möglichen Luftschnittstellentreibern ausgewählt.

Dieser Parameter hat nur Bedeutung, wenn er vom Identifikationssystem unterstützt wird, ansonsten kann er einen beliebigen Wert (im Normalfall "0") erhalten.

## ET 200S

### ET 200S konfigurieren

#### Einführung

Für die Familie ET 200S finden Sie im Hardware-Katalog unter "Dezentrale Peripherie" ein breites Modulspektrum.

#### Parametrierung

Informationen zum Konfigurieren und Parametrieren finden Sie unter "Siehe auch".

## Frequenzumrichter

### Einsatz des Frequenzumrichters

#### Frequenzumrichter

Die Frequenzumrichter ICU24 und ICU24F (als Fail-safe-Ausführung) sind modular aufgebaute Frequenzumrichter, die vollständig in das dezentrale Peripheriesystem ET 200S eingebettet sind. Im Folgenden erfahren Sie, was Sie bei der Parametrierung beider Baugruppen beachten müssen.

#### Telegramm

Die Telegrammnummer und Betriebsart der Baugruppe werden nur angezeigt und sind nicht änderbar.

#### Anwendungs-ID

Mit der Anwendungs-ID kennzeichnen Sie die Gesamtheit der im Frequenzumrichter gespeicherten Parameter. Tragen Sie eine Anwendungs-ID aus dem Wertebereich 0 bis 65535 ein. Beim Anlauf (bzw. Ziehen/Stecken) wird diese Kennung mit der auf dem Umrichter gespeicherten Anwendungs-ID verglichen.

Umrichter, die identische Applikationen bearbeiten, sind normalerweise auch identisch parametrierung und sollten mit derselben Anwendungs-ID gekennzeichnet werden. Umrichter mit derselben Anwendungs-ID dürfen untereinander getauscht werden. Ebenso wird das Kopieren der kompletten Parametrierung von einem Umrichter auf einen anderen, beispielsweise über eine MMC, nur dann akzeptiert, wenn beide dieselbe Anwendungs-ID haben.

Umrichter, die unterschiedliche Anwendungen bearbeiten und unterschiedlich parametrierung sind, müssen durch verschiedene Anwendungs-IDs gekennzeichnet werden. Damit wird verhindert, dass ein Umrichter mit unpassender Parametrierung auf einem falschen Steckplatz, d. h. an der falschen Anwendung, anläuft. Ebenso wird verhindert, dass die im Umrichter gespeicherte Parametrierung zufälligerweise mit einer beliebigen Parametrierung, die auf einer MMC gespeichert ist, überschrieben werden kann.

#### Diagnosealarm freigeben

Sie können für die Frequenzumrichter den Diagnosealarm freigeben. Bei freigegebenem Diagnosealarm muss in der CPU ein OB 82 vorhanden sein, der die Diagnoseereignisse verarbeitet.

#### Siehe auch

Dokumentation zum Frequenzumrichter (<http://support.automation.siemens.com/WW/view/de/26291825>)



## ET 200pro

### Einsatz des Frequenzumrichters

#### Frequenzumrichter

Die Frequenzumrichter ET 200pro FC und ET 200pro F-FC (als Fail-safe-Ausführung) sind modular aufgebaute Frequenzumrichter, die vollständig in das dezentrale Peripheriesystem ET 200pro eingebettet sind. Im Folgenden erfahren Sie, was Sie bei der Parametrierung beider Baugruppen beachten müssen.

#### Telegramm

Die Telegrammnummer und Betriebsart der Baugruppe werden nur angezeigt und sind nicht änderbar.

#### Anwendungs-ID

Mit der Anwendungs-ID kennzeichnen Sie die Gesamtheit der im Frequenzumrichter gespeicherten Parameter. Tragen Sie eine Anwendungs-ID aus dem Wertebereich 0 bis 65535 ein. Beim Anlauf (bzw. Ziehen/Stecken) wird diese Kennung mit der auf dem Umrichter gespeicherten Anwendungs-ID verglichen.

Umrichter, die identische Applikationen bearbeiten, sind normalerweise auch identisch parametrierung und sollten mit derselben Anwendungs-ID gekennzeichnet werden. Umrichter mit derselben Anwendungs-ID dürfen untereinander getauscht werden. Ebenso wird das Kopieren der kompletten Parametrierung von einem Umrichter auf einen anderen, beispielsweise über eine MMC, nur dann akzeptiert, wenn beide dieselbe Anwendungs-ID haben.

Umrichter, die unterschiedliche Anwendungen bearbeiten und unterschiedlich parametrierung sind, müssen durch verschiedene Anwendungs-IDs gekennzeichnet werden. Damit wird verhindert, dass ein Umrichter mit unpassender Parametrierung auf einem falschen Steckplatz, d. h. an der falschen Anwendung, anläuft. Ebenso wird verhindert, dass die im Umrichter gespeicherte Parametrierung zufälligerweise mit einer beliebigen Parametrierung, die auf einer MMC gespeichert ist, überschrieben werden kann.

#### Diagnosealarm freigeben

Sie können für die Frequenzumrichter den Diagnosealarm freigeben. Bei freigegebenem Diagnosealarm muss in der CPU ein OB 82 vorhanden sein, der die Diagnoseereignisse verarbeitet.

## 8.2 Geräte und Netze diagnostizieren

### 8.2.1 Hardware diagnostizieren

#### 8.2.1.1 Überblick über die Hardware-Diagnose

#### Prinzipielle Möglichkeiten, Hardware zu diagnostizieren

#### Prinzipielle Möglichkeiten der Hardware-Diagnose

Sie können die Hardware wie folgt diagnostizieren:

- Über die Online- und Diagnosesicht
- Über die Task Card "Online-Tools"
- Über den Bereich "Diagnose > Geräte-Information" des Inspektorfensters
- Über Diagnosesymbole z. B. in der Gerätesicht und in der Projektnavigation

#### Aufbau der Online- und Diagnosesicht

Die Online- und Diagnosesicht besteht aus zwei nebeneinander liegenden Fenstern:

- Das linke Fenster zeigt eine Baumstruktur mit Ordnern und - falls Sie die Ordner aufgeklappt haben - Gruppen.
- Das rechte Fenster enthält Detailinformationen zum selektierten Ordner bzw. zur selektierten Gruppe.

Es gibt die Gruppe "Online-Zugänge" und die Ordner "Diagnose" und "Funktionen":

- Gruppe "Online-Zugänge": Sie enthält die Anzeige, ob momentan eine Online-Verbindung zum zugehörigen Ziel besteht oder nicht. Sie können darüber hinaus die Online-Verbindung herstellen bzw. trennen.
- "Diagnose": enthält mehrere Diagnosegruppen zur selektierten Baugruppe.
- "Funktionen": enthält mehrere Gruppen, in denen Sie Einstellungen auf der selektierten Baugruppe vornehmen bzw. Kommandos an die Baugruppe ausgeben.

#### Funktion und Aufbau der Task Card "Online-Tools"

In der Task Card "Online-Tools" können Sie bei Baugruppen mit einem eigenen Betriebszustand (z. B. bei CPUs) aktuelle Diagnoseinformationen auslesen und Kommandos an die Baugruppe übertragen.

Falls Sie vor der Aktivierung der Task Card "Online-Tools" eine Baugruppe ohne eigenen Betriebszustand oder mehrere Baugruppen selektiert haben, bezieht sich die Task Card auf die zugehörige CPU.

Die Task Card "Online-Tools" besteht aus folgenden Paletten:

- CPU-Bedienpanel
- Zykluszeit
- Speicher

---

**Hinweis**

Eine Palette ist nur dann mit Inhalt gefüllt, wenn die Baugruppe die zugehörigen Funktionen beherrscht sowie eine Online-Verbindung besteht.

Falls keine Online-Verbindung zur zugehörigen Baugruppe besteht, erfolgt in jeder Palette die Anzeige "Keine Online-Verbindung" in blauer Schrift. Wenn eine bestehende Online-Verbindung getrennt wurde, wird "Dieses Ziel ist nicht erreichbar" angezeigt.

---

### Aufbau des Registers "Diagnose" des Inspektorfensters

Das Register "Diagnose" des Inspektorfensters besteht selbst wiederum aus mehreren Registern. Davon ist das folgende für die Hardware-Diagnose relevant:

- Geräte-Information  
Dieses Register bezieht sich auf alle CPUs des Projekts, zu denen eine Online-Verbindung besteht. Hier werden Meldungen ausgegeben, falls eine oder mehrere CPUs gestört sind oder sich nicht im Betriebszustand RUN befinden.

### Siehe auch

Grundlagen zu den Task Cards (Seite 194)

Inspektorfenster (Seite 192)

### Unter den online verbundenen Geräten diejenigen ermitteln, die gestört sind

### Übersicht über die gestörten Geräte

Im Bereich "Diagnose > Geräte-Information" des Inspektorfensters erhalten Sie eine Übersicht über die gestörten Geräte, zu denen eine Online-Verbindung besteht oder bestanden hat.

Der Bereich "Diagnose > Geräte-Information" des Inspektorfensters besteht aus folgenden Elementen:

- Kopfzeile mit der Anzahl der gestörten Geräte
- Tabelle mit Detailinformationen zu jedem gestörten Gerät

Wenn Sie den Aufbau der Online-Verbindung zu einem Gerät angestoßen haben, gilt es als gestört, wenn es nicht erreichbar ist oder mindestens einen Fehler meldet oder nicht im Betriebszustand "RUN" ist.

### **Aufbau der Tabelle mit Detailinformationen zu den gestörten Geräten**

Die Tabelle besteht aus den folgenden Spalten:

- Online-Status: enthält den Online-Status als Diagnosesymbol und in Worten
- Betriebszustand: enthält den Betriebszustand als Symbol und in Worten
- Gerät / Baugruppe: Name des betroffenen Geräts bzw. der betroffenen Baugruppe
- Meldung: erläutert den Eintrag der vorhergehenden Spalte
- Details: Der Link öffnet die zum Gerät gehörende Online- und Diagnosesicht bzw. legt sie in den Vordergrund. Falls keine Online-Verbindung mehr besteht, öffnet der Link den Dialog zum Aufbau der Verbindung.
- Hilfe: Der Link liefert weitere Informationen zur aufgetretenen Störung.

### **Siehe auch**

Diagnose- und Vergleichsstatus mittels Symbolen anzeigen (Seite 966)

### **Diagnose- und Vergleichsstatus mittels Symbolen anzeigen**

### **Diagnosestatus online ermitteln und mittels Symbolen anzeigen**

Beim Aufbau der Online-Verbindung zu einem Gerät wird auch sein Diagnosestatus und ggf. der seiner unterlagerten Komponenten und ggf. sein Betriebszustand ermittelt.

Im Folgenden wird beschrieben, in welcher Sicht welche Symbole angezeigt werden.

- Gerätesicht
  - Bei jeder Hardware-Komponente (mit Ausnahme des Signalboards auf der CPU) wird das zugehörige Diagnosesymbol angezeigt.
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Bei Hardware-Komponenten mit eigenem Betriebszustand wird zusätzlich das Betriebszustandssymbol links oder oberhalb vom Diagnosesymbol angezeigt.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es werden keine Diagnosesymbole angezeigt (wegen der Projektierung als GSDML-Gerät).
- Geräteübersicht
  - Bei jeder Hardware-Komponente wird das zugehörige Diagnosesymbol angezeigt.
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Für die der CPU zugeordneten Module wird das zugehörige Diagnosesymbol angezeigt (Die nicht zugeordneten Module erhalten kein Diagnosesymbol.). Für die steckbaren Submodule eines zugeordneten Moduls wird das zugehörige Diagnosesymbol angezeigt (Die nicht steckbaren Submodule sind nicht sichtbar und erhalten folglich auch kein Diagnosesymbol.).
- Netzsicht
  - Bei jedem Gerät wird das zugehörige Diagnosesymbol angezeigt.
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es wird ein Diagnosesymbol angezeigt. Es gehört zu demjenigen Teil der Station, der der CPU zugeordnet ist.

- Netzübersicht
  - Bei jeder Hardware-Komponente wird das zugehörige Diagnosesymbol angezeigt.
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es wird ein Diagnosesymbol angezeigt. Es gehört zu demjenigen Teil der Station, der der CPU zugeordnet ist.
- Topologiesicht
  - Bei jedem Gerät wird das zugehörige Diagnosesymbol angezeigt.
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Für jeden Port wird das zugehörige Diagnosesymbol angezeigt. Die Bedeutung der einzelnen Farben ist weiter unten beschrieben.
  - Jede Leitung zwischen zwei Ports, die online sind, erhält die ihrem Diagnosezustand entsprechende Farbe.  
Die Farbe der Leitung zwischen zwei Ports ist abhängig vom Status der einzelnen Ports:

| Farbe des ersten Ports | Farbe des zweiten Ports | Farbe der Verbindungsleitung |
|------------------------|-------------------------|------------------------------|
| grün                   | grün                    | grün                         |
| grün                   | grau                    | grau                         |
| grün                   | rot                     | rot                          |
| grau                   | rot                     | rot                          |

- Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es wird ein Diagnosesymbol angezeigt. Es gehört zu demjenigen Teil der Station, der der CPU zugeordnet ist.
- Topologieübersicht
  - Bei jeder Hardware-Komponente wird das zugehörige Diagnosesymbol angezeigt.
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Es wird ein Diagnosesymbol angezeigt. Es gehört zu demjenigen Teil der Station, der der CPU zugeordnet ist.

- Projektnavigation
  - Hinter jeder Hardware-Komponente wird das zugehörige Diagnosesymbol angezeigt.
  - Wenn bei einer Hardware-Komponente mit unterlagerten Komponenten (z. B. Dezentrale Peripherie, Slave\_1) in mindestens einer unterlagerten Komponente ein Hardware-Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" eingeblendet.
  - Bei Hardware-Komponenten mit eigenem Betriebszustand wird zusätzlich das Betriebszustandssymbol in die rechte obere Ecke des Diagnosesymbols eingeblendet.
  - Falls auf einer CPU Forcen aktiv ist, wird am linken Rand des Diagnosesymbols ein rotes F eingeblendet.
  - Hinter dem Ordner "Lokale Baugruppen" wird das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" angezeigt, wenn in mindestens einer der zugehörigen Baugruppen ein Hardware-Fehler vorliegt.
  - Hinter dem Ordner "Dezentrale Peripherie" wird das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" angezeigt, wenn in mindestens einer der zugehörigen Komponenten ein Hardware-Fehler vorliegt.
  - Hinter dem Projektordner wird das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" angezeigt, wenn hinter mindestens einem der Ordner "Lokale Baugruppen" und "Dezentrale Peripherie" das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" angezeigt wird.
  - Für die Module bzw. Submodule eines Shared Device bei einer S7-1500-CPU gilt: Für die der CPU zugeordneten Module wird das zugehörige Diagnosesymbol angezeigt (Die nicht zugeordneten Module sind gegraut und erhalten kein Diagnosesymbol.). Für die steckbaren Submodule eines zugeordneten Moduls wird das zugehörige Diagnosesymbol angezeigt (Die nicht steckbaren Submodule sind nicht sichtbar und erhalten folglich auch kein Diagnosesymbol.).

---



#### Hinweis















Wenn bei einer Hardware-Komponente die Diagnose "ist von der CPU aus nicht erreichbar" gestellt wird, wird das Diagnosesymbol "Hardware-Fehler in unterlagerter Komponente" nicht zusätzlich eingeblendet.

---

### Diagnosesymbole für Baugruppen und Geräte

Die folgende Tabelle zeigt die möglichen Symbole und ihre jeweilige Bedeutung.

| Symbol  | Bedeutung   |
|---|---|
|  | Die Verbindung zu einer CPU wird gerade aufgebaut.            |
|  | Unter der eingestellten Adresse ist die CPU nicht erreichbar. |

| Symbol  | Bedeutung  |
|---|--|
|    | Die projektierte CPU und die tatsächlich vorhandene CPU sind vom Typ her inkompatibel.   |
|    | Beim Aufbau der Online-Verbindung zu einer geschützten CPU wurde der Passwort-Dialog ohne Eingabe des korrekten Passworts abgebrochen.   |
|    | Keine Störung  |
|    | Wartungsbedarf   |
|    | Wartungsanforderung  |
|    | Fehler   |
|    | Die Baugruppe bzw. das Gerät ist deaktiviert.  |
|    | Die Baugruppe bzw. das Gerät ist von der CPU aus nicht erreichbar (gültig für Baugruppen und Geräte unterhalb einer CPU).  |
|    | Es sind keine Diagnosedaten verfügbar, da die aktuellen Online-Konfigurationsdaten sich von den Offline-Konfigurationsdaten unterscheiden.   |
|    | Die projektierte Baugruppe bzw. das projektierte Gerät und die tatsächlich vorhandene Baugruppe bzw. das tatsächlich vorhandene Gerät sind inkompatibel (gültig für Baugruppen bzw. Geräte unterhalb einer CPU). |
|    | Die projektierte Baugruppe unterstützt die Anzeige des Diagnosezustands nicht (gültig für Baugruppen unterhalb einer CPU).   |
|   | Die Verbindung ist aufgebaut, aber der Zustand der Baugruppe wird momentan noch ermittelt bzw. ist unbekannt..   |
|  | Die projektierte Baugruppe unterstützt die Anzeige des Diagnosezustands nicht.   |
|  | Hardware-Fehler in unterlagerter Komponente: In mindestens einer unterlagerten Hardware-Komponente liegt ein Hardware-Fehler vor. (kommt als eigenständiges Symbol nur in der Projektnavigation vor)             |







**Hinweis**

Manche Baugruppen wie z. B. FM 450-1 werden beim Vorliegen eines Fehlers nur dann als fehlerhaft gekennzeichnet, wenn Sie bei der Parametrierung der Baugruppeneigenschaften den Diagnosealarm freigegeben haben.



## Symbole für den Vergleichsstatus

Die Diagnose-Symbole können rechts unten mit kleineren Zusatzsymbolen kombiniert werden, die das Ergebnis des Online-/Offline-Vergleichs anzeigen. Die folgende Tabelle zeigt die möglichen Vergleichssymbole und ihre Bedeutung.

| Symbol  | Bedeutung   |
|---|---|
|  | Hardware-Fehler in unterlagerter Komponente: In mindestens einer unterlagerten Hardware-Komponente sind Online- und Offline-Version verschieden (nur in der Projektnavigation). |
|  | Software-Fehler in unterlagerter Komponente: In mindestens einer unterlagerten Software-Komponente sind Online- und Offline-Version verschieden (nur in der Projektnavigation). |
|  | Online- und Offline-Version des Objekts sind verschieden  |
|  | Objekt nur online vorhanden   |
|  | Objekt nur offline vorhanden  |
|  | Online- und Offline-Version des Objekts sind gleich   |

### Hinweis



Wenn in der Gerätesicht sowohl ein Vergleichssymbol als auch das Diagnosesymbol " Fehler in unterlagerter Komponente " rechts unten eingeblendet werden sollen, gilt die folgende Regel: Das Diagnosesymbol für die unterlagerte Hardware-Komponente hat eine höhere Priorität als das Vergleichssymbol. Das hat zur Folge, dass ein Vergleichssymbol nur dann angezeigt wird, wenn in den unterlagerten Hardware-Komponenten keine Störung vorliegt.

## Anzeige von Software-Fehlern in der Projektnavigation

- Hinter jedem Baustein wird das zugehörige Vergleichssymbol eingeblendet.
- Hinter jedem Ordner, unter dem sich ausschließlich Bausteine verbergen, wird das Diagnosesymbol "Software-Fehler in unterlagerter Komponente" angezeigt, wenn in mindestens einem der zugehörigen Bausteine ein Software-Fehler vorliegt.
- Wenn bei einer Hardware-Komponente mit unterlagerten Software-Komponenten kein Hardware-Fehler vorliegt und in mindestens einer unterlagerten Software-Komponente ein Fehler vorliegt, sieht das Diagnosesymbol wie folgt aus: Das Diagnosesymbol der Hardware-Komponente wird blass dargestellt, und zusätzlich wird in die rechte untere Ecke das Diagnosesymbol "Software-Fehler in unterlagerter Komponente" eingeblendet.








### Kombinierte Diagnose- und Vergleichssymbole

Die folgende Tabelle zeigt Beispiele für Symbole, die in das Diagnosesymbol eingeblendet werden.

| Symbol  | Bedeutung   |
|---|---|
|  | Ordner enthält Objekte, deren Online- und Offline-Version verschieden sind (nur in der Projektnavigation) |
|  | Objekt nur online vorhanden   |

### Betriebszustandssymbole für CPUs und CPs

Die folgende Tabelle zeigt die möglichen Symbole und die zugehörigen Betriebszustände.





| Symbol  | Betriebszustand  |
|---|--|
|    | RUN  |
|    | STOP   |
|   | ANLAUF   |
|  | HALT   |
|  | DEFEKT   |
|  | Unbekannter Betriebszustand  |
|  | Die projektierte Baugruppe unterstützt die Anzeige des Betriebszustands nicht. |

#### Hinweis

Falls auf einer CPU Forcen aktiv ist, wird in das Betriebszustandssymbol rechts unten ein rotes F auf rosafarbenem Grund eingeblendet.

### Farbliche Kennzeichnung von Ports und Ethernet-Leitungen

Die folgende Tabelle zeigt die möglichen Farben und ihre jeweilige Bedeutung.

| Farbe   | Bedeutung                         |
|---|-----------------------------------|
|  | Keine Störung oder Wartungsbedarf |
|  | Wartungsanforderung               |
|  | Kommunikation gestört             |
|  | nicht diagnosefähig               |

## Online- und Diagnosesicht starten

### Übersicht über die Möglichkeiten, die Online- und Diagnosesicht zu starten

Sie können die Online- und Diagnosesicht einer zu diagnostizierenden Baugruppe an folgenden Stellen starten:

- Übersicht
- Projektnavigation
- Gerätesicht
- Geräteübersicht
- Netzsicht
- Netzübersicht
- Topologiesicht

Im Folgenden wird anhand von Beispielen dargestellt, wie dabei vorzugehen ist.

### Voraussetzung

Das Projekt mit der zu diagnostizierenden Baugruppe ist geöffnet.

---

#### Hinweis

Diese Voraussetzung ist nicht erforderlich, wenn Sie die Online- und Diagnosesicht aus der Projektnavigation heraus aufrufen, nachdem Sie die erreichbaren Teilnehmer ermittelt haben.

---

### Vorgehen

Um die Online- und Diagnosesicht einer Baugruppe zu starten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den betroffenen Geräteordner.
2. Doppelklicken Sie "Online & Diagnose".

Oder:

1. Selektieren Sie in der Projektnavigation den betroffenen Geräteordner.
2. Wählen Sie im Kontextmenü oder im Hauptmenü "Online" den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie in der Projektnavigation den Ordner "Online-Zugänge".
2. Öffnen Sie den Ordner der Schnittstelle, über die Sie die Online-Verbindung herstellen wollen.
3. Doppelklicken Sie auf "Erreichbare Teilnehmer anzeigen / aktualisieren".
4. Selektieren Sie die zu diagnostizierende Baugruppe.
5. Wählen Sie im Kontextmenü oder im Hauptmenü "Online" den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie in der Projektnavigation den Ordner "Lokale Baugruppen".
2. Selektieren Sie das betroffene Gerät bzw. die zu diagnostizierende Baugruppe.
3. Wählen Sie im Kontextmenü oder im Hauptmenü den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie die Gerätesicht der Gerätekonfiguration.
2. Selektieren Sie die zu diagnostizierende Baugruppe.
3. Wählen Sie im Kontextmenü oder im Hauptmenü "Online" den Befehl "Online & Diagnose".

Oder:

1. Öffnen Sie die Gerätesicht der Gerätekonfiguration.
2. Stellen Sie eine Online-Verbindung mit der zu diagnostizierenden Baugruppe her.
3. Doppelklicken Sie auf das Diagnosesymbol oberhalb der Baugruppe.

Oder:

1. Öffnen Sie die Netzsicht der Gerätekonfiguration.
2. Selektieren Sie die Station mit der zu diagnostizierenden Baugruppe.
3. Wählen Sie im Kontextmenü oder im Hauptmenü "Online" den Befehl "Online & Diagnose".

Oder:

## Ergebnis

Die Online- und Diagnosesicht der zu diagnostizierenden Baugruppe wird gestartet. Falls vorher eine Online-Verbindung zur zugehörigen CPU bestand, wird die Titelzeile der Online- und Diagnosesicht mit der Online-Farbe orange hinterlegt.

---

### Hinweis

Falls beim Start der Online- und Diagnosesicht keine Online-Verbindung besteht, können keine Online-Informationen angezeigt werden und die entsprechenden Anzeigefelder bleiben leer.

---

## Aktivieren der Task Card "Online-Tools"

### Aktivieren der Task Card "Online-Tools"

Sie können diese Task Card wie folgt aktivieren:

1. Starten Sie die Online- und Diagnosesicht.
2. Klicken sie auf die Task Card "Online-Tools".

Oder:

1. Starten Sie die Gerätesicht.
  2. Klicken sie auf die Task Card "Online-Tools".
- Oder:
1. Starten Sie die Netzsicht.
  2. Klicken sie auf die Task Card "Online-Tools".

### 8.2.1.2 Nicht änderbare und aktuelle Werte von parametrierbaren Baugruppeneigenschaften anzeigen

#### Allgemeine Eigenschaften und anlagenrelevante Informationen einer Baugruppe anzeigen

##### Wo befinden sich die gesuchten Informationen?

Die allgemeinen Eigenschaften und anlagenrelevanten Informationen einer Baugruppe finden Sie in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Allgemein".

##### Aufbau der Gruppe "Allgemein"

Die Gruppe "Allgemein" besteht aus folgenden Bereichen:

- Baugruppe
- Baugruppeninformation
- Herstellerinformation

##### Bereich "Baugruppe"

Dieser Bereich zeigt die folgenden Daten der Baugruppe an:

- Kurzbezeichnung, z. B. CPU 1214C DC/DC/DC
- Bestell-Nr.
- Hardware
- Firmware
- Baugruppenträger
- Steckplatz

##### Bereich " Baugruppeninformation"

Dieser Bereich zeigt die folgenden Daten der Baugruppe an, die Sie bei der Konfiguration der Hardware projiziert haben:

- Baugruppenname
- Einbaudatum (wird nicht bei allen Baugruppen angezeigt)
- Zusatzinformation (wird nicht bei allen Baugruppen angezeigt)

### Bereich "Herstellerinformation"

Dieser Bereich zeigt die folgenden Daten der Baugruppe an:

- Hersteller
- Seriennummer
- Profil: Profil-ID als Hexadezimalzahl

---

#### Hinweis

Den zugehörigen Profilnamen entnehmen Sie der Profile ID Table bei PROFIBUS International (siehe "www.profibus.com").

---

- Profildetails: Profilspezifischer Typ als Hexadezimalzahl

---

#### Hinweis

Den zugehörigen profilspezifischen Typnamen entnehmen Sie der Profile Specific Type Table bei PROFIBUS International (siehe "www.profibus.com").

---

### Parametrierte Zykluszeiten anzeigen

#### Wo befinden sich die gesuchten Informationen?

Die gesuchten Informationen finden Sie an den folgenden Stellen:

- In der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Zykluszeit"
- In der Task Card "Online-Tools" in der Palette "Zykluszeit"

#### Aufbau der Gruppe "Zykluszeit" im Ordner "Diagnose" der Online- und Diagnosesicht

Die Gruppe "Zykluszeit" besteht aus folgenden Bereichen:

- Zykluszeit-Diagramm (Grafische Anzeige der parametrisierten und der gemessenen Zykluszeiten)
- Zykluszeit parametrisiert (Anzeige der parametrisierten Zykluszeiten als Absolutwerte)
- Zykluszeiten gemessen (Anzeige der gemessenen Zykluszeiten als Absolutwerte)

#### Aufbau der Palette "Zykluszeit" der Task Card "Online-Tools"

Die Palette "Zykluszeit" zeigt das Zykluszeit-Diagramm und darunter die gemessenen Zykluszeiten als Absolutwerte.

## Parametrierte Zykluszeiten

Folgende parametrisierten Zykluszeiten werden im Zykluszeit-Diagramm und im Bereich "Zykluszeit parametrieren" dargestellt:

- Mindestzykluszeit
- Zyklusüberwachungszeit

Im Zykluszeit-Diagramm entsprechen die Mindestzykluszeit und die Zyklusüberwachungszeit den beiden Markierungen auf der Zeitachse.

Im Bereich "Zykluszeit parametrieren" werden die parametrisierten Zykluszeiten als Absolutwerte dargestellt.

## Schnittstellen und Schnittstelleneigenschaften einer Baugruppe anzeigen

### Wo befinden sich die gesuchten Informationen?

Die Schnittstellen und Schnittstelleneigenschaften einer Baugruppe finden Sie in der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der folgenden Gruppe:

- PROFINET-Schnittstelle

### Gruppe "PROFINET-Schnittstelle"

Diese Gruppe gliedert sich in folgende Bereiche:

- "Ethernet-Adresse" mit den Unterbereichen "Netzwerkverbindung" und "IP-Parameter"
- "Ports"

### Unterbereich "Netzwerkverbindung" des Bereichs "Ethernet-Adresse"

Dieser Unterbereich zeigt die folgenden Daten der Baugruppe an:

- MAC-Adresse:  
MAC-Adresse der Schnittstelle.  
Die MAC-Adresse besteht aus zwei Teilen. Der erste Teil ("Basis-MAC-Adresse") kennzeichnet den Hersteller (Siemens, 3COM, ...). Der zweite Teil der MAC-Adresse unterscheidet die verschiedenen Ethernet-Teilnehmer. Jeder Ethernet-Baugruppe ist eine eindeutige MAC-Adresse zugeordnet.

### Unterbereich "IP-Parameter" des Bereichs "Ethernet-Adresse"

Dieser Unterbereich zeigt die folgenden Daten der Baugruppe an:

- IP-Adresse:  
Internet-Protokolladresse des Teilnehmers am Bus (TCP/IP)
- Subnetzmaske:  
Die Subnetzmaske gibt an, welcher Teil der IP-Adresse die Zugehörigkeit zu einem bestimmten Subnetz bestimmt.

8.2 Geräte und Netze diagnostizieren

- **Default-Router:**  
Falls das Subnetz über Router an andere Subnetze angeschlossen ist, muss die IP-Adresse des Standard-Routers bekannt sein. Nur so kann ein Datagramm mit einer nicht übereinstimmenden Subnetzadresse weitergeleitet werden.
- **IP-Einstellungen:**  
Kennung für den Weg, über den der Teilnehmer seine IP-Einstellungen (IP-Adresse, Subnetzmaske, Default-Router) bekommen hat.

| Kennung | Bedeutung   |
|---------|---|
| 0       | IP-Adresse ist nicht initialisiert  |
| 1       | Über Projektierung (d. h. über die Konfiguration, die aus der Geräte- oder Netzsicht in den Teilnehmer geladen wurde)   |
| 2       | Über die Gruppe "IP-Adresse zuweisen" der Online- und Diagnosesicht   |
| 3       | Über DHCP-Server (d. h. die IP-Parameter wurden über einen speziellen Dienst von einem DHCP-Server (Dynamic Host Configuration Protocol) bezogen und für begrenzte Zeit zugewiesen) |

- **IP-Einstellungszeit:**  
Zeitstempel der letzten Änderung der IP-Adresse direkt über den Ethernetanschluss der Baugruppe

**Bereich "Ports"**

Dieser Bereich zeigt die folgenden Daten der Baugruppe an:

- **Ethernet-Ports**  
Physikalische Eigenschaften der PROFINET-Schnittstelle

| Eigenschaft der PROFINET-Schnittstelle | Bedeutung  |
|--|--|
| Port-Nr.                               | Port-Nummer. In Klammern ist die Kurzbezeichnung von Schnittstelle (X + Nr. der Schnittstelle) und Port (P + Nr. des Ports) angegeben. Ein "R" in der Kurzbezeichnung eines Ports bedeutet, dass es sich um einen Ringport handelt.  |
| Status                                 | Anzeige des Zustands der zum Port gehörigen LINK-LED: <ul style="list-style-type: none"> <li>• Bei "OK" ist ein anderes Gerät (z. B. ein Switch) an den Port angeschlossen, und die physikalische Verbindung steht.</li> <li>• Bei "Getrennt" ist kein weiteres Gerät an den Port angeschlossen.</li> <li>• Bei "Abgeschaltet" ist der Zugang zum Port unterbunden.</li> </ul> |
| Einstellungen                          | Individuelle Netzwerkeinstellungen des Teilnehmers (automatisch oder manuell)  |
| Betriebsart                            | Netzwerkeinstellungen hinsichtlich der Geschwindigkeit und des Übertragungsverfahrens  |

Wenn Sie in der Port-Tabelle eine Zeile markieren, dann werden zum zugehörigen Port zusätzliche Hilfeinformationen bereitgestellt.



## **IO-Controller anzeigen, die auf Module eines Shared Device zugreifen**

### **Wo befinden sich die gesuchten Informationen?**

Die Anzeige derjenigen IO-Controller, die auf Module eines Shared Device zugreifen, finden Sie in der Online- und Diagnosesicht des Interfacemoduls des Shared Device im Ordner "Diagnose" im folgenden Bereich der Gruppe "PROFINET-Schnittstelle":

- IO-Controller

## **Sync-Domain-Eigenschaften eines PROFINET-Geräts anzeigen**

### **Wo befinden sich die gesuchten Informationen?**

Die Sync-Domain-Eigenschaften eines PROFINET-Geräts finden Sie in der Online- und Diagnosesicht des zu diagnostizierenden Geräts im Ordner "Diagnose" im folgenden Bereich der Gruppe "PROFINET-Schnittstelle":

- Domain

### **Bereich "Domain"**

Dieser Bereich gliedert sich in die folgenden Unterbereiche:

- Sync-Domain
- MRP-Domain

### **Was ist eine Sync-Domain?**

Eine Sync-Domain ist eine Gruppe von PROFINET-Geräten, die auf einen gemeinsamen Takt synchronisiert sind. Genau ein Gerät hat die Rolle des Sync-Masters (Taktgeber), alle anderen Geräte nehmen die Rolle eines Sync-Slaves ein. Der Sync-Master ist meist ein IO-Controller oder ein Switch.

Nicht synchronisierte PROFINET-Geräte sind nicht Teil einer Sync-Domain.

### **Unterbereich "Sync-Domain" des Bereichs "Domain"**

Dieser Unterbereich zeigt die folgenden Eigenschaften der Sync-Domain an:

- Name:  
Name der Sync-Domain
- Rolle:  
Rolle des PROFINET-Geräts in der Sync-Domain. Folgende Rollen sind möglich:
  - Sync-Master
  - Sync-Slave
- Synchronisationsintervall:  
Intervall, in dem die Synchronisation durchgeführt wird

- Sendetakt  
kleinstmögliches Sende-Intervall für den Datenaustausch
- Jittergenauigkeit des Sendetakts
- Reservierte Bandbreite für zyklische Kommunikation

## MRP-Domain-Eigenschaften eines PROFINET-Geräts anzeigen

### Wo befinden sich die gesuchten Informationen?

Die MRP-Domain-Eigenschaften eines PROFINET-Geräts finden Sie in der Online- und Diagnosesicht des zu diagnostizierenden Geräts im Ordner "Diagnose" im folgenden Bereich der Gruppe "PROFINET-Schnittstelle":

- Domain

### Bereich "Domain"

Dieser Bereich gliedert sich in die folgenden Unterbereiche:

- Sync-Domain
- MRP-Domain

### Was ist eine MRP-Domain?

Über das sogenannte Media Redundancy Protocol (MRP) ist es möglich, redundante Netze aufzubauen. Redundante Übertragungstrecken (Ringtopologie) sorgen dafür, dass bei Ausfall einer Übertragungstrecke ein alternativer Kommunikationsweg zur Verfügung steht. Diejenigen PROFINET-Geräte, die Teil dieses redundanten Netzes sind, bilden eine MRP-Domain.

### Unterbereich "MRP-Domain" des Bereichs "Domain"

Dieser Unterbereich zeigt die folgenden Eigenschaften der MRP-Domain an:

- Name:  
Name der MRP-Domain
- Rolle:  
Rolle des PROFINET-Geräts in der MRP-Domain. Folgende Rollen sind möglich:
  - Manager
  - Manager (Auto)
  - Client
  - Nicht Teilnehmer des Rings
- Ringport 1:  
derjenige Port des PROFINET-Geräts, der die Eigenschaft "Ringport 1" hat

- Ringport 2:  
derjenige Port des PROFINET-Geräts, der die Eigenschaft "Ringport 2" hat
- Zustand des MRP-Rings:  
Hier wird angezeigt, ob der Ring unterbrochen ist (Zustand "offen") oder nicht (Zustand "geschlossen").

## Aktuelle Firmware einer Baugruppe anzeigen

### Firmware anzeigen

Sie können sich die aktuell installierte Firmware einer Baugruppe anzeigen lassen.

### Voraussetzungen

- Die Baugruppe unterstützt ein Firmware-Update.
- Die Baugruppe ist online verbunden.

### Vorgehen

Um die aktuelle Firmware anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Baugruppe in der Online- und Diagnosesicht.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Firmware-Update".
3. Die aktuelle Firmware lesen Sie im Bereich "Online-Daten" unter "Firmware" ab.

## 8.2.1.3 Aktuelle Werte von dynamischen Baugruppeneigenschaften anzeigen

### Gemessene Zykluszeiten anzeigen

#### Wo befinden sich die gesuchten Informationen?

Die gemessenen Zykluszeiten finden Sie an den folgenden beiden Stellen:

- In der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" in der Gruppe "Zykluszeit"
- In der Task Card "Online-Tools" in der Palette "Zykluszeit"

### Aufbau der Gruppe "Zykluszeit" im Ordner "Diagnose" der Online- und Diagnosesicht

Die Gruppe "Zykluszeit" besteht aus folgenden Bereichen:

- Zykluszeit-Diagramm (Grafische Anzeige der parametrierten und der gemessenen Zykluszeiten)
- Zykluszeit parametriert (Anzeige der parametrierten Zykluszeiten als Absolutwerte)
- Zykluszeiten gemessen (Anzeige der gemessenen Zykluszeiten als Absolutwerte)

### Aufbau der Palette "Zykluszeit" der Task Card "Online-Tools"

Die Palette "Zykluszeit" zeigt das Zykluszeit-Diagramm und darunter die gemessenen Zykluszeiten als Absolutwerte.

### Grafische Darstellung der gemessenen Zykluszeiten

Folgende gemessenen Zykluszeiten werden im Zykluszeit-Diagramm dargestellt:

- Kürzeste Zykluszeit: Dauer des kürzesten Zyklus seit dem letzten Übergang von STOP nach RUN  
Im Diagramm entspricht das dem linken gestrichelten grauen Pfeil.
- Aktuelle / letzte Zykluszeit: Dauer des letzten Zyklus  
Im Diagramm entspricht das dem grünen Pfeil. Wenn die aktuelle / letzte Zykluszeit die Zyklusüberwachungszeit überschreitet, wird der Pfeil rot.

---

#### Hinweis

Wenn die Dauer des letzten Zyklus nahe an die Zyklusüberwachungszeit heranreicht, ist es möglich, dass diese überschritten wird. Abhängig vom CPU-Typ, Ihrer Parametrierung und Ihrem Anwenderprogramm kann die CPU in den Betriebszustand STOP gehen. Wenn Sie z. B. Variablen Ihres Programms beobachten, verlängern Sie damit die Zykluszeit.

Wenn der Zyklus länger als die doppelte Zyklusüberwachungszeit dauert und Sie im Anwenderprogramm die Zyklusüberwachungszeit nicht neu starten (durch Aufruf der erweiterten Anweisung RE\_TRIGR), geht die CPU in STOP.

---

- Längste Zykluszeit: Dauer des längsten Zyklus seit dem letzten Übergang von STOP nach RUN  
Im Diagramm entspricht das dem rechten gestrichelten blauen Pfeil.

Zwischen den beiden gestrichelten Pfeilen erstreckt sich ein blauer Streifen, der dem gesamten Bereich der gemessenen Zykluszeiten entspricht. Wenn eine gemessene Zykluszeit größer als die Zyklusüberwachungszeit ist, wird derjenige Teil des Streifens, der außerhalb der parametrierten Grenzen liegt, rot eingefärbt.

### **Darstellung der gemessenen Zykluszeiten als Absolutwerte**

Folgende gemessenen Zeiten werden im Bereich "Zykluszeiten gemessen" und in der Palette "Zykluszeit" angezeigt:

- Kürzeste Zykluszeit seit dem letzten Übergang von STOP nach RUN
- Aktuelle/letzte Zykluszeit
- Längste Zykluszeit seit dem letzten Übergang von STOP nach RUN

### **Aktuellen Zustand der LEDs einer CPU anzeigen**

#### **Wo befinden sich die gesuchten Informationen?**

Den aktuellen Zustand der LEDs einer CPU finden Sie in der Task Card "Online-Tools" im Anzeigebereich der Palette "CPU-Bedienpanel".

### **Anzeigebereich der Palette "CPU-Bedienpanel" in der Task Card "Online-Tools"**

Dieser Bereich enthält die folgenden Anzeigen:

- Stationsname und CPU-Typ (Kurzbezeichnung)
- RUN / STOP (entspricht der LED "RUN / STOP" der CPU)
- ERROR (entspricht der LED "ERROR" der CPU)
- MAINT (entspricht der LED "MAINT" der CPU)

### **Füllstand aller Speicherarten einer CPU anzeigen**

#### **Wo befinden sich die gesuchten Informationen?**

Den Füllstand aller Speicherarten einer CPU finden Sie an den folgenden beiden Stellen:

- In der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" im Anzeigebereich der Gruppe "Speicher"
- In der Task Card "Online-Tools" im Anzeigebereich der Palette "Speicher"

### **Anzeigebereich der Gruppe "Speicher" im Ordner "Diagnose" der Online- und Diagnosesicht**

Dieser Bereich enthält die aktuelle Speicherauslastung der zugehörigen Baugruppe und Details der einzelnen Speicherbereiche.

Die Speicherauslastung wird sowohl als Balkendiagramm als auch als Zahlenwert (prozentual) angezeigt.

Die folgenden Speicherauslastungen werden angezeigt:

- Ladespeicher  
Falls keine Memory Card steckt, wird der interne Ladespeicher angezeigt.  
Falls eine Memory Card steckt, benutzt das Betriebssystem als Ladespeicher ausschließlich den gesteckten Ladespeicher. Dieser wird hier angezeigt.
- Arbeitsspeicher
- Remanenzspeicher

### Anzeigebereich der Palette "Speicher" in der Task Card "Online-Tools"

Dieser Bereich enthält die aktuelle Speicherauslastung der zugehörigen Baugruppe. Der freie Speicher wird sowohl als Balkendiagramm als auch als Zahlenwert (prozentual) angezeigt. Der Zahlenwert ist auf ganzzahlige Werte gerundet.

---

#### Hinweis

Wenn ein Speicherbereich weniger als 1 % belegt ist, wird beim freien Teil dieses Speicherbereichs "99 %" angezeigt.

---

Die folgenden Speicherauslastungen werden angezeigt:

- Ladespeicher  
Falls keine Memory Card steckt, wird der interne Ladespeicher angezeigt.  
Falls eine Memory Card steckt, benutzt das Betriebssystem als Ladespeicher ausschließlich den gesteckten Ladespeicher. Dieser wird hier angezeigt.
- Arbeitsspeicher
- Remanenzspeicher

### Siehe auch

Ladespeicher (Seite 842)

Arbeitsspeicher (Seite 842)

Remanente Speicherbereiche (Seite 844)

### Füllstand aller Speicherarten einer S7-1500-CPU anzeigen

#### Wo befinden sich die gesuchten Informationen?

Den Füllstand aller Speicherarten einer S7-1500-CPU finden Sie an den folgenden beiden Stellen:

- In der Online- und Diagnosesicht der zu diagnostizierenden Baugruppe im Ordner "Diagnose" im Anzeigebereich der Gruppe "Speicher"
- In der Task Card "Online-Tools" im Anzeigebereich der Palette "Speicher"

### Anzeigebereich der Gruppe "Speicher" im Ordner "Diagnose" der Online- und Diagnosesicht

Dieser Bereich enthält die aktuelle Speicherauslastung der zugehörigen Baugruppe und Details der einzelnen Speicherbereiche.

Die Speicherauslastung wird sowohl als Balkendiagramm als auch als Zahlenwert (prozentual) angezeigt.

Die folgenden Speicherauslastungen werden angezeigt:

- Ladespeicher

---

**Hinweis**

Der Ladespeicher befindet sich auf der SIMATIC Memory Card.

---

- Code-Arbeitsspeicher: Arbeitsspeicher für Programmcode
- Daten-Arbeitsspeicher: Arbeitsspeicher für Datenbausteine
- Remanenzspeicher

### Anzeigebereich der Palette "Speicher" in der Task Card "Online-Tools"

Dieser Bereich enthält die aktuelle Speicherauslastung der zugehörigen Baugruppe. Der freie Speicher wird sowohl als Balkendiagramm als auch als Zahlenwert (prozentual) angezeigt. Der Zahlenwert ist auf ganzzahlige Werte gerundet.

---

**Hinweis**

Wenn ein Speicherbereich weniger als 1 % belegt ist, wird beim freien Teil dieses Speicherbereichs "99 %" angezeigt.

---

Die folgenden Speicherauslastungen werden angezeigt:

- Ladespeicher

---

**Hinweis**

Der Ladespeicher befindet sich auf der SIMATIC Memory Card.

---

- Code-Arbeitsspeicher: Arbeitsspeicher für Programmcode
- Daten-Arbeitsspeicher: Arbeitsspeicher für Datenbausteine
- Remanenzspeicher

#### 8.2.1.4 Eine Baugruppe auf Störungen überprüfen

##### Diagnosestatus einer Baugruppe ermitteln

##### Wo wird der Diagnosestatus einer Baugruppe angezeigt?

Der Diagnosestatus einer Baugruppe wird in der Online- und Diagnosesicht der Baugruppe im Ordner "Diagnose" in der Gruppe "Diagnosestatus" angezeigt.

Die Gruppe "Diagnosestatus" besteht aus folgenden Bereichen:

- Status
- Standarddiagnose (bei S7-300 und S7-400 nur bei Nicht-CPU-Baugruppen)

##### Bereich "Status"

In diesem Bereich werden die folgenden Statusinformationen angezeigt:

- Status der Baugruppe aus Sicht der CPU, z. B.:
  - Baugruppe vorhanden und OK.
  - Baugruppe gestört.  
Falls die Baugruppe eine Störung hat und Sie bei der Projektierung den Diagnosealarm freigegeben haben, wird der Status "Baugruppe gestört" angezeigt.
  - Baugruppe projektiert, aber nicht vorhanden.  
Bsp.: Es sind keine Diagnosedaten verfügbar, weil die aktuelle Online-Konfiguration sich von der Offline-Konfiguration unterscheidet.
- Erkannte Unterschiede zwischen projektierte und gesteckter Baugruppe. Soweit ermittelbar wird für Soll- und Isttyp die Bestellnummer angezeigt.

Der Umfang der angezeigten Informationen ist abhängig von der selektierten Baugruppe.

##### Bereich "Standarddiagnose"

In diesem Bereich werden die folgenden Diagnoseinformationen von Nicht-CPU-Baugruppen angezeigt:

- Interne und externe Störungen, die sich auf die gesamte Baugruppe beziehen
- Zugehörige Diagnoseereignisse

Beispiele für solche Diagnoseinformationen sind:



- Gesamte Pufferung ausgefallen
- Baugruppe gestört

---

#### **Hinweis**

#### **Diagnosealarme**

Ein Diagnosealarm kann nur dann zur CPU gemeldet werden, wenn die Baugruppe diagnosealarmfähig ist und Sie den Diagnosealarm freigegeben haben.

Die Anzeige des Diagnosealarms ist eine Momentaufnahme. Sporadische Störungen einer Baugruppe erkennen Sie im Diagnosepuffer der zugehörigen CPU.

---

## **Den Diagnosepuffer einer CPU auslesen**

### **Wo lesen Sie den Diagnosepuffer einer CPU aus?**

Sie lesen den Diagnosepuffer einer CPU in deren Online- und Diagnosesicht im Ordner "Diagnose" in der Gruppe "Diagnosepuffer" aus.

### **Aufbau der Gruppe "Diagnosepuffer"**

Die Gruppe "Diagnosepuffer" besteht aus folgenden Bereichen:

- "Ereignisse"
- "Einstellungen"

### **Diagnosepuffer**

Der Diagnosepuffer dient als Log-Datei für die auf der CPU und den ihr zugeordneten Baugruppen aufgetretenen Diagnoseereignisse. Diese werden in der Reihenfolge ihres Auftretens eingetragen, wobei das jüngste Ereignis an oberster Stelle dargestellt wird.

### **Bereich "Ereignisse"**

Der Bereich "Ereignisse" besteht aus folgenden Elementen:

- Optionskästchen "CPU-Zeitstempel berücksichtigt lokale PG/PC-Zeit"
- Ereignistabelle
- Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben"
- Details zum Ereignis: Ereignis-Nr., Ereignis-ID, Beschreibung, Zeitstempel, Kommend/ Gehend-Information
- Schaltflächen "Hilfe zum Ereignis", "Im Editor öffnen", "Speichern unter ..."

### Optionskästchen " CPU-Zeitstempel berücksichtigt lokale PG/PC-Zeit "

Wenn Sie das Optionskästchen nicht aktiviert haben, werden die Diagnosepuffereinträge mit der Baugruppenzeit angezeigt.

Wenn Sie das Optionskästchen aktiviert haben, werden die Diagnosepuffereinträge mit derjenigen Uhrzeit angezeigt, die sich aus folgender Formel ergibt:

$$\text{Angezeigte Uhrzeit} = \text{Baugruppenzeit} + \text{Zeitzone-Offset Ihres PG / PC}$$

Dabei wird vorausgesetzt, dass die Baugruppenzeit identisch mit UTC ist.

Sie verwenden diese Einstellung, wenn sie bei den Diagnosepuffereinträgen der Baugruppe als Uhrzeit die Lokalzeit Ihres PG / PC sehen wollen.

Wenn Sie das Optionskästchen aktivieren bzw. deaktivieren, werden die Zeitangaben der Diagnosepuffereinträge sofort angepasst.

---

#### Hinweis

Wenn Sie in Ihrem Programm die Anweisung "WR\_SYS\_T" einsetzen oder die Echtzeituhr der CPU über ein Bediengerät stellen und dabei nicht UTC verwenden, wird empfohlen, das Optionskästchen " CPU-Zeitstempel berücksichtigt lokale PG/PC-Zeit " zu deaktivieren. In diesem Fall orientieren Sie sich nur an der Baugruppenzeit.

---

### Ereignistabelle

In der Tabelle werden zu jedem Diagnoseereignis folgende Informationen angezeigt:

- Laufende Nummer des Eintrags  
Der erste Eintrag enthält das neueste Ereignis.
- Datum und Uhrzeit des Diagnoseereignisses  
Werden kein Datum und keine Uhrzeit ausgegeben, besitzt die Baugruppe keine integrierte Uhr.
- Kurzbezeichnung des Ereignisses und ggf. die Reaktion der CPU

---



#### Hinweis



Wenn ein einzelner Parameter eines Textes nicht ermittelt werden kann, wird an dieser Stelle die Zeichenfolge "####" ausgegeben.

Wenn für neue Baugruppen oder neue Ereignisse noch kein Anzeigetext vorhanden ist, werden die Nummern der Ereignisse und die einzelnen Parameter als Hexadezimalwerte ausgegeben.





---

- Symbol für die Kommend-/Gehend-Information  
Die folgende Tabelle zeigt die möglichen Symbole und ihre jeweilige Bedeutung.

| Symbol  | Bedeutung          |
|---|--------------------|
|  | Kommendes Ereignis |
|  | Gehendes Ereignis  |

| Symbol  | Bedeutung  |
|---|--|
|  | Kommendes Ereignis, zu dem es kein eigenständiges gehendes Ereignis gibt |
|  | Anwenderdefiniertes Diagnoseereignis                                     |

- Nur bei S7-1200- und S7-1500-CPU: Symbol für das Gewicht des Ereignisses  
Die folgende Tabelle zeigt die möglichen Symbole und ihre jeweilige Bedeutung.

| Symbol  | Bedeutung                        |
|---|----------------------------------|
|  | keine Wartung bzw. keine Störung |
|  | Wartungsbedarf                   |
|  | Wartungsanforderung              |
|  | Fehler                           |

Sie können in der Ereignistabelle die Spaltenreihenfolge ändern, die Spaltenbreiten ändern und einzelne Spalten entfernen und wieder hinzufügen. Darüber hinaus können Sie wie folgt sortieren: nach der laufenden Nummer, nach "Datum und Uhrzeit" und nach "Ereignis".

### Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben"

Die Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben" ist nur aktiv, wenn eine Online-Verbindung zur CPU besteht.

Die Voreinstellung ist "Anzeige einfrieren".

Wenn Sie auf die Schaltfläche "Anzeige einfrieren" klicken, geschieht Folgendes:

- Die aktuelle Anzeige der Diagnosepuffereinträge wird eingefroren.
- Die Beschriftung der Schaltfläche wechselt zu "Einfrieren aufheben".

Wenn ein Fehler in Ihrer Anlage aufgetreten ist, kann es zu schnell aufeinanderfolgenden Diagnoseereignissen kommen. Daraus resultiert eine hohe Aktualisierungsrate der Anzeige. Das Einfrieren der Anzeige bietet Ihnen in diesem Fall die Möglichkeit, den Sachverhalt in Ruhe näher zu untersuchen.

Wenn die Anzeige eingefroren ist und Sie auf die Schaltfläche "Einfrieren aufheben" klicken, geschieht Folgendes:

- Die Anzeige der Diagnosepuffereinträge wird wieder aktualisiert.
- Die Beschriftung der Schaltfläche wechselt zu "Anzeige einfrieren".

---

#### Hinweis

Beim Einfrieren der Diagnosepufferanzeige trägt die CPU weiterhin Ereignisse in den Diagnosepuffer ein.

---

### Details zum Ereignis

Wenn Sie in der Liste der Ereignisse eine Zeile selektieren, erhalten Sie hier Detailinformationen zum zugehörigen Ereignis:

- Laufende Nummer des Ereignisses im Diagnosepuffer
- Ereignis-ID
- Beschreibung des Ereignisses mit ereignisabhängigen Zusatzinformationen. Beispiele für diese Zusatzinformationen sind:
  - Befehl, der das Ereignis verursacht hat
  - Betriebszustandsübergang, der durch das Diagnoseereignis verursacht wurde
- Zeitstempel
- Nur bei S7-1200- und S7-1500-CPU: Zugehörige I&M-Daten (Modul, Rack/Steckplatz, Anlagenkennzeichen, Ortskennzeichen)
- Priorität des Ereignisses
- Information, ob es sich um ein kommendes oder ein gehendes Ereignis handelt

### Schaltfläche "Hilfe zum Ereignis"

Wenn Sie auf diese Schaltfläche klicken, wird das selektierte Ereignis näher erläutert und es werden ggf. Abhilfen genannt.

---

#### Hinweis

Bei einigen wenigen Ereignissen ist die Schaltfläche "Hilfe zum Ereignis" grau.

---

### Schaltfläche "Im Editor öffnen"

Die folgende Tabelle zeigt, wann die Schaltfläche "Öffne Baustein" aktiv ist und welche Funktion sich dahinter verbirgt.

| Wann ist die Schaltfläche "Im Editor öffnen" aktiv?   | Was bewirkt ein Klicken auf diese Schaltfläche?   |
|---|---|
| Wenn im Diagnoseereignis auf eine Relativadresse eines Bausteins verwiesen wird.<br>Dies ist die Adresse des Ereignis verursachenden Befehls. | Die Funktion "Im Editor öffnen" öffnet den referenzierten Baustein in der Offline-Ansicht an der Fehler verursachenden Programmieranweisung. Dadurch können Sie den Quellcode des Bausteins an der angegebenen Stelle überprüfen, ggf. ändern und anschließend wieder in die CPU laden. |
| Wenn das Diagnoseereignis von einem Modul ausgelöst wurde.  | Die Funktion "Im Editor öffnen" öffnet die Gerätesicht des betroffenen Moduls.  |

### Schaltfläche "Speichern unter ..."

Wenn Sie auf diese Schaltfläche klicken, wird der Inhalt des Diagnosepuffers in einer Textdatei gespeichert. Als Dateiname wird sprachabhängig "Diagnose" mit der Erweiterung ".txt" vorgeschlagen. Sie können diesen Namen jedoch ändern.

## **Bereich "Einstellungen"**

Der Bereich "Einstellungen" besteht aus folgenden Elementen:

- Liste "Ereignisse anzeigen"
- Schaltfläche "Einstellungen als Standard übernehmen"
- Optionskästchen "Ereignisinformation hexadezimal ausgeben"

### **Liste "Ereignisse anzeigen:"**

In dieser Liste gibt es zu jeder Ereignisklasse ein Optionskästchen (Voreinstellung: Alle Optionskästchen sind aktiviert.). Wenn Sie ein Optionskästchen deaktivieren, werden die Ereignisse der zugehörigen Ereignisklasse im Bereich "Ereignisse" nicht mehr angezeigt. Wenn Sie das Optionskästchen wieder aktivieren, werden die zugehörigen Ereignisse wieder angezeigt.

### **Schaltfläche "Einstellungen als Standard übernehmen"**

Wenn Sie auf diese Schaltfläche klicken, gelten die Einstellungen auch bei künftigen Aufrufen des Registers "Ereignisse".

### **Optionskästchen "Ereignisinformation hexadezimal ausgeben"**

Wenn Sie das Optionskästchen aktivieren, wird in der Ereignisliste des Bereichs "Ereignisse" die Ereignis-ID hexadezimal angegeben. Wenn Sie das Optionskästchen deaktivieren, wird die Ereignisinformation in Textform angegeben.

## **Siehe auch**

Grundlagen zum Diagnosepuffer (Seite 1009)

## **Service­daten speichern**

### **Zweck**

Im Servicefall kann es notwendig sein, dass der Customer Support von SIEMENS zu Diagnosezwecken sehr spezielle Informationen über den Zustand einer Baugruppe Ihrer Anlage benötigt.

Wenn ein solcher Fall in Ihrer Anlage auftritt, werden Sie vom Customer Support gebeten, die Service­daten der Baugruppe zu speichern und ihm die erstellte Datei zukommen zu lassen.

### **Wo führen Sie das Speichern von Service­daten einer Baugruppe durch?**

Das Speichern von Service­daten einer Baugruppe führen Sie in deren Online- und Diagnosesicht an der folgenden Stelle durch: im Ordner "Funktionen" in der Gruppe "Service­daten speichern"

Die Gruppe "Servicedaten speichern" besteht aus folgenden Bereichen:

- Online-Daten
- Servicedaten speichern

### **Bereich "Online-Daten"**

Dieser Bereich zeigt die folgenden Daten der Baugruppe an:

- Bestellnummer
- Firmware-Stand
- Baugruppenname (Diesen haben Sie bei der Konfiguration der Hardware projiziert.)
- Baugruppenträger
- Steckplatz

### **Bereich "Service-Daten speichern"**

Um eine Datei mit speziellen Servicedaten zu erstellen und zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie die Stelle im Dateisystem, an der Sie die Datei speichern wollen:
  - Sie benutzen den im Feld "Pfad" voreingestellten Pfad.
  - Sie klicken auf die Schaltfläche mit den drei Punkten. Im sich öffnenden Dialog geben Sie den gewünschten Pfad vor und legen den Dateinamen fest.
2. Klicken Sie auf die Schaltfläche "Daten speichern".

## **8.2.1.5 Eigenschaften einer Baugruppe bzw. des PG/PC ändern**

### **Den Betriebszustand einer CPU umschalten**

#### **Voraussetzung**

Es besteht eine Online-Verbindung zur CPU, deren Betriebszustand Sie umschalten wollen.

## Vorgehen

Um den Betriebszustand einer CPU umzuschalten, gehen Sie wie folgt vor:

1. Aktivieren Sie die Task Card "Online Tools" der CPU.
2. Klicken Sie in der Palette "CPU-Bedienpanel" auf die Schaltfläche "RUN", wenn Sie die CPU in den Betriebszustand RUN setzen wollen, bzw. auf die Schaltfläche "STOP", wenn Sie die CPU in den Betriebszustand STOP setzen wollen.

---

### **Hinweis**

Es ist nur diejenige Schaltfläche aktiv, die Sie im aktuellen Betriebszustand der CPU anwählen können.

---

3. Beantworten Sie die Sicherheitsabfrage mit "OK".

Oder:

1. Öffnen Sie das Menü "Online".
2. Wählen Sie den Menübefehl "CPU starten", wenn Sie die CPU in den Betriebszustand RUN setzen wollen, bzw. "CPU stoppen", wenn Sie die CPU in den Betriebszustand STOP setzen wollen.

---

### **Hinweis**

Es ist nur derjenige Menübefehl aktiv, den Sie im aktuellen Betriebszustand der CPU wählen können.

---

3. Beantworten Sie die Sicherheitsabfrage mit "OK".

Oder:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "CPU starten", wenn Sie die CPU in den Betriebszustand RUN setzen wollen, bzw. "CPU stoppen", wenn Sie die CPU in den Betriebszustand STOP setzen wollen.

---

### **Hinweis**

Es ist nur diejenige Schaltfläche aktiv, die Sie im aktuellen Betriebszustand der CPU wählen können.

---

2. Beantworten Sie die Sicherheitsabfrage mit "OK".

## Ergebnis

Die CPU wird in den gewünschten Betriebszustand umgeschaltet.

## Urlöschen durchführen

### Voraussetzung

- Es besteht eine Online-Verbindung zur CPU, auf der Urlöschen durchgeführt werden soll.
- Diese CPU befindet sich im Betriebszustand STOP.

---

#### Hinweis

Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Urlöschens nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

---

### Vorgehen

Um auf einer CPU Urlöschen durchzuführen, gehen Sie wie folgt vor:

1. Aktivieren Sie die Task Card "Online Tools" der CPU.
2. Klicken Sie in der Palette "CPU-Bedienpanel" auf die Schaltfläche "MRES".
3. Beantworten Sie die Sicherheitsabfrage mit "OK".

### Ergebnis

Die CPU wird ggf. in den Betriebszustand STOP gesetzt, und auf der CPU wird Urlöschen durchgeführt.

### Siehe auch

Grundlagen zum Urlöschen (Seite 839)

## Die Uhrzeit einer CPU ermitteln und einstellen

### Wo befinden sich die gesuchten Funktionen?

Die Ermittlung und die Änderung der Uhrzeit einer CPU führen Sie in deren Online- und Diagnosesicht im Ordner "Funktionen" in der Gruppe "Uhrzeit einstellen" durch. Dies ist nur bei einer bestehenden Online-Verbindung möglich.

### Aufbau der Gruppe "Uhrzeit einstellen"

Die Gruppe "Uhrzeit einstellen" besteht aus folgenden Bereichen:

- Bereich zum Auslesen und Einstellen der Uhrzeit
- Zeitsystem (Diesen Bereich gibt es nicht bei S7-1200. Er wird hier nicht betrachtet.)



## Aufbau des Bereichs zum Auslesen und Einstellen der Uhrzeit

Dieser Bereich besteht aus den folgenden Teilen:

- PG / PC-Zeit  
Hier werden die eingestellte Zeitzone, das aktuelle Datum und die aktuelle Uhrzeit Ihres PG / PC angezeigt.
- Baugruppenzeit  
Hier werden die aktuell von der Baugruppe (z. B. CPU) gelesenen und auf Lokalzeit umgerechneten Werte von Datum und Uhrzeit angezeigt.  
Bei aktiviertem Optionskästchen "Von PG / PC übernehmen" bewirkt das Klicken auf die Schaltfläche "Übernehmen" die Übertragung des Datums und der auf UTC-Zeit umgerechneten PG / PC-Zeit auf die Baugruppe.  
Bei deaktiviertem Optionskästchen "Von PG / PC übernehmen" können Sie das Datum und die Uhrzeit für die integrierte Uhr der Baugruppe vorgeben. Nach Klicken auf die Schaltfläche "Übernehmen" werden das Datum und die auf UTC-Zeit umgerechnete Uhrzeit auf die Baugruppe übertragen.

## Firmware einer Baugruppe aktualisieren

### Firmware-Update durchführen

Mit Hilfe von Firmware-Dateien können Sie die Firmware einer Baugruppe aktualisieren.

### Voraussetzungen

- Die Baugruppe ist online verbunden.
- Die Baugruppe unterstützt ein Firmware-Update.
- Für diejenigen Module, die zur korrekten Durchführung des Firmware-Updates eine anliegende Versorgungsspannung benötigen: Die Versorgungsspannung des Moduls ist sichergestellt. Genaueres entnehmen Sie bitte der Dokumentation des Moduls.

### Vorgehen

Um ein Firmware-Update durchzuführen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Baugruppe in der Online- und Diagnosesicht.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Firmware-Update".

---

#### **Hinweis**

Bei S7-1500-CPUs ist diese Gruppe unterteilt in "PLC" und "Display".

---

3. Klicken Sie im Bereich "Firmware-Lader" auf die Schaltfläche "Durchsuchen", um den Pfad zu den Firmware-Update-Dateien zu wählen.
4. Wählen Sie eine dieser Dateien aus. In der Tabelle werden dann alle Baugruppen aufgelistet, für die mit der gewählten Firmware-Datei ein Update möglich ist.

5. Optional: Aktivieren Sie das Optionskästchen "Aktiviere Firmware nach Aktualisierung", um die Baugruppe nach dem Ladevorgang automatisch zurückzusetzen und die neue Firmware zu starten.
6. Klicken Sie auf die Schaltfläche "Starte Aktualisierung". Wenn die ausgewählte Datei von der Baugruppe interpretiert werden kann, wird sie in die Baugruppe geladen. Falls dazu der Betriebszustand der CPU geändert werden muss, werden Sie über Dialoge dazu aufgefordert.



**WARNUNG**

**Unzulässige Anlagenzustände möglich**

Eine S7-1500-CPU geht beim Start der Firmware-Aktualisierung sofort in den Betriebszustand STOP, was sich auf den Betrieb eines Online-Prozesses oder einer Maschine auswirken kann.

Unerwarteter Betrieb eines Prozesses oder einer Maschine kann zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen.

**Hinweis**

Nach der Durchführung eines Firmware-Updates müssen Sie in der Hardware-Konfiguration Ihres Projekts die betroffene Baugruppe durch die gleiche Baugruppe mit dem aktuellem Firmware-Stand ersetzen. Dann entspricht die projektierte Konfiguration wieder der tatsächlich vorhandenen Konfiguration.

**Optionskästchen "Firmware nach Aktualisierung aktivieren"**

Wenn Sie das Optionskästchen "Aktiviere Firmware nach Aktualisierung" nicht aktiviert haben, bleibt die bisherige Firmware so lange aktiv, bis die Baugruppe zurückgesetzt wird (z. B. durch einen Übergang NETZ AUS - NETZ EIN). Erst nach dem Rücksetzen der Baugruppe wird die neue Firmware aktiv.

Wenn Sie das Optionskästchen aktiviert haben, wird die Baugruppe nach erfolgreichem Ladevorgang automatisch zurückgesetzt und läuft anschließend mit der neuen Firmware.

Die Aktivierung der Firmware nach der Aktualisierung hat die folgenden Konsequenzen:

- Eine Station führt einen Neustart durch. Dieser hat den Ausfall sämtlicher Module der Station zur Folge.
- Wenn die zugehörige CPU im Betriebszustand RUN ist, kann die Aktivierung der Firmware zu Zugriffsfehlern oder anderen Beeinträchtigungen des Anwenderprogramms führen bis hin zum dauerhaften STOP der CPU.

---

**Hinweis**

Bei einigen CPUs ist das Optionskästchen "Aktiviere Firmware nach Aktualisierung" gegraut dargestellt und deaktiviert. In diesem Fall müssen Sie die CPU manuell neu starten.

Bei S7-1500-CPU's ist das Optionskästchen "Aktiviere Firmware nach Aktualisierung" gegraut dargestellt und aktiviert. In diesem Fall wird die neue Firmware unmittelbar nach dem Ladevorgang aktiviert.

---

**Siehe auch**

Hardware-Komponente tauschen (Seite 408)

**Eine S7-1200-CPU auf Werkseinstellungen zurücksetzen**

**Voraussetzung**

- Es steckt keine Memory Card in der CPU.
- Es besteht eine Online-Verbindung zur CPU, die auf Werkseinstellungen zurückgesetzt werden soll.
- Diese CPU befindet sich im Betriebszustand STOP.

---

**Hinweis**

Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Rücksetzvorgangs nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

---

**Vorgehen**

Um eine S7-1200-CPU auf Werkseinstellungen zurückzusetzen, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU.
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Rücksetzen auf Werkseinstellungen".

3. Aktivieren Sie das Optionsfeld "IP-Adresse beibehalten", falls Sie die IP-Adresse beibehalten wollen, bzw. das Optionsfeld "IP-Adresse löschen", wenn Sie die IP-Adresse löschen wollen.

---

**Hinweis**

Die beiden genannten Optionsfelder sind nur dann vorhanden, wenn die zurückzusetzende Baugruppe die Wahlmöglichkeit beherrscht, die IP-Adresse beizubehalten oder zu löschen.

---

4. Klicken Sie auf die Schaltfläche "Rücksetzen".
5. Beantworten Sie die Sicherheitsabfrage mit "OK".

## Ergebnis

Die Baugruppe wird ggf. in den Betriebszustand STOP gesetzt und auf Werkseinstellungen zurückgesetzt. Das bedeutet:

- Der Arbeitsspeicher und der interne Ladespeicher und alle Operandenbereiche werden gelöscht.
- Alle Parameter werden auf ihre Voreinstellung zurückgesetzt.
- Der Diagnosepuffer wird gelöscht.
- Die Uhrzeit wird zurückgesetzt.
- Die IP-Adresse wird beibehalten oder gelöscht, je nachdem, welche Einstellung Sie getroffen haben.

## Eine S7-1500-CPU auf Werkseinstellungen zurücksetzen

### Voraussetzung

- Wenn Sie das Rücksetzen auf Werkseinstellungen aus dem Projektkontext aufrufen, muss eine Online-Verbindung zur betroffenen CPU vorhanden sein.
- Die betroffene CPU befindet sich im Betriebszustand STOP.

---

**Hinweis**

Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Rücksetzvorgangs nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

---

### Vorgehen

Um eine S7-1500-CPU auf Werkseinstellungen zurückzusetzen, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU (entweder aus dem Projektkontext oder über "Erreichbare Teilnehmer").
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Rücksetzen auf Werkseinstellungen".

3. Aktivieren Sie das Optionsfeld "IP-Adresse beibehalten", falls Sie die IP-Adresse beibehalten wollen, bzw. das Optionsfeld "IP-Adresse löschen", wenn Sie die IP-Adresse löschen wollen.

---

**Hinweis**

Bei "IP-Adresse löschen" werden alle IP-Adressen gelöscht. Dies gilt unabhängig davon, wie Sie die Online-Verbindung hergestellt haben.

Wenn eine Memory Card steckt, bewirkt die Aktivierung des Optionsfelds "IP-Adresse löschen" das Folgende: Die IP-Adressen werden gelöscht, und die CPU wird auf Werkseinstellungen zurückgesetzt. Anschließend wird die auf der Memory Card gespeicherte Konfiguration (einschließlich IP-Adresse) in die CPU übertragen (siehe unten). Wurde die Memory Card vor dem Rücksetzen auf Werkseinstellungen formatiert oder ist sie leer, wird keine IP-Adresse in die CPU übertragen.

---

4. Klicken Sie auf die Schaltfläche "Rücksetzen".
5. Beantworten Sie die Sicherheitsabfrage mit "OK".

**Ergebnis**

Die Baugruppe wird ggf. in den Betriebszustand STOP gesetzt und auf Werkseinstellungen zurückgesetzt. Das bedeutet:

- Der Arbeitsspeicher und der interne remanente Systemspeicher und alle Operandenbereiche werden gelöscht.
- Alle Parameter werden auf ihre Voreinstellung zurückgesetzt.
- Der Diagnosepuffer wird gelöscht.
- Die Uhrzeit wird zurückgesetzt.
- Die I&M-Daten werden mit Ausnahme der I&M0-Daten gelöscht.
- Die Betriebsstundenzähler werden zurückgesetzt.
- Die IP-Adresse wird beibehalten oder gelöscht, je nachdem, welche Einstellung Sie getroffen haben.
- Wenn vor dem Rücksetzen auf Werkseinstellungen eine Memory Card gesteckt war, wird die auf der Memory Card enthaltene Konfiguration (Hardware und Software) in die CPU geladen.

## Eine S7-1500-Memory Card formatieren

### Voraussetzung

- Wenn Sie die Formatierung der Memory Card aus dem Projektkontext aufrufen, muss eine Online-Verbindung zur betroffenen CPU vorhanden sein.
- Die betroffene CPU befindet sich im Betriebszustand STOP.

---

#### Hinweis

Falls sich die CPU noch im Betriebszustand RUN befindet, können Sie sie beim Anstoßen des Formatiervorgangs nach positiver Beantwortung einer Sicherheitsabfrage in den Betriebszustand STOP setzen.

---

### Vorgehen

Um eine S7-1500-Memory Card zu formatieren, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU (entweder aus dem Projektkontext oder über "Erreichbare Teilnehmer").
2. Wählen Sie im Ordner "Funktionen" die Gruppe "Memory Card formatieren".
3. Klicken Sie auf die Schaltfläche "Formatieren".
4. Beantworten Sie die Sicherheitsabfrage mit "Ja".

### Ergebnis

- Die Memory Card wird formatiert.
- Die CPU ist vorübergehend nicht erreichbar.
- Die Projektdaten auf der CPU werden mit Ausnahme der IP-Adresse gelöscht.
- Wenn Sie die Formatierung der Memory Card aus dem Projektkontext aufrufen, bleibt die Online- und Diagnosesicht geöffnet, beim Aufruf über "Erreichbare Teilnehmer" wird sie geschlossen.

## Einem PROFINET IO-Device eine IP-Adresse zuweisen

### Grundsätzliches zur Vergabe einer IP-Adresse an ein PROFINET IO-Device

#### Übersicht

Alle PROFINET IO-Devices beherrschen das TCP/IP-Protokoll und benötigen daher für den Betrieb am Industrial Ethernet eine IP-Adresse. Nachdem ein IO-Device eine IP-Adresse erhalten hat, ist es über diese Adresse erreichbar. Sie können dann beispielsweise Projektdaten laden oder eine Diagnose durchführen.

### Voraussetzung

- Der Anschluss zum Ethernet LAN muss hergestellt sein.
- Die Ethernet-Schnittstelle ihres PG/PC muss erreichbar sein.
- Das IO-Device, das eine IP-Adresse erhalten soll, muss sich in demselben IP-Band befinden wie das PG/PC.

### Die Adresszuweisung über "Erreichbare Teilnehmer" aufrufen

#### Voraussetzung

- Sie haben die Online- und Diagnosesicht des PROFINET IO-Device über "Erreichbare Teilnehmer aktualisieren" (in der Projektnavigation) oder über "Erreichbare Teilnehmer..." (Menü "Online") aufgerufen.

#### Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "IP-Adresse zuweisen". Im Feld "MAC-Adresse" wird die MAC-Adresse des PROFINET IO-Devices angezeigt. Die Schaltfläche "Erreichbare Teilnehmer" ist gegraut.
2. Tragen Sie die gewünschte IP-Adresse ein.
3. Tragen Sie die Subnetzmaske ein.
4. Falls ein Router verwendet werden soll, aktivieren Sie das Optionskästchen "Router verwenden" und tragen anschließend dessen IP-Adresse ein.
5. Klicken Sie auf die Schaltfläche "IP-Adresse zuweisen".

#### Ergebnis

Dem IO-Device bzw. der betroffenen PROFINET-Schnittstelle des IO-Device wird die IP-Adresse dauerhaft zugewiesen. Sie bleibt auch über einen Anlauf oder über einen Spannungsausfall hinaus erhalten.

---

#### Hinweis

Bei einer S7-1500-CPU können Sie über den oben beschriebenen Weg die IP-Adresse einer PROFINET-Schnittstelle auch dann ändern, wenn über diese Schnittstelle auf die CPU bereits ein Projekt geladen wurde. Die über das Projekt geladene IP-Adresse wird dadurch überschrieben.

---

#### Siehe auch

Remanenz von IP-Adressparametern und Gerätenamen (Seite 805)

## Die Adresszuweisung aus dem Projektkontext aufrufen

### Voraussetzung

- Es besteht eine Online-Verbindung zum PROFINET IO-Device.
- Sie haben die Online- und Diagnosesicht des PROFINET IO-Device aus dem Projektkontext aufgerufen.
- Das PROFINET IO-Device ist keinem IO-Controller zugeordnet.

### Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "IP-Adresse zuweisen".
2. Klicken Sie auf die Schaltfläche "Erreichbare Teilnehmer", um die erreichbaren Geräte zu ermitteln.  
Hinweis: Bei einer S7-1500-CPU gibt es hier zwei Einträge, da sie zwei PROFINET-Schnittstellen besitzt.
3. Wählen Sie das IO-Device aus. Die Felder "IP-Adresse", "Subnetzmaske", das Optionskästchen "Router verwenden" und das Feld "Routeradresse" sind gegraut und enthalten die Knoteneigenschaften, über die Sie den aktuellen Online-Zugang hergestellt haben.
4. Klicken Sie auf die Schaltfläche "IP-Adresse zuweisen".

### Ergebnis

Dem IO-Device bzw. der betroffenen PROFINET-Schnittstelle des IO-Device wird die IP-Adresse dauerhaft zugewiesen. Sie bleibt auch über einen Anlauf oder über einen Spannungsausfall hinaus erhalten.

### Siehe auch

Remanenz von IP-Adressparametern und Gerätenamen (Seite 805)

## PROFINET-Gerätename vergeben

### Grundsätzliches zur Vergabe eines Namens an ein PROFINET IO-Device

#### Gerätename

Bevor ein IO-Device von einem IO-Controller angesprochen werden kann, muss es einen Gerätenamen haben. Bei PROFINET hat man diese Vorgehensweise gewählt, weil Namen einfacher zu handhaben sind als komplexe IP-Adressen.

Das Zuweisen eines Gerätenamens an ein PROFINET IO-Device ist mit dem Einstellen der PROFIBUS-Adresse bei einem DP-Slave vergleichbar.



Im Auslieferungszustand hat ein IO-Device keinen Gerätenamen. Erst nach der Zuweisung eines Gerätenamens mit dem PG/PC ist ein IO-Device für einen IO-Controller adressierbar, z. B. für die Übertragung der Projektierungsdaten (u. a. die IP-Adresse) im Anlauf oder für den Nutzdatenaustausch im zyklischen Betrieb.

### Regeln für den Gerätenamen

Der Gerätename unterliegt folgenden Einschränkungen:

- Beschränkung auf 240 Zeichen insgesamt (Kleinbuchstaben, Ziffern, Bindestrich oder Punkt)
- Ein Namensbestandteil innerhalb des Gerätenamens, d. h. eine Zeichenkette zwischen zwei Punkten, darf maximal 63 Zeichen lang sein.
- Keine Sonderzeichen wie Umlaute, Klammern, Unterstrich, Schrägstrich, Blank etc. Der Bindestrich ist das einzige erlaubte Sonderzeichen.
- Der Gerätename darf nicht mit dem Zeichen "-" beginnen und auch nicht mit diesem Zeichen enden.
- Der Gerätename darf nicht mit Ziffern beginnen.
- Der Gerätename darf nicht die Form n.n.n.n haben (n = 0, ... 999).
- Der Gerätename darf nicht mit der Zeichenfolge "port-xyz" oder "port-xyz-abcde" beginnen (a, b, c, d, e, x, y, z = 0, ... 9).

### Wo befindet sich die gesuchte Funktion?

Die Namensvergabe an ein PROFINET IO-Device führen Sie in dessen Online- und Diagnosesicht im Ordner "Funktionen" in der Gruppe "Name zuweisen" durch. Die Benutzeroberfläche dieser Gruppe hängt davon ab, wie Sie die Online- und Diagnosesicht aufrufen:

- Aufruf über "Erreichbare Teilnehmer"
- Aufruf aus dem Projektkontext

### Siehe auch

Die Namensvergabe über "Erreichbare Teilnehmer" aufrufen (Seite 1004)

Die Namensvergabe aus dem Projektkontext aufrufen (Seite 1004)

## Die Namensvergabe über "Erreichbare Teilnehmer" aufrufen

### Voraussetzung

- Sie haben die Online- und Diagnosesicht des PROFINET IO-Device über "Erreichbare Teilnehmer aktualisieren" (in der Projektnavigation) oder über "Erreichbare Teilnehmer..." (Menü "Online") aufgerufen.

### Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "Name zuweisen". Im Feld "Typ" wird der Baugruppentyp des PROFINET IO-Devices angezeigt.
2. Geben Sie im Eingabefeld "PROFINET-Gerätename" den gewünschten Gerätenamen ein.
3. Optional: Aktivieren Sie das Optionskästchen "LED blinken", um einen LED-Blinktest auf dem PROFINET IO-Device durchzuführen. Damit überprüfen Sie, ob Sie sich auf dem gewünschten IO-Device befinden.

---

#### Hinweis

Der LED-Blinktest wird nicht von allen PROFINET IO-Devices unterstützt.

---

Der LED-Blinktest läuft so lange, bis sie ihn abbrechen. Dies geschieht z. B. durch Deaktivieren des Optionskästchens "LED blinken", durch Selektieren eines anderen IO-Devices in der Tabelle, durch Schließen der Online- und Diagnosesicht.

4. Klicken Sie auf die Schaltfläche "Name zuweisen".

### Ergebnis

Dem PROFINET IO-Device wird der vorgegebene Name zugewiesen.

## Die Namensvergabe aus dem Projektkontext aufrufen

### Voraussetzung

- Eine Online-Verbindung zum PROFINET IO-Device ist nicht erforderlich.
- Sie haben die Online- und Diagnosesicht des PROFINET IO-Device aus dem Projektkontext aufgerufen.
- Das PROFINET IO-Device kann über mindestens eine PG/PC-Schnittstelle erreicht werden.

## Vorgehen

1. Öffnen Sie den Ordner "Funktionen" und darin die Gruppe "Name zuweisen". In der Klappliste "PROFINET-Gerätename" wird der aktuell im Offline-Projekt vorhandene Name und im Feld "Typ" der Baugruppentyp des PROFINET IO-Device angezeigt.

---

**Hinweis**

Bei S7-1500-CPUs werden die im Offline-Projekt vorhandenen Namen der beiden PROFINET-Schnittstellen angezeigt.

---

2. Wählen Sie aus der Klappliste ggf. einen anderen Namen.

---

**Hinweis**

In den Schritten 3 bis 5 ermitteln Sie die am PROFINET-Subnetz vorhandenen IO-Devices.

---

3. Wählen Sie in der Klappliste "PG/PC-Schnittstelle für die Zuweisung" diejenige PG/PC-Schnittstelle, über die Sie die Online-Verbindung herstellen wollen.
4. Optional: Treffen Sie über die drei Optionskästchen eine Auswahl unter allen online vorhandenen IO-Devices.
5. Klicken Sie auf das Symbol zur Ermittlung der am PROFINET-Subnetz vorhandenen IO-Devices. Anschließend wird die Tabelle aktualisiert.
6. Markieren Sie in der Tabelle das gewünschte IO-Device.
7. Optional: Aktivieren Sie das Optionskästchen "LED blinken", um einen LED-Blinktest auf dem PROFINET IO-Device durchzuführen. Damit überprüfen Sie, ob Sie sich auf dem gewünschten IO-Device befinden.

---

**Hinweis**

Der LED-Blinktest wird nicht von allen PROFINET IO-Devices unterstützt.

---

Der LED-Blinktest läuft so lange, bis sie ihn abbrechen. Dies geschieht z. B. durch Deaktivieren des Optionskästchens "LED blinken", durch Selektieren eines anderen IO-Devices in der Tabelle, durch Schließen der Online- und Diagnosesicht.

8. Klicken Sie auf die Schaltfläche "Name zuweisen".

## Ergebnis

Dem PROFINET IO-Device wird der ausgewählte Name zugewiesen.

---

**Hinweis**

Bei S7-1500-CPU wird der oben ausgewählten PROFINET-Schnittstelle der ausgewählte Name zugewiesen.

---

## Ein S7-1500-Analogmodul kalibrieren

### Ein S7-1500-Analogmodul kalibrieren - Überblick

#### Wo kalibrieren Sie ein S7-1500-Analogmodul?

Die Kalibrierung eines S7-1500-Analogmoduls führen Sie in dessen Online- und Diagnosesicht im Ordner "Funktionen" in der Gruppe "Kalibrieren" durch.

#### Übersicht über den Funktionsumfang der Kalibrierfunktion

In der Gruppe "Kalibrieren" können Sie die folgenden Funktionen bei einem S7-1500-Analogmodul durchführen:

- Feststellen der aktuellen Kalibrierung aller Kanäle
- Kalibrieren eines Kanals
- Abbrechen eines laufenden Kalibriervorgangs
- Rücksetzen der Kalibrierung eines Kanals auf die Werkseinstellungen

#### Voraussetzung für die im Folgenden beschriebene Kalibrierfunktion

Für die im Folgenden beschriebene Kalibrierfunktion wird das Folgende vorausgesetzt:

- Sie haben die Online- und Diagnosesicht aus dem Projektkontext aufgerufen (also nicht aus der Projektnavigation oder über das Menü "Online").
- Es besteht eine Online-Verbindung zum Analogmodul, das kalibriert werden soll.
- Offline- und Online-Projektierung sind identisch.

## Ein S7-1500-Analogmodul kalibrieren

### Übersicht über die Kalibrierung eines Kanals eines S7-1500-Analogmoduls

Die Kalibrierung eines Kanals eines S7-1500-Analogmoduls besteht aus folgenden Schritten:

1. Kalibriervorgang starten
2. Zweiten bis vorletzten Schritt des Kalibriervorgangs durchführen
3. Kalibriervorgang abschließen

Darauf wird im Folgenden näher eingegangen.

## Voraussetzung

- Sie haben die Online- und Diagnosesicht des S7-1500-Analogmoduls aus dem Projektkontext aufgerufen und befinden sich im Ordner "Funktionen" in der Gruppe "Kalibrieren".
- Die zugehörige CPU ist Online.
- Aktuell läuft kein Kalibriervorgang auf dem Analogmodul (wenn Sie den Kalibriervorgang starten wollen) bzw. der zuletzt angestoßene Schritt wurde erfolgreich durchgeführt (wenn Sie den Kalibriervorgang fortsetzen oder abschließen wollen).

## Vorgehen für den Start des Kalibriervorgangs

Um den Kalibriervorgang zu starten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Übersichtstabelle diejenige Zeile, die zum zu kalibrierenden Kanal gehört.
2. Klicken Sie auf die Schaltfläche "Manuelle Kalibrierung starten".

Anschließend ändert sich die Bedienoberfläche wie folgt:

- Die Übersichtstabelle und die Schaltflächen "Manuelle Kalibrierung starten" und "Auf Werkseinstellungen setzen" werden inaktiv.
- Die Schrittanzeige wird aktiviert und zeigt die Nummer des aktuellen und des letzten Schritts an.
- Das Feld "Befehl" wird aktiv und zeigt an, was der Anwender im nächsten Kalibrierungsschritt zu tun hat.
- Das Feld "Status" wird aktiv und zeigt den aktuellen Zustand im Kalibrierungsvorgang an, z. B. "Kalibrierung erfolgreich gestartet".
- Das Feld "Messwert" wird aktiv. Bei einem Eingabemodul wird hier ein Wert angezeigt, bei einem Ausgabemodul müssen Sie hier einen Wert eingeben.
- Die Schaltfläche "Abbrechen" wird aktiv.
- Die Schaltfläche "Weiter" wird aktiv. Damit kann der nächste Schritt im Kalibriervorgang durchgeführt werden.

## Vorgehen für den zweiten bis vorletzten Schritt des Kalibriervorgangs

Sie gehen folgendermaßen vor:

1. Klicken sie auf die Schaltfläche "Weiter".

Anschließend werden die oben beschriebenen Felder der Bedienoberfläche aktualisiert.

## Vorgehen für den letzten Schritt des Kalibriervorgangs

Sie gehen folgendermaßen vor:

1. Klicken sie auf die Schaltfläche "Weiter".

Anschließend ändert sich die Bedienoberfläche wie folgt:

- Die Übersichtstabelle wird aktiv.
- Die Kalibrierungsanzeige des kalibrierten Kanals wird aktualisiert.
- Die Schaltflächen "Manuelle Kalibrierung starten" und "Auf Werkseinstellungen setzen" werden aktiv.
- Die Schrittanzeige wird deaktiviert und die Nummern des aktuellen und des letzten Schritts sind leer.
- Das Feld "Befehl" wird inaktiv und ist leer.
- Das Feld "Status" wird inaktiv und zeigt den letzten Zustand im Kalibrierungsvorgang an, z. B. "Kalibrierung erfolgreich beendet".
- Das Feld "Messwert" wird inaktiv und ist leer.
- Die Schaltfläche "Abbrechen" wird inaktiv.
- Die Schaltfläche "Weiter" wird inaktiv.

### Auftreten eines Fehlers

Wenn während des Kalibriervorgangs ein Fehler auftritt, bricht das Modul den Kalibriervorgang ab. Der Kanal, der kalibriert werden sollte, hat danach dieselben Einstellungen wie vor Beginn des Kalibriervorgangs.

Die Bedienoberfläche sieht nach dem Auftreten eines Fehlers mit Ausnahme des Felds "Status" aus wie vor Beginn des Kalibriervorgangs. Das Feld "Status" zeigt den Fehler an, den das Modul beim Kalibriervorgang erkannt hat.

### Einen laufenden Kalibriervorgang eines S7-1500-Analogmoduls abbrechen

#### Voraussetzung

- Sie haben die Online- und Diagnosesicht des S7-1500-Analogmoduls aus dem Projektkontext aufgerufen und befinden sich im Ordner "Funktionen" in der Gruppe "Kalibrieren".
- Die zugehörige CPU ist Online.
- Aktuell läuft ein Kalibriervorgang auf dem Analogmodul.

#### Vorgehen

Um einen laufenden Kalibriervorgang abzubrechen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche "Abbrechen".

#### Ergebnis

Der laufende Kalibriervorgang wird abgebrochen, und der zu kalibrierende Kanal hat danach dieselben Einstellungen wie vor Beginn des Kalibriervorgangs.

In der Bedienoberfläche werden alle Bedienelemente deaktiviert, bis der Abbruch abgeschlossen ist. Die Bedienoberfläche sieht danach mit Ausnahme des Felds "Status" aus wie vor Beginn des Kalibriervorgangs. Das Feld "Status" zeigt das Ergebnis des Abbruchs an.

## Ein S7-1500-Analogmodul auf Werkseinstellungen zurücksetzen

### Voraussetzung

- Sie haben die Online- und Diagnosesicht des S7-1500-Analogmoduls aus dem Projektkontext aufgerufen und befinden sich im Ordner "Funktionen" in der Gruppe "Kalibrieren".
- Die zugehörige CPU ist Online.

### Vorgehen

Um einen Kanal eines S7-1500-Analogmoduls auf Werkseinstellungen zurückzusetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Übersichtstabelle die zum rückzusetzenden Kanal gehörige Zeile.
2. Klicken Sie auf die Schaltfläche "Auf Werkseinstellungen setzen".

### Ergebnis

In der Bedienoberfläche werden alle Bedienelemente deaktiviert, bis der Rücksetzvorgang abgeschlossen ist. Die Bedienoberfläche sieht danach mit Ausnahme des Felds "Status" aus wie vor Beginn des Rücksetzvorgangs. Das Feld "Status" zeigt das Ergebnis des Rücksetzvorgangs an.

## 8.2.1.6 Diagnose im Betriebszustand STOP

### Grundlagen zum Diagnosepuffer

#### Funktion

Das Betriebssystem der CPU trägt die von der CPU und diagnosefähigen Baugruppen erkannten Fehler in der Reihenfolge ihres Auftretens in den Diagnosepuffer ein. Dazu gehören u. a. die folgenden Ereignisse:

- Jeder Betriebszustandswechsel der CPU (z. B. NETZEIN, Übergang in den Betriebszustand STOP, Übergang in den Betriebszustand RUN)
- Jeder Prozess- und jeder Diagnosealarm

Der Eintrag an oberster Stelle enthält das neueste Ereignis. Die Einträge im Diagnosepuffer werden dauerhaft gespeichert: Sie bleiben auch beim Ausfall der Spannungsversorgung erhalten und können nur durch Zurücksetzen der CPU auf Werkseinstellungen gelöscht werden.

Ein Diagnosepuffereintrag enthält die folgenden Elemente:

- Zeitstempel
- Fehler-ID
- Zusätzliche Fehler-ID-spezifische Information

### Vorteile des Diagnosepuffers

Der Diagnosepuffer bietet folgende Vorteile:

- Nachdem die CPU in den Betriebszustand STOP gegangen ist, können Sie die letzten Ereignisse vor STOP auswerten und so die STOP-Ursache eingekreisen bzw. ermitteln.
- Sie können Fehlerursachen schneller erkennen und beheben und dadurch die Verfügbarkeit der Anlage erhöhen.
- Sie können das dynamische Anlagenverhalten auswerten und anschließend optimieren.

### Organisation des Diagnosepuffers

Der Diagnosepuffer ist ein Ringpuffer. Die maximale Anzahl der Einträge ist bei den S7-1200-CPUs 50. Wenn der Diagnosepuffer voll ist und ein weiterer Eintrag erfolgen soll, werden alle vorhandenen Einträge um eine Stelle weitergerückt (mit der Folge, dass der älteste Eintrag gelöscht wird), und der neue Eintrag gelangt an die frei gewordene erste Stelle (FIFO-Prinzip: first in, first out).

### Auswertung des Diagnosepuffers

Sie haben folgende Möglichkeit, auf Inhalte des Diagnosepuffers zuzugreifen:

- Über die Online- und Diagnosesicht

Durch die Auswertung der Ereignisse vor dem Fehlerereignis (z. B. Betriebszustandsübergang nach STOP) verschaffen Sie sich ein Bild über die mögliche Ursache bzw. kreisen sie näher ein oder spezifizieren sie genauer (abhängig vom Fehlertyp).

Lesen Sie die Detailangaben zu den Ereignissen sorgfältig und nutzen Sie die Schaltfläche "Hilfe zum Ereignis", um weitere Informationen und mögliche Ursachen einzelner Einträge zu erhalten.

---

#### Hinweis

Um die Zeitangaben der Diagnosepuffer-Einträge bei zeitkritischen Anlagen sinnvoll nutzen zu können, wird empfohlen, die Uhrzeit und das Datum auf der CPU gelegentlich zu überprüfen und gegebenenfalls zu korrigieren.

Alternativ besteht die Möglichkeit der Uhrzeitsynchronisation über einen NTP-Zeitserver.

---



## Siehe auch

- Eine S7-1200-CPU auf Werkseinstellungen zurücksetzen (Seite 997)
- Die STOP-Ursache einer CPU ermitteln (Seite 1011)
- Die Uhrzeit einer CPU ermitteln und einstellen (Seite 994)
- Parametrieren der Uhr (Seite 863)

## Die STOP-Ursache einer CPU ermitteln

### Voraussetzung

Die zu analysierende CPU befindet sich im Betriebszustand STOP.

### Vorgehen

Um die STOP-Ursache einer CPU zu ermitteln, gehen Sie wie folgt vor:

1. Öffnen Sie die Online- und Diagnosesicht der CPU.
2. Wählen Sie im Ordner "Diagnose" die Gruppe "Diagnosepuffer".
3. Werten Sie die Ereignisse vor dem Übergang in den Betriebszustand STOP aus. Verschaffen Sie sich damit ein Bild über die mögliche Ursache bzw. kreisen Sie sie näher ein oder spezifizieren Sie sie genauer (abhängig vom Fehlertyp). Lesen Sie die Detailangaben zu den Ereignissen sorgfältig und nutzen Sie die Schaltfläche "Hilfe zum Ereignis", um weitere Informationen und mögliche Ursachen einzelner Einträge zu erhalten.

### Ergebnis

Sie konnten die STOP-Ursache der CPU näher eingekreisen bzw. genau ermitteln.

---

#### Hinweis

Wenn Sie bei der Analyse nicht weiterkommen, wenden Sie sich an den Customer Support. Sichern Sie in diesem Fall den Inhalt des Diagnosepuffers mit der Schaltfläche "Speichern unter" in eine Textdatei und stellen Sie diese dem Customer Support zur Verfügung.

---

## Siehe auch

- Den Diagnosepuffer einer CPU auslesen (Seite 987)

### 8.2.1.7 Online-Zugänge in der Online- und Diagnosesicht

#### Status der Online-Verbindung anzeigen

##### Voraussetzung

- Das zugehörige Gerät kann über mindestens eine PG/PC-Schnittstelle erreicht werden.

##### Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht desjenigen Geräts, dessen Status der Online-Verbindung angezeigt werden soll.
2. Wählen Sie die Gruppe "Online-Zugänge".

---

##### Hinweis

Die Gruppe "Online-Zugänge" existiert nur bei CPUs. Falls Sie die Online- und Diagnosesicht über die Funktion "Erreichbare Teilnehmer anzeigen / aktualisieren" aufgerufen haben, wird sie nicht angezeigt.

---

##### Ergebnis

Der Status der Online-Verbindung wird im Bereich "Status" sowohl grafisch als auch in Worten angezeigt.

#### PG/PC-Schnittstelle festlegen, Online verbinden

##### Voraussetzung

- Das zugehörige Gerät kann über mindestens eine PG/PC-Schnittstelle erreicht werden.
- Zum zugehörigen Gerät besteht momentan keine Online-Verbindung.

##### Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht desjenigen Geräts, zu dem Sie eine Online-Verbindung herstellen wollen.
2. Wählen Sie die Gruppe "Online-Zugänge" und darin den Bereich "Online-Zugänge".

---

##### Hinweis

Die Gruppe "Online-Zugänge" existiert bei CPUs und einigen CPs. Falls Sie die Online- und Diagnosesicht über die Funktion "Erreichbare Teilnehmer anzeigen / aktualisieren" aufgerufen haben, wird sie nicht angezeigt.

---

3. Wenn zum Gerät bereits zuvor eine Online-Verbindung bestanden hat, sind in den Klapplisten die zugehörigen Werte dieser Online-Verbindung voreingestellt. In diesem Fall können Sie sofort mit dem letzten Schritt dieser Handlungsanweisung fortfahren, sofern Sie zwischenzeitlich die IP-Adresse nicht über die Online- und Diagnosesicht geändert haben.
4. Wählen Sie in der Klappliste "Typ der PG/PC-Schnittstelle" den Schnittstellentyp. Abhängig von dieser Auswahl werden in der Klappliste "PG/PC-Schnittstelle für den Online-Zugang" nur noch diejenigen Schnittstellen des PG/PC angezeigt, die dem gewählten Schnittstellentyp entsprechen.
5. Wählen Sie in der Klappliste "PG/PC-Schnittstelle für den Online-Zugang" diejenige PG/PC-Schnittstelle, über die Sie die Online-Verbindung herstellen wollen.
6. Optional: Klicken Sie auf die Schaltfläche "Eigenschaften", um die Eigenschaften des zugehörigen CP zu verändern.
7. Wählen Sie in der Klappliste "Verbindung mit Subnetz" dasjenige Subnetz aus, über welches das Gerät mit der PG/PC-Schnittstelle verbunden ist.

---

**Hinweis**

Die PG/PC-Schnittstelle sei mit einer Schnittstelle eines Geräts verbunden.

Wenn Sie ausschließlich auf dieses Gerät zugreifen wollen, dann wählen Sie in der Klappliste die Einstellung "Direkt am Steckplatz <Schnittstellenbezeichnung>".

Wenn Sie hingegen mittels Routing auf ein anderes Gerät zugreifen wollen, dann legen Sie in der Hardware-Konfiguration an dieser Schnittstelle ein Subnetz an und wählen Sie anschließend in der Klappliste dieses Subnetz aus.

---

8. Falls das Gerät über ein Gateway erreichbar ist, wählen Sie aus der Klappliste "1. Gateway" dasjenige Gateway aus, das die betroffenen Subnetze miteinander verbindet.
9. Geben Sie im Eingabefeld "Geräteadresse" ggf. die IP-Adresse des Geräts ein, zu dem Sie eine Online-Verbindung herstellen wollen.

---

**Hinweis**

Bei CPUs mit mehreren IP-Adressen wählen Sie über die Klappliste "Geräteadresse" die IP-Adresse derjenigen PROFINET-Schnittstelle, über die Sie eine Online-Verbindung herstellen wollen.

---

10. Alternativ: Klicken Sie auf die Schaltfläche "Erreichbare Teilnehmer anzeigen" und wählen Sie das Gerät, zu dem Sie eine Online-Verbindung herstellen wollen, aus der Liste der erreichbaren Teilnehmer.
11. Klicken Sie auf die Schaltfläche "Online verbinden".

## Ergebnis

Die Online-Verbindung zum gewünschten Gerät wird aufgebaut.

## Online-Verbindung trennen

### Voraussetzung

- Zum zugehörigen Gerät besteht momentan eine Online-Verbindung.

### Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht desjenigen Geräts, dessen Online-Verbindung Sie trennen wollen.
2. Wählen Sie die Gruppe "Online-Zugänge" und darin den Bereich "Online-Zugänge".

---

#### Hinweis

Die Gruppe "Online-Zugänge" existiert nur bei CPUs. Falls Sie die Online- und Diagnosesicht über die Funktion "Erreichbare Teilnehmer anzeigen / aktualisieren" aufgerufen haben, wird sie nicht angezeigt.

---

3. Klicken Sie auf die Schaltfläche "Online-Verbindung trennen".

### Ergebnis

Die Online-Verbindung zum gewünschten Gerät wird getrennt.

## Den Blinktest bei einem online verbundenen Gerät durchführen

### Voraussetzung

- Zum zugehörigen Gerät besteht momentan eine Online-Verbindung.
- Die Funktion FORCEN ist nicht aktiv.

### Vorgehen

1. Öffnen Sie die Online- und Diagnosesicht desjenigen Geräts, bei dem Sie einen Blinktest durchführen wollen.
2. Wählen Sie die Gruppe "Online-Zugänge" und darin den Bereich "Status".

---

#### Hinweis

Die Gruppe "Online-Zugänge" existiert nur bei CPUs. Falls Sie die Online- und Diagnosesicht über die Funktion "Erreichbare Teilnehmer anzeigen / aktualisieren" aufgerufen haben, wird sie nicht angezeigt.

---

3. Aktivieren Sie das Optionskästchen "LED-Blinktest".

## Ergebnis

- Bei einer S7-1200-CPU blinken die LEDs RUN/STOP, ERROR und MAINT.
- Bei einer S7-1500-CPU blinken die LEDs RUN/STOP, ERROR und MAINT.
- Bei einer S7-300- oder S7-400-CPU blinkt die FRCE-LED.

Die LEDs blinken so lange, bis Sie den Blinktest abbrechen. Dies geschieht z. B. durch Deaktivieren des Optionskästchens "LED-Blinktest", durch Wechsel zu einer anderen Gruppe der Online- und Diagnosesicht, durch Änderung von Einstellungen im Bereich "Online-Zugänge".

### 8.2.1.8 PROFIBUS DP-Subnetze auf Störungen überprüfen

## Grundlagen zum Diagnose-Repeater

### Was ist der Diagnose-Repeater?

Der Diagnose-Repeater ist ein Repeater mit der Fähigkeit, ein Segment eines RS 485-PROFIBUS-Subnetzes (Kupferleitung) im laufenden Betrieb zu überwachen und Leitungsfehler per Diagnosetelegramm an den DP-Master zu melden.

Der Diagnose-Repeater ermöglicht durch seine Leitungsdiagnose im laufenden Betrieb, Leitungsfehler frühzeitig zu erkennen, zu lokalisieren und zu visualisieren. Damit werden Anlagenstörungen frühzeitig erkannt und Anlagenstillstände minimiert.

### Funktionsweise des Diagnose-Repeaters

Der Diagnose-Repeater kann an den Segmenten DP2 und DP3 eine Leitungsdiagnose durchführen, weil er für diese Segmente eine Messschaltung besitzt.

Die Leitungsdiagnose läuft in zwei Schritten ab:

- 1. Schritt: Topologieermittlung  
Sie starten die Topologieermittlung, indem Sie in Ihrem Programm die erweiterte Anweisung "DP\_TOPOL" aufrufen.  
Der Diagnose-Repeater ermittelt dabei die PROFIBUS-Adressen und die Entfernung der Teilnehmer und erstellt eine Topologietabelle.
- 2. Schritt: Störstellenermittlung  
Der Diagnose-Repeater prüft im laufenden Busbetrieb die Leitungen. Er ermittelt die Entfernung der Störstelle, bestimmt die Fehlerursache und setzt eine Diagnosemeldung mit relativer Angabe des Fehlerorts ab.

### Anzeige detaillierter Informationen zur ermittelten Störstelle

In der Online- und Diagnosesicht des Diagnose-Repeaters erhalten Sie detaillierte Informationen zur ermittelten Störstelle:

- Mittels Symbolen
- Mittels einer grafischen und einer textuellen Anzeige

### Siehe auch

Status der Segmentdiagnose mittels Symbolen anzeigen (Seite 1016)

Status der Segmentdiagnose grafisch und textuell anzeigen (Seite 1016)

### Status der Segmentdiagnose mittels Symbolen anzeigen

#### Wo befinden sich die gesuchten Informationen?




Die Symbole zum Status der Segmentdiagnose finden Sie an der folgenden Stelle:

- Im Navigationsfenster der Online- und Diagnosesicht des zugehörigen Diagnose-Repeater im aufgeklappten Ordner "Segmentdiagnose"

Das zum Segment gehörige Diagnosesymbol wird hinter der Segmentbezeichnung angezeigt. Dabei ist zu beachten, dass Leitungsfehler nur bei den Segmenten DP2 und DP3 angezeigt werden. Die Segmente DP1 und PG zeigen keine Fehler in Form eines Diagnosesymbols an, sondern melden nur wenige Busfehler.

### Diagnosesymbole

Die folgende Tabelle zeigt die möglichen Symbole und ihre Bedeutung.

| Symbol  | Bedeutung                |
|---|--------------------------|
|  | Segment ist fehlerfrei   |
|  | Segment enthält Fehler   |
|  | Segment ist abgeschaltet |

### Status der Segmentdiagnose grafisch und textuell anzeigen

#### Wo wird der Status der Segmentdiagnose grafisch und textuell angezeigt?

Die grafische und textuelle Anzeige des Status der Segmentdiagnose erfolgt in der Online- und Diagnosesicht des zugehörigen Diagnose-Repeater im Ordner "Segmentdiagnose" in den Gruppen "DP1", "DP2", "DP3" und "PG".

### Aufbau der Gruppen "DP1", "DP2" "DP3" und "PG"

Die Gruppen "DP1", "DP2", "DP3" und "PG" bestehen aus folgenden Elementen:

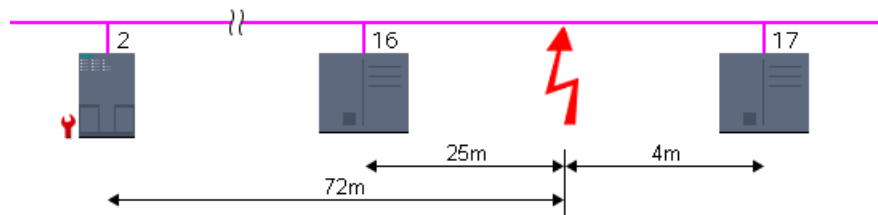
- Feld "Fehlerort"
- Feld "Fehler"
- Feld "Behebung"

- Schaltfläche "Hilfe zum Ereignis"
- Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben"

### Feld "Fehlerort"

Hier wird der Fehlerort grafisch dargestellt, sofern der Diagnose-Repeater den Ort ermitteln kann.

Das folgende Bild zeigt ein Beispiel für einen aufgetretenen Leitungsfehler im Segment DP2.



In diesem Beispiel hat der Diagnose-Repeater die PROFIBUS-Adresse 2, und zwischen den Teilnehmern mit den PROFIBUS-Adressen 16 und 17 ist ein Leitungsfehler aufgetreten. Dieser Leitungsfehler befindet sich 25 m vom Teilnehmer 16 entfernt, 4 m vom Teilnehmer 17 entfernt und 72 m vom Diagnose-Repeater entfernt.

### Feld "Fehler"

Hier wird der Fehler im Klartext erläutert.

### Feld "Behebung"

Hier finden Sie Maßnahmen zur Behebung des Fehlers.

### Schaltfläche "Hilfe zum Ereignis"

Wenn Sie auf diese Schaltfläche klicken, wird der aufgetretene Fehler näher erläutert und es werden ggf. weitere Einzelheiten zu seiner Behebung gegeben.

### Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben"

Die Schaltfläche "Anzeige einfrieren" bzw. "Einfrieren aufheben" ist nur aktiv, wenn eine Online-Verbindung zum Diagnose-Repeater besteht.

Die Voreinstellung ist "Anzeige einfrieren".

Wenn Sie auf die Schaltfläche "Anzeige einfrieren" klicken, geschieht Folgendes:

- Die aktuelle Anzeige der Segmentdiagnose wird eingefroren.
- Die Beschriftung der Schaltfläche wechselt zu "Einfrieren aufheben".

Wenn die Anzeige eingefroren ist und Sie auf die Schaltfläche "Einfrieren aufheben" klicken, geschieht Folgendes:

- Die Anzeige der Segmentdiagnose wird wieder aktualisiert.
- Die Beschriftung der Schaltfläche wechselt zu "Anzeige einfrieren".

## 8.2.2 Verbindungen diagnostizieren

### 8.2.2.1 Überblick über die Verbindungsdiagnose

#### Grundlagen

Unter Verbindungsdiagnose wird im Folgenden die Diagnose von Kommunikationsverbindungen verstanden.

Die Verbindungsdiagnose wird bei jedem Aufbau der Online-Verbindung zu einer Baugruppe (CPU oder CP), die an einem oder mehreren Kommunikationsdiensten beteiligt ist, gestartet. Aktualisierungen des Verbindungsstatus erfolgen automatisch im Hintergrund.

Bei einseitigen Verbindungen muss zu demjenigen Kommunikationspartner eine Online-Verbindung bestehen, der die Kommunikationsverbindung aufgebaut hat.

Bei zweiseitigen Verbindungen sind die folgenden beiden Fälle zu unterscheiden:

- Wenn zu genau einem Verbindungsendpunkt eine Online-Verbindung besteht, kann nur der zu diesem Verbindungsendpunkt gehörende Verbindungsteil diagnostiziert werden.
- Wenn zu beiden Verbindungsendpunkten eine Online-Verbindung besteht, können beide Verbindungsteile (und damit die gesamte Verbindung) diagnostiziert werden.

#### Prinzipielle Möglichkeiten der Verbindungsdiagnose

Sie können Verbindungen wie folgt diagnostizieren:

- Über die Anzeige des Verbindungsstatus mittels Symbolen  
Diese Anzeige erfolgt in der Verbindungstabelle.
- Durch eine detaillierte Verbindungsdiagnose  
Diese erfolgt im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters.

#### Voraussetzung für die im Folgenden beschriebene Verbindungsdiagnose

In der Verbindungstabelle können Sie sich die Details entweder aller im Projekt angelegten Kommunikationsverbindungen (Voreinstellung) oder auswahlbezogener Kommunikationsverbindungen anzeigen lassen.

Für die im Folgenden beschriebene Verbindungsdiagnose wird vorausgesetzt, dass Sie sich die Details auswahlbezogener Kommunikationsverbindungen anzeigen lassen. Deaktivieren Sie hierzu im Kontextmenü die Option "Alle Verbindungen anzeigen".



### 8.2.2.2 Verbindungsstatus mittels Symbolen anzeigen

#### Inhalt der Verbindungstabelle bei nicht aufgebauter Online-Verbindung

- Im Kontext einer CPU oder eines CP werden in der Verbindungstabelle bei nicht aufgebauter Online-Verbindung die offline projektierten Kommunikationsverbindungen (inkl. Eigenschaften) aufgelistet.

#### Inhalt der Verbindungstabelle bei aufgebauter Online-Verbindung




Nach dem Aufbau der Online-Verbindung werden die Eigenschaften der offline aufgelisteten Kommunikationsverbindungen um Diagnosesymbole für den Verbindungsstatus (Spalte "Online-Status") ergänzt.

Zusätzlich enthält die Verbindungstabelle jetzt auch Einträge zu allen Kommunikationsverbindungen, die nur online vorhanden sind (z. B. Verbindungen bei den Anweisungen zur Open User Communication, PG- und OP-Verbindungen, Verbindungen für den Webserver-Zugriff).

Bei Verbindungen, die ausschließlich online oder offline vorhanden sind, wird das Diagnosesymbol rechts unten mit einem kleineren Zusatzsymbol für den Vergleichsstatus kombiniert.



#### Diagnosesymbole für Kommunikationsverbindungen

Die folgende Tabelle zeigt die Diagnosesymbole für Kommunikationsverbindungen.

| Symbol  | Bedeutung                                   |
|---|---|
|  | Verbindung aufgebaut                        |
|  | Verbindung nicht aufgebaut / wird aufgebaut |
|  | Verbindung nicht verfügbar                  |

#### Diagnosesymbole für den Vergleichsstatus

Die Diagnosesymbole für Kommunikationsverbindungen können rechts unten mit kleineren Zusatzsymbolen kombiniert werden, die das Ergebnis des Online-/Offline-Vergleichs anzeigen. Die folgende Tabelle zeigt die möglichen Vergleichssymbole und ihre Bedeutung.

| Symbol  | Bedeutung                        |
|---|----------------------------------|
|  | Verbindung nur online vorhanden  |
|  | Verbindung nur offline vorhanden |

### 8.2.2.3 Verbindungen im Detail diagnostizieren

#### Verbindungen im Detail diagnostizieren - Überblick

##### Wo diagnostizieren Sie Verbindungen im Detail?

Die detaillierte Diagnose von Verbindungen führen Sie im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters durch.

##### Wie öffnen Sie den Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters?

Sie haben die folgenden Möglichkeiten, um das Register "Verbindungsinformation" des Inspektorfensters zu öffnen:

- Markieren Sie in der Verbindungstabelle diejenige Zeile, die zu der betrachteten Verbindung gehört. Klicken Sie dann im Inspektorfenster nacheinander auf die Register "Diagnose" und "Verbindungsinformation".
- Doppelklicken Sie in der Verbindungstabelle auf das Diagnosesymbol der betrachteten Verbindung.
- Sie befinden sich im Programmiereditor bei einer Anweisung der S7-Kommunikation oder der Open User Communication. Doppelklicken Sie auf das Diagnosesymbol der Anweisung (Stethoskop).

##### Aufbau des Bereichs "Diagnose > Verbindungsinformation" des Inspektorfensters

Voraussetzung dafür, dass der Inhalt des Registers "Verbindungsinformation" befüllt ist, ist eine bestehende Online-Verbindung zu mindestens einem Endpunkt der betrachteten Verbindung.

Bei selektierter Baugruppe (Netzsicht) enthält das Register die folgende Gruppe:

- Verbindungsressourcen (bei S7-1200 und S7-1500)

Bei selektierter Verbindung (Verbindungstabelle) enthält es die folgenden Gruppen:

- Verbindungsdetails
- Adressdetails der Verbindung (bei S7-1200 und S7-1500)

##### Online-Verbindungsressourcen bei S7-1200 ermitteln

##### Wo ermitteln Sie die Online-Verbindungsressourcen?

Die Online-Verbindungsressourcen entnehmen Sie der Gruppe "Verbindungsressourcen". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters. Sie wird nur dann angezeigt, wenn Sie in der Netzsicht eine Baugruppe ausgewählt haben, zu der eine Online-Verbindung besteht.

### Anzahl von Verbindungsressourcen

- Maximale Anzahl: Gibt die maximal mögliche Anzahl verfügbarer Verbindungsressourcen der Baugruppe an.
- Nicht belegt: Gibt an, wie viele Verbindungsressourcen noch nicht belegt sind. Wenn für bestimmte Kommunikationsarten bereits Verbindungsressourcen reserviert sind, können die nicht belegten Verbindungsressourcen nicht immer für beliebige Verbindungstypen verwendet werden.

### Reservierte und aktuell belegte Verbindungsressourcen

Für die im Folgenden genannten Kommunikationsarten werden die von der Baugruppe reservierten und aktuell belegten Verbindungsressourcen angezeigt.

| Kommunikationsart       | Bedeutung  |
|-------------------------|--|
| PG-Kommunikation        | Ressourcen für Verbindungen zwischen der Baugruppe und PGs (z. B. für den Verbindungsaufbau aus der Projektnavigation, für Online-Diagnose etc.) |
| HMI-Kommunikation       | Ressourcen für Verbindungen zwischen der Baugruppe und HMI-Geräten   |
| Open User Communication | Ressourcen für Verbindungen von Anweisungen der Open User Communication  |
| S7-Kommunikation        | Ressourcen für projektierte S7-Verbindungen, über die durch Aufruf von Anweisungen im Anwenderprogramm Daten ausgetauscht werden können          |
| Sonstige Kommunikation  | Gibt weitere belegte Verbindungsressourcen an, für die keine Verbindungsressourcen reserviert sind   |

### Online-Verbindungsressourcen bei S7-1500 ermitteln

#### Wo ermitteln Sie die Online-Verbindungsressourcen?

Die Online-Verbindungsressourcen entnehmen Sie der Gruppe "Verbindungsressourcen". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters. Sie wird nur dann angezeigt, wenn Sie in der Netzsicht eine Baugruppe ausgewählt haben, zu der eine Online-Verbindung besteht.

#### Beschreibung der detaillierten Anzeige der Verbindungsressourcen

Zur detaillierten Anzeige der Verbindungsressourcen gehören u. a.:

- die Anzahl der zur Verfügung stehenden Verbindungsressourcen
- die Anzahl der projektierten Verbindungsressourcen
- die Anzahl der noch zur Verfügung stehenden Verbindungsressourcen.

Deren Beschreibung finden Sie hier .

## Verbindungsdetails ermitteln

### Wo ermitteln Sie die Verbindungsdetails?

Die Verbindungsdetails entnehmen Sie der Gruppe "Verbindungsdetails". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters.

### Wann ist die Gruppe "Verbindungsdetails" befüllt?

Damit die Gruppe "Verbindungsdetails" des Registers "Verbindungsinformation" befüllt ist, müssen folgende Voraussetzungen erfüllt sein:

- Es besteht eine Online-Verbindung zum Endpunkt der betrachteten Verbindung.
- Sie haben in der Verbindungstabelle eine Zeile markiert.

### Aufbau der Gruppe "Verbindungsdetails"

Die Gruppe "Verbindungsdetails" besteht aus folgenden Elementen:

- Lokale ID (hex)
- Verbindungstyp (bei S7-1200 und S7-1500)
- Protokoll
- Verbindungsstatus: Symbol und Beschreibung
- Details
- Letzte Zustandsänderung (nur bei S7-300 und S7-400)

## Adressdetails einer Verbindung ermitteln

### Wo ermitteln Sie die Adressdetails einer Verbindung?

Die Adressdetails einer Verbindung entnehmen Sie der Gruppe "Adressdetails der Verbindung". Diese befindet sich im Bereich "Diagnose > Verbindungsinformation" des Inspektorfensters.

### Bei welchen CPUs ist die Gruppe "Adressdetails der Verbindung" vorhanden?

Die Gruppe "Adressdetails der Verbindung" des Registers "Verbindungsinformation" gibt es bei S7-1200- und S7-1500-CPU's.

### Wann ist die Gruppe "Adressdetails der Verbindung" befüllt?

Damit die Gruppe "Adressdetails der Verbindung" des Registers "Verbindungsinformation" befüllt ist, müssen folgende Voraussetzungen erfüllt sein:

- Es besteht eine Online-Verbindung zu den Endpunkten der betrachteten Verbindung.
- Sie haben in der Verbindungstabelle eine Zeile markiert.

### **Aufbau der Gruppe "Adressdetails der Verbindung"**

Für die beiden Kommunikationspartner werden die für den Verbindungstyp relevanten Adressdetails angegeben.



# PLC programmieren

## 9.1 Anwenderprogramm erstellen

### 9.1.1 Grundlagen zur Programmierung

#### 9.1.1.1 Betriebssystem und Anwenderprogramm

##### Betriebssystem

##### Funktion

Das Betriebssystem ist in jeder CPU enthalten und organisiert alle Funktionen und Abläufe der CPU, die nicht mit einer spezifischen Steuerungsaufgabe verbunden sind.

Zu den Aufgaben des Betriebssystems gehören z. B.:

- Abwickeln von Neustart (Warmstart)
- Aktualisieren des Prozessabbilds der Eingänge und des Prozessabbilds der Ausgänge
- Aufrufen des Anwenderprogramms
- Erfassen von Alarmen und Aufrufen der Alarm-OBs
- Erkennen und Behandeln von Fehlern
- Verwalten von Speicherbereichen

Das Betriebssystem ist Bestandteil der CPU und ist bei der Auslieferung bereits auf dieser enthalten.

##### Siehe auch

Anwenderprogramm (Seite 1025)

##### Anwenderprogramm

##### Funktion

Das Anwenderprogramm enthält alle Funktionen, die zur Bearbeitung Ihrer spezifischen Automatisierungsaufgabe erforderlich sind.

Zu den Aufgaben des Anwenderprogramms gehören:

- Prüfung der Vorbedingungen für einen Neustart (Warmstart) mithilfe von Anlauf-OBs, z. B. Endschalte in richtiger Stellung oder Sicherheitseinrichtung aktiv
- Bearbeiten von Prozessdaten, z. B. Binärsignale verknüpfen, Analogwerte einlesen und auswerten, Binärsignale für die Ausgabe festlegen, Analogwerte ausgeben
- Reaktion auf Alarme, z. B. Diagnosealarm bei Unterschreitung des Grenzwerts einer analogen Erweiterungsbaugruppe
- Bearbeiten von Störungen im normalen Programmablauf

Das Anwenderprogramm wird von Ihnen erstellt und in die CPU geladen.

**Siehe auch**

Betriebssystem (Seite 1025)

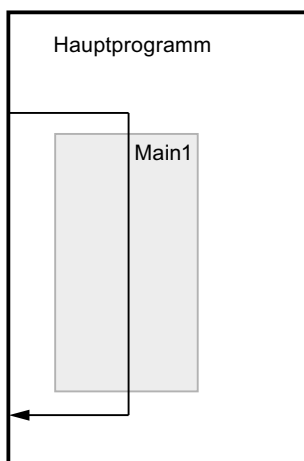
**9.1.1.2 Bausteine im Anwenderprogramm**

**Lineare und strukturierte Programmierung**

**Lineare Programmierung**

Kleine Automatisierungsaufgaben können Sie lösen, indem Sie das gesamte Anwenderprogramm linear in einen Zyklus-OB schreiben. Dies ist nur empfehlenswert bei einfachen Programmen.

Das folgende Bild zeigt schematisch ein lineares Programm: Der Zyklus-OB "Main1" enthält das gesamte Anwenderprogramm.



**Strukturierte Programmierung**

Komplexe Automatisierungsaufgaben können einfacher realisiert und gewartet werden, wenn sie in kleinere Teilaufgaben untergliedert werden, die den technologischen Funktionen des Automatisierungsprozesses entsprechen oder mehrfach verwendet werden können. Im

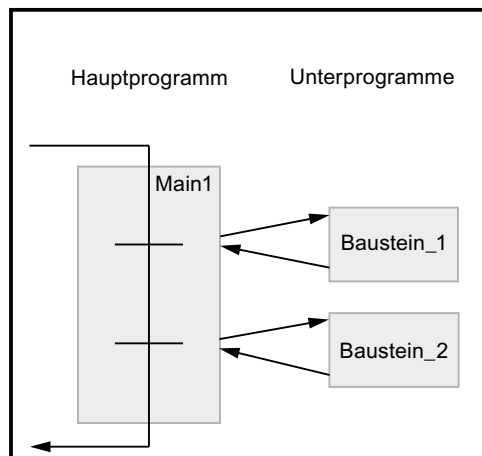


Anwenderprogramm werden diese Teilaufgaben durch Bausteine repräsentiert. Dabei ist jeder Baustein ein unabhängiger Abschnitt des Anwenderprogramms.

Die Strukturierung des Programms bietet folgende Vorteile:

- Umfangreiche Programme lassen sich übersichtlich programmieren.
- Einzelne Programmteile können standardisiert und mehrfach mit wechselnden Parametern verwendet werden.
- Die Programmorganisation wird vereinfacht.
- Änderungen des Programms lassen sich leichter durchführen.
- Der Programmtest wird vereinfacht, weil er abschnittsweise erfolgen kann.
- Die Inbetriebnahme wird erleichtert.

Das folgende Bild zeigt schematisch ein strukturiertes Programm: Der Zyklus-OB "Main1" ruft nacheinander Unterprogramme auf, die definierte Teilaufgaben ausführen.



## Übersicht über die Bausteinararten

### Bausteinararten

Um die Aufgaben innerhalb eines Automatisierungssystems zu erledigen, gibt es unterschiedliche Bausteinararten. Die folgende Tabelle zeigt die Bausteinararten, die zur Verfügung stehen:

| Bausteinart                              | Kurzbeschreibung  |
|--|---|
| Organisationsbausteine (Seite 1028) (OB) | Organisationsbausteine legen die Struktur des Anwenderprogramms fest.   |
| Funktionen (Seite 1028) (FC)             | Funktionen enthalten Programmroutinen für wiederkehrende Aufgaben. Sie haben kein "Gedächtnis".   |
| Funktionsbausteine (Seite 1029) (FB)     | Funktionsbausteine sind Codebausteine, die ihre Werte dauerhaft in Instanz-Datenbausteinen ablegen, so dass sie auch nach der Bausteinbearbeitung zur Verfügung stehen. |

| Bausteinart                         | Kurzbeschreibung   |
|-------------------------------------|--|
| Instanz-Datenbausteine (Seite 1032) | Instanz-Datenbausteine werden beim Aufruf eines Funktionsbausteins dem Baustein zugeordnet und dienen zur Speicherung der Programmdateien. |
| Globale Datenbausteine (Seite 1030) | Globale Datenbausteine sind Datenbereiche zur Speicherung von Daten, die von beliebigen Bausteinen genutzt werden können.                  |

## Organisationsbausteine (OB)

### Definition

Organisationsbausteine (OBs) bilden die Schnittstelle zwischen dem Betriebssystem und dem Anwenderprogramm. Sie werden vom Betriebssystem aufgerufen und steuern z. B. folgende Vorgänge:

- Anlaufverhalten des Automatisierungssystems
- Zyklische Programmbearbeitung
- Alarmgesteuerte Programmbearbeitung
- Behandlung von Fehlern

Sie können die Organisationsbausteine programmieren und so das Verhalten der CPU bestimmen. Abhängig von der verwendeten CPU stehen Ihnen unterschiedliche Organisationsbausteine zur Verfügung.

Weitere Informationen zu den Organisationsbausteinen finden Sie in den Beschreibungen der Arbeitsweisen der CPUs im Kapitel "Zusatzinformationen zu Konfigurationen" von "Geräte und Netze konfigurieren".

### Startinformation von Organisationsbausteinen

Beim Start einiger Organisationsbausteine stellt das Betriebssystem Informationen bereit, die im Anwenderprogramm ausgewertet werden können. Ob und welche Informationen bereitgestellt werden, finden Sie in den Beschreibungen der Organisationsbausteine.

### Siehe auch

Organisationsbausteine anlegen (Seite 1212)

## Funktionen (FC)

### Definition

Funktionen (FCs) sind Codebausteine ohne Gedächtnis. Sie haben keinen Datenspeicher, in denen Werte von Bausteinparametern gespeichert werden könnten. Deshalb müssen beim Aufruf einer Funktion alle Formalparameter mit Aktualparametern versorgt werden. Um Daten dauerhaft zu speichern, stehen den Funktionen globale Datenbausteine zur Verfügung.

## Anwendungsbereich

Eine Funktion enthält ein Programm, das immer dann ausgeführt wird, wenn die Funktion von einem anderen Codebaustein aufgerufen wird. Funktionen können z. B. zu folgenden Zwecken eingesetzt werden:

- Funktionswerte an den aufrufenden Baustein zurückgeben, z. B. bei mathematischen Funktionen
- Technologische Funktionen ausführen, z. B. Einzelsteuerungen mit Binärverknüpfungen

Eine Funktion kann auch mehrmals an verschiedenen Stellen innerhalb eines Programms aufgerufen werden. Sie erleichtern so die Programmierung häufig wiederkehrender Funktionen.

---

### Hinweis

#### Parameterübergabe beim Aufruf von Funktionen

Um Fehler beim Arbeiten mit Funktionen zu vermeiden, beachten Sie das Kapitel "Auto-Hotspot".

---

## Siehe auch

Funktionen und Funktionsbausteine anlegen (Seite 1213)

## Funktionsbausteine (FB)

### Definition

Funktionsbausteine sind Codebausteine, die ihre Ein-, Aus- und Durchgangparameter dauerhaft in Instanz-Datenbausteinen ablegen, sodass sie auch nach der Bausteinbearbeitung zur Verfügung stehen. Deshalb werden sie auch als "Bausteine mit Gedächtnis" bezeichnet.

Funktionsbausteine können auch mit temporären Variablen arbeiten. Temporäre Variablen werden jedoch nicht im Instanz-DB abgespeichert, sondern stehen nur einen Zyklus lang zur Verfügung.

## Anwendungsbereich

Funktionsbausteine enthalten Unterprogramme, die immer dann ausgeführt werden, wenn ein Funktionsbaustein von einem anderen Codebaustein aufgerufen wird. Ein Funktionsbaustein kann auch mehrmals an verschiedenen Stellen innerhalb eines Programms aufgerufen werden. Sie erleichtern so die Programmierung häufig wiederkehrender Funktionen.

## Instanzen von Funktionsbausteinen

Ein Aufruf eines Funktionsbausteins wird als Instanz bezeichnet. Für jede Instanz eines Funktionsbausteins wird ein Instanzdatenbaustein benötigt, in dem für die im FB deklarierten Formalparameter instanzspezifische Werte gespeichert werden.

Der Funktionsbaustein kann seine instanzspezifischen Daten in einem eigenen Instanz-Datenbaustein ablegen oder im Instanz-Datenbaustein des aufrufenden Bausteins.

## Zugriffsarten

S7-1200 und S7-1500 bieten zwei verschiedene Zugriffsmöglichkeiten für die Instanz-Datenbausteine, die einem Funktionsbaustein beim Aufruf zugeordnet werden:

- Datenbausteine mit optimiertem Zugriff  
Datenbausteine mit optimiertem Zugriff haben keine fest definierte Speicherstruktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins.
- Datenbausteine mit Standardzugriff (kompatibel mit S7-300/400)  
Datenbausteine mit Standardzugriff haben eine feste Speicherstruktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins.

---

### Hinweis

Um Fehler beim Arbeiten mit Funktionsbausteinen zu vermeiden, beachten Sie das Kapitel "Auto-Hotspot".

---

## Siehe auch

Funktionen und Funktionsbausteine anlegen (Seite 1213)

Multiinstanzen (Seite 1042)

Instanz-Datenbausteine (Seite 1032)

Grundlagen zum Bausteinzugriff (Seite 1034)

## Globale Datenbausteine (DB)

### Definition

Datenbausteine dienen der Speicherung von Programmdateien. In Datenbausteinen stehen also variable Daten, mit denen das Anwenderprogramm arbeitet. Globale Datenbausteine nehmen Daten auf, die von allen anderen Bausteinen aus verwendet werden können.

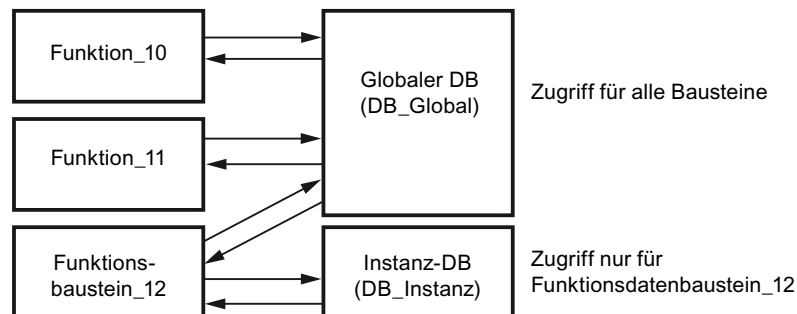
Die maximale Größe von Datenbausteinen variiert abhängig von der CPU. Die Struktur globaler Datenbausteine können Sie beliebig festlegen.

Sie haben auch die Möglichkeit, PLC-Datentypen (UDT) als Vorlage für die Erstellung von globalen Datenbausteinen zu verwenden.

## Globale Datenbausteine im Anwenderprogramm

Jeder Funktionsbaustein, jede Funktion oder jeder Organisationsbaustein kann die Daten aus einem globalen Datenbaustein lesen oder selbst Daten in einen globalen Datenbaustein schreiben. Diese Daten bleiben im Datenbaustein auch dann erhalten, wenn der Datenbaustein verlassen wird. Ein globaler Datenbaustein und ein Instanz-Datenbaustein können gleichzeitig geöffnet sein.

Das folgende Bild zeigt die verschiedenen Zugriffe auf Datenbausteine:



## Zugriffsarten

S7-1200 und S7-1500 bieten zwei verschiedene Zugriffsmöglichkeiten für globale Datenbausteine:

- Datenbausteine mit optimiertem Zugriff  
Datenbausteine mit optimiertem Zugriff haben keine fest definierte Struktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins.
- Datenbausteine mit Standardzugriff (kompatibel mit S7-300/400)  
Datenbausteine mit Standardzugriff haben eine feste Struktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins.

## ARRAY-Datenbausteine (S7-1500)

Eine besondere Art von globalen Datenbausteinen sind ARRAY-Datenbausteine. Diese bestehen aus einem ARRAY eines beliebigen Datentyps. Es kann z. B. ein ARRAY eines PLC-Datentyps (UDT) sein. Neben dem ARRAY enthält der DB keine weiteren Elemente. ARRAY-Datenbausteine erleichtern durch ihre flache Struktur den Zugriff auf die ARRAY-Elemente sowie deren Übergabe an aufgerufene Bausteine.

Für ARRAY-Datenbausteine ist das Attribut "Optimierter Bausteinzugriff" immer aktiviert. ARRAY-Datenbausteine mit Standardzugriff sind nicht möglich.

Möglichkeiten zur Adressierung von ARRAY-DBs bietet die Task Card "Anweisungen" im Abschnitt "Verschieben".

## Siehe auch

- Datenbausteine anlegen (Seite 1214)
- Grundlagen zum Bausteinzugriff (Seite 1034)

## Instanz-Datenbausteine

### Definition

Der Aufruf eines Funktionsbausteins wird als Instanz bezeichnet. Die Daten, mit denen die Instanz arbeitet, werden in einem Instanz-Datenbaustein gespeichert.

Die maximale Größe von Instanz-Datenbausteinen variiert abhängig von der CPU. Die im Funktionsbaustein deklarierten Variablen bestimmen die Struktur des Instanz-Datenbausteins.

### Zugriffsarten

S7-1200 und S7-1500 bieten zwei verschiedene Zugriffsmöglichkeiten für die Instanz-Datenbausteine, die einem Funktionsbaustein beim Aufruf zugeordnet werden:

- Datenbausteine mit optimiertem Zugriff  
Datenbausteine mit optimiertem Zugriff haben keine fest definierte Struktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins.
- Datenbausteine mit Standardzugriff (kompatibel mit S7-300/400)  
Datenbausteine mit Standardzugriff haben eine feste Struktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins.

Siehe auch: Auto-Hotspot

## Siehe auch

- Datenbausteine anlegen (Seite 1214)
- Grundlagen zum Bausteinzugriff (Seite 1034)

## CPU-Datenbausteine

### Definition

CPU-Datenbausteine sind Datenbausteine, die von der CPU zur Laufzeit erzeugt werden. Dazu fügen Sie in Ihr Anwenderprogramm die Anweisung "CREATE\_DB" ein. Nach dem Erzeugen des Datenbausteins zur Laufzeit können Sie den Datenbaustein zum Speichern Ihrer Daten verwenden.

CPU-Datenbausteine werden innerhalb eines erreichbaren Teilnehmers im Ordner "Programmbausteine" mit einem kleinen CPU-Symbol angezeigt. Sie können die aktuellen

Werte der Variablen eines CPU-Datenbausteins wie die eines anderen Datenbausteintyps online beobachten.

CPU-Datenbausteine können Sie nicht offline in Ihrem Projekt anlegen.

### Laden von CPU-Datenbausteinen

Nach dem das Anwenderprogramm über die Anweisung "CREATE\_DB" den CPU-Datenbaustein erzeugt hat, steht er zunächst nur online auf dem Gerät zur Verfügung. Beim nächsten vollständigen Ladevorgang, bei dem Sie die Bausteine vom Gerät in Ihr Projekt laden, werden auch alle CPU-Datenbausteine mit in das Projekt geladen. Dabei werden die CPU-Datenbausteine mit einem kleinen CPU-Symbol gekennzeichnet. Sie können die CPU-Datenbausteine jedoch nicht mehr zurück in Ihr Gerät laden.

### Beschränkungen von CPU-Datenbausteinen im Projekt

Wenn die CPU-Datenbausteine durch einen Ladevorgang in Ihr offline Projekt geladen wurden, können Sie sie öffnen und sich ihren Inhalt ansehen. Beachten Sie jedoch, dass CPU-Datenbausteine im Projekt schreibgeschützt sind. CPU-Datenbausteine unterliegen im Projekt daher folgenden Beschränkungen:

- Sie können CPU-Datenbausteine weder bearbeiten noch in einen anderen Datenbausteintyp konvertieren.
- Sie können CPU-Datenbausteine nicht mit einem Know-how-Schutz versehen.
- Sie können die Programmiersprache eines CPU-Datenbausteins nicht ändern.
- CPU-Datenbausteine können nicht übersetzt und auch nicht in ein Gerät geladen werden.

### Vergleichen von CPU-Datenbausteinen

Wenn die CPU-Datenbausteine durch einen Ladevorgang in Ihr offline Projekt geladen wurden, können Sie für die geladenen CPU-Datenbausteine einen Online/Offline-Vergleich durchführen. Im Vergleichseditor erhalten Sie eine Übersicht über die Unterschiede. Sie können CPU-Datenbausteine, die sich in ihrer Online- und Offline-Version unterscheiden, jedoch nicht durch Laden der Offline-Version in das Gerät synchronisieren.

### Löschen von CPU-Datenbausteinen

Sie können CPU-Datenbausteine sowohl aus Ihrem Projekt als auch in der CPU löschen.

### Siehe auch

CPU-Datenbausteine löschen (Seite 1234)

## Bausteine mit optimiertem Zugriff

### Grundlagen zum Bausteinzugriff

#### Einführung

STEP 7 bietet Datenbausteine mit unterschiedlichen Zugriffsmöglichkeiten an:

- Datenbausteine mit optimiertem Zugriff (S7-1200 / S7-1500)
- Datenbausteine mit Standardzugriff (S7-300 / S7-400 / S7-1200 / S7-1500)

Innerhalb eines Programms können Sie beide Arten von Bausteinen kombinieren.

#### Datenbausteine mit optimiertem Zugriff

Datenbausteine mit optimiertem Zugriff haben keine fest definierte Struktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins. Die Elemente werden im verfügbaren Speicherbereich des Bausteins automatisch so angeordnet, dass keine Lücken im Speicher entstehen. Somit wird die Speicherkapazität optimal ausgeschöpft.

Variablen in diesen Datenbausteinen werden durch ihren symbolischen Namen identifiziert. Um die Variablen zu adressieren, geben Sie den symbolischen Namen der Variablen an. Zum Beispiel greifen Sie auf die Variable "Füllstand" im DB "Daten" folgendermaßen zu:

```
"Daten".Füllstand
```

Bausteine mit optimiertem Zugriff bieten folgende Vorteile:

- Sie können Datenbausteine mit beliebiger Struktur aufbauen, ohne auf die physikalische Anordnung der einzelnen Datenelemente zu achten.
- Der Zugriff auf optimierte Daten erfolgt immer schnellstmöglich, da die Datenablage vom System optimiert und verwaltet wird.
- Zugriffsfehler, z. B. bei indirekter Adressierung oder aus HMI heraus, sind nicht möglich.
- Sie können gezielt einzelne Variablen als remanent definieren.
- Optimierte Bausteine verfügen standardmäßig über eine Speicherreserve, die es Ihnen ermöglicht, die Schnittstellen von Funktions- oder Datenbausteinen im laufenden Betrieb zu erweitern. Die geänderten Bausteine können geladen werden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen.

---

#### Hinweis

Das Attribut "optimierter Bausteinzugriff" ist für folgende Bausteine immer aktiviert und kann nicht abgewählt werden:

- GRAPH-Bausteine
  - ARRAY-Datenbausteine
-



## Datenbausteine mit Standardzugriff

Datenbausteine mit Standardzugriff haben eine feste Struktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins. Die Adresse wird in der Spalte "Offset" angezeigt.

Variablen in diesen Datenbausteinen können Sie sowohl symbolisch als auch absolut adressieren:

```
"Daten".Füllstand
DB1.DBW2
```

## Remanenz bei optimiertem Zugriff oder Standardzugriff

Wenn Sie Daten als remanent definieren, bleiben deren Werte nach einem Spannungsausfall oder Netz-Aus erhalten. Eine remanente Variable wird nach dem Wiederanlauf nicht initialisiert, sondern behält den Wert, den sie vor dem Spannungsausfall hatte. Wenn eine DB-Variable als remanent definiert wird, wird sie im remanenten Speicherbereich des Datenbausteins abgelegt.

Die Möglichkeiten zur Einstellung der Remanenz hängen von der Zugriffsart des Bausteins ab:

- In Datenbausteinen mit Standardzugriff können Sie das Remanenzverhalten einzelner Variablen nicht einstellen. Die Remanenzeinstellung gilt für alle Variablen des Datenbausteins.
- In Datenbausteinen mit optimiertem Zugriff können Sie das Remanenzverhalten einzelner Variablen bestimmen.  
Bei Variablen strukturierten Datentyps gilt die Remanenzeinstellung immer für die gesamte Struktur. Für einzelne Elemente innerhalb des Datentyps können Sie keine individuelle Remanenzeinstellung vornehmen.

## Adressierungsmöglichkeiten bei optimiertem Zugriff und Standardzugriff

Bausteine mit optimiertem Zugriff lassen nur sogenannte "typsichere" Zugriffe zu. Typsichere Zugriffe adressieren Variablen nur über den symbolischen Namen. So kann es auch bei Änderungen am Baustein oder an der Bausteinschnittstelle nicht zu Inkonsistenzen im Programm oder Zugriffsfehlern kommen.

Die folgende Tabelle zeigt die zulässigen Adressierungsmöglichkeiten für optimierte Daten:

| Adressierung                                    | Baustein mit Standardzugriff | Baustein mit optimiertem Zugriff |
|---|------------------------------|----------------------------------|
| Symbolische Adressierung                        | x                            | x                                |
| Indizierte Adressierung von ARRAYS              |                              | x                                |
| Slice-Zugriffe                                  | x                            | x                                |
| Überlagerung mit AT                             | x                            | -                                |
| Absolute Adressierung                           | x                            | -                                |
| Indirekte Adressierung über ANY                 | x                            | -                                |
| Indirekte Adressierung über POINTER und VARIANT | x                            | nur bei symbolischer Notation    |

## Siehe auch

Bausteinzugriff einstellen (Seite 1036)

## Bausteinzugriff einstellen

### Einführung

Der Bausteinzugriff wird beim Anlegen eines Bausteins automatisch eingestellt:

- Auf CPUs der Baureihe S7-1200/1500 haben neu erzeugte Bausteine per Voreinstellung optimierten Zugriff.
- Auf CPUs der Baureihe S7-300/S7-400 haben neu erzeugte Bausteine per Voreinstellung Standardzugriff.

Wenn Sie einen Baustein in eine CPU einer anderen Baureihe kopieren oder migrieren, wird der Bausteinzugriff nicht automatisch umgestellt. Es kann jedoch sinnvoll sein, den Bausteinzugriff manuell zu ändern, z. B. um den vollen Funktionsumfang der CPU nutzen zu können.

Nach der Umstellung des Bausteinzugriffs müssen Sie das Programm in den meisten Fällen neu übersetzen und laden.

#### **ACHTUNG**

##### **Optimierter Bausteinzugriff bei GRAPH-Bausteinen**

Das Attribut "optimierter Bausteinzugriff" ist für GRAPH-Bausteine in S7-1500 immer aktiviert und kann nicht abgewählt werden.

### Vorgehensweise

Um den Bausteinzugriff einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf den Baustein, dessen Bausteinzugriff Sie ändern möchten.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Der Eigenschaftsdialog des Bausteins wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf "Attribute".
5. Aktivieren oder deaktivieren Sie die Option "Optimierter Bausteinzugriff".
6. Bestätigen Sie Ihre Eingaben mit "OK".

### Einschränkungen und Besonderheiten

Grundsätzlich können Sie den Bausteinzugriff nur bei CPUs der Baureihe S7-1200/1500 umstellen, da nur diese die Zugriffsart "optimiert" unterstützen.

Dabei gibt es folgende Einschränkungen oder Besonderheiten:


- **Instanzenbausteine**  
Der Bausteinzugriff von Instanzdatenbausteinen richtet sich immer nach dem zugeordneten Funktionsbaustein und kann nicht manuell geändert werden. Wenn Sie die Zugriffsart eines Funktionsbausteins ändern, müssen Sie die zugeordneten Instanzdatenbausteine aktualisieren. Bei der Aktualisierung wird die Zugriffsart des Instanzdatenbausteins angepasst.
- **Systembausteine und know-how-geschützte Bausteine**  
Bei Systembausteinen und know-how-geschützten Bausteinen können Sie den Bausteinzugriff nicht manuell ändern.
- **Organisationsbausteine**  
Die Startinformationen eines OB mit Standardzugriff liegen in der Bausteinschnittstelle immer in den ersten 20 Byte des Abschnitts "Temp". Die Startinformationen eines OB mit optimiertem Zugriff hingegen werden im Abschnitt "Input" abgelegt. Deshalb ändert sich bei OBs durch die Umstellung des Bausteinzugriffs auch die Bausteinschnittstelle. Details sind in den nachfolgenden Abschnitten beschrieben.

### Umstellen des Bausteinzugriffs von "standard" nach "optimiert"

Wenn Sie einen Baustein von einer CPU der Baureihe S7-300/400 auf eine CPU der Baureihe S7-1200/1500 kopieren oder migrieren, behält er zunächst die Zugriffsart "standard". Die Programmbearbeitung wird jedoch durch die Verwendung von Bausteinen mit optimiertem Zugriff deutlich performanter. Aus diesem Grund kann es sinnvoll sein, die Zugriffsart manuell zu ändern.

Bei der Umstellung werden die Bausteine wie folgt angepasst:

- **Funktionsbausteine**  
Alle Schnittstellenparameter erhalten die Remanenzeinstellung "nicht remanent".
- **Globale Datenbausteine**  
Die Remanenzeinstellung, die zuvor zentral für den gesamten Datenbaustein eingestellt war, wird auf die einzelnen Schnittstellenparameter übertragen. Anschließend können Sie die Remanenzeinstellung der einzelnen Parameter manuell einstellen.  
Es gilt jedoch weiterhin folgende Regel: Bei Variablen strukturierten Datentyps gilt die Remanenzeinstellung immer für die gesamte Struktur. Für einzelne Elemente innerhalb eines strukturierten Datentyps können Sie keine individuelle Remanenzeinstellung vornehmen. Daraus folgt, dass auch in Datenbausteinen, die auf PLC-Datentypen basieren, keine individuellen Remanenzeinstellungen für die einzelnen Variablen möglich sind.
- **Organisationsbausteine**  
Alle Schnittstellenparameter, die in den ersten 20 Byte des Abschnitts "Temp" liegen, werden gelöscht. Im Abschnitt "Input" werden neue CPU-spezifische Startinformationen angelegt. Wenn dabei Namenskonflikte mit anwenderdefinierten Schnittstellenparametern auftreten, werden die anwenderdefinierten Schnittstellenparameter umbenannt.

|  |
|--|
|  <b>VORSICHT</b>  |
| <b>Die Umstellung des Bausteinzugriffs hat folgende Konsequenzen:</b>  |
| <ul style="list-style-type: none"><li>• Nach der Umstellung des Bausteinzugriffs auf "optimiert" können Sie die Schnittstellenparameter des Bausteins nicht mehr absolut adressieren.<br/>Beispiel: #L0.1 ist nicht mehr zulässig.</li><li>• Durch die Umstellung des Bausteinzugriffs von Organisationsbausteinen auf "optimiert" ändert sich die Schnittstelle des OB.</li></ul> |
| Sie müssen das Programm aufgrund dieser Änderungen möglicherweise anpassen, neu übersetzen und laden.  |
| Siehe auch: Auto-Hotspot   |

### Umstellen des Bausteinzugriffs von "optimiert" nach "standard"

Wenn Sie einen Baustein von einer CPU der Baureihe S7-1200/1500 auf eine CPU der Baureihe S7-300/400 kopieren oder verschieben möchten, müssen Sie zunächst den Bausteinzugriff auf "standard" setzen.

Bei der Umstellung werden die Bausteine wie folgt angepasst:

- Funktionsbausteine und globale Datenbausteine:  
Im Funktionsbaustein kann keine Remanenzeinstellung mehr vorgenommen werden. Die Einstellung erfolgt am Instanz-Datenbaustein.  
Alle Schnittstellenparameter im Instanzdatenbaustein oder globalen Datenbaustein erhalten dieselbe Remanenzeinstellung. Die Umstellung erfolgt nach folgender Regel:
  - Wenn alle Schnittstellenparameter im Ausgangsbaustein remanent waren, wird der gesamte Baustein nach der Umstellung remanent.
  - Wenn alle Schnittstellenparameter im Ausgangsbaustein nicht-remanent waren, wird der gesamte Baustein nach der Umstellung nicht-remanent.
  - Wenn die Schnittstellenparameter im Ausgangsbaustein unterschiedliche Remanenzeinstellungen hatten, wird der gesamte Baustein nach der Umstellung nicht-remanent.
- Organisationsbausteine  
Alle Schnittstellenparameter, die im Abschnitt "Input" liegen, werden gelöscht. Im Abschnitt "Temp" werden neue CPU-spezifische Startinformationen angelegt. Dabei werden die ersten 20 Byte belegt. Wenn dabei Namenskonflikte mit anwenderdefinierten Schnittstellenparametern auftreten, werden die anwenderdefinierten Schnittstellenparameter umbenannt.

 **VORSICHT**

**Die Umstellung des Bausteinzugriffs hat folgende Konsequenzen:**

Bei der Umstellung des Bausteinzugriffs auf "standard" können sich die Remanenzeinstellungen der Schnittstellenparameter ändern. Sie müssen das Programm aufgrund dieser Änderungen möglicherweise anpassen, neu übersetzen und laden.

Siehe auch: Auto-Hotspot

**Siehe auch**

Grundlagen zum Bausteinzugriff (Seite 1034)

### 9.1.1.3 Bausteinaufrufe

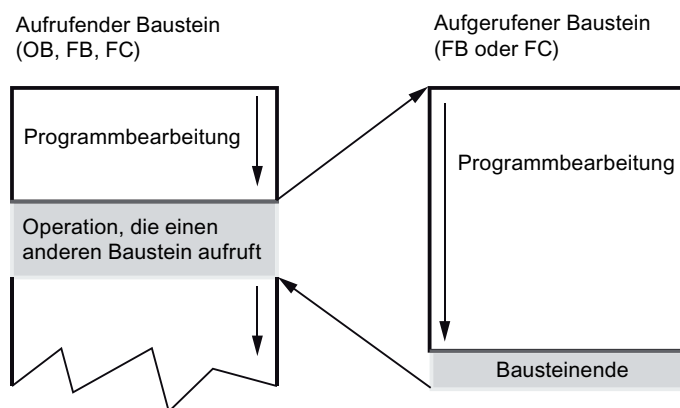
#### Grundlagen zu Bausteinaufrufen

#### Funktion von Bausteinaufrufen

Damit Ihre Bausteine im Anwenderprogramm ausgeführt werden, müssen sie von einem anderen Baustein aus aufgerufen werden.

Ruft ein Baustein einen weiteren Baustein auf, werden die Anweisungen des aufgerufenen Bausteins bearbeitet. Erst wenn die Bearbeitung des aufgerufenen Bausteins beendet ist, wird die Bearbeitung des aufrufenden Bausteins wieder aufgenommen. Die Bearbeitung wird mit der Anweisung fortgesetzt, die auf den Bausteinaufruf folgt.

Das folgende Bild zeigt den Ablauf eines Bausteinaufrufs innerhalb eines Anwenderprogramms:



### Parameterübergabe

Beim Bausteinaufruf versorgen Sie die Parameter in der Schnittstelle des Bausteins mit Werten. Durch die Versorgung der Eingangsparameter legen Sie fest, mit welchen Daten die Bearbeitung des Bausteins erfolgt. Durch die Versorgung der Ausgangsparameter legen Sie fest, wo die Bearbeitungsergebnisse abgelegt werden.

### Siehe auch

Aufrufhierarchie (Seite 1040)

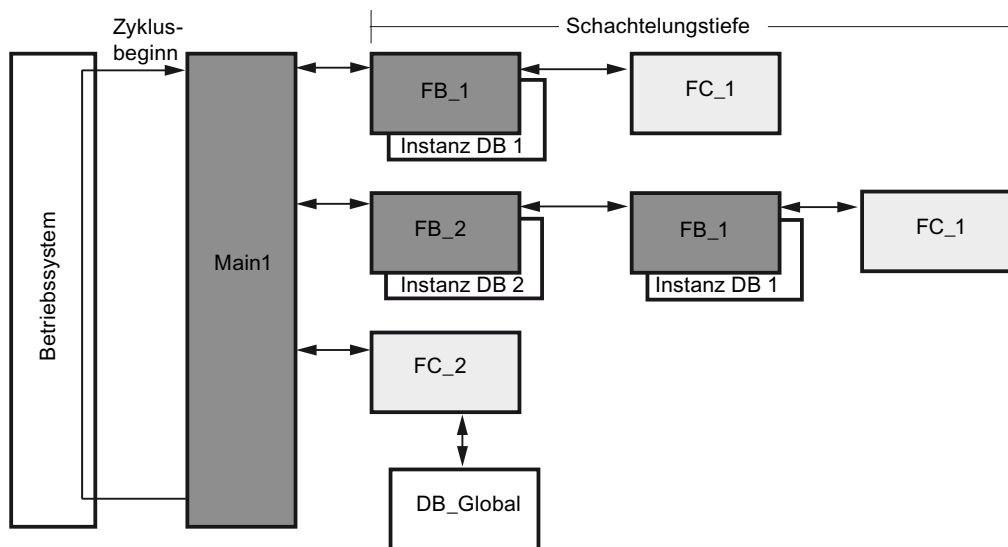
Grundlagen zu Einzel- und Multiinstanzen (Seite 1041)

### Aufrufhierarchie

#### Definition

Als Aufrufhierarchie wird die Reihenfolge und Schachtelung der Bausteinaufrufe bezeichnet. Die zulässige Schachtelungstiefe ist von der CPU abhängig.

Das folgende Bild zeigt ein Beispiel für die Reihenfolge und Schachtelung der Bausteinaufrufe innerhalb eines Bearbeitungszyklus:



### Siehe auch

Grundlagen zu Einzel- und Multiinstanzen (Seite 1041)

Grundlagen zu Bausteinaufrufen (Seite 1039)

## Aufruf von Funktionsbausteinen als Einzel- oder Multiinstanz

### Grundlagen zu Einzel- und Multiinstanzen

### Verwendung von Einzel- und Multiinstanzen

Funktionsbausteine (FB) legen ihre Daten in Instanz-Datenbausteinen ab. Instanz-Datenbausteine speichern die Werte der Bausteinparameter und die statischen Lokaldaten der Funktionsbausteine.

Sie können Instanz-Datenbausteine folgendermaßen zuordnen:

- Einzelinstanz:  
Ein Instanz-Datenbaustein pro Instanz eines Funktionsbausteins
- Multiinstanz:  
Ein Instanz-Datenbaustein für die Instanz eines Funktionsbausteins und alle Instanzen von in ihm aufgerufenen Funktionsbausteinen.

### Siehe auch

Grundlagen zu Bausteinaufrufen (Seite 1039)

Multiinstanzen (Seite 1042)

Einzelinstanzen (Seite 1041)

Aufrufhierarchie (Seite 1040)

## Einzelinstanzen

### Definition

Der Aufruf eines Funktionsbausteins, dem ein eigener Instanz-Datenbaustein zugeordnet wird, wird als Einzelinstanz bezeichnet.

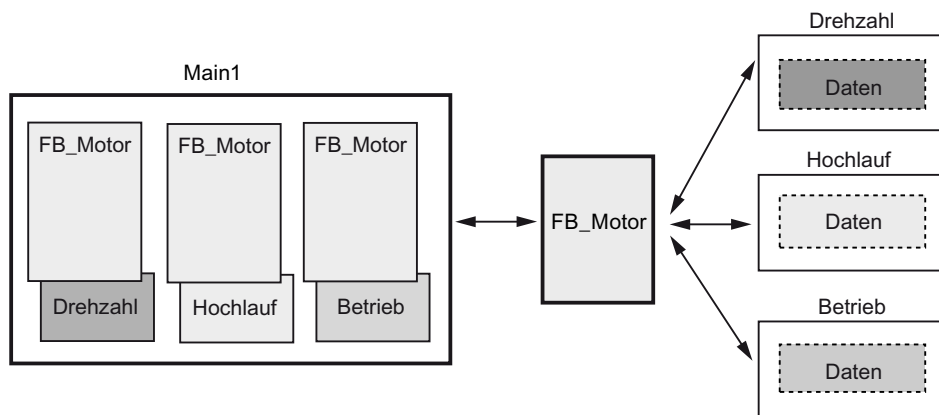
Durch Zuordnung des Instanz-Datenbausteins geben Sie an, wo die Instanzdaten des FB gespeichert werden sollen. Indem Sie jedem Aufruf einen anderen Instanz-Datenbaustein zuordnen, können sie denselben FB mehrfach mit jeweils anderen Instanzdaten einsetzen.

### Beispiel für eine Einzelinstanz

Sie können mit einem Funktionsbaustein mehrere Motoren steuern. Ordnen Sie dazu jedem Aufruf des Funktionsbausteins für die Motorsteuerung einen anderen Instanz-Datenbaustein zu.

Die unterschiedlichen Daten für die einzelnen Motoren, z. B. Drehzahl, Hochlaufzeit, Gesamtbetriebszeit, werden in den verschiedenen Instanz-Datenbausteinen gespeichert. Abhängig vom zugeordneten Instanz-Datenbaustein wird ein anderer Motor gesteuert.

Das folgende Bild zeigt die Steuerung von drei Motoren mit einem Funktionsbaustein und drei verschiedenen Datenbausteinen:



### Siehe auch

Grundlagen zu Einzel- und Multiinstanzen (Seite 1041)

Multiinstanzen (Seite 1042)

### Multiinstanzen

#### Definition

Multiinstanzen bieten einem aufgerufenen Funktionsbaustein die Möglichkeit, seine Daten in den Instanz-Datenbaustein des aufrufenden Funktionsbausteins zu legen.

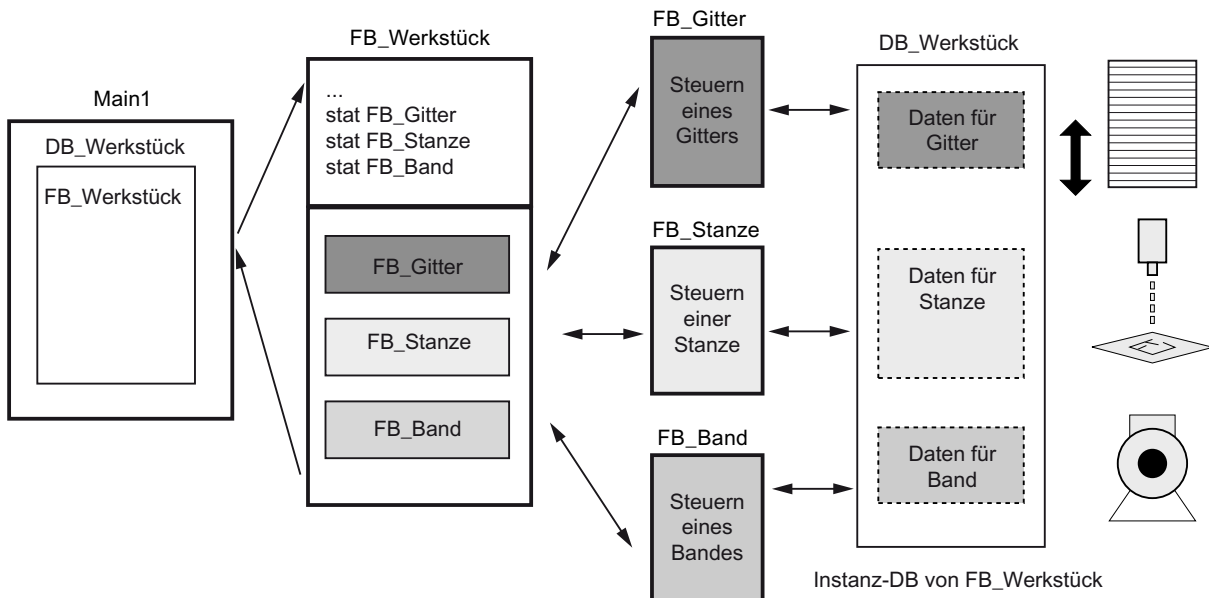
Damit erreichen Sie die Konzentration der Instanzdaten in einem Instanz-Datenbaustein und können die verfügbare Anzahl der Instanz-Datenbausteine besser ausnutzen.

#### Ein Instanz-Datenbaustein für die Instanzen unterschiedlicher Funktionsbausteine

Das folgende Bild zeigt, wie mehrere verschiedene Funktionsbausteine ihre Daten in einem aufrufenden Baustein ablegen. Der FB\_Werkstück ruft nacheinander FB\_Gitter, FB\_Stanze

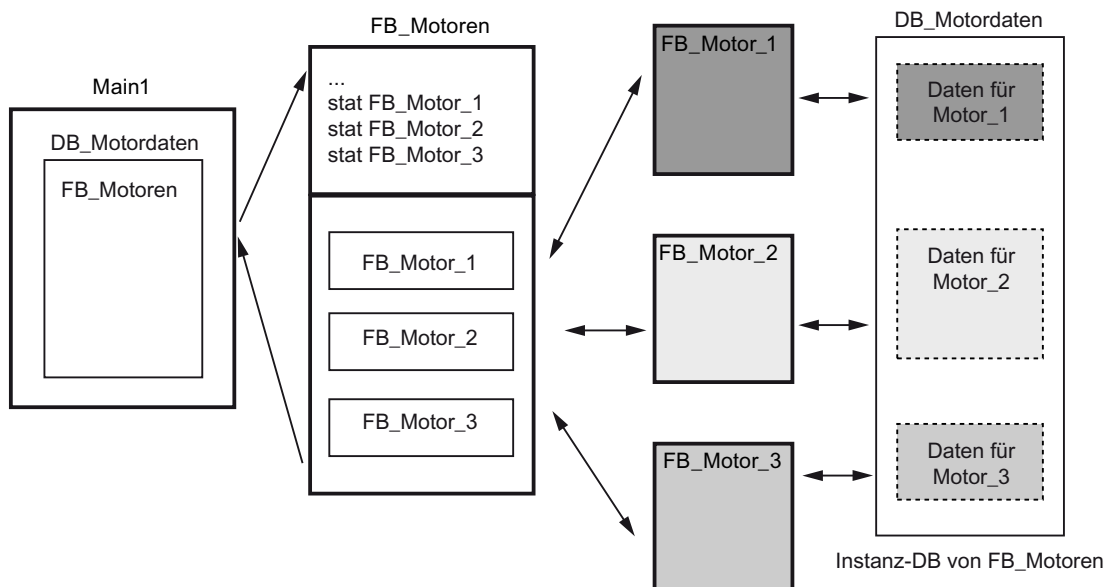


und FB\_Band auf. Die aufgerufenen Bausteine legen ihre Daten im DB\_Werkstück, dem Instanzdatenbaustein des aufrufenden Bausteins ab.



### Ein Instanz-Datenbaustein für mehrere Instanzen eines Funktionsbausteins

Das folgende Bild zeigt, wie ein Funktionsbaustein, der in mehreren Instanzen aufgerufen wird, die Daten für alle Instanzen in einem Instanz-Datenbaustein ablegt.



Der Funktionsbaustein FB\_Motoren ruft drei Instanzen des FB\_Motor auf. Die Instanzen sind "Motor\_1", "Motor\_2" und "Motor\_3". Für jeden Aufruf werden unterschiedliche Instanzdaten verwendet. Alle Instanzdaten liegen jedoch in einem einzigen Instanz-Datenbaustein, dem DB\_Motordaten.

## Siehe auch

Grundlagen zu Einzel- und Multiinstanzen (Seite 1041)  
Einzelinstanzen (Seite 1041)

## Parameterübergabe beim Bausteinaufruf

### Grundlagen zu Bausteinparametern

#### Einführung

Der aufrufende Baustein gibt dem aufgerufenen Baustein Werte mit, mit denen er arbeiten soll. Diese Werte werden als Bausteinparameter bezeichnet. Über die Eingangspartner erhält der aufgerufene Baustein die Werte, die er verarbeiten soll. Die Ergebnisse liefert der Baustein über die Ausgangspartner zurück.

Bausteinparameter stellen also die Schnittstelle zwischen dem aufrufenden und dem aufgerufenen Baustein dar.

Sie verwenden Eingangspartner, wenn Sie Werte nur abfragen oder lesen wollen, und Ausgangspartner, wenn Sie diese nur setzen oder schreiben wollen. Werden Bausteinparameter gelesen und geschrieben, müssen Sie sie als Durchgangspartner anlegen.

#### Formal- und Aktualparameter

Die Bausteinparameter sind in der Schnittstelle des aufgerufenen Bausteins definiert. Diese Parameter werden als Formalparameter bezeichnet. Sie sind Platzhalter für die Werte, die dem Baustein beim Aufruf übergeben werden. Die beim Aufruf übergebenen Werte werden als Aktualparameter bezeichnet.

#### Regel zur Verwendung der Bausteinparameter

Für die Verwendung der Bausteinparameter innerhalb des Bausteins gelten die folgenden Regeln:

- Eingangspartner dürfen nur gelesen werden.
- Ausgangspartner dürfen nur geschrieben werden.
- Durchgangspartner dürfen gelesen und geschrieben werden.

## Siehe auch

Parameterversorgung von Funktionsbausteinen (Seite 1048)  
Parameterversorgung von Funktionen (Seite 1046)  
Allgemeine Regeln zur Parameterversorgung (Seite 1045)  
Variablen (Seite 1056)  
Schlüsselwörter (Seite 1059)

## Bausteinparameter beim Aufruf versorgen

### Allgemeine Regeln zur Parameterversorgung

#### Einführung

Beim Aufruf eines Bausteins mit Bausteinparametern versorgen Sie dessen Formalparameter mit Aktualparametern. Dabei gelten die im Folgenden genannten Regeln.

#### Passende Datentypen

Die Datentypen von Aktual- und Formalparameter müssen gleich oder nach den Regeln der Datentypkonvertierung konvertierbar sein.

#### ARRAYs übergeben

Sie können ARRAYs als Parameter übergeben. Wenn ein Baustein einen Eingangsparameter vom Typ ARRAY hat, müssen Sie als Aktualparameter ein ARRAY mit identischem Aufbau übergeben. Sie können auch einzelne Elemente eines ARRAYs als Aktualparameter übergeben, wenn das Element dem Datentyp des Formalparameters entspricht.

#### PLC-Datentypen übergeben

Sie können auch Variablen, die als PLC-Datentyp deklariert sind, als Aktualparameter übergeben. Wenn der Formalparameter in der Variablendeklaration als PLC-Datentyp deklariert ist, müssen Sie eine Variable als Aktualparameter übergeben, die den gleichen PLC-Datentyp hat.

Auch ein Element einer mittels PLC-Datentyp deklarierten Variablen kann beim Bausteinaufruf als Aktualparameter übergeben werden, sofern der Datentyp des Elements der Variablen dem Datentyp des Formalparameters entspricht.

#### Strukturen (STRUCT) übergeben

Sie können Strukturen als Parameter übergeben. Wenn ein Baustein einen Eingangsparameter vom Typ STRUCT hat, müssen Sie als Aktualparameter ein STRUCT mit identischem Aufbau übergeben. Das bedeutet, dass die Namen und Datentypen aller Strukturkomponenten identisch sein müssen.

Sie können auch einzelne Elemente eines STRUCT als Aktualparameter übergeben, wenn das Element dem Datentyp des Formalparameters entspricht.

---

#### Hinweis

Wir empfehlen, Strukturen als PLC-Datentypen zu programmieren. PLC-Datentypen erleichtern die Programmierung, da sie mehrfach verwendet und zentral geändert werden können.

---

## Siehe auch

Parameterversorgung von Funktionsbausteinen (Seite 1048)

Parameterversorgung von Funktionen (Seite 1046)

Grundlagen zu Bausteinparametern (Seite 1044)

## Parameterversorgung von Funktionen

### Parameter von Funktionen (FC)

Funktionen haben keinen Datenspeicher, in denen Werte von Bausteinparametern gespeichert werden könnten. Deshalb müssen beim Aufruf einer Funktion alle Formalparameter mit Aktualparametern versorgt werden.

### Eingangsparameter (Input)

Eingangsparameter werden nur einmalig pro Zyklus gelesen, nämlich vor dem Bausteinaufruf. Deshalb gilt die Regel, dass das Schreiben eines Eingangsparameters innerhalb des Bausteins sich nicht auf den Aktualparameter auswirkt. Es wird nur der Formalparameter geschrieben.

### Ausgangsparameter (Output)

Ausgangsparameter werden nur einmalig pro Zyklus geschrieben, nämlich nach dem Bausteinaufruf. Deshalb gilt die Regel, dass Ausgangsparametern innerhalb des Bausteins nicht gelesen werden sollten. Wenn Sie dennoch einen Ausgangsparameter lesen, beachten Sie, dass dabei nur der Wert des Formalparameters gelesen wird. Der Wert des Aktualparameters kann innerhalb des Bausteins nicht gelesen werden.

Wenn ein Ausgangsparameter einer Funktion in dieser Funktion nicht geschrieben wird, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert. Strukturierte Ausgangsparameter werden jedoch nicht mit einem Wert vorbelegt.

Um nicht ungewollt den vordefinierten Wert oder einen undefinierten Wert weiterzuverarbeiten, beachten Sie bei der Programmierung des Bausteins Folgendes:

- Achten Sie darauf, dass die Ausgangsparameter für jeden möglichen Programmpfad innerhalb des Bausteins mit Werten beschrieben werden. Beachten Sie hierbei z. B., dass Sprungbefehle möglicherweise über Anweisungssequenzen springen, in denen Ausgänge gesetzt werden.
- Beachten Sie, dass Setze- und Rücksetze-Befehle vom Verknüpfungsergebnis abhängig sind. Wird der Wert eines Ausgangsparameters mit diesen Befehlen ermittelt, wird bei VKE = 0 kein Wert gebildet.
- Vergeben Sie für Ausgangsparameter von Funktionen wenn möglich einen Defaultwert.

## Durchgangparameter (InOut)

Durchgangparameter werden vor dem Bausteinaufruf gelesen und nach dem Bausteinaufruf geschrieben. Wenn Sie den Parameter innerhalb des Bausteins lesen oder schreiben, greifen Sie nur auf dessen Formalparameter zu.

Eine Ausnahme bilden die Durchgangparameter mit strukturiertem Datentyp. Strukturierte Datentypen sind die Datentypen, die aus mehreren Datenelementen bestehen, z. B. ARRAY oder STRUCT. Diese werden durch einen POINTER an den aufgerufenen Baustein übergeben. Sie greifen also immer auf den Aktualparameter zu, wenn Sie einen strukturierten Durchgangparameter innerhalb eines Bausteins lesen oder schreiben.

Wenn ein Durchgangparameter einer Funktion in dieser Funktion nicht geschrieben wird, wird der alte Ausgangswert bzw. der Eingangswert als Wert verwendet. Dennoch sollten Sie auch hier die oben für Ausgangsparameter gegebenen Hinweise beachten, um nicht unbeabsichtigt alte Werte weiter zu verarbeiten.

## Temporäre Lokaldaten (Temp)

Temporäre Lokaldaten stehen nur innerhalb eines Zyklus zur Verfügung. Sie werden abhängig von der Bausteinart unterschiedlich behandelt:

- Standardzugriff  
In Codebausteinen mit Standardzugriff sowie für alle Variablen mit der Remanenzeinstellung "Im IDB setzen" gilt die folgende Regel:  
Wenn Sie temporäre Lokaldaten verwenden, müssen Sie sicherstellen, dass die Werte innerhalb des Zyklus geschrieben werden, in dem Sie sie lesen wollen. Ansonsten sind die Werte zufällig. Eine Ausnahme sind temporäre Daten vom Datentyp STRING: Sie werden automatisch mit den richtigen Längeninformationen vorbelegt.
- Optimierter Zugriff  
In Codebausteinen mit optimiertem Zugriff gilt die folgende Regel:  
Wenn eine temporäre Variable innerhalb einer Funktion nicht geschrieben wird, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert. Elemente von PLC-Datentypen werden mit dem Defaultwert vorbelegt, der in der Deklaration des PLC-Datentyps (UDT) angegeben ist. ARRAY-Elemente werden mit dem Wert "0" vorbelegt, auch wenn sie innerhalb eines PLC-Datentyps verwendet werden. Strings werden mit den richtigen Längeninformationen vorbelegt.

## Funktionswert (Return)

Funktionen berechnen üblicherweise einen Funktionswert. Dieser Funktionswert kann über den Ausgangsparameter RET\_VAL an den aufrufenden Baustein zurückgeliefert werden. Dazu muss der Ausgangsparameter RET\_VAL in der Schnittstelle der Funktion deklariert sein. RET\_VAL ist immer der erste Ausgangsparameter einer Funktion. Für den Parameter RET\_VAL sind alle Datentypen zulässig mit Ausnahme von ARRAY und STRUCT sowie der Parametertypen TIMER und COUNTER.

In der Programmiersprache SCL können Funktionen direkt in einem Ausdruck aufgerufen werden. Das Ergebnis des Ausdrucks wird dann mit dem berechneten Funktionswert gebildet. In SCL ist deshalb der Datentyp ANY für den Funktionswert nicht zulässig.

**Siehe auch**

- Parameterversorgung von Funktionsbausteinen (Seite 1048)
- Grundlagen zu Bausteinparametern (Seite 1044)
- Allgemeine Regeln zur Parameterversorgung (Seite 1045)
- Aufruf von Funktionen (Seite 1389)
- Beispiele zum Aufruf von Funktionen in SCL (Seite 1392)

**Parameterversorgung von Funktionsbausteinen**

**Versorgung der Parameter von Funktionsbausteinen (FB)**

Bei Funktionsbausteinen werden die Parameterwerte in den Instanzdaten gespeichert.

Wenn Ein-, Aus- oder Durchgangparameter eines Funktionsbausteins nicht mit Werten versorgt wurden, werden die gespeicherten Werte verwendet.

In einigen Fällen wird zwingend die Angabe eines Aktualparameters verlangt.

Die folgende Tabelle zeigt, welchen Parametern eines Funktionsbausteins Aktualparameter zugeordnet werden müssen:

| Parameter         | Elementarer Datentyp  | Strukturierter Datentyp   | Parametertyp   |
|-------------------|---|---|--|
| Eingang (Input)   | optional  | optional  | erforderlich   |
| Ausgang (Output)  | optional  | optional  | erforderlich   |
| Durchgang (InOut) | optional  | erforderlich  | Nur bei S7-1200 zulässig, Parameterversorgung erforderlich |
| Temporär (Temp)   | erforderlich<br>S7-1500: Bei optimiertem Bausteinzugriff optional | erforderlich<br>S7-1500: Bei optimiertem Bausteinzugriff optional | erforderlich   |

**Siehe auch**

- Grundlagen zu Bausteinparametern (Seite 1044)
- Allgemeine Regeln zur Parameterversorgung (Seite 1045)
- Parameterversorgung von Funktionen (Seite 1046)

**Zugriff auf Bausteinparameter während der Programmbearbeitung**

**Einführung**

Bausteinparameter von Funktionen und Funktionsbausteinen werden während des Programmablaufs unterschiedlich bearbeitet. Die Art des Zugriffs variiert je nach CPU-Familie, Bausteinart und Datentyp des Bausteinparameters.

Grundsätzlich wird zwischen den beiden folgenden Zugriffsarten unterschieden:

- Parameter werden als Parameterpointer übergeben  
 An den aufgerufenen Baustein wird ein Parameterpointer übergeben.  
 Das bedeutet, dass der aufgerufene Baustein direkt auf den Operanden zugreifen kann, der als Aktualparameter angegeben ist. Das Beschreiben eines Parameters im aufgerufenen Baustein führt zur Veränderung des Aktualparameters im aufrufenden Baustein. Lesende Zugriffe auf einen Bausteinparameter lesen direkt den Aktualparameter.
- Parameter werden als Kopie übergeben  
 Beim Bausteinaufruf wird der Wert des Aktualparameters in die temporären Daten des aufgerufenen Bausteins kopiert.  
 Das bedeutet, dass der aufgerufene Baustein immer mit dem Wert arbeitet, den der Aktualparameter beim Bausteinaufruf hatte. Er kann nicht direkt auf den Operanden zugreifen, der als Aktualparameter angegeben ist. Das Beschreiben eines Parameters im aufgerufenen Baustein führt nicht zu einer Veränderung des Aktualparameters im aufrufenden Baustein. Lesende Zugriffe greifen auf den Formalparameter zu, nicht auf den Aktualparameter.  
 Beim Kopieren wird wie folgt vorgegangen:
  - Eingangsparmeter:  
 Der Wert des Aktualparameters wird beim Bausteinaufruf in den Formalparameter des aufgerufenen Bausteins kopiert.
  - Ausgangsparmeter:  
 Der Wert des Formalparameters wird nach dem Verlassen des Bausteins in den Aktualparameter kopiert.
  - Durchgangsparmeter:  
 Der Wert des Aktualparameters wird vor dem Bausteinaufruf in den Formalparameter des augerufenen Bausteins kopiert, und nach dem Verlassen der Funktion wieder in den Aktualparameter zurück kopiert.

### Parameterübergabe bei elementaren Datentypen

Die folgende Tabelle zeigt, wie die unterschiedlichen CPU-Familien Bausteinparameter mit elementarem Datentyp übergeben. Elementare Datentypen sind z.B. Bool, Int, Byte.

| Aufgerufener Baustein | Aktualparameter im Bereich  | S7-300/400 | S7-1500  |
|-----------------------|---|------------|--|
| FC                    | E, A, M, P, L   | Pointer    | Pointer  |
|                       | Teilqualifizierte DB-Adresse (z. B. DBW 2)  | Pointer    | teilqualifizierte DB-Adressen werden in S7-1500 nicht unterstützt. |
|                       | Vollqualifizierte DB-Adresse (z. B. "MyDB".Wert),<br>weiteregereichte Parameter des Aufrufers,<br>Statische Parameter des Aufrufers | Kopie      | Kopie  |
| FB                    | Alle Aktualparameter  | Kopie      | Kopie  |

### Parameterübergabe bei strukturierten Datentypen

Die folgende Tabelle zeigt, wie die unterschiedlichen CPU-Familien Bausteinparameter mit strukturiertem Datentyp übergeben. Strukturierte Datentypen sind Datentypen, die aus mehreren Datenelementen bestehen, z. B. ARRAY oder STRUCT.

| Aufgerufener Baustein | Aktualparameter im Bereich | S7-300/400 | S7-1500 |
|-----------------------|----------------------------|------------|---------|
| FC                    | IN, OUT, InOut             | Pointer    | Pointer |
| FB                    | IN, OUT                    | Kopie      | Kopie   |
|                       | InOut                      | Pointer    | Pointer |

### Weiterreichen von Bausteinparametern

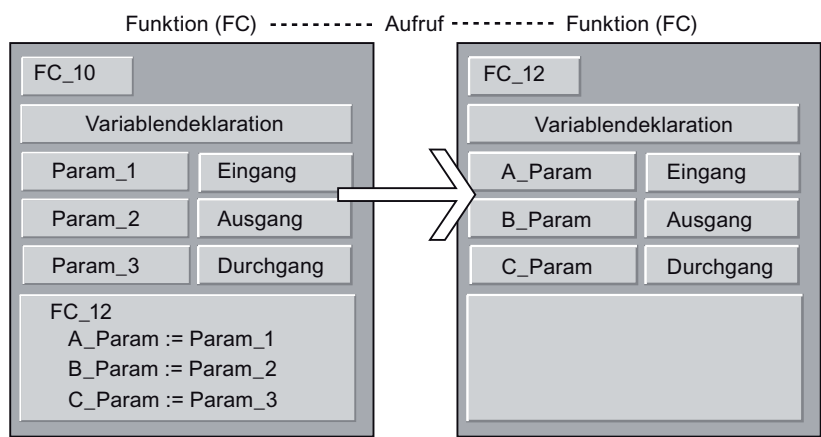
#### Grundlagen zum Weiterreichen von Bausteinparametern

#### Einführung

#### Definition

Eine besondere Art der Parameterverwendung ist das "Weiterreichen" von Bausteinparametern. Dabei werden die Bausteinparameter des aufrufenden Bausteins an die Parameter des aufgerufenen Bausteins weitergereicht. Der aufgerufene Baustein verwendet als Aktualparameter die Werte, die aktuell an den Bausteinparametern des aufrufenden Bausteins anstehen.

Das folgende Bild zeigt, wie die Parameter der Funktion FC\_10, an die Funktion FC\_12 weitergereicht werden:





## Regeln für KOP/FUP

In KOP und FUP gelten folgende allgemeine Regeln:

- Eingangsparameter können nur an Eingangsparameter weitergereicht werden.
- Ausgangsparameter können nur an Ausgangsparameter weitergereicht werden.
- Durchgangsparameter können an alle Parameterarten weitergereicht werden.
- In S7-300/400 müssen beide Bauteinparameter vom gleichen Datentyp sein.
- In S7-1200/1500 können die Parameter auch nach den Regeln der impliziten Konvertierung konvertiert werden.

## Regeln für AWL

In AWL gelten folgende allgemeine Regeln:

- Eingangsparameter können nur an Eingangsparameter weitergereicht werden.
- Ausgangsparameter können nur an Ausgangsparameter weitergereicht werden.
- Durchgangsparameter können an alle Parameterarten weitergereicht werden.
- Beide Bauteinparameter müssen vom gleichen Datentyp sein. Diese Regel gilt bei AWL in allen CPU-Familien.

## Regeln für SCL

Für SCL gelten weniger strenge Regeln. Damit Programme aus früheren SCL-Versionen leichter übernommen werden können, sind weitere Möglichkeiten der Parameterübergabe zwar zulässig, werden jedoch verwarnet. So können Sie z. B. einen Durchgangsparameter an einen Eingangsparameter weiterreichen, erhalten dabei jedoch eine Warnung, da der übergebene Durchgangsparameter vom Programm nicht beschrieben werden kann.

Weitere Regeln sind detailliert in den nachfolgenden Kapiteln beschrieben.

## Siehe auch

Aufruf einer Funktion durch eine andere Funktion (Seite 1052)

Aufruf einer Funktion durch einen Funktionsbaustein (Seite 1053)

Aufruf eines Funktionsbausteins durch eine Funktion (Seite 1054)

Aufruf eines Funktionsbausteins durch einen anderen Funktionsbaustein (Seite 1055)

**Aufruf einer Funktion durch eine andere Funktion**

**Zulässige Datentypen beim Aufruf einer Funktion durch eine andere Funktion**

Das Weiterreichen von Formalparametern unterliegt bestimmten Regeln. Die folgende Tabelle zeigt nach welchen Regeln Parameter in den unterschiedlichen CPU-Familien weitergereicht werden können:

| FC ruft FC                                |  | Datentypen                       |                                 |                 |                    |  |                             |
|---|--|----------------------------------|---------------------------------|-----------------|--------------------|--|-----------------------------|
| Aktualparameter<br>(aufrufender Baustein) | Formalparameter<br>(aufgerufener Baustein) | Standard-Datentypen              | ARRAY,<br>STRUCT,<br>STRING, DT | ANY,<br>POINTER | VARIANT            | Parametertypen<br>(TIMER,<br>COUNTER,<br>BLOCK_XX) | DB_Any                      |
| Input                                     | Input                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | S7-1500         | S7-1200<br>S7-1500 | S7-1500  | S7-1200 ab<br>V2<br>S7-1500 |
| Output                                    | Output                                     | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | -               | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | Input                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | S7-1500         | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | Output                                     | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | -               | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | InOut                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | S7-1500         | S7-1200<br>S7-1500 | -  | -                           |

**Siehe auch**

Grundlagen zum Weiterreichen von Bausteinparametern (Seite 1050)

## Aufruf einer Funktion durch einen Funktionsbaustein

### Zulässige Datentypen beim Aufruf einer Funktion durch einen Funktionsbaustein

Das Weiterreichen von Formalparametern unterliegt bestimmten Regeln. Die folgende Tabelle zeigt nach welchen Regeln Parameter in den unterschiedlichen CPU-Familien weitergereicht werden können:

| FB ruft FC                                |  | Datentypen                       |                                  |                 |                    |  |                             |
|---|--|----------------------------------|----------------------------------|-----------------|--------------------|--|-----------------------------|
| Aktualparameter<br>(aufrufender Baustein) | Formalparameter<br>(aufgerufener Baustein) | Standard-Datentypen              | ARRAY,<br>STRUCT,<br>STRING, DT  | ANY,<br>POINTER | VARIANT            | Parametertypen<br>(TIMER,<br>COUNTER,<br>BLOCK_XX) | DB_Any                      |
| Input                                     | Input                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-300/400<br>S7-1200<br>S7-1500 | S7-1500         | S7-1200<br>S7-1500 | S7-1500  | S7-1200 ab<br>V2<br>S7-1500 |
| Output                                    | Output                                     | S7-300/400<br>S7-1200<br>S7-1500 | S7-300/400<br>S7-1200<br>S7-1500 | -               | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | Input                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500               | S7-1500         | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | Output                                     | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500               | -               | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | InOut                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500               | S7-1500         | S7-1200<br>S7-1500 | -  | -                           |

### Siehe auch

Grundlagen zum Weiterreichen von Bausteinparametern (Seite 1050)

**Aufruf eines Funktionsbausteins durch eine Funktion**

**Zulässige Datentypen beim Aufruf eines Funktionsbausteins durch eine Funktion**

Das Weiterreichen von Formalparametern unterliegt bestimmten Regeln. Die folgende Tabelle zeigt nach welchen Regeln Parameter in den unterschiedlichen CPU-Familien weitergereicht werden können:

| FC ruft FB                                |  | Datentypen                       |                                 |                 |                    |  |                             |
|---|--|----------------------------------|---------------------------------|-----------------|--------------------|--|-----------------------------|
| Aktualparameter<br>(aufrufender Baustein) | Formalparameter<br>(aufgerufener Baustein) | Standard-Datentypen              | ARRAY,<br>STRUCT,<br>STRING, DT | ANY,<br>POINTER | VARIANT            | Parametertypen<br>(TIMER,<br>COUNTER,<br>BLOCK_XX) | DB_Any                      |
| Input                                     | Input                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | S7-1500         | S7-1200<br>S7-1500 | S7-300/400<br>S7-1500                              | S7-1200 ab<br>V2<br>S7-1500 |
| Output                                    | Output                                     | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | -               | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | Input                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | S7-1500         | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | Output                                     | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | -               | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | InOut                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500              | S7-1500         | S7-1200<br>S7-1500 | -  | -                           |

**Siehe auch**

Grundlagen zum Weiterreichen von Bausteinparametern (Seite 1050)

## Aufruf eines Funktionsbausteins durch einen anderen Funktionsbaustein

### Zulässige Datentypen beim Aufruf eines Funktionsbausteins durch einen anderen Funktionsbaustein

Das Weiterreichen von Formalparametern unterliegt bestimmten Regeln. Die folgende Tabelle zeigt nach welchen Regeln Parameter in den unterschiedlichen CPU-Familien weitergereicht werden können:

| FB ruft FB                                |  | Datentypen                       |                                  |                 |                    |  |                             |
|---|--|----------------------------------|----------------------------------|-----------------|--------------------|--|-----------------------------|
| Aktualparameter<br>(aufrufender Baustein) | Formalparameter<br>(aufgerufener Baustein) | Standard-Datentypen              | ARRAY,<br>STRUCT,<br>STRING, DT  | ANY,<br>POINTER | VARIANT            | Parametertypen<br>(TIMER,<br>COUNTER,<br>BLOCK_XX) | DB_Any                      |
| Input                                     | Input                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-300/400<br>S7-1200<br>S7-1500 | S7-1500         | S7-1200<br>S7-1500 | S7-300/400<br>S7-1500                              | S7-1200 ab<br>V2<br>S7-1500 |
| Output                                    | Output                                     | S7-300/400<br>S7-1200<br>S7-1500 | S7-300/400<br>S7-1200<br>S7-1500 | -               | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | Input                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500               | S7-1500         | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | Output                                     | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500               | -               | S7-1200<br>S7-1500 | -  | -                           |
| InOut                                     | InOut                                      | S7-300/400<br>S7-1200<br>S7-1500 | S7-1200<br>S7-1500               | S7-1500         | S7-1200<br>S7-1500 | -  | -                           |

### Siehe auch

Grundlagen zum Weiterreichen von Bausteinparametern (Seite 1050)

#### 9.1.1.4 Operanden verwenden und adressieren

### Grundlagen zu Operanden

#### Einführung

Bei der Programmierung von Anweisungen müssen Sie angeben, welche Datenwerte die Anweisung verarbeiten soll. Diese Werte werden als Operanden bezeichnet. Sie können z. B. folgende Elemente als Operanden verwenden:

- PLC-Variablen
- Konstanten
- Variablen in Instanzdatenbausteinen
- Variablen in globalen Datenbausteinen

#### Absolute Adresse und symbolischer Name

Operanden werden über eine absolute Adresse und einen symbolischen Namen identifiziert. Die Namen und Adressen definieren Sie in der PLC-Variablen-tabelle oder in der Variablen-deklaration der Bausteine.

#### Datenbausteine mit optimiertem Zugriff (S7-1200, S7-1500)

Datenelemente in Datenbausteinen mit optimiertem Zugriff erhalten in der Deklaration nur einen symbolischen Namen, keine absolute Adresse. Weitere Informationen hierzu finden Sie unter "Siehe auch".

#### Siehe auch

Symbolische und absolute Operanden anzeigen (Seite 1246)

Grundlagen zum Bausteinzugriff (Seite 1034)

#### Variablen

#### Definition

Eine Variable ist ein Platzhalter für einen im Programm veränderbaren Datenwert. Das Format des Datenwerts ist festgelegt. Durch die Verwendung von Variablen wird Ihr Programm flexibler. Sie können z. B. Variablen, die Sie in der Schnittstelle von Bausteinen deklariert haben, bei jedem Aufruf des Bausteins mit anderen Werten versorgen. So können Sie einen einmal programmierten Baustein für verschiedene Zwecke wiederverwenden.

Eine Variable besteht aus folgenden Elementen:

- Name
- Datentyp

- Absolute Adresse
  - PLC-Variablen und DB-Variablen in Bausteinen mit Standardzugriff haben eine absolute Adresse.
  - DB-Variablen in Bausteinen mit optimiertem Zugriff haben keine absolute Adresse.
- Wert (optional)

### Deklaration von Variablen

Sie können für Ihr Programm Variablen mit unterschiedlichem Geltungsbereich definieren:

- PLC-Variablen, die CPU-weit gelten
- DB-Variablen in globalen Datenbausteinen, die CPU-weit von allen Bausteinen genutzt werden können.
- DB-Variablen in Instanzdatenbausteinen, die vorrangig in dem Baustein genutzt werden, in dem sie deklariert sind.

Die folgende Tabelle zeigt den Unterschied zwischen den Variablenarten:

|                    | PLC-Variablen  | Variablen in Instanz-DBs   | Variablen in globalen DBs   |
|--------------------|--|--|---|
| Gültigkeitsbereich | <ul style="list-style-type: none"> <li>• Gelten in der gesamten CPU.</li> <li>• Können von allen Bausteinen in der CPU benutzt werden.</li> <li>• Der Name ist innerhalb der CPU eindeutig.</li> </ul> | <ul style="list-style-type: none"> <li>• Werden vorrangig in dem Baustein genutzt, in dem sie definiert sind.</li> <li>• Der Name ist innerhalb des Instanz-DB eindeutig.</li> </ul> | <ul style="list-style-type: none"> <li>• Können von allen Bausteinen in der CPU benutzt werden.</li> <li>• Der Name ist innerhalb des globalen DB eindeutig.</li> </ul> |
| Zulässige Zeichen  | <ul style="list-style-type: none"> <li>• Buchstaben, Ziffern, Sonderzeichen</li> <li>• Anführungszeichen sind nicht zulässig.</li> <li>• Reservierte Schlüsselwörter sind nicht zulässig.</li> </ul>   | <ul style="list-style-type: none"> <li>• Buchstaben, Ziffern, Sonderzeichen</li> <li>• Reservierte Schlüsselwörter sind nicht zulässig.</li> </ul>                                   | <ul style="list-style-type: none"> <li>• Buchstaben, Ziffern, Sonderzeichen</li> <li>• Reservierte Schlüsselwörter sind nicht zulässig.</li> </ul>                      |
| Einsatz            | <ul style="list-style-type: none"> <li>• E/A-Signale (E, EB, EW, ED, A, AB, AW, AD)</li> <li>• Merker (M, MB, MW, MD)</li> </ul>   | <ul style="list-style-type: none"> <li>• Bausteinparameter (Eingangs-, Ausgangs- und Durchgangparameter),</li> <li>• Statische Daten eines Bausteins</li> </ul>                      | <ul style="list-style-type: none"> <li>• Statische Daten</li> </ul>   |
| Ort der Definition | PLC-Variablentabelle   | Schnittstelle des Bausteins  | Deklarationstabelle des globalen DB   |

## Siehe auch

- Schlüsselwörter (Seite 1059)
- Grundlagen zu Operanden (Seite 1056)
- Symbolische und absolute Operanden anzeigen (Seite 1246)
- Zulässige Namen von PLC-Variablen (Seite 1188)
- Zulässige Adressen und Datentypen von PLC-Variablen (Seite 1189)

## Konstanten

### Definition

Eine Konstante definiert einen unveränderlichen Datenwert. Konstanten können während der Programmbearbeitung von verschiedenen Programmelementen ausgelesen, aber nicht überschrieben werden. Eine Änderung des Konstantenwerts zur Laufzeit des Programms kann zu Syntax- oder Laufzeitfehlern führen.

### Symbolische Konstanten (S7-1200, S7-1500)

In S7-1200 und S7-1500 haben Sie die Möglichkeit, symbolische Namen für Konstanten zu deklarieren, und so statische Werte unter einem Namen im Programm verfügbar zu machen. Diese symbolischen Konstanten sind CPU-weit gültig. Die Deklaration der Konstanten erfolgt im Register "Konstanten" der PLC-Variablen-tabelle.

### Elemente von Konstanten

Eine Konstante besteht aus folgenden Elementen:

- Name (bei symbolischen Konstanten)  
Zulässige Zeichen im Konstantennamen sind Buchstaben, Ziffern und Sonderzeichen, unzulässige Zeichen sind Anführungszeichen und reservierte Schlüsselwörter.
- Datentyp
- Konstantenwert  
Das Eingabeformat und der Wertebereich des Konstantenwerts hängen dabei vom Datentyp der Konstanten ab.

---

#### Hinweis

#### Konstanten vom Typ BOOL

Konstanten vom Typ BOOL dürfen in S7-300/400 nicht als Eingänge in KOP oder FUP verwendet werden.

---



## Weitere Informationen

Weitere Informationen zu Datentypen von Konstanten sowie deren Eingabeformaten und Wertebereichen finden Sie im Kapitel "Datentypen" unter "Siehe auch".

## Siehe auch

Regeln für symbolische Konstanten (Seite 1202)

Konstanten eingeben (Seite 1085)

Konstanten deklarieren (Seite 1202)

## Schlüsselwörter

SIMATIC kennt eine Reihe fest definierter Schlüsselwörter, die im Programm eine bestimmte Bedeutung haben. Sie sollten diese Schlüsselwörter nicht als Namen für Variablen oder Konstanten verwenden.

## Tabelle der Schlüsselwörter

Die folgende Tabelle zeigt alle Schlüsselwörter.

| Schlüsselwörter<br>Deutsche Mnemonik | Schlüsselwörter<br>Englische Mnemonik | Beschreibung   |
|--------------------------------------|---------------------------------------|--|
| &                                    | &                                     | Und-Verknüpfung logischer Ausdrücke  |
| A                                    | Q                                     | Ausgang, Bit   |
| A1                                   | CC1                                   | Anzeigebit   |
| A0                                   | CC0                                   | Anzeigebit   |
| AB                                   | QB                                    | Ausgang, Byte  |
| AD                                   | QD                                    | Ausgang, Doppelwort  |
| AND                                  | AND                                   | Und-Verknüpfung logischer Ausdrücke  |
| ANY                                  | ANY                                   | Datentyp, Zeiger   |
| AR1                                  | AR1                                   | Adressregister 1   |
| AR2                                  | AR2                                   | Adressregister 2   |
| ARRAY                                | ARRAY                                 | Einleitung der Spezifikation eines Arrays, danach folgt zwischen "[" und "]" die Indexliste      |
| AT                                   | AT                                    | Überlagernde Variablendeklaration  |
| AUTHOR                               | AUTHOR                                | Name des Autors, Firmenname, Abteilungsname oder andere Namen (max. 8 Zeichen, ohne Leerzeichen) |
| AW                                   | QW                                    | Ausgang, Wort  |
| B                                    | B                                     | Byte   |

| Schlüsselwörter<br>Deutsche Mnemonik | Schlüsselwörter<br>Englische Mnemonik | Beschreibung   |
|--------------------------------------|---------------------------------------|--|
| BEGIN                                | BEGIN                                 | Einleitung Anweisungsteil bei Codebausteinen oder Initialisierungsteil bei Datenbaustein   |
| BIE                                  | BR                                    | Binäres Ergebnis   |
| BLOCK_FB                             | BLOCK_FB                              | Parametertyp zur Angabe eines FB   |
| BLOCK_FC                             | BLOCK_FC                              | Parametertyp zur Angabe einer FC   |
| BLOCK_SDB                            | BLOCK_SDB                             | Parametertyp zur Angabe eines SDB  |
| BOOL                                 | BOOL                                  | Datentyp   |
| BY                                   | BY                                    | Schrittweite der FOR-Schleife  |
| BYTE                                 | BYTE                                  | Datentyp   |
| CALL                                 | CALL                                  | Aufruf   |
| CASE                                 | CASE                                  | Einleitung der CASE-Anweisung  |
| CHAR                                 | CHAR                                  | Elementarer Datentyp   |
| CODE_VERSION1                        | CODE_VERSION1                         | Kennung, ob ein FB multiinstanzfähig ist oder nicht. Wenn Sie Multiinstanzen deklarieren möchten, darf der FB diese Eigenschaft nicht haben. |
| CONST                                | CONST                                 | Beginn der Konstantendeklaration   |
| CONTINUE                             | CONTINUE                              | Anweisung zum Verlassen einer Schleife in SCL  |
| COUNTER                              | COUNTER                               | Parametertyp zur Angabe eines Zählers  |
| DATA_BLOCK                           | DATA_BLOCK                            | Einleitung des Datenbausteins  |
| DATE                                 | DATE                                  | Datentyp   |
| DATE_AND_TIME                        | DATE_AND_TIME                         | Datentyp   |
| DB                                   | DB                                    | Datenbaustein  |
| DB_ANY                               | DB_ANY                                | Datentyp   |
| DBB                                  | DBB                                   | Datenbaustein, Datenbyte   |
| DBD                                  | DBD                                   | Datenbaustein, Datendoppelwort   |
| DBLG                                 | DBLG                                  | Datenbausteinlänge   |
| DBNO                                 | DBNO                                  | Datenbausteinnummer  |
| DBW                                  | DBW                                   | Datenbaustein, Datenwort   |
| DBX                                  | DBX                                   | Datenbaustein, Datenbit  |
| DI                                   | DI                                    | Instanz-Datenbaustein  |
| DIB                                  | DIB                                   | Instanz-Datenbaustein, Datenbyte   |
| DID                                  | DID                                   | Instanz-Datenbaustein, Datendoppelwort   |
| DILG                                 | DILG                                  | Instanzdatenbausteinlänge  |

| Schlüsselwörter<br>Deutsche Mnemonik | Schlüsselwörter<br>Englische Mnemonik | Beschreibung  |
|--------------------------------------|---------------------------------------|---|
| DINO                                 | DINO                                  | Instanzdatenbausteinnummer                                    |
| DINT                                 | DINT                                  | Datentyp  |
| DIW                                  | DIW                                   | Instanz-Datenbaustein,<br>Datenwort                           |
| DIX                                  | DIX                                   | Instanz-Datenbaustein, Datenbit                               |
| DO                                   | DO                                    | Einleitung des Anweisungsteils<br>in FOR- und WHILE-Anweisung |
| DT                                   | DT                                    | Datentyp  |
| DTL                                  | DTL                                   | Datentyp  |
| DWORD                                | DWORD                                 | Datentyp  |
| E                                    | I                                     | Eingang (über Prozessabbild),<br>Bit                          |
| EB                                   | IB                                    | Eingang (über Prozessabbild),<br>Byte                         |
| ED                                   | ID                                    | Eingang (über Prozessabbild),<br>Doppelwort                   |
| ELSE                                 | ELSE                                  | Alternativer Zweig in IF- und<br>CASE-Anweisung               |
| ELSIF                                | ELSIF                                 | Alternative Bedingung der IF-<br>Anweisung                    |
| EN                                   | EN                                    | Systemoperand des EN/ENO-<br>Mechanismus                      |
| ENO                                  | ENO                                   | Systemoperand des EN/ENO-<br>Mechanismus                      |
| END_CASE                             | END_CASE                              | Abschluss der CASE-Anweisung                                  |
| END_DATA_BLOCK                       | END_DATA_BLOCK                        | Abschluss des Datenbausteins                                  |
| END_FOR                              | END_FOR                               | Abschluss der FOR-Anweisung                                   |
| END_FUNCTION                         | END_FUNCTION                          | Abschluss der Funktion  |
| END_FUNCTION_BLOCK                   | END_FUNCTION_BLOCK                    | Abschluss des<br>Funktionsbausteins                           |
| END_IF                               | END_IF                                | Abschluss der IF-Anweisung                                    |
| END_ORGANIZATION_BLOCK               | END_ORGANIZATION_BLOCK                | Abschluss des<br>Organisationsbausteins                       |
| END_REPEAT                           | END_REPEAT                            | Abschluss der REPEAT-<br>Anweisung                            |
| END_STRUCT                           | END_STRUCT                            | Abschluss der Spezifikation<br>einer Struktur                 |
| END_SYSTEM_FUNCTION                  | END_SYSTEM_FUNCTION                   | Abschluss der Systemfunktion                                  |
| END_SYSTEM_FUNCTION_BLOCK            | END_SYSTEM_FUNCTION_BLOCK             | Abschluss des<br>Systemfunktionsbausteins                     |
| END_TYPE                             | END_TYPE                              | Abschluss des PLC-Datentyps                                   |
| END_VAR                              | END_VAR                               | Abschluss eines<br>Deklarationsblocks                         |
| END_WHILE                            | END_WHILE                             | Abschluss der WHILE-<br>Anweisung                             |

| Schlüsselwörter<br>Deutsche Mnemonik | Schlüsselwörter<br>Englische Mnemonik | Beschreibung   |
|--------------------------------------|---------------------------------------|--|
| EW                                   | IW                                    | Eingang (über Prozessabbild), Wort   |
| EXIT                                 | EXIT                                  | Anweisung zum Verlassen einer Schleife in SCL  |
| FALSE                                | FALSE                                 | Vordefinierte boolesche Konstante: Logische Bedingung nicht erfüllt, Wert gleich 0         |
| FAMILY                               | FAMILY                                | Name der Bausteinfamilie: z. B. Regler   |
| FB                                   | FB                                    | Funktionsbaustein  |
| FC                                   | FC                                    | Funktion   |
| FOR                                  | FOR                                   | Einleitung der FOR-Anweisung   |
| FUNCTION                             | FUNCTION                              | Einleitung der Funktion  |
| FUNCTION_BLOCK                       | FUNCTION_BLOCK                        | Einleitung des Funktionsbausteins  |
| GOTO                                 | GOTO                                  | Einleitung der GOTO-Anweisung  |
| IF                                   | IF                                    | Einleitung der IF-Anweisung  |
| INSTANCE                             | INSTANCE                              | Datentyp   |
| INT                                  | INT                                   | Datentyp   |
| KNOW_HOW_PROTECT                     | KNOW_HOW_PROTECT                      | Bausteinschutz   |
| L                                    | L                                     | Lokaldatenbit  |
| LB                                   | LB                                    | Lokaldatenbyte   |
| LD                                   | LD                                    | Lokaldatendoppelwort   |
| LDT                                  | LDT                                   | Datentyp   |
| LINT                                 | LINT                                  | Datentyp   |
| LTIME                                | LTIME                                 | Datentyp   |
| LTOD                                 | LTOD                                  | Datentyp   |
| LW                                   | LW                                    | Lokaldatenwort   |
| LWORD                                | LWORD                                 | Datentyp   |
| M                                    | M                                     | Merkerbit  |
| MB                                   | MB                                    | Merkerbyte   |
| MD                                   | MD                                    | Merkerdoppelwort   |
| MOD                                  | MOD                                   | Modulo-Operator  |
| MW                                   | MW                                    | Merkerwort   |
| NAME                                 | NAME                                  | Bausteinname   |
| NETWORK                              | NETWORK                               | Netzwerk   |
| NOT                                  | NOT                                   | Logische Inversion   |
| NULL                                 | NULL                                  | Null-Pointer   |
| OB                                   | OB                                    | Organisationsbaustein  |
| OF                                   | OF                                    | Einleitung der Datentypspezifikation/<br>Einleitung des Anweisungsteils der CASE-Anweisung |

| Schlüsselwörter<br>Deutsche Mnemonik | Schlüsselwörter<br>Englische Mnemonik | Beschreibung  |
|--------------------------------------|---------------------------------------|---|
| OR                                   | OR                                    | Oder-Verknüpfung logischer Ausdrücke  |
| ORGANIZATION_BLOCK                   | ORGANIZATION_BLOCK                    | Einleitung des Organisationsbausteins   |
| OS                                   | OS                                    | Überlauf, speichernd  |
| OV                                   | OV                                    | Überlauf  |
| PA                                   | PQ                                    | Ausgang (Peripherie direkt), Bit  |
| PAB                                  | PQB                                   | Ausgang (Peripherie direkt), Byte   |
| PAD                                  | PQD                                   | Ausgang (Peripherie direkt), Doppelwort   |
| PAW                                  | PQW                                   | Ausgang (Peripherie direkt), Wort   |
| PE                                   | PI                                    | Eingang (Peripherie direkt), Bit  |
| PEB                                  | PIB                                   | Eingang (Peripherie direkt), Byte   |
| PED                                  | PID                                   | Eingang (Peripherie direkt), Doppelwort   |
| PEW                                  | PIW                                   | Eingang (Peripherie direkt), Wort   |
| POINTER                              | POINTER                               | Datentyp  |
| READ_ONLY                            | READ_ONLY                             | Schreibschutz für Datenbausteine  |
| REAL                                 | REAL                                  | Datentyp  |
| REPEAT                               | REPEAT                                | Einleitung der REPEAT-Anweisung   |
| RET_VAL                              | RET_VAL                               | Rücksprungwert  |
| RETURN                               | RETURN                                | RETURN-Anweisung in SCL   |
| S5T                                  | S5T                                   | Syntax für Datentyp S5TIME  |
| S5TIME                               | S5TIME                                | Datentyp  |
| S7_                                  | S7_                                   | Schlüsselwörter für Systemattribute   |
| SDB                                  | SDB                                   | Systemdatenbaustein   |
| SFB                                  | SFB                                   | Systemfunktionsbaustein   |
| SFC                                  | SFC                                   | Systemfunktion  |
| SINT                                 | SINT                                  | Datentyp  |
| STRING                               | STRING                                | Datentyp  |
| STRUCT                               | STRUCT                                | Einleitung der Spezifikation einer Struktur, danach folgt Liste der Komponenten |
| STW                                  | STW                                   | Statuswort  |
| SYSTEM_FUNCTION                      | SYSTEM_FUNCTION                       | Systemfunktion  |
| SYSTEM_FUNCTION_BLOCK                | SYSTEM_FUNCTION_BLOCK                 | Systemfunktionsbaustein   |
| T                                    | T                                     | Zeitglied (Timer)   |
| THEN                                 | THEN                                  | Einleitung des Anweisungsteils einer IF-Anweisung                               |

| Schlüsselwörter<br>Deutsche Mnemonik | Schlüsselwörter<br>Englische Mnemonik | Beschreibung   |
|--------------------------------------|---------------------------------------|--|
| THIS                                 | THIS                                  | Syntax für den Zugriff auf einen ARRAY-Datenbaustein                           |
| TIME                                 | TIME                                  | Elementarer Datentyp für Zeitangaben   |
| TIME_OF_DAY                          | TIME_OF_DAY                           | Datentyp   |
| TIMER                                | TIMER                                 | Parametertyp zur Angabe einer Zeit   |
| TITLE                                | TITLE                                 | Optionalen Baustein- oder Netzwerktitel  |
| TO                                   | TO                                    | Definition des Endwerts einer FOR-Anweisung                                    |
| TOD                                  | TOD                                   | Datentyp   |
| TRUE                                 | TRUE                                  | Vordefinierte boolesche Konstante: Logische Bedingung erfüllt, Wert ungleich 0 |
| TYPE                                 | TYPE                                  | Einleitung des PLC-Datentyps   |
| UDT                                  | UDT                                   | globaler bzw. PLC-Datentyp   |
| UDINT                                | UDINT                                 | Datentyp   |
| UINT                                 | UINT                                  | Datentyp   |
| ULINT                                | ULINT                                 | Datentyp   |
| UNLINKED                             | UNLINKED                              | Kennzeichnung 'nicht ablaufrelevant'   |
| UNTIL                                | UNTIL                                 | Abschluss des Anweisungsteils einer REPEAT-Anweisung                           |
| USINT                                | USINT                                 | Datentyp   |
| UO                                   | UO                                    | Abfrage nach (A1=1) UND (A0=1)   |
| VAR                                  | VAR                                   | Einleitung eines Deklarationsblocks  |
| VAR_IN_OUT                           | VAR_IN_OUT                            | Einleitung eines Deklarationsblocks  |
| VAR_INPUT                            | VAR_INPUT                             | Einleitung eines Deklarationsblocks  |
| VAR_OUTPUT                           | VAR_OUTPUT                            | Einleitung eines Deklarationsblocks  |
| VAR_TEMP                             | VAR_TEMP                              | Einleitung eines Deklarationsblocks  |
| VARIANT                              | VARIANT                               | Datentyp   |
| VERSION                              | VERSION                               | Versionsnummer des Bausteins   |
| VOID                                 | VOID                                  | Funktion hat keinen Rückgabewert   |
| WCHAR                                | WCHAR                                 | Datentyp   |
| WSTRING                              | WSTRING                               | Datentyp   |
| WHILE                                | WHILE                                 | Einleitung einer WHILE-Anweisung   |
| WORD                                 | WORD                                  | Datentyp   |

| Schlüsselwörter<br>Deutsche Mnemonik | Schlüsselwörter<br>Englische Mnemonik | Beschreibung       |
|--------------------------------------|---------------------------------------|--------------------|
| XOR                                  | XOR                                   | Logische Operation |
| Z                                    | C                                     | Zähler (Counter)   |

## Operanden adressieren

### Globale Variablen adressieren

### Globale Variablen adressieren

Zur Adressierung einer globalen PLC-Variablen können Sie die absolute Adresse oder den symbolischen Namen verwenden.

#### Hinweis

Die Datentypen LWORD, LINT, ULINT, LREAL, LTIME, LTOD und LDT können nur mit ihrem symbolischen Namen adressiert werden.

### Globale Variablen symbolisch adressieren

Bei der symbolischen Adressierung geben Sie den Variablennamen aus der PLC-Variablen-tabelle ein. Der symbolische Name von globalen Variablen wird automatisch in Anführungszeichen eingeschlossen.

Strukturierte Variablen, die auf einem PLC-Datentyp basieren, adressieren Sie mit dem symbolischen Namen der PLC-Variablen. Getrennt durch einen Punkt können Sie auch die Namen der einzelnen Komponenten angeben.

### Globale Variablen absolut adressieren

Bei der absoluten Adressierung geben Sie die Adresse der Variablen aus der PLC-Variablen-tabelle ein. Die absolute Adresse verwendet numerische Adressen beginnend bei Null für jeden Operandenbereich. Der absoluten Adresse von globalen Variablen wird automatisch das Adresskennzeichen % vorangestellt.

## Beispiele

Die folgenden Beispiele zeigen Anwendungen der symbolischen und absoluten Adressierung:

| Adressierung | Erläuterung                      |
|--------------|----------------------------------|
| %A1.0        | Absolute Adresse: Ausgang 1.0    |
| %E16.4       | Absolute Adresse: Eingang 16.4   |
| %EW4         | Absolute Adresse: Eingangswort 4 |
| "Motor"      | Symbolische Adresse "Motor"      |
| "Value"      | Symbolische Adresse "Value"      |

| Adressierung                        | Erläuterung  |
|-------------------------------------|--|
| "Strukturierte_Variable"            | Symbolische Adresse einer Variable, die auf einem PLC-Datentyp basiert |
| "Strukturierte_Variable".Komponente | Symbolische Adresse der Komponente einer strukturierten Variable.      |

Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 1189)

### Siehe auch

Symbolische und absolute Operanden anzeigen (Seite 1246)

Peripherie adressieren (Seite 1066)

### Peripherie adressieren

#### Beschreibung


In einem Zyklus wird das Prozessabbild der CPU einmal aktualisiert. In zeitkritischen Anwendungen kann es jedoch sein, dass Sie den aktuellen Zustand eines digitalen Eingangs oder Ausgangs häufiger als einmal pro Zyklus lesen oder übertragen müssen. Zu diesem Zweck können Sie mithilfe einer Peripheriezugriffskennung am Operanden direkt auf die Peripherie zugreifen.

Wenn Sie den Eingang direkt aus der Peripherie lesen möchten, verwenden Sie den Speicherbereich Peripherieeingänge (PE) im Gegensatz zum Prozessabbild der Eingänge (E). Der Speicherbereich der Peripherie kann als Bit, Byte, Wort oder Doppelwort gelesen werden.

Wenn Sie den Ausgang direkt schreiben möchten, verwenden Sie den Speicherbereich der Peripherieausgänge (PA) im Gegensatz zum Prozessabbild der Ausgänge (A). Der Speicherbereich der Peripherieausgänge kann als Bit, Byte, Wort und Doppelwort beschrieben werden.

Um ein Signal direkt von einem Peripherieeingang zu lesen oder zu schreiben, kann ein Operand um die Peripheriezugriffskennung ":P" erweitert werden.

Komponenten von strukturierten PLC-Variablen können ebenfalls mit ":P" adressiert werden. Ein Zugriff auf die übergeordnete Variable mit ":P" ist jedoch nicht möglich.

|   |
|---|
|  <b>WARNUNG</b>  |
| <b>Direktes Schreiben der Peripherie</b>  |
| Das direkte Schreiben der Peripherie kann gefährliche Zustände hervorrufen, z. B. wenn ein Peripherieausgang in einem Programmzyklus mehrfach schreibend adressiert wird. |

#### Syntax

<Operand>:P



## Beispiel

Das folgende Beispiel zeigt Anwendungen der Peripheriezugriffskennung:

| Adressierung                          | Erläuterung   |
|---------------------------------------|---|
| "Motor"                               | Adressiert die Variable "Motor" im Prozessabbild.   |
| "Motor":P                             | Adressiert die Variable "Motor" im Speicherbereich Peripherie (PE oder PA).                             |
| "Strukturierte_Variable".Komponente   | Adressiert die Komponente einer strukturierten PLC-Variable im Prozessabbild.                           |
| "Strukturierte_Variable".Komponente:P | Adressiert die Komponente einer strukturierten PLC-Variable im Speicherbereich Peripherie (PE oder PA). |

## Siehe auch

Globale Variablen adressieren (Seite 1065)

## Variablen in Datenbausteinen adressieren

## Variablen in globalen Datenbausteinen adressieren

## Beschreibung

Variablen in globalen Datenbausteinen können symbolisch oder absolut adressiert werden. Bei symbolischer Adressierung verwenden Sie den Namen des Datenbausteins und den Namen der Variablen, getrennt durch einen Punkt. Der Name des Datenbausteins wird in Anführungszeichen gesetzt.

Bei absoluter Adressierung verwenden Sie die Nummer des Datenbausteins und die absolute Adresse der Variablen im Datenbaustein, getrennt durch einen Punkt. Der absoluten Adresse wird automatisch das Adresskennzeichen % vorangestellt.

In S7-1200/1500 haben Sie die Möglichkeit, auf einen Datenbaustein zuzugreifen, der bei der Programmierung noch nicht bekannt ist. Dazu erstellen Sie in der Bausteinschnittstelle des zugreifenden Bausteins einen Bausteinparameter vom Datentyp DB\_Any. An diesem Parameter wird zur Laufzeit der Datenbausteinname oder die Datenbausteinnummer übergeben. Um auf die Variablen innerhalb des Datenbausteins zuzugreifen, verwenden Sie den Namen des Bausteinparameters vom Datentyp DB\_Any und die absolute Adresse der Variablen, getrennt durch einen Punkt.

---

### Hinweis

Folgende Variablen können nicht absolut adressiert werden:

- Variablen in Bausteinen mit optimiertem Zugriff.
- Variablen der Datentypen LWORD, LINT, ULINT, LREAL, LTIME, LTOD und LDT.

Verwenden Sie für diese Variablen die komfortablere symbolische Adressierung.

---

### ARRAY-Datenbausteine

Eine besondere Art von globalen Datenbausteinen sind ARRAY-Datenbausteine. Diese bestehen aus einem ARRAY eines beliebigen Datentyps. Es kann z. B. ein ARRAY eines PLC-Datentyps (UDT) sein.

Elemente in ARRAY-Datenbausteinen adressieren Sie mithilfe des Schlüsselworts "THIS". Anschließend wird der Index in eckigen Klammern angegeben. Der Index kann sowohl eine Konstante als auch eine Variable sein. Als Variablen für den Index sind Ganzzahlen von einer Breite bis 32 Bit zulässig.

Erweiterte Möglichkeiten zur Adressierung von ARRAY-DBs bietet die Task Card "Anweisungen" im Abschnitt "Verschieben". Diese Anweisungen bieten z. B. die Möglichkeit, auch den DB-Namen indirekt zu adressieren.

### Syntax

```
"<DBname>".Variablenname
%<DBnummer>.absoluteAdresse
#<DBAny_name>.%absoluteAdresse
"<ArrayDBname".THIS[#i].<Component>.<ComponentElement>
```

SCL:

```
"<ArrayDBname"."THIS"[#i].<Component>.<ComponentElement>
```

Die folgende Tabelle zeigt die möglichen absoluten Adressen von Variablen in Datenbausteinen:

| Datentyp                       | Absolute Adresse | Beispiel    | Erläuterung              |
|--------------------------------|------------------|-------------|--------------------------|
| BOOL                           | %DBn.DBXx.y      | %DB1.DBX1.0 | Datenbit 1.0 im DB1      |
| BYTE, CHAR, SINT, USINT        | %DBn.DBBy        | %DB1.DBB1   | Datenbyte 1 im DB1       |
| WORD, INT, UINT                | %DBn.DBWy        | %DB1.DBW1   | Datenwort 1 im DB1       |
| DWORD, DINT, UDINT, REAL, TIME | %DBn.DBBy        | %DB1.DBD1   | Datendoppelwort 1 im DB1 |

### Beispiel

Die folgenden Beispiele zeigen die Adressierung von Variablen in globalen Datenbausteinen:

| Adressierung   | Erläuterung  |
|--|--|
| "Motor".Value  | Symbolische Adressierung der Variablen "Value" im globalen Datenbaustein "Motor".  |
| %DB1.DBX1.0  | Absolute Adressierung der Variablen "DBX1.0" im globalen Datenbaustein "DB1".  |
| #MyDBAny.%DBX30.0  | Absolute Adressierung der Variablen "DBX30.0" im dem globalen Datenbaustein, der zur Laufzeit am Parameter "MyDBAny" übergeben wird.     |
| "MyARRAY_DB".THIS[#MyIndex].MyComponent.MyComponentElement | Adressierung eines ARRAY-Datenbausteins. Der ARRAY-Index wird über die Variable "MyIndex" angegeben. Das ARRAY-Element enthält noch zwei |

| Adressierung | Erläuterung  |
|--------------|--|
|              | weitere Unterstrukturen: "MyComponent" und "MyComponentElement". |

## Siehe auch

- Strukturierte Variablen adressieren (Seite 1070)
- Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 1072)
- Grundlagen zur indirekten Adressierung (Seite 1077)
- Instanzdaten adressieren (Seite 1069)

## Instanzdaten adressieren

### Beschreibung

Sie können Datenelemente aus der Schnittstelle des aktuellen Bausteins adressieren. Diese Variablen werden im Instanz-Datenbaustein gespeichert.

---

#### Hinweis

Variablen in Bausteinen mit optimiertem Zugriff können nur symbolisch adressiert werden.

---

Um eine Variable aus der Schnittstelle des aktuellen Bausteins zu adressieren, geben Sie das Zeichen # gefolgt von dem symbolischen Variablennamen ein.

Sie können auch auf die Variablen eines Multiinstanzbausteins zugreifen. Innerhalb des Multiinstanzbausteins adressieren Sie die Daten ebenfalls durch das Zeichen # gefolgt von dem Variablennamen. Vom aufrufenden Baustein greifen Sie auf die Daten des Multiinstanzbausteins über #<Multiinstanzname.Variablenname> zu.

### Syntax

Für die Adressierung von Variablen in Instanz-Datenbausteinen wird die folgende Syntax verwendet:

```
#<Variablenname>
#<Multiinstanzname.Variablenname>
```

### Beispiele

Die folgenden Beispiele zeigen die Adressierung von Variablen in Instanzdatenbausteinen:

| Adressierung | Erläuterung   |
|--------------|---|
| #Value       | Adressierung der Variablen "Value" im Instanz-Datenbaustein |

| Addressierung | Erläuterung  |
|---------------|--|
| #On           | Adressierung der Variablen "On" innerhalb des Multiinstanzbausteins                    |
| #Multi.On     | Adressierung der Variablen "On" des Multiinstanzbausteins vom aufrufenden Baustein aus |

## Siehe auch

Variablen in globalen Datenbausteinen adressieren (Seite 1067)

Strukturierte Variablen adressieren (Seite 1070)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 1072)

Grundlagen zur indirekten Adressierung (Seite 1077)

## Strukturierte Variablen adressieren

### Datenelemente eines ARRAYs adressieren

Sie greifen auf ein Element in einem ARRAY über die Syntax `ArrayName[i, j, k...]` zu.

In der eckigen Klammer wird der Index des Elements angegeben. Der Index enthält pro ARRAY-Dimension einen ganzzahligen Wert (-2147483648 ... 2147483647).

Zugriffsfehler entstehen, wenn Sie zur Laufzeit auf ein Element zugreifen, das außerhalb der deklarierten ARRAY-Grenzen liegt. Die verschiedenen CPU-Familien reagieren unterschiedlich auf Überschreitungen der ARRAY-Grenzen:

- S7-300/400
  - Die CPU wechselt in den Betriebszustand "STOP".
  - Um dies zu vermeiden, können Sie den Programmablauffehler-OB (OB 85) programmieren.
  - In SCL haben Sie zusätzlich die Möglichkeit, in den Bausteineigenschaften das Attribut "ARRAY-Grenzen prüfen" zu aktivieren. Es bewirkt, dass bei ARRAY-Zugriffsfehlern der Freigabeausgang ENO auf FALSE gesetzt wird.
- S7-1200
  - Die CPU erzeugt einen Diagnosepuffereintrag und verbleibt im Betriebszustand "RUN".
- S7-1500
  - Die CPU wechselt in den Betriebszustand "STOP".
  - Um dies zu vermeiden, können Sie den Programmierfehler-OB (OB 121) programmieren.
  - Außerdem haben Sie die Möglichkeit, die lokale Fehlerbehandlung mit den Anweisungen "GET\_ERROR: Fehler lokal abfragen" oder "GET\_ERROR\_ID: Fehler-ID lokal abfragen" zu programmieren.

---

**Hinweis****Überwachung von ARRAY-Zugriffsfehlern durch ENO**

Der Freigabeausgang ENO wird nicht auf den Signalzustand FALSE gesetzt, wenn bei der Ausführung einer Anweisung die ARRAY-Grenzen überschritten werden. Eine Ausnahme bilden nur SCL-Bausteine auf CPUs der Baureihen S7-300/400, für die die Bausteineigenschaft "ARRAY-Grenzen prüfen" gesetzt ist.

---

Siehe auch:

Auto-Hotspot

Indirektes Indizieren von ARRAY-Komponenten (Seite 1079)

**ARRAY-Datenbausteine adressieren**

Eine besondere Form des ARRAYs sind ARRAY-Datenbausteine. ARRAY-Datenbausteine sind globale Datenbausteine, die aus genau einem ARRAY bestehen. Elemente in ARRAY-Datenbausteinen adressieren Sie über die folgende Syntax:

```
"<GlobArrayDBname".THIS[#i].<componentname>."<elementname>"
```

SCL:

```
"<GlobArrayDBname"."THIS"[#i].<componentname>."<elementname>".
```

Erweiterte Möglichkeiten zur Adressierung von ARRAY-DBs bietet die Task Card "Anweisungen" im Abschnitt "Verschieben". Diese Anweisungen bieten z. B. die Möglichkeit, auch den DB-Namen indirekt zu adressieren.

**Datenelemente in Strukturen adressieren**

Sie greifen auf die einzelnen Elemente einer Struktur über `Strukturname.Elementname` zu.

Siehe auch:

Auto-Hotspot

**Datenelemente eines PLC-Datentyps adressieren**

Für den Zugriff auf Elemente eines PLC-Datentyps wird die Syntax `PLCDatentypName.Elementname` verwendet.

Siehe auch:

Auto-Hotspot

**Einzelne Zeichen eines STRINGs adressieren**

Sie greifen auf ein einzelnes Zeichen eines STRINGs über die Syntax `StringName[i]` zu.

Zugriffsfehler entstehen, wenn Sie zur Laufzeit auf ein Zeichen zugreifen, das außerhalb der deklarierten STRING-Länge liegt. Die verschiedenen CPU-Familien reagieren unterschiedlich auf Überschreitungen der STRING-Länge:

- S7-300/400
  - Wenn die Anweisung über den Freigabeausgang ENO verfügt, wird ENO auf den Signalzustand FALSE gesetzt.
  - In allen anderen Fällen wird das Überschreiten der STRING-Länge nicht überwacht. Wenn der Zugriff außerhalb des STRINGS liegt, wird auf den falschen Speicherbereich zugegriffen. Falls der Zugriff außerhalb des Datenbausteins liegt, geht die CPU in STOP.
  - Um dies zu vermeiden, können Sie den Programmablauffehler-OB (OB 85) programmieren.
- S7-1200/1500
  - Wenn die Anweisung über den Freigabeausgang ENO verfügt, wird ENO auf den Signalzustand FALSE gesetzt.
  - In allen anderen Fällen erfolgt keine Fehlerreaktion. Bei lesendem Zugriff erhalten Sie das Zeichen '\$00', ein schreibender Zugriff wird nicht ausgeführt.

**Beispiele:**

Die folgenden Beispiele zeigen die Adressierung von Variablen strukturierten Datentyps:

| Adressierung             | Erläuterung   |
|--------------------------|---|
| Motor.Value_1x3[2]       | Adressierung eines eindimensionalen Arrays  |
| Motor.Value_2x4[2,4]     | Adressierung eines zweidimensionalen Arrays   |
| Motor.Value_4x7[2,4,1,3] | Adressierung eines vierdimensionalen Arrays   |
| Batch_1.Temperature      | Adressierung des Elements "Temperature" in der Struktur "Batch_1"                                     |
| Values.Temperature       | Adressierung des Elements "Temperature" in der Variable "Values", die auf einem PLC-Datentyp basiert. |
| STRING[3]                | Adressiert das dritte Zeichen des Strings.  |

**Siehe auch**

- Grundlagen zur indirekten Adressierung (Seite 1077)
- STRING (Seite 1112)

**Bereiche einer Variablen mit Slice-Zugriffen adressieren**

**Beschreibung**

Sie haben die Möglichkeit, gezielt Bereiche innerhalb deklarierter Variablen zu adressieren. Dabei können Sie auf Bereiche der Breiten 1 Bit, 8 Bit, 16 Bit oder 32 Bit zugreifen. Diese Art des Zugriffs wird als "Slice-Zugriff" bezeichnet.

**Syntax**

Für die Adressierung wird die folgende Syntax verwendet:

<Variable>.X<BIT-Nummer>  
 <Variable>.B<BYTE-Nummer>  
 <Variable>.W<WORD-Nummer>  
 <Variable>.D<DWORD-Nummer>

Die Syntax besteht aus folgenden Teilen:

| Teil           | Beschreibung  |
|----------------|---|
| <Variable>     | Variable, auf die Sie zugreifen. Die Variable muss vom Datentyp "Bitfolge" sein. Bei aktivierter IEC-Prüfung ist auch der Zugriff auf Variablen des Datentyps "Ganzzahl" möglich. |
| X              | Kennung für die Zugriffsbreite "Bit (1Bit)"   |
| B              | Kennung für die Zugriffsbreite "Byte (8 Bit)"   |
| W              | Kennung für die Zugriffsbreite "Word (16 Bit)"  |
| D              | Kennung für die Zugriffsbreite "DWord (32 Bit)"   |
| <BIT-Nummer>   | Nummer des Bits innerhalb von <Variable>, auf das zugegriffen wird. Die Nummer 0 greift auf das niederwertigste BIT zu.   |
| <BYTE-Nummer>  | Nummer des Bytes innerhalb von <Variable>, auf das zugegriffen wird. Die Nummer 0 greift auf das niederwertigste BYTE zu.   |
| <WORD-Nummer>  | Nummer des Words innerhalb von <Variable>, auf das zugegriffen wird. Die Nummer 0 greift auf das niederwertigste WORD zu.   |
| <DWORD-Nummer> | Nummer des DWords innerhalb von <Variable>, auf das zugegriffen wird. Die Nummer 0 greift auf das niederwertigste DWORD zu.   |

## Beispiele

Die folgenden Beispiele zeigen die Adressierung einzelner Bits:

| Adressierung                           | Erläuterung   |
|--|---|
| "Engine".Motor.X0<br>"Engine".Motor.X7 | "Motor" ist eine Variable vom Datentyp BYTE, WORD, DWORD oder LWORD im globalen Datenbaustein "Engine".<br>X0 adressiert die Bitadresse 0, X7 die Bitadresse 7 innerhalb von "Motor". |
| "Engine".Speed.B0<br>"Engine".Speed.B1 | "Speed" ist eine Variable vom Datentyp WORD, DWORD oder LWORD im globalen Datenbaustein "Engine".<br>B0 adressiert die Byteadresse 0, B1 die Byteadresse 1 innerhalb von "Speed".     |
| "Engine".Fuel.W0<br>"Engine".Fuel.W1   | "FUEL" ist eine Variable vom Datentyp DWORD oder LWORD im globalen Datenbaustein "Engine".<br>W0 adressiert die Wortadresse 0, W1 die Wortadresse 1 innerhalb von "Fuel".             |
| "Engine".Data.D0<br>"Engine".Data.D1   | "Data" ist eine Variable vom Datentyp LWORD im globalen Datenbaustein "Engine".<br>D0 adressiert die Doppelwortadresse 0, D1 die Doppelwortadresse 1 innerhalb von "Data".            |

## Variablen mit AT überlagern

### Beschreibung

Um auf Datenbereiche innerhalb einer deklarierten Variablen zuzugreifen, können Sie die deklarierten Variablen mit einer weiteren Deklaration überlagern. Sie haben so die Möglichkeit, eine bereits deklarierte Variable mit einem anderen Datentyp anzusprechen. Sie können z. B. die einzelnen Bits einer Variablen vom Datentyp WORD mit einem ARRAY of BOOL ansprechen.

### Regeln

Folgende allgemeine Regeln gelten für das Überlagern von Variablen:

- In AWL, KOP, FUP und GRAPH ist das Überlagern in S7-1200 und S7-1500 möglich.
- In SCL ist das Überlagern in allen CPU-Familien möglich.
- Das Überlagern von Variablen ist in folgenden Bausteinen möglich:
  - In Codebausteinen mit Standardzugriff
  - In Codebausteinen mit optimiertem Zugriff für Variablen mit der Remanenzeinstellung "Im IDB setzen"
- Die Datenbreite der überlagernden Variablen muss gleich oder kleiner als die der überlagerten Variablen sein.
- Die Datentypen VARIANT und INSTANCE können nicht überlagert werden.
- Bausteine aus Bibliotheken, die als Parameter in der Schnittstelle deklariert sind, können nicht überlagert werden.
- Strukturierte PLC-Variablen, die als Parameter in der Schnittstelle deklariert sind, können nicht überlagert werden.

---

#### Hinweis

##### S7-1200/1500: AT in FCs verwenden

Bei FCs in S7-1200/1500 müssen die Datenbreiten der überlagernden Variablen und der überlagerten Variablen gleich sein. Falls dies in Ihrem Programm nicht möglich ist, können Sie auch Bereiche innerhalb existierender Variablen mit der Syntax .X, .B, .W oder .D adressieren.

Siehe auch: Einzelne Bereiche einer Variablen adressieren (Seite 1072)

---

Darüber hinaus gelten folgende Kombinationsregeln:

|           |                | Überlagernde Variable | Überlagerte Variable |   |   |   |
|-----------|----------------|-----------------------|----------------------|---|---|---|
| Elementar | Strukturiert * | Any/Pointer           | DB_ANY               |   |   |   |
| FB        | Input          | Elementar             | x                    | x |   | x |
|           |                | Strukturiert *        | x                    | x | x | x |



|    |                      | Überlagernde Variable   | Überlagerte Variable |   |   |   |
|----|----------------------|---|----------------------|---|---|---|
|    |                      | Any/Pointer   |                      | x |   |   |
|    | Temp                 | Elementar   | x                    | x |   |   |
|    |                      | Strukturiert  | x                    | x | x |   |
|    |                      | Any/Pointer   |                      | x |   |   |
|    | Static, Output       | Elementar   | x                    | x |   | x |
|    |                      | Strukturiert  | x                    | x |   | x |
|    |                      | Any/Pointer   |                      |   |   |   |
|    | InOut                | Elementar   | x                    |   |   | x |
|    |                      | Strukturiert  |                      | x |   |   |
|    |                      | Any/Pointer   |                      |   |   |   |
| FC | Temp                 | Elementar   | x                    | x |   |   |
|    |                      | Strukturiert  | x                    | x | x |   |
|    |                      | Any/Pointer   |                      | x |   |   |
|    | Input, Output, InOut | Elementar<br>(beide Variablen müssen die gleiche Bitbreite haben) | x                    |   |   | x |
|    |                      | Strukturiert  |                      | x | x |   |
|    |                      | Any/Pointer   |                      |   |   |   |
| OB | Temp                 | Elementar   | x                    | x |   |   |
|    |                      | Strukturiert  | x                    | x | x |   |
|    |                      | Any/Pointer   |                      | x |   |   |

\* Strukturierte Datentypen sind Datentypen, die aus mehreren Datenelementen bestehen, z. B. ARRAY oder STRUCT.

## Deklaration

Um eine Variable zu überlagern, deklarieren Sie eine weitere Variable direkt nach der zu überlagernden Variable und kennzeichnen sie mit dem Schlüsselwort "AT".

### Beispiel

Das folgende Bild zeigt die Deklaration einer überlagerten Variable in der Schnittstelle eines FB:

|          |                                  |
|----------|----------------------------------|
| ▼ Input  |                                  |
| ■ MyByte | Byte                             |
| ▼ AT     | AT "MyByte" Array [0..7] of Bool |
| ■ AT[0]  | Bool                             |
| ■ AT[1]  | Bool                             |
| ■ AT[2]  | Bool                             |
| ■ AT[3]  | Bool                             |
| ■ AT[4]  | Bool                             |
| ■ AT[5]  | Bool                             |
| ■ AT[6]  | Bool                             |
| ■ AT[7]  | Bool                             |

Beim Aufruf eines Bausteins mit der gezeigten Variablendeklaration wird die Variable "MyByte" versorgt. Innerhalb des Bausteins stehen nun zwei Möglichkeiten zur Verfügung, die Daten zu interpretieren:


- als Byte
- als eindimensionales ARRAY of BOOL

### Operanden indirekt adressieren

#### Grundlagen zur indirekten Adressierung

#### Einführung

Die indirekte Adressierung bietet die Möglichkeit, Operanden zu adressieren, deren Adresse erst zur Laufzeit berechnet wird. Sie können mit indirekter Adressierung auch Programmteile mehrfach bearbeiten lassen und bei jedem Durchlauf einen anderen Operanden verwenden.

|   |
|---|
|  <b>WARNUNG</b>  |
| <b>Gefahr von Zugriffsfehlern</b>   |
| Da bei der indirekten Adressierung die Operanden erst zur Laufzeit berechnet werden, besteht die Gefahr, dass Zugriffsfehler auftreten und das Programm mit falschen Werten arbeitet. Außerdem können ungewollt Speicherbereiche mit falschen Werten überschrieben werden. Das Automatisierungssystem kann dann unerwartet reagieren. |
| Setzen Sie deshalb die indirekte Adressierung nur mit Vorsicht ein.   |

## Indirekte Adressierung

### Grundlagen zur indirekten Adressierung

#### Allgemeine Möglichkeiten der indirekten Adressierung in S7-1200 und S7-1500

Die folgenden Möglichkeiten der indirekten Adressierung stehen in allen Programmiersprachen zur Verfügung:

- Indirekte Adressierung über Pointer
- Indirektes Indizieren von ARRAY-Komponenten
- Indirekte Adressierung eines Datenbausteins über den Datentyp DB\_ANY.

#### Sprachspezifische Möglichkeiten der indirekten Adressierung

Darüberhinaus gibt es in den einzelnen Programmiersprachen folgende spezifische Adressierungsmöglichkeiten:

- In AWL können Sie über die Adressregister Operanden indirekt adressieren.
- In SCL können Sie einen variablen Speicherbereich mit den folgenden Anweisungen lesen oder schreiben:
  - POKE - Speicheradresse schreiben
  - POKE\_BOOL - Speicherbit schreiben
  - PEEK - Speicheradresse lesen
  - PEEK\_BOOL - Speicherbit lesen
  - POKE\_BLK - Speicherbereich schreiben

Eine detaillierte Beschreibung dieser Adressierungsmöglichkeiten finden Sie unter "Siehe auch".

#### Siehe auch

Variablen in globalen Datenbausteinen adressieren (Seite 1067)

POKE: Speicheradresse schreiben (Seite 2245)

POKE\_BOOL: Speicherbit schreiben (Seite 2247)

PEEK: Speicheradresse lesen (Seite 2241)

PEEK\_BOOL: Speicherbit lesen (Seite 2244)

POKE\_BLK: Speicherbereich schreiben (Seite 2249)

Indirektes Adressieren über Pointer (Seite 1078)

Indirektes Indizieren von ARRAY-Komponenten (Seite 1079)

Indirektes Adressieren in AWL (Seite 1083)

## Indirektes Adressieren über Pointer

### Beschreibung

Für die indirekte Adressierung wird ein besonderes Datenformat benötigt, das die Adresse und gegebenenfalls auch den Bereich und den Datentyp eines Operanden enthält. Dieses Datenformat wird als Pointer (Zeiger) bezeichnet. Folgende Typen von Zeigern stehen Ihnen zur Verfügung:

- POINTER (S7-1500)
- ANY (S7-1500, nur Bausteine mit Standardzugriff)
- VARIANT (S7-1200/1500)

Weitere Informationen zu den Zeiger-Datentypen finden Sie unter "Siehe auch".

---

#### Hinweis

In SCL ist die Verwendung von POINTER eingeschränkt. Es besteht lediglich die Möglichkeit, ihn an aufgerufene Bausteine weiterzugeben.

---

### Beispiel

Das folgende Beispiel zeigt eine indirekte Adressierung mit einem bereichsinternen Pointer:

| Adressierung in AWL | Erläuterung                                 |
|---------------------|---|
| L P#10.0            | // Pointer (P#10.0) in Akkumulator 1 laden  |
| T MD20              | // Pointer in den Operanden MD20 übertragen |
| L MW [MD20]         | // MW10 in Akkumulator 1 laden              |
| ....                | // Beliebiges Programm                      |
| L MD [MD20]         | // MD10 in Akkumulator 1 laden              |
| ....                | // Beliebiges Programm                      |
| = M [MD20]          | // Bei VKE=1 das Merkerbit M10.0 setzen     |

Der Pointer P#10.0 wird in den Operanden MD20 übertragen. Wenn der Operand MD20 in eckigen Klammern programmiert ist, wird dieser zur Laufzeit durch die Adresse ersetzt, die im Pointer enthalten ist.

### Siehe auch

Grundlagen zur indirekten Adressierung (Seite 1077)

## Indirektes Indizieren von ARRAY-Komponenten

### Beschreibung

Zur Adressierung der Komponenten eines ARRAYS können Sie als Index neben Konstanten auch Variablen vom Datentyp Ganzzahl angeben. Dabei sind Ganzzahlen mit einer Länge bis 32 Bit zulässig. Bei Verwendung von Variablen wird der Index zur Laufzeit berechnet. So können Sie z. B. in Programmschleifen bei jedem Durchlauf einen anderen Index verwenden.

#### Hinweis

Wenn Sie einen Baustein aufrufen und diesem eine indirekt indizierte ARRAY-Komponente ("`<Datenbaustein>.<ARRAY>[\"i\"]`") als Durchgangsparameter (InOut) übergeben, können Sie während der Bausteinbearbeitung den Wert der Indexvariable [i] nicht ändern. Der Wert wird also immer in dieselbe ARRAY-Komponente zurückgeschrieben, aus der er gelesen wurde.

### Syntax

Für das indirekte Indizieren eines ARRAYS wird die folgende Syntax verwendet:

```
"<Datenbaustein>.<ARRAY>[\"i\"] // eindimensionales ARRAY
"<Datenbaustein>.<ARRAY>[\"i\"].a // eindimensionales ARRAY of STRUCT
"<Datenbaustein>.<ARRAY>[\"i\", \"j\"] // mehrdimensionales ARRAY
"<Datenbaustein>.<ARRAY>[\"i\", \"j\"].a // mehrdimensionales ARRAY of STRUCT
```

Die Syntax besteht aus folgenden Teilen:

| Teil          | Beschreibung   |
|---------------|--|
| Datenbaustein | Name des Datenbausteins, in dem sich das ARRAY befindet              |
| ARRAY         | Variable vom Datentyp ARRAY  |
| i, j          | PLC-Variablen vom Datentyp Ganzzahl, die als Zeiger verwendet werden |
| a             | Weitere Teilvariable der Struktur                                    |

### Beispiele

Das folgende Beispiel zeigt das indirekte Indizieren einer ARRAY-Komponente in AWL:

Es sollen mehrere Achsen in verschiedenen Winkel gefahren werden. Die Werte für Achsennummer und Winkel sind in dem zweidimensionalen ARRAY "control\_axis" abgelegt.

Mit der Anweisung "SEL" wählen Sie aus, welche Komponente des ARRAYS "control\_axis" an den Ausgangsparameter "#out" geschrieben wird.

Die Achsennummer wird über die Konstanten "Constant\_Axis\_NoX" und "Constant\_Axis\_NoY" definiert, der Winkel wird über die Variable "#Angle" bestimmt.

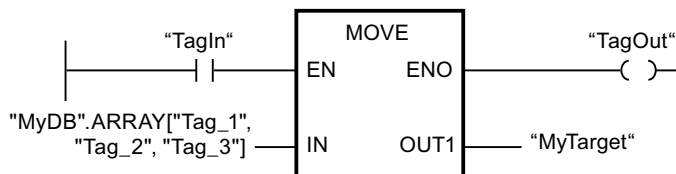
**Adressierung in AWL**

```
CALL SEL
  value_type:=Int
  G := "Select"
  IN0 :=#control_axis["Constant_Axis_NoX", #Angle]
  IN1 :=#control_axis["Constant_Axis_NoY", #Angle]
  OUT :=#out
```

Die folgenden Beispiele zeigen das indirekte Indizieren einer ARRAY-Komponente am Beispiel von SCL. "MOTOR" ist ein eindimensionales ARRAY\_of\_INT mit drei Zeilen. "VALUES" ist eine PLC-Variable vom Datentyp "Integer".

| Adressierung in SCL                 | Erläuterung   |
|-------------------------------------|---|
| MOTOR[2] := VALUES;                 | (*Direkte Adressierung: Zuweisung von VALUES zur zweiten Zeile des ARRAYS MOTOR*)   |
| MOTOR["Tag_1"] := VALUES;           | (*Indirekte Adressierung: Zuweisung von VALUES zur durch "Tag_1" spezifizierten Zeile des ARRAYS MOTOR*)                      |
| #MOTOR["Tag_2"+"Tag_3"] := #Values; | (*Indirekte Adressierung: Zuweisung von VALUES zur durch den Ausdruck "Tag_2"+"Tag_3" spezifizierten Zeile des ARRAYS MOTOR*) |

Das folgende Beispiel zeigt das indirekte Indizieren einer ARRAY-Komponente am Beispiel von KOP. "ARRAY" ist ein dreidimensionales ARRAY. "Tag\_1", "Tag\_2" und "Tag\_3" sind PLC-Variablen vom Datentyp "Integer". Abhängig von deren Werten wird eine der Komponenten von "ARRAY" in die Variable "MyTarget" kopiert.



**ARRAY-Komponenten mit den Anweisungen "FieldRead" und "FieldWrite" indizieren**

Zur indirekten Indizierung von ARRAY-Komponenten in KOP und FUP können Sie auch die folgenden Anweisungen verwenden:

- FieldWrite - Feld schreiben
  - FieldRead - Feld lesen
- Eine detaillierte Beschreibung dieser Anweisungen finden Sie im Kapitel "Referenzen".

## Weitere Informationen

Weitere Informationen zum Datentyp ARRAY finden Sie unter "Siehe auch".

## Siehe auch

Grundlagen zur indirekten Adressierung (Seite 1077)

Strukturierte Variablen adressieren (Seite 1070)

## Indirektes Adressieren einzelner Zeichen eines STRINGS

### Beschreibung

Zur Adressierung der einzelnen Zeichen eines STRINGS können Sie als Index neben Konstanten auch Variablen vom Datentyp Ganzzahl angeben. Bei Verwendung von Variablen wird der Index zur Laufzeit berechnet. So können Sie z. B. in Programmschleifen bei jedem Durchlauf einen anderen Index verwenden.

Die Indexvariable [i] wird einmal zu Beginn des Bausteinaufrufes gelesen und kann während der Bearbeitung des aufgerufenen Bausteins von diesem nicht geändert werden.

---

#### Hinweis

##### Überwachung von STRING-Zugriffen zur Laufzeit

Wenn zur Laufzeit ein STRING geschrieben wird, der die definierte Länge der STRING-Variablen überschreitet, kann es zu unerwünschten Reaktionen im Programm kommen. In S7-1200/1500 wird das Überschreiten der STRING-Länge überwacht. Sie können wählen, ob Sie auf Überschreitungen mit der globalen Fehlerbehandlung der CPU oder mit einer eigenen lokalen Fehlerbehandlung reagieren wollen.

---

### Syntax

Für das indirekte Indizieren eines STRINGS wird die folgende Syntax verwendet:  
 "<Datenbaustein>".<STRING>["i"]

### Beispiel

Das folgenden Beispiel zeigt das indirekte Indizieren eines STRINGS am Beispiel von SCL. "STRING" ist eine Variable vom Datentyp STRING. "Tag\_1" ist eine PLC-Variable vom Datentyp "Integer".

| Adressierung in SCL           | Erläuterung   |
|-------------------------------|---|
| STRING["Tag_1"] := CHARACTER; | (*Indirekte Adressierung: Zuweisung von "CHARACTER" an das durch "Tag_1" spezifizierten Zeichen des STRINGS*) |

## Weitere Informationen

Weitere Informationen zum Datentyp STRING finden Sie unter "Siehe auch".

## Siehe auch

STRING (Seite 1112)

## Indirekte Adressierung in AWL

## Grundlagen zu Adressregistern

### Einführung

Zur indirekten Adressierung von Operanden stehen Ihnen zwei Adressregister zur Verfügung, das Adressregister 1 (AR1) und das Adressregister 2 (AR2). Die Adressregister sind gleichwertig und haben eine Länge von 32 Bits. In den Adressregistern können Sie bereichsinterne und bereichsübergreifende Zeiger speichern. Die gespeicherten Daten können Sie im Programm aufrufen, um die Adresse eines Operanden zu bestimmen.

Der Datenaustausch zwischen den Registern und den anderen verfügbaren Speicherbereichen erfolgt mithilfe von Lade- und Transferanweisungen.

---

### Hinweis

In S7-1500 gelten besondere Regeln zum Datenaustausch über Adress- und Datenbausteinregister:

- Die Werte in den Registern bleiben nicht über Bausteingrenzen hinweg bestehen.
  - Bei Sprachwechsel innerhalb eines Bausteins werden die Register rückgesetzt.
  - Auf Daten in Bausteinen mit optimiertem Zugriff können Sie nur verweisen, wenn diese die Remanenzeinstellung "Im IDB setzen" haben.
  - Es ist nicht möglich, mit Hilfe der Adressregister (bereichsübergreifend) auf Lokaldaten in Bausteinen mit optimiertem Zugriff zu verweisen.
- 

## Weitere Informationen

Weitere Informationen zu den Anweisungen, die Adressregister verwenden, und der indirekten Adressierung finden Sie unter "Siehe auch".

## Siehe auch

Indirektes Adressieren in AWL (Seite 1083)

Bereiche einer Variablen mit Slice-Zugriffen adressieren (Seite 1072)



## Indirektes Adressieren in AWL

In AWL stehen Ihnen folgende Möglichkeiten der indirekten Adressierung zur Verfügung:

- Speicherindirekte Adressierung
- Registerindirekte bereichsinterne Adressierung
- Registerindirekte bereichsübergreifende Adressierung

## Speicherindirekte Adressierung

Bei speicherindirekter Adressierung legen Sie die Adresse in einer Variablen ab. Diese Variable kann vom Datentyp WORD oder DWORD sein. Die Variable kann in den Speicherbereichen "Daten" (DB oder DI), "Merker" (M) oder "temporäre Lokaldaten" (L) liegen. In S7-1500 können auch FB-Parameter zum Ablegen der Adresse verwendet werden. Wenn die Variable in einem Datenbaustein liegt, muss es ein Datenbaustein mit Standardzugriff sein.

Das folgende Beispiel zeigt Anwendungen der speicherindirekten Adressierung:

| Adressierung in AWL | Erläuterung  |
|---------------------|--|
| U E [MD 2]          | // Führe eine UND-Verknüpfung mit einem variablen Eingangsbit aus. Die Adresse des Eingangsbits befindet sich in Merkerdoppelwort MD2. |
| = DIX [DBD 2]       | // Weise das VKE einem variablen Datenbit zu. Die Adresse des Datenbits befindet sich im Datendoppelwort DBD2.                         |
| L EB [DID 4]        | // Lade ein variables Eingangsbyte in AKKU 1. Die Adresse des Eingangsbytes befindet sich im Instanzdoppelwort DID4.                   |
| AUF DB [LW 2]       | // Öffne einen variablen Datenbaustein. Die Nummer des Datenbausteins befindet sich im Lokaldatenwort LW2.                             |

## Registerindirekte bereichsinterne Adressierung

Die registerindirekte Adressierung verwendet eines der Adressregister (AR1 oder AR2), um die Adresse des Operanden aufzunehmen.

Bei der registerindirekten bereichsinternen Adressierung indizieren Sie lediglich die Bitadresse und die Byteadresse über das Adressregister (z. B. P#10.0). Den Speicherbereich, für den die Adresse im Adressregister gelten soll, geben Sie erst bei der Programmierung der Anweisung an. Die Adresse im Adressregister bewegt sich dann in dem Speicherbereich, der in der Anweisung angegeben ist.

Mögliche Speicherbereiche sind "Eingänge" (E), "Ausgänge" (A), "Peripherie" (PE oder PA), "Merker" (M), "temporäre Lokaldaten" (L) und "Daten" (DB oder DI). Wenn der Operand in einem Datenbaustein liegt, muss es ein Datenbaustein mit Standardzugriff sein.

Bei der Eingabe der registerindirekten bereichsinternen Adressierung geben Sie nach der Angabe des Adressregisters einen Versatz an, der zum Inhalt des Adressregisters addiert wird, ohne das Adressregister zu verändern. Dieser Versatz hat ebenfalls das Format eines Zeigers. Die Angabe des Zeigers ist zwingend und muss als Konstante erfolgen (z. B. P#0.0 oder P#2.0).

Das folgende Beispiel zeigt eine Anwendung der registerindirekten bereichsinternen Adressierung:

| AWL               | Erläuterung  |
|-------------------|--|
| LAR1 P#10.0       | // Zeiger (P#10.0) in Adressregister 1 laden   |
| L EW [AR1, P#2.0] | // Inhalt des Adressregisters 1 (P#10.0) um Versatz P#2.0 erhöhen.<br>// Inhalt des Eingangswort EW12 in Akkumulator 1 laden |
| L EW [AR1, P#0.0] | // Inhalt des Adressregisters 1 (P#10.0) um Versatz P#0.0 erhöhen.<br>// Inhalt des Eingangswort EW10 in Akkumulator 1 laden |

### Registerindirekte bereichsübergreifende Adressierung

Bei der registerindirekten bereichsübergreifenden Adressierung indizieren Sie die gesamte Adresse des Operanden, d. h. die Bit- und Byteadresse sowie den Speicherbereich, über das Adressregister. Mögliche Speicherbereiche sind "Eingänge" (E), "Ausgänge" (A), "Peripherie" (P), "Merker" (M), "temporäre Lokaldaten" (L) und "Daten" (DB oder DI). Wenn der Operand in einem Datenbaustein liegt, muss es ein Datenbaustein mit Standardzugriff sein oder der Operand muss die Remanenzeinstellung "Im IDB setzen" haben.

In der Anweisung programmieren Sie nur die Operandenbreite. Mögliche Operandenbreiten sind Bit, Byte, Wort und Doppelwort.

Das folgende Beispiel zeigt eine Anwendung der registerindirekten bereichsübergreifenden Adressierung:

|                  |   |
|------------------|---|
| LAR1 P#M10.0     | // Bereichsübergreifenden Zeiger (P#M10.0) in Adressregister 1 laden  |
| L W [AR1, P#2.0] | // Inhalt des Adressregisters 1 (P#M10.0) um Versatz P#2.0 erhöhen.<br>// Inhalt des Merkerwort MW12 in Akkumulator 1 laden     |
| LAR1 P#A10.0     | // Bereichsübergreifenden Zeiger (P#A10.0) in Adressregister 1 laden  |
| L W [AR1, P#2.0] | // Inhalt des Adressregisters 1 (P#A10.0) um Versatz P#2.0 addieren<br>// Inhalt des Ausgangswort AW12.0 in Akkumulator 1 laden |

**Hinweis****Besonderheiten in S7-1500**

In S7-1500 gelten besondere Regeln zum Datenaustausch über Adress- und Datenbausteinregister:

- Die Werte in den Registern bleiben nicht über Bausteingrenzen hinweg bestehen. Auch bei Sprachwechsel innerhalb eines Bausteins werden die Register rückgesetzt.
- Wenn Sie über registerindirekte Adressierung auf einen Operanden vom Typ BYTE, WORD oder DWORD zugreifen, muss die Adresse auf einer Byte-Grenze beginnen.

Beispiele:

LAR1 P#0.0

L MW [AR1, P#0.0] //  $P\#0.0 + P\#0.0 = P\#0.0$  - Die Adressierung ist zulässig, da der Zeiger P#0.0 auf eine Bytegrenze zeigt.

L MW [AR1, P#2.1] //  $P\#0.0 + P\#2.1 = P\#2.1$  - Die Adressierung ist nicht zulässig, da der Zeiger P#2.1 nicht auf eine Bytegrenze zeigt.

---

**Siehe auch**

Grundlagen zur indirekten Adressierung (Seite 1077)

Strukturierte Variablen adressieren (Seite 1070)

Grundlagen zu Adressregistern (Seite 1082)

**Konstanten eingeben****Beschreibung**

Sie haben folgende Möglichkeiten, Konstanten im Programm zu verwenden:

- Eingabe des Wertes. Sie können einen reinen Wert eingeben oder optional eine Datentypangabe voranstellen. Wenn Sie nur einen Wert eingeben, interpretiert das Programm die Konstante selbstständig in einem passenden Datentyp. Wenn Sie eine Datentypangabe machen, wird die Konstante immer in dem angegebenen Datentyp interpretiert.
- Eingabe eines in der PLC-Variablentabelle definierten symbolischen Namens (bei S7-1200/S7-1500). Der symbolische Name einer Konstanten wird automatisch in Anführungszeichen eingeschlossen.

Beide Arten von Konstanten werden im Programm blau dargestellt.

### Syntax

- Eingabe eines Wertes:  
    <Wert>  
    <Datentyp>#<Wert>
- Eingabe eines symbolischen Konstantennamens aus der PLC-Variablen-tabelle:  
    "<Name>"

### Beispiel

Die folgenden Beispiele zeigen die Verwendung von Konstanten:

| Adressierung | Erläuterung   |
|--------------|---|
| 4            | Werteingabe für eine Konstante ohne Datentypangabe. Das Programm interpretiert den Wert selbstständig in einem Datentyp, der zum aktuellen Kontext passt. |
| INT#4        | Werteingabe für eine Konstante vom Typ Integer mit Angabe des Datentyps.  |
| FALSE        | Werteingabe für eine Konstante vom Typ Bool   |
| "Name"       | Symbolische Konstante aus der PLC-Variablen-tabelle   |
| "Offset"     | Symbolische Konstante aus der PLC-Variablen-tabelle   |

#### Hinweis

##### Konstanten vom Typ BOOL

Konstanten vom Typ BOOL dürfen in S7-300/400 nicht als Eingänge in KOP oder FUP verwendet werden.

### Weitere Informationen

Weitere Informationen zu Datentypen von Konstanten sowie deren Eingabeformaten und Wertebereichen finden Sie im Kapitel "Datentypen" unter "Siehe auch".

### Siehe auch

Konstanten (Seite 1058)

Konstanten deklarieren (Seite 1202)

### 9.1.1.5 Datentypen

#### Übersicht über die gültigen Datentypen

#### Gültigkeit der Datentypgruppen

Die Datentypen legen die Eigenschaften der Daten fest, z. B. die Darstellung des Inhalts und die zulässigen Speicherbereiche. Im Anwenderprogramm können Sie vordefinierte Datentypen verwenden oder auch eigene definieren.

Die folgenden Tabellen zeigen die Verfügbarkeit der vordefinierten Datentypen in den verschiedenen S7-CPU's:

Tabelle 9-1 Binärzahlen

| <b>Binärzahlen</b> | <b>S7-300/400</b> | <b>S7-1200</b> | <b>S7-1500</b> |
|--------------------|-------------------|----------------|----------------|
| BOOL (Seite 1091)  | X                 | X              | X              |
| <b>Bitfolgen</b>   |                   |                |                |
| BYTE (Seite 1091)  | X                 | X              | X              |
| WORD (Seite 1092)  | X                 | X              | X              |
| DWORD (Seite 1093) | X                 | X              | X              |
| LWORD (Seite 1094) | -                 | -              | X              |

Tabelle 9-2 Ganzzahlen

| <b>Ganzzahlen</b>  | <b>S7-300/400</b> | <b>S7-1200</b> | <b>S7-1500</b> |
|--------------------|-------------------|----------------|----------------|
| SINT (Seite 1095)  | -                 | X              | X              |
| INT (Seite 1096)   | X                 | X              | X              |
| DINT (Seite 1098)  | X                 | X              | X              |
| USINT (Seite 1095) | -                 | X              | X              |
| UINT (Seite 1097)  | -                 | X              | X              |
| UDINT (Seite 1099) | -                 | X              | X              |
| LINT (Seite 1099)  | -                 | -              | X              |
| ULINT (Seite 1101) | -                 | -              | X              |

Tabelle 9-3 Gleitpunktzahlen

| <b>Gleitpunktzahlen</b> | <b>S7-300/400</b> | <b>S7-1200</b> | <b>S7-1500</b> |
|-------------------------|-------------------|----------------|----------------|
| REAL (Seite 1102)       | X                 | X              | X              |
| LREAL (Seite 1103)      | -                 | X              | X              |

Tabelle 9-4 Zeiten

| <b>Zeiten</b>       | <b>S7-300/400</b> | <b>S7-1200</b> | <b>S7-1500</b> |
|---------------------|-------------------|----------------|----------------|
| S5TIME (Seite 1105) | X                 | -              | X              |
| TIME (Seite 1107)   | X                 | X              | X              |
| LTIME (Seite 1107)  | -                 | -              | X              |

Tabelle 9-5 Datum und Uhrzeit

| Datum und Uhrzeit                | S7-300/400 | S7-1200 | S7-1500 |
|----------------------------------|------------|---------|---------|
| DATE (Seite 1108)                | X          | X       | X       |
| TOD (TIME_OF_DAY) (Seite 1108)   | X          | X       | X       |
| LTOD (LTIME_OF_DAY) (Seite 1109) | -          | -       | X       |
| DT (DATE_AND_TIME) (Seite 1109)  | X          | -       | X       |
| LDT (Seite 1110)                 | -          | -       | X       |
| DTL (Seite 1111)                 | -          | X       | X       |

Tabelle 9-6 Zeichen

| Zeichen             | S7-300/400 | S7-1200 | S7-1500 |
|---------------------|------------|---------|---------|
| CHAR (Seite 1112)   | X          | X       | X       |
| STRING (Seite 1112) | X          | X       | X       |

Tabelle 9-7 Array

| Array                              | S7-300/400 | S7-1200 | S7-1500 |
|------------------------------------|------------|---------|---------|
| ARRAY [...] OF <type> (Seite 1114) | X          | X       | X       |

Tabelle 9-8 Strukturen

| Strukturen          | S7-300/400 | S7-1200 | S7-1500 |
|---------------------|------------|---------|---------|
| STRUCT (Seite 1119) | X          | X       | X       |

Tabelle 9-9 Zeiger

| Zeiger               | S7-300/400 | S7-1200 | S7-1500 |
|----------------------|------------|---------|---------|
| POINTER (Seite 1120) | X          | -       | X       |
| ANY (Seite 1122)     | X          | -       | X       |
| VARIANT (Seite 1125) | -          | X       | X       |

Tabelle 9-10 Parametertypen

| Parametertypen         | S7-300/400 | S7-1200 | S7-1500 |
|------------------------|------------|---------|---------|
| TIMER (Seite 1126)     | X          | -       | X       |
| COUNTER (Seite 1126)   | X          | -       | X       |
| BLOCK_FC (Seite 1126)  | X          | -       | X       |
| BLOCK_FB (Seite 1126)  | X          | -       | X       |
| BLOCK_DB (Seite 1126)  | X          | -       | -       |
| BLOCK_SDB (Seite 1126) | X          | -       | -       |
| BLOCK_SFB (Seite 1126) | X          | -       | -       |
| BLOCK_SFC (Seite 1126) | X          | -       | -       |
| VOID (Seite 1126)      | X          | X       | X       |

Tabelle 9-11 PLC-Datentypen

| PLC-Datentypen            | S7-300/400 | S7-1200 | S7-1500 |
|---------------------------|------------|---------|---------|
| PLC-Datentyp (Seite 1127) | X          | X       | X       |

Tabelle 9-12 Systemdatentypen

| Systemdatentypen   | S7-300/400      | S7-1200 | S7-1500 |
|--|-----------------|---------|---------|
| IEC_TIMER (Seite 1128)   | X <sup>1)</sup> | X       | X       |
| IEC_LTIMER (Seite 1128)  | -               | -       | X       |
| IEC_SCOUNTER (Seite 1128)  | -               | X       | X       |
| IEC_USCOUNTER (Seite 1128)   | -               | X       | X       |
| IEC_COUNTER (Seite 1128)   | X <sup>2)</sup> | X       | X       |
| IEC_UCOUNTER (Seite 1128)  | -               | X       | X       |
| IEC_DCOUNTER (Seite 1128)  | -               | X       | X       |
| IEC_UDCOUNTER (Seite 1128)   | -               | X       | X       |
| IEC_LCOUNTER (Seite 1128)  | -               | -       | X       |
| IEC_ULCOUNTER (Seite 1128)   | -               | -       | X       |
| ERROR_STRUCT (Seite 1128)  | -               | X       | X       |
| NREF (Seite 1128)  | -               | X       | X       |
| CREF (Seite 1128)  | -               | X       | X       |
| FBTREF (Seite 1128)  | -               | -       | -       |
| VREF (Seite 1128)  | -               | -       | -       |
| STARTINFO (Seite 1128)   | X               | -       | X       |
| SSL_HEADER (Seite 1128)  | X               | -       | X       |
| CONDITIONS (Seite 1128)  | -               | X       | X       |
| TADDR_Param (Seite 1128)   | -               | X       | X       |
| TCON_Param (Seite 1128)  | -               | X       | X       |
| <sup>1)</sup> Bei S7-300/400 CPUs wird der Datentyp durch TP, TON und TOF repräsentiert.   |                 |         |         |
| <sup>2)</sup> Bei S7-300/400 CPUs wird der Datentyp durch CTU, CTD und CTUD repräsentiert. |                 |         |         |

Tabelle 9-13 Hardwaredatentypen

| Hardwaredatentypen        | S7-300/400 | S7-1200 | S7-1500 |
|---------------------------|------------|---------|---------|
| REMOTE (Seite 1130)       | -          | X       | X       |
| GEOADDR (Seite 1130)      | -          | -       | X       |
| HW_ANY (Seite 1130)       | -          | X       | X       |
| HW_DEVICE (Seite 1130)    | -          | X       | X       |
| HW_DPMaster (Seite 1130)  | -          | -       | X       |
| HW_DPSLAVE (Seite 1130)   | -          | X       | X       |
| HW_IO (Seite 1130)        | -          | X       | X       |
| HW_IOSYSTEM (Seite 1130)  | -          | X       | X       |
| HW_SUBMODULE (Seite 1130) | -          | X       | X       |
| HW_MODULE (Seite 1130)    | -          | -       | X       |
| HW_INTERFACE (Seite 1130) | -          | X       | X       |

| Hardwaredatentypen        | S7-300/400 | S7-1200 | S7-1500 |
|---------------------------|------------|---------|---------|
| HW_IEPORT (Seite 1130)    | -          | X       | X       |
| HW_HSC (Seite 1130)       | -          | X       | X       |
| HW_PWM (Seite 1130)       | -          | X       | X       |
| HW_PTO (Seite 1130)       | -          | X       | X       |
| AOM_AID (Seite 1130)      | -          | X       | X       |
| AOM_IDENT (Seite 1130)    | -          | X       | X       |
| EVENT_ANY (Seite 1130)    | -          | X       | X       |
| EVENT_ATT (Seite 1130)    | -          | X       | X       |
| EVENT_HWINT (Seite 1130)  | -          | X       | X       |
| OB_ANY (Seite 1130)       | -          | X       | X       |
| OB_DELAY (Seite 1130)     | -          | X       | X       |
| OB_TOD (Seite 1130)       | -          | X       | X       |
| OB_CYCLIC (Seite 1130)    | -          | X       | X       |
| OB_ATT (Seite 1130)       | -          | X       | X       |
| OB_PCYCLE (Seite 1130)    | -          | X       | X       |
| OB_HWINT (Seite 1130)     | -          | X       | X       |
| OB_DIAG (Seite 1130)      | -          | X       | X       |
| OB_TIMEERROR (Seite 1130) | -          | X       | X       |
| OB_STARTUP (Seite 1130)   | -          | X       | X       |
| PORT (Seite 1130)         | -          | X       | X       |
| RTM (Seite 1130)          | -          | X       | X       |
| PIP (Seite 1130)          | -          | -       | X       |
| CONN_ANY (Seite 1130)     | -          | X       | X       |
| CONN_PRG (Seite 1130)     | -          | X       | X       |
| CONN_OUC (Seite 1130)     | -          | X       | X       |
| CONN_R_ID (Seite 1130)    | -          | -       | X       |
| DB_ANY (Seite 1130)       | -          | X       | X       |
| DB_WWW (Seite 1130)       | -          | X       | X       |

**Hinweis**

Abhängig von der CPU-Version können die tatsächlich gültigen Datentypen geringfügig von der Tabelle abweichen.



## Binärzahlen

## BOOL (Bit)

### Beschreibung

Ein Operand vom Datentyp BOOL stellt einen Bitwert dar und enthält einen der folgenden Werte:

- TRUE
- FALSE

Die folgende Tabelle zeigt die Eigenschaften des Datentyps BOOL:

| Länge (Bit) | Format                     | Wertebereich  | Beispiele für Werteingaben  |
|-------------|----------------------------|---|-----------------------------|
| 1           | Boolesch                   | FALSE oder TRUE<br>BOOL#0 oder BOOL#1<br>BOOL#FALSE oder<br>BOOL#TRUE | TRUE<br>BOOL#1<br>BOOL#TRUE |
|             | Ganzzahlen ohne Vorzeichen | 0 oder 1  | 1                           |
|             | Dualzahlen                 | 2#0 oder 2#1  | 2#0                         |
|             | Oktalzahlen                | 8#0 oder 8#1  | 8#1                         |
|             | Hexadezimalzahlen          | 16#0 oder 16#1  | 16#1                        |

### Hinweis

#### Gültig für CPUs der Baureihe S7-1500

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" hat das Bit eine Länge von 1 Byte.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

## Bitfolgen

## BYTE (Byte)

### Beschreibung

Ein Operand vom Datentyp BYTE ist eine Bitfolge aus 8 Bit.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps BYTE:

| Länge (Bit) | Format                     | Wertebereich                       | Beispiele für Werteingaben                      |
|-------------|----------------------------|------------------------------------|---|
| 8           | Ganzzahlen ohne Vorzeichen | -128 bis 255                       | 15, BYTE#15, B#15                               |
|             | Dualzahlen                 | 2#0 bis 2#11111111                 | 2#00001111,<br>BYTE#2#00001111,<br>B#2#00001111 |
|             | Oktalzahlen                | 8#0 bis 8#377                      | 8#17, BYTE#8#17, B#8#17,                        |
|             | Hexadezimalzahlen          | B#16#0 bis B#16#FF, 16#0 bis 16#FF | 16#0F, BYTE#16#0F, B#16#0F                      |

**Hinweis**

Der Datentyp BYTE kann nicht auf größer oder kleiner verglichen werden. Er kann dezimal nur mit den gleichen Daten versorgt werden, die auch die Datentypen SINT und USINT verarbeiten können.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

**WORD**

**Beschreibung**

Ein Operand vom Datentyp WORD ist eine Bitfolge aus 16 Bit.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps WORD:

| Länge (Bit) | Format                     | Wertebereich                           | Beispiele für Werteingabe   |
|-------------|----------------------------|--|---|
| 16          | Ganzzahlen ohne Vorzeichen | -32768 bis 65535                       | 61680, WORD#61680, W#61680  |
|             | Dualzahlen                 | 2#0 bis 2#1111111111111111             | 2#1111000011110000,<br>WORD#2#1111000011110000,<br>W#2#1111000011110000 |
|             | Oktalzahlen                | 8#0 bis 8#177777                       | 8#170360,<br>WORD#8#170360,<br>W#8#170360                               |
|             | Hexadezimalzahlen          | W#16#0 bis W#16#FFFF, 16#0 bis 16#FFFF | 16#F0F0, WORD#16#F0F0, W#16#F0F0  |
|             | BCD                        | C#0 bis C#999                          | C#55  |
|             | Dezimalfolge               | B#(0,0) bis B#(255,255)                | B#(127,200)   |

**Hinweis**

Der Datentyp WORD kann nicht auf größer oder kleiner verglichen werden. Er kann dezimal nur mit den gleichen Daten versorgt werden, die auch die Datentypen INT und UINT verarbeiten können.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

**DWORD**

**Beschreibung**

Ein Operand vom Datentyp DWORD ist eine Bitfolge aus 32 Bit.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DWORD:

| Länge (Bit) | Format                     | Wertebereich   | Beispiele für Werteingaben   |
|-------------|----------------------------|--|--|
| 32          | Ganzzahlen ohne Vorzeichen | -2147483648 bis 4294967295                                     | 15793935, DWORD#15793935, DW#15793935  |
|             | Dualzahlen                 | 2#0 bis 2#11111111111111111111111111111111                     | 2#0000000011110000111111110001111, DWORD#2#00000000111100001111111100001111, DW#2#000000001111000011111100001111 |
|             | Oktalzahlen                | 8#0 bis 8#3777777777   | 8#74177417, DWORD#8#74177417, DW#8#74177417  |
|             | Hexadezimalzahlen          | DW#16#00000000 bis DW#16#FFFFFFFF, 16#00000000 bis 16#FFFFFFFF | 16#00F0FF0F, DWORD#16#00F0FF0F, DW#16#00F0FF0F   |
|             | Dezimalfolge               | B#(0,0,0,0) bis B#(255,255,255,255)                            | B#(127,200,127,200)  |

**Hinweis**

Der Datentyp DWORD kann nicht auf größer oder kleiner verglichen werden. Er kann dezimal nur mit den gleichen Daten versorgt werden, die auch die Datentypen DINT und UDINT verarbeiten können.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)  
 Konstanten (Seite 1058)

**LWORD**

**Beschreibung**

Ein Operand vom Datentyp LWORD ist eine Bitfolge aus 64 Bit.  
 Die folgende Tabelle zeigt die Eigenschaften des Datentyps LWORD:

| Länge (Bit) | Format                     | Wertebereich  | Beispiele für Werteingaben  |
|-------------|----------------------------|---|---|
| 64          | Ganzzahlen ohne Vorzeichen | -9223372036854775808 bis 18446744073709551615   | 26123590360715,<br>LWORD#26123590360715,<br>LW#26123590360715   |
|             | Dualzahlen                 | 2#0 bis<br>2#11111111111111111111111111111111<br>11111111111111111111111111111111<br>11111111111111111111111111111111 | 2#0000000000000000000010111110000100101111010100101101111010001011,<br>LWORD#2#00000000000000000000001011110000100101111010101010111010001011,<br>LW#2#000000000000000000001011110000100101111010100101111010001011 |
|             | Oktalzahlen                | 8#0 bis<br>8#17777777777777777777777777777777   | 8#13724557213,<br>LWORD#8#13724557213,<br>LW#8#13724557213  |
|             | Hexadezimalzahlen          | LW#16#00000000 bis<br>LW#16#FFFFFFFFFFFFFFF, 16#00000000 bis<br>16#FFFFFFFFFFFFFFF                                    | 16#00000005F52DE8B,<br>LWORD#16#00000005F52DE8B,<br>LW#16#00000005F52DE8B   |
|             | Dezimalfolge               | B#(0,0,0,0,0,0,0,0) bis<br>B#(255,255,255,255,255,255,255,255)  | B#(127,200,127,200,127,200,127,200)   |

**Hinweis**

Der Datentyp LWORD kann nicht auf größer oder kleiner verglichen werden. Er kann dezimal nur mit den gleichen Daten versorgt werden, die auch die Datentypen LINT und ULINT verarbeiten können.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)  
 Konstanten (Seite 1058)

## Ganzzahlen

### SINT (8-Bit-Ganzzahlen)

#### Beschreibung

Ein Operand vom Datentyp SINT (Short INT) hat eine Länge von 8 Bit und besteht aus zwei Komponenten: einem Vorzeichen und einem Zahlenwert im Zweierkomplement. Die Signalzustände der Bits 0 bis 6 stehen für die Größe der Zahl. Der Signalzustand von Bit 7 stellt das Vorzeichen dar. Das Vorzeichen kann den Signalzustand "0" für positiv oder "1" für negativ annehmen.

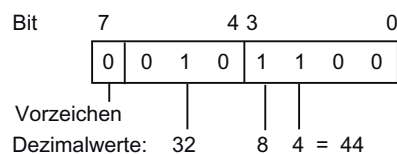
Ein Operand vom Datentyp SINT belegt im Speicher ein BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps SINT:

| Länge (Bit) | Format                          | Wertebereich       | Beispiele für Werteingaben     |
|-------------|---------------------------------|--------------------|--------------------------------|
| 8           | Ganzzahlen mit Vorzeichen       | -128 bis 127       | +44, SINT#+44                  |
|             | Dualzahlen                      | 2#0 bis 2#01111111 | 2#00101100,<br>SINT#2#00101100 |
|             | Oktalzahlen (nur positiv)       | 8#0 bis 8#177      | 8#54, SINT#8#54                |
|             | Hexadezimalzahlen (nur positiv) | 16#0 bis 16#7F     | 16#2C, SINT#16#2C              |

#### Beispiel

Das folgende Bild zeigt die Ganzzahl +44 als Dualzahl:



#### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

### USINT (8-Bit-Ganzzahlen)

#### Beschreibung

Ein Operand vom Datentyp USINT (Unsigned Short INT) hat eine Länge von 8 Bit und enthält Zahlenwerte ohne Vorzeichen.

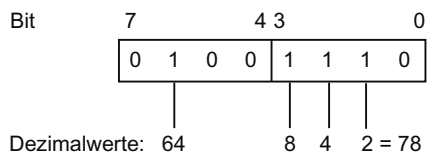
Ein Operand vom Datentyp USINT belegt im Speicher ein BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps USINT:

| Länge (Bit) | Format                     | Wertebereich       | Beispiele für Werteingaben      |
|-------------|----------------------------|--------------------|---------------------------------|
| 8           | Ganzzahlen ohne Vorzeichen | 0 bis 255          | 78, USINT#78                    |
|             | Dualzahlen                 | 2#0 bis 2#11111111 | 2#01001110,<br>USINT#2#01001110 |
|             | Oktalzahlen                | 8#0 bis 8#377      | 8#116, USINT#8#116              |
|             | Hexadezimalzahlen          | 16#0 bis 16#FF     | 16#4E, USINT#16#4E              |

### Beispiel

Das folgende Bild zeigt die Ganzzahl 78 als Dualzahl:



### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

### INT (16-Bit-Ganzzahlen)

#### Beschreibung

Ein Operand vom Datentyp INT hat eine Länge von 16 Bit und besteht aus zwei Komponenten: einem Vorzeichen und einem Zahlenwert im Zweierkomplement. Die Signalzustände der Bits 0 bis 14 stehen für die Größe der Zahl. Der Signalzustand von Bit 15 stellt das Vorzeichen dar. Das Vorzeichen kann den Signalzustand "0" für positiv oder "1" für negativ annehmen.

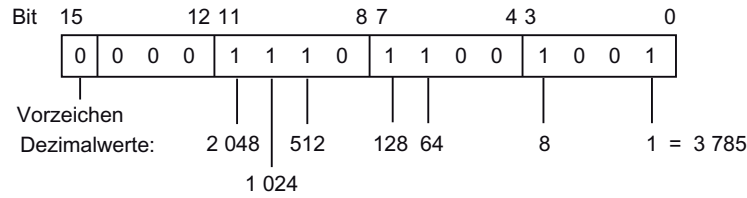
Ein Operand vom Datentyp INT belegt im Speicher zwei BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps INT:

| Länge (Bit) | Format                          | Wertebereich               | Beispiele für Werteingaben                    |
|-------------|---------------------------------|----------------------------|---|
| 16          | Ganzzahlen mit Vorzeichen       | -32768 bis 32767           | +3785, INT#+3785                              |
|             | Dualzahlen (nur positiv)        | 2#0 bis 2#0111111111111111 | 2#0000111011001001,<br>INT#2#0000111011001001 |
|             | Oktalzahlen                     | 8#0 bis 8#77777            | 8#7311, INT#8#7311                            |
|             | Hexadezimalzahlen (nur positiv) | 16#0 bis 16#7FFF           | 16#0EC9, INT#16#0EC9                          |

**Beispiel**

Das folgende Bild zeigt die Ganzzahl +3785 als Dualzahl:



**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Konstanten (Seite 1058)

**UINT (16-Bit-Ganzzahlen)**

**Beschreibung**

Ein Operand vom Datentyp UINT (Unsigned INT) hat eine Länge von 16 Bit und enthält Zahlenwerte ohne Vorzeichen.

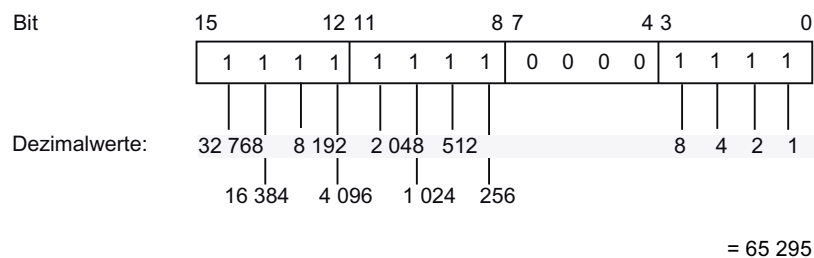
Ein Operand vom Datentyp UINT belegt im Speicher zwei BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps UINT:

| Länge (Bit) | Format                     | Wertebereich               | Beispiele für Werteingaben                  |
|-------------|----------------------------|----------------------------|---|
| 16          | Ganzzahlen ohne Vorzeichen | 0 bis 65535                | 65295, UINT#65295                           |
|             | Dualzahlen                 | 2#0 bis 2#1111111111111111 | 2#1111111100001111, UINT#2#1111111100001111 |
|             | Oktalzahlen                | 8#0 bis 8#177777           | 8#177417, UINT#8#177417                     |
|             | Hexadezimalzahlen          | 16#0 bis 16#FFFF           | 16#FF0F, UINT#16#FF0F                       |

**Beispiel**

Das folgende Bild zeigt die Ganzzahl 65295 als Dualzahl:



**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Konstanten (Seite 1058)

**DINT (32-Bit-Ganzzahlen)**

**Beschreibung**

Ein Operand vom Datentyp DINT (Double INT) hat eine Länge von 32 Bit und besteht aus zwei Komponenten: einem Vorzeichen und einem Zahlenwert im Zweierkomplement. Die Signalzustände der Bits 0 bis 30 stehen für die Größe der Zahl. Der Signalzustand von Bit 31 stellt das Vorzeichen dar. Das Vorzeichen kann den Signalzustand "0" für positiv oder "1" für negativ annehmen.

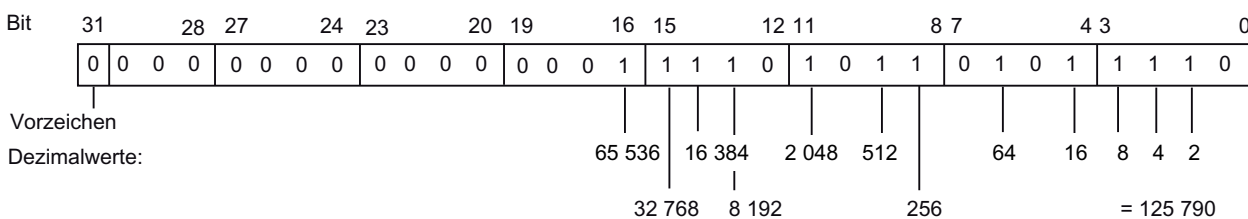
Ein Operand vom Datentyp DINT belegt im Speicher vier BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DINT:

| Länge (Bit) | Format                    | Wertebereich                                  | Beispiele für Werteingaben  |
|-------------|---------------------------|---|---|
| 32          | Ganzzahlen mit Vorzeichen | -2147483648 bis +2147483647                   | 125790, DINT#125790, L#275  |
|             | Dualzahlen (nur positiv)  | 2#0 bis<br>2#01111111111111111111111111111111 | 2#000000000000000011110101101011110,<br>DINT#2#00000000000000001110101101011110 |
|             | Oktalzahlen (nur positiv) | 8#0 bis 8#1777777777                          | 8#365536, DINT#8#365536   |
|             | Hexadezimalzahlen         | 16#00000000 bis<br>16#7FFFFFFF                | 16#0001EB5E,<br>DINT#16#0001EB5E  |

**Beispiel**

Das folgende Bild zeigt die Ganzzahl +125790 als Dualzahl:



**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Konstanten (Seite 1058)



## UDINT (32-Bit-Ganzzahlen)

### Beschreibung

Ein Operand vom Datentyp UDINT (Unsigned Double INT) hat eine Länge von 32 Bit und enthält Zahlenwerte ohne Vorzeichen.

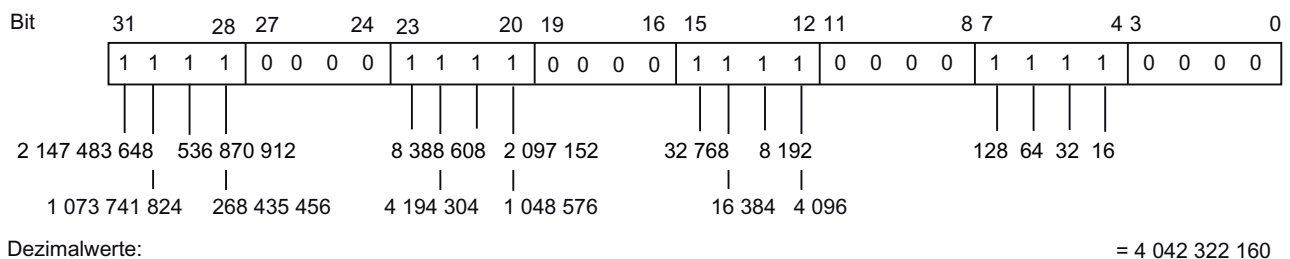
Ein Operand vom Datentyp UDINT belegt im Speicher vier BYTE.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps UDINT:

| Länge (Bit) | Format                     | Wertebereich                                  | Beispiele für Werteingaben  |
|-------------|----------------------------|---|---|
| 32          | Ganzzahlen ohne Vorzeichen | 0 bis 4294967295                              | 4042322160,<br>UDINT#4042322160   |
|             | Dualzahlen                 | 2#0 bis<br>2#11111111111111111111111111111111 | 2#1111000011110000111100001111000011110000,<br>UDINT#2#11110000111100001111000011110000 |
|             | Oktalzahlen                | 8#0 bis 8#377777777777                        | 8#36074170360,<br>UDINT#8#36074170360   |
|             | Hexadezimalzahlen          | 16#00000000 bis<br>16#FFFFFFFF                | 16#F0F0F0F0,<br>UDINT#16#F0F0F0F0   |

### Beispiel

Das folgende Bild zeigt die Ganzzahl 4042322160 als Dualzahl:



### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

## LINT (64-Bit-Ganzzahlen)

### Beschreibung

Ein Operand vom Datentyp LINT (Long INT) hat eine Länge von 64 Bit und besteht aus zwei Komponenten: einem Vorzeichen und einem Zahlenwert im Zweierkomplement. Die Signalzustände der Bits 0 bis 62 stehen für die Größe der Zahl. Der Signalzustand von Bit 63





**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

**Gleitpunktzahlen**

**REAL**

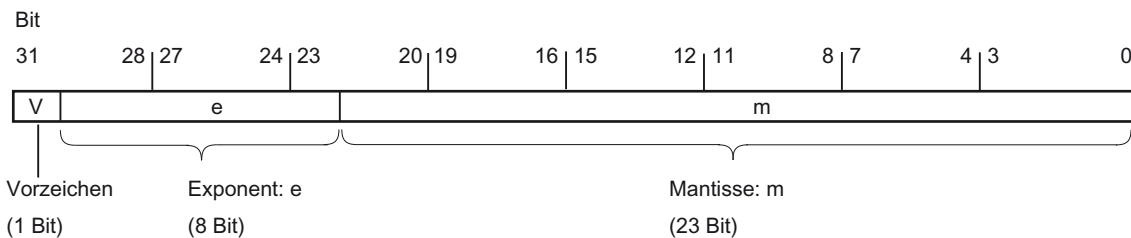
**Beschreibung**

Operanden vom Datentyp REAL haben eine Länge vom 32 Bits und werden zur Darstellung von Gleitpunktzahlen verwendet. Ein Operand vom Datentyp REAL besteht aus den folgenden drei Komponenten:

- Vorzeichen: Das Vorzeichen wird durch den Signalzustand von Bit 31 bestimmt. Das Bit 31 kann die Werte "0" (positiv) und "1" (negativ) annehmen.
- 8-Bit-Exponenten zur Basis 2: Der Exponent wird um eine Konstante (Basis, +127) erhöht, so dass er einen Wertebereich von 0 bis 255 aufweist.
- 23-Bit-Mantisse: Nur der gebrochene Anteil der Mantisse wird dargestellt. Der ganzzahlige Anteil der Mantisse ist bei normalisierten Gleitpunktzahlen immer 1 und wird nicht gespeichert.

Der Datentyp REAL wird mit einer Genauigkeit von 7 Stellen nach dem Komma verarbeitet.

Das folgende Bild zeigt den Aufbau des Datentyps REAL:



**Hinweis**

Bei Gleitpunktzahlen werden nur die von der IEEE754 Norm definierten Genauigkeiten gespeichert. Zusätzlich angegebene Dezimalstellen werden nach IEEE754 gerundet.

Bei häufig geschachtelten arithmetischen Berechnungen kann sich die Anzahl der Dezimalstellen verringern.

Werden mehr Dezimalstellen eingegeben als der Datentyp speichern kann, dann wird die Zahl auf den, der möglichen Genauigkeit in diesem Wertebereich, entsprechenden Wert gerundet.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps REAL:

| Länge (Bit) | Format                        | Wertebereich                           | Beispiele für Werteingabe |
|-------------|-------------------------------|--|---------------------------|
| 32          | Gleitpunktzahlen nach IEEE754 | -3.402823e+38 bis -1.175 495e-38<br>±0 | 1.0e-5, REAL#1.0e-5       |
|             | Gleitpunktzahlen              | +1.175 495e-38 bis +3.402823e+38       | 1.0, REAL#1.0             |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

**LREAL**

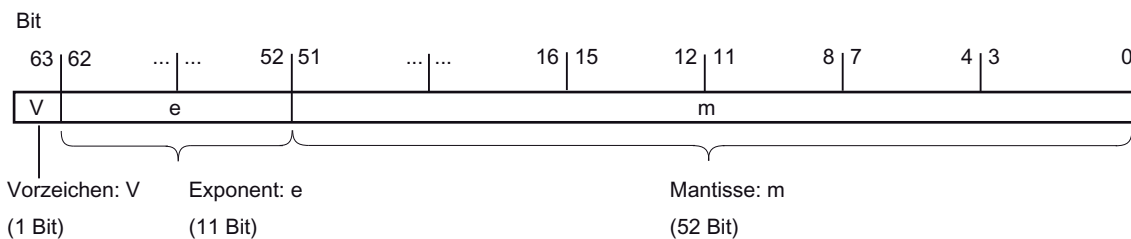
**Beschreibung**

Operanden vom Datentyp LREAL haben eine Länge von 64 Bits und werden zur Darstellung von Gleitpunktzahlen verwendet. Ein Operand vom Datentyp LREAL besteht aus den folgenden drei Komponenten:

- Vorzeichen: Das Vorzeichen wird durch den Signalzustand von Bit 63 bestimmt. Das Bit 63 kann die Werte "0" (positiv) und "1" (negativ) annehmen.
- 11-Bit-Exponenten zur Basis 2: Der Exponent wird um eine Konstante (Basis, +1023) erhöht, so dass er einen Wertebereich von 0 bis 2047 aufweist.
- 52-Bit-Mantisse: Nur der gebrochene Anteil der Mantisse wird dargestellt. Der ganzzahlige Anteil der Mantisse ist bei normalisierten Gleitpunktzahlen immer 1 und wird nicht gespeichert.

Der Datentyp LREAL wird mit einer Genauigkeit von 15 Stellen nach dem Komma verarbeitet.

Das folgende Bild zeigt den Aufbau des Datentyps LREAL:



Die folgende Tabelle zeigt die Eigenschaften des Datentyps LREAL:

| Länge (Bit) | Format                        | Wertebereich   | Beispiele für Werteingabe |
|-------------|-------------------------------|--|---------------------------|
| 64          | Gleitpunktzahlen nach IEEE754 | -1.7976931348623158e+308 bis<br>-2.2250738585072014e-308       | 1.0e-5, LREAL#1.0e-5      |
|             | Gleitpunktzahlen              | ±0<br>+2.2250738585072014e-308 bis<br>+1.7976931348623158e+308 | 1.0, LREAL#1.0            |

---

### Hinweis

Bei Gleitpunktzahlen werden nur die von der IEEE754 Norm definierten Genauigkeiten gespeichert. Zusätzlich angegebene Dezimalstellen werden nach IEEE754 gerundet.

Bei häufig geschachtelten arithmetischen Berechnungen kann sich die Anzahl der Dezimalstellen verringern.

Werden mehr Dezimalstellen eingegeben als der Datentyp speichern kann, dann wird die Zahl auf den, der möglichen Genauigkeit in diesem Wertebereich, entsprechenden Wert gerundet.

---

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

### Ungültige Gleitpunktzahlen

#### Beschreibung

Bei den Datentypen REAL und LREAL unterscheidet man vier Zahlenbereiche:

- normalisierte Zahlen, die mit der vollen Genauigkeit gespeichert werden
- denormalisierte Zahlen, die nicht mit der vollen Genauigkeit gespeichert werden können
- Unendliche Zahlen: +Inf/-Inf (Infinity)
- Ungültige Zahlen: NaN (Not a Number)

---

### Hinweis

Gleitpunktzahlen werden wie in der IEEE754 Norm abgelegt. Ergebnisse von Konvertierungen bzw. arithmetischen Funktionen mit einer denormalisierten, unendlichen oder NaN (Not a Number) Gleitpunktzahl sind CPU abhängig.

---

Wenn Sie in mathematischen Funktionen nicht mit normalisierten Gleitpunktzahlen arbeiten, dann gibt es im Ergebnis, abhängig davon welche CPU aus welcher Baureihe Sie verwenden, starke Unterschiede.

Eine CPU kann nicht mit denormalisierten Gleitpunktzahlen rechnen, mit Ausnahme von älteren CPU Versionen der Baureihen S7-300 und S7-400. Das Bitmuster einer denormalisierten Zahl wird wie eine Null interpretiert. Fällt ein Rechenergebnis in diesen Bereich, wird mit Null weiter gemacht, wobei die Statusbits OV und OS gesetzt werden (Zahlenbereichsunterschreitung).

Obwohl die Werte von ungültigen Gleitpunktzahlen nur mit einer eingeschränkten Genauigkeit bei mathematischen Funktionen dargestellt werden können, können Zahlen mit einem Exponenten von -39 (z. B. 2.4408e-039) im TIA-Portal beobachtet werden und stellen daher

nicht unbedingt ein fehlerhaftes Ergebnis dar. Daher kann es vorkommen, dass Gleitpunktzahlenwerte außerhalb der gültigen Zahlenwerte liegen.

---

**Hinweis****Für CPUs der Baureihe S7-1200 V1, V2 und V3 gilt:**

Die Vergleichsoperation "Gleich" verwendet das Bitmuster der ungültigen Gleitpunktzahl. Wenn zwei "NaN-Zahlen" mit dem gleichen Bitmuster verglichen werden, dann liefert der Ausgang der Vergleichsoperation "Gleich" das Ergebnis TRUE.

---

**Hinweis****Für CPUs der Baureihe S7-1200 V4 und S7-1500 gilt:**

Werden zwei ungültige Zahlen (NaN) miteinander verglichen, so ist das Ergebnis immer FALSE, unabhängig vom Bitmuster der ungültigen Zahl oder der Relation (>, >, ...).

---

**Hinweis****Vergleich von denormalisierten Gleitpunktzahlen**

Bei der Vergleichsoperation "Gleich" mit zwei denormalisierten Gleitpunktzahlen wird der Ausgang bei CPUs der Baureihen S7-300/400 auf den Signalzustand "0" und bei CPUs der Baureihen S7-1200/1500 auf den Signalzustand "1" gesetzt.

---

Wenn die Eingangsvariablen einer mathematischen Funktion eine ungültige Gleitpunktzahl darstellen, dann wird auch eine ungültige Gleitpunktzahl als Ergebnis ausgegeben.

Zur Auswertung möglicher Fehler durch ungültige Gleitpunktzahlen, haben Sie folgende Möglichkeiten:

- In KOP/FUP und SCL können Sie den Freigabeausgang ENO auf FALSE abfragen
- In AWL können Sie das Statusbit OV auswerten

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

**Zeiten****S5TIME (Zeitdauer)****Format**

Der Datentyp S5TIME speichert die Zeitdauer im BCD-Format. Die Zeitdauer ist das Produkt aus einem Zeitwert im Bereich von 0 bis 999 und einer Zeitbasis. Die Zeitbasis gibt das Intervall an, in dem eine Zeit den Zeitwert um je eine Einheit vermindert, bis er "0" erreicht. Über die Zeitbasis kann die Auflösung der Zeitwerte gesteuert werden.





**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)  
 Konstanten (Seite 1058)

**TIME (IEC-Zeit)**

**Beschreibung**

Der Inhalt eines Operanden vom Datentyp TIME wird als Millisekunden interpretiert. Die Darstellung enthält Angaben für Tage (d), Stunden (h), Minuten (m), Sekunden (s) und Millisekunden (ms).

Die folgende Tabelle zeigt die Eigenschaften des Datentyps TIME:

| Länge (Bit) | Format                   | Wertebereich                                  | Beispiele für Werteingabe                   |
|-------------|--------------------------|---|---|
| 32          | Zeitdauer mit Vorzeichen | T#-24d20h31m23s648ms bis T#+24d20h31m23s647ms | T#10d20h30m20s630ms, TIME#10d20h30m20s630ms |

Es ist nicht erforderlich, alle Zeiteinheiten anzugeben. So ist z. B. T#5h10s gültig. Wenn nur eine Einheit angegeben wird, darf der absolute Wert an Tagen, Stunden und Minuten die oberen oder unteren Grenzwerte nicht überschreiten. Wenn mehr als eine Zeiteinheit angegeben wird, darf der Wert die Einheit 24 Tage, 23 Stunden, 59 Minuten, 59 Sekunden oder 999 Millisekunden nicht überschreiten.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)  
 Konstanten (Seite 1058)

**LTIME (IEC-Zeit)**

**Beschreibung**

Der Inhalt eines Operanden vom Datentyp LTIME wird als Nanosekunden interpretiert. Die Darstellung enthält Angaben für Tage (d), Stunden (h), Minuten (m), Sekunden (s), Millisekunden (ms), Mikrosekunden (us) und Nanosekunden (ns).

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LTIME:

| Länge (Bit) | Format                   | Wertebereich  | Beispiele für Werteingabe   |
|-------------|--------------------------|---|---|
| 64          | Zeitdauer mit Vorzeichen | LT#-106751d23h47m16s854ms775us808ns bis LT#+106751d23h47m16s854ms775us807ns | LT#11350d20h25m14s830ms652us315ns, LTIME#11350d20h25m14s830ms652us315ns |

Es ist nicht erforderlich, alle Zeiteinheiten anzugeben. So ist z. B. LT#5h10s gültig. Wenn nur eine Einheit angegeben wird, darf der absolute Wert an Tagen, Stunden und Minuten die

oberen oder unteren Grenzwerte nicht überschreiten. Wenn mehr als eine Zeiteinheit angegeben wird, darf der Wert die Einheit 106751 Tage, 23 Stunden, 59 Minuten, 59 Sekunden, 999 Millisekunden, 999 Mikrosekunden oder 999 Nanosekunden nicht überschreiten.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)  
Konstanten (Seite 1058)

**Datum und Uhrzeit**

**DATE**

**Format**

Der Datentyp DATE speichert ein Datum als vorzeichenlose Ganzzahl. Die Darstellung enthält das Jahr, den Monat und den Tag.

Der Inhalt eines Operanden vom Datentyp DATE entspricht im Hexadezimalformat der Anzahl der Tage seit 01.01.1990 (16#0000).

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DATE:

| Länge (Byte) | Format                        | Wertebereich                     | Beispiele für Werteingabe     |
|--------------|-------------------------------|----------------------------------|-------------------------------|
| 2            | IEC-Datum<br>(Jahr-Monat-Tag) | D#1990-01-01 bis<br>D#2168-12-31 | D#2009-12-31, DATE#2009-12-31 |
|              | Hexadezimalzahlen             | 16#0000 bis 16#FF62              | 16#00F2                       |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)  
Konstanten (Seite 1058)

**TOD (TIME\_OF\_DAY)**

**Format**

Der Datentyp TOD (TIME\_OF\_DAY) belegt ein Doppelwort und speichert die Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr) als vorzeichenlose Ganzzahl.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps TOD:

| Länge (Byte) | Format                                  | Wertebereich                             | Beispiele für Werteingabe                         |
|--------------|---|--|---|
| 4            | Tageszeit<br>(Stunden:Minuten:Sekunden) | TOD#00:00:00.000 bis<br>TOD#23:59:59.999 | TOD#10:20:30.400,<br>TIME_OF_DAY#10:20:30.40<br>0 |

Die Angabe von Stunden, Minuten und Sekunden ist erforderlich. Die Angabe der Millisekunden ist optional.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

## LTOD (LTIME\_OF\_DAY)

### Format

Der Datentyp LTOD (LTIME\_OF\_DAY) belegt zwei Doppelworte und speichert die Anzahl der Nanosekunden seit Tagesbeginn (0:00 Uhr) als vorzeichenlose Ganzzahl.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LTOD:

| Länge (Byte) | Format   | Wertebereich   | Beispiele für Werteingabe   |
|--------------|--|--|---|
| 8            | Tageszeit<br>(Stunden:Minuten:Sekunden.<br>Millisekunden.Mikrosekunden.<br>Nanosekunden) | LTOD#00:00:00.0000000<br>00 bis<br>LTOD#23:59:59.9999999<br>99 | LTOD#10:20:30.400_365_21<br>5,<br>LTIME_OF_DAY#10:20:30.4<br>00_365_215 |

Die Angabe von Stunden, Minuten und Sekunden ist erforderlich. Die Angabe der Millisekunden, Mikrosekunden und Nanosekunden ist optional.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

## DT (DATE\_AND\_TIME)

### Format

Der Datentyp DT (DATE\_AND\_TIME) speichert Angaben zu Datum und Uhrzeit im BCD-Format.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DT:

| Länge (Byte) | Format   | Wertebereich   | Beispiel für Werteingabe  |
|--------------|--|--|---|
| 8            | Datum und Uhrzeit<br>(Jahr-Monat-Tag-Stunde:Minute:Sekunde <sup>3)</sup> ) | Min.: DT#1990-01-01-0:0:0<br>Max.: DT#2089-12-31-23:59:59.99 | DT#2008-10-25-8:12:34.567,<br>DATE_AND_TIME#2008-10-25-08:12:34.567 |

Die folgende Tabelle zeigt die Struktur des Datentyps DT:

| Byte   | Inhalt                                     | Wertebereich  |
|--|--|---|
| 0  | Jahr                                       | 0 bis 99<br>(Jahre 1990 bis 2089)<br>BCD#90 = 1990<br>...<br>BCD#0 = 2000<br>...<br>BCD#89 = 2089 |
| 1  | Monat                                      | BCD#0 bis BCD#12  |
| 2  | Tag  | BCD#1 bis BCD# 31   |
| 3  | Stunde                                     | BCD#0 bis BCD#23  |
| 4  | Minute                                     | BCD#0 bis BCD#59  |
| 5  | Sekunde                                    | BCD#0 bis BCD#59  |
| 6  | Die beiden höchstwertigen Ziffern von MSEC | BCD#0 bis BCD#999   |
| 7 (4MSB) <sup>1)</sup>   | Die niederwertige Ziffer von MSEC          | BCD#0 bis BCD#9   |
| 7 (4LSB) <sup>2)</sup>   | Wochentag                                  | BCD#1 bis BCD#7<br>BCD#1 = Sonntag<br>...<br>BCD#7 = Samstag                                      |
| <sup>1)</sup> MSB: Most Significant Bit<br><sup>2)</sup> LSB: Least Significant Bit<br><sup>3)</sup> Festpunktzahl |  |   |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Konstanten (Seite 1058)

**LDT (DATE\_AND\_LTIME)**

**Format**

Der Datentyp LDT (DATE\_AND\_LTIME) speichert Angaben zu Datum und Uhrzeit in Nanosekunden seit dem 01.01.1970 0:0.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps LDT:

| Länge (Byte) | Format  | Wertebereich   | Beispiel für Werteingabe   |
|--------------|---|--|----------------------------|
| 8            | Datum und Uhrzeit<br>(Jahr-Monat-Tag-Stunde:Minute:Sekunde) | Min.:<br>LDT#1970-01-01-0:0:0.000<br>000000, 16#0<br><br>Max.:<br>LDT#2262-04-11-23:47:16.<br>854775807,<br>16#7FFF_FFFF_FFFF_FFF<br>F | LDT#2008-10-25-8:12:34.567 |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

**DTL**

**Beschreibung**

Ein Operand vom Datentyp DTL hat eine Länge von 12 Byte und speichert Angaben zu Datum und Uhrzeit in einer vordefinierten Struktur.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps DTL:

| Länge (Byte) | Format  | Wertebereich   | Beispiel für Werteingabe        |
|--------------|---|--|---------------------------------|
| 12           | Datum und Uhrzeit<br>(Jahr-Monat-Tag-Stunde:Minute:Sekunde.Nanosekunde) | Min.:<br>DTL#1970-01-01-00:00:00.0<br><br>Max.:<br>DTL#2554-12-31-23:59:59.9<br>99999999 | DTL#2008-12-16-20:30:20<br>.250 |

Die Struktur des Datentyps DTL setzt sich aus mehreren Komponenten zusammen, die jeweils einen unterschiedlichen Datentyp und Wertebereich aufweisen können. Der Datentyp eines angegebenen Werts muss dabei zum Datentyp der jeweiligen Komponente passen.

Die folgende Tabelle zeigt die Strukturkomponenten des Datentyps DTL und deren Eigenschaften:

| Byte | Komponente | Datentyp | Wertebereich  |
|------|------------|----------|---|
| 0    | Jahr       | UINT     | 1970 bis 2554   |
| 1    |            |          |   |
| 2    | Monat      | USINT    | 1 bis 12  |
| 3    | Tag        | USINT    | 1 bis 31  |
| 4    | Wochentag  | USINT    | 1 (Sonntag) bis 7 (Samstag)<br>Der Wochentag wird bei der Werteingabe nicht berücksichtigt. |

| Byte | Komponente  | Datentyp | Wertebereich    |
|------|-------------|----------|-----------------|
| 5    | Stunde      | USINT    | 0 bis 23        |
| 6    | Minute      | USINT    | 0 bis 59        |
| 7    | Sekunde     | USINT    | 0 bis 59        |
| 8    | Nanosekunde | UDINT    | 0 bis 999999999 |
| 9    |             |          |                 |
| 10   |             |          |                 |
| 11   |             |          |                 |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)  
 Konstanten (Seite 1058)

**Zeichenfolgen**

**CHAR (Zeichen)**

**Beschreibung**

Ein Operand vom Datentyp CHAR hat eine Länge von 8 Bit und belegt ein BYTE im Speicher.

Der Datentyp CHAR speichert ein einziges Zeichen, das im ASCII-Format abgelegt ist. Informationen zur Kodierung von Sonderzeichen finden Sie unter "Siehe auch".

Die folgende Tabelle zeigt den Wertebereich des Datentyps CHAR:

| Länge (Bit) | Format        | Wertebereich      | Beispiel für Werteingaben |
|-------------|---------------|-------------------|---------------------------|
| 8           | ASCII-Zeichen | ASCII-Zeichensatz | 'A', CHAR#'A'             |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)  
 Konstanten (Seite 1058)

**STRING**

**Beschreibung**

Ein Operand vom Datentyp STRING speichert mehrere Zeichen in einer Zeichenkette, die aus maximal 254 Zeichen bestehen kann. In einer Zeichenkette sind alle Zeichen des ASCII-Codes zugelassen. Die Angabe der Zeichen erfolgt dabei in einfachen Anführungszeichen.

Die folgende Tabelle zeigt die Eigenschaften einer STRING-Variablen:

| Länge (Byte)  | Format  | Wertebereich      | Beispiel für Werteingabe |
|---|---|-------------------|--------------------------|
| n + 2 *   | ASCII-Zeichenkette einschließlich Sonderzeichen | 0 bis 254 Zeichen | 'Name', STRING#'NAME'    |
| * Ein Operand vom Datentyp STRING belegt im Speicher zwei Bytes mehr als die angegebene Maximallänge. |   |                   |                          |

Eine Zeichenkette kann auch Sonderzeichen enthalten. Für die Angabe von Steuerungszeichen, Dollarzeichen und einfachen Anführungszeichen wird das Fluchtsymbol \$ verwendet.

Die folgende Tabelle zeigt Beispiele für die Notation von Sonderzeichen:

| Zeichen      | Hex       | Bedeutung  | Beispiel                    |
|--------------|-----------|--|-----------------------------|
| \$L oder \$l | 0A        | Zeilenvorschub   | '\$LText', '\$0AText'       |
| \$N oder \$n | 0A und 0D | Zeilenumbruch<br>Der Zeilenumbruch belegt 2 Zeichen in der Zeichenkette. | '\$NText', '\$0A\$0DText'   |
| \$P oder \$p | 0C        | Seitenvorschub   | '\$PText', '\$0CText'       |
| \$R oder \$r | 0D        | Wagenrücklauf (CR)   | '\$RText', '\$0DText'       |
| \$T oder \$t | 09        | Tabulator  | '\$TText', '\$09Text'       |
| \$\$         | 24        | Dollarzeichen  | '100\$\$t', '100\$26'       |
| \$'          | 27        | Einfaches Anführungszeichen  | '\$'Text\$', '\$27Text\$27' |

Die Maximallänge der Zeichenkette kann bei der Deklaration eines Operanden mithilfe von eckigen Klammern nach dem Schlüsselwort STRING angegeben werden (z. B. STRING[4]). Wenn die Angabe der Maximallänge weggelassen wird, wird für den jeweiligen Operanden die Standardlänge von 254 Zeichen eingestellt.

Wenn die tatsächliche Länge einer angegebenen Zeichenkette kürzer als die deklarierte Maximallänge ist, werden die Zeichen rechtsbündig in die Zeichenkette geschrieben, die restlichen Zeichenplätze bleiben undefiniert. Bei der Wertbearbeitung werden nur belegte Zeichenplätze berücksichtigt.

---

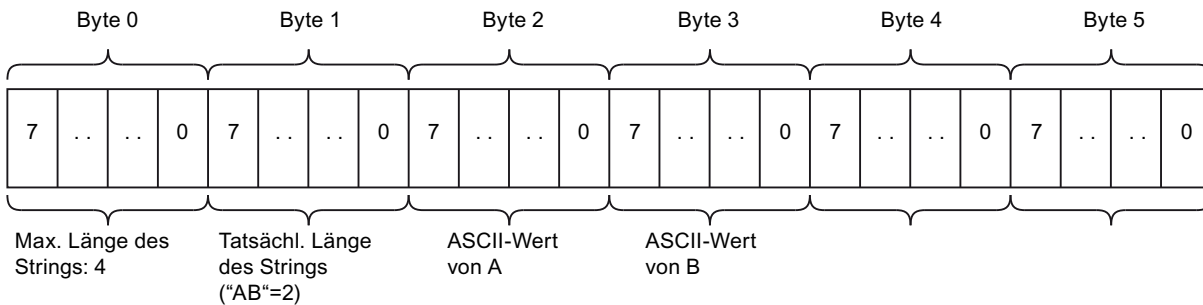
#### Hinweis

Für S7-300/400 CPUs ist zu beachten: Wurde eine temporäre Variable vom Datentyp STRING definiert, muss das BYTE "Max. Länge des Strings" vor der Verwendung der Variablen im Anwenderprogramm mit der definierten Länge beschrieben werden.

---

**Beispiel**

Das folgende Beispiel zeigt die Reihenfolge der Bytes bei der Angabe des Datentyps STRING[4] mit dem Ausgangswert 'AB':



**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

**Array**

**Format von Array (16-Bit-Grenzen)**

**Beschreibung**

Der Datentyp Array repräsentiert eine Datenstruktur, die aus einer festen Anzahl von Komponenten des gleichen Datentyps besteht. Es sind mit der Ausnahme von Array alle Datentypen zugelassen.

Eine Variable mit dem Datentyp Array beginnt immer an einer WORD-Grenze.

Die Adressierung der Feldkomponenten erfolgt über einen Index. Die Indexgrenzen werden bei der Deklaration des Feldes nach dem Schlüsselwort Array in eckigen Klammern definiert. Der untere Grenzwert muss kleiner oder gleich dem oberen Grenzwert sein. Ein Feld kann bis zu sechs Dimensionen enthalten, deren Grenzen durch je ein Komma getrennt angegeben werden.

Die folgende Tabelle zeigt die Eigenschaft des Datentyps Array:

| Länge                                   | Format   | Indexgrenzen                  | Datentyp   |
|---|--|-------------------------------|--|
| Komponentenanzahl * Länge des Datentyps | Array [unterer Grenzwert.. oberer Grenzwert] of <Datentyp> | [-32768..32767] of <Datentyp> | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeilen, Zeichenfolgen, Strukturen |



## Beispiel

Das folgende Beispiel zeigt, wie Operanden vom Datentyp Array deklariert werden können:

| Name     | Deklaration               | Kommentar                                |
|----------|---------------------------|--|
| Messwert | Array[1..20] of REAL      | Eindimensionales Feld mit 20 Komponenten |
| Uhrzeit  | Array[-5..5] of INT       | Eindimensionales Feld mit 11 Komponenten |
| Zeichen  | Array[1..2, 3..4] of CHAR | Zweidimensionales Feld mit 4 Komponenten |

## Maximale Array-Grenzen

Die maximalen Array-Grenzen hängen von folgenden Faktoren ab:

- Datentyp der Array-Elemente
- Maximale Speicherkapazität der CPU (Weitere Informationen finden Sie im jeweiligen Gerätehandbuch)

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Indirektes Indizieren von ARRAY-Komponenten (Seite 1079)

## Format von Array (32-Bit-Grenzen)

## Beschreibung

Der Datentyp Array repräsentiert eine Datenstruktur, die aus einer festen Anzahl von Komponenten des gleichen Datentyps besteht. Es sind mit der Ausnahme von Array alle Datentypen zugelassen.

Die Adressierung der Feldkomponenten erfolgt über einen Index. Die Indexgrenzen werden bei der Deklaration des Feldes nach dem Schlüsselwort Array in eckigen Klammern definiert. Der untere Grenzwert muss kleiner oder gleich dem oberen Grenzwert sein. Ein Feld kann bis zu sechs Dimensionen enthalten, deren Grenzen durch je ein Komma getrennt angegeben werden.

---

### Hinweis

Abhängig von der CPU, ist die Speicherkapazität eines DBs begrenzt und damit auch die Komponentenanzahl des Array. Jedoch kann die Adressierung der Feldkomponenten an jeder beliebigen Stelle innerhalb der Indexgrenzen initialisiert werden.

---

Die folgende Tabelle zeigt die Eigenschaft des Datentyps Array:

| Länge   | Format   | Indexgrenzen                                | Datentyp  |
|---|--|---|---|
| Komponentenanzahl<br>* Länge des<br>Datentyps | Array [unterer<br>Grenzwert.. oberer<br>Grenzwert] of <Datentyp> | [-2147483648..214748<br>3647] of <Datentyp> | Bitfolgen, Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten, Zeichenfolgen,<br>Strukturen |

**Hinweis**

Die Länge des Arrays ist abhängig davon, ob der Baustein mit der Bausteineigenschaft "Standard" oder "mit optimiertem Zugriff" angelegt wurde.

**Hinweis**

**Gültig für CPUs der Baureihe S7-1500**

Bei einem Baustein mit der Bausteineigenschaft "Optimierter Bausteinzugriff" benötigt ein Bit einen Speicherplatz von 1 Byte. Das gilt auch, wenn Sie ein Array of <Datentyp> verwenden. So benötigt z. B. ein Array [0..1] of BOOL in einem optimierten Baustein 8 Byte.

**Beispiel**

Das folgende Beispiel zeigt, wie Operanden vom Datentyp Array deklariert werden können:

| Name     | Deklaration               | Kommentar                                |
|----------|---------------------------|--|
| Messwert | Array[1..20] of REAL      | Eindimensionales Feld mit 20 Komponenten |
| Uhrzeit  | Array[-5..5] of INT       | Eindimensionales Feld mit 11 Komponenten |
| Zeichen  | Array[1..2, 3..4] of CHAR | Zweidimensionales Feld mit 4 Komponenten |

**Maximale Array-Grenzen**

Die maximalen Array-Grenzen hängen von folgenden Faktoren ab:

- Datentyp der Array-Elemente
- Speicherreserve (nur in Bausteinen mit optimiertem Zugriff)  
Weitere Informationen dazu finden Sie unter "Bausteinänderungen ohne Reinitialisierung laden".
- Maximale Größe eines Datenbausteins einer CPU (Weitere Informationen finden Sie im jeweiligen Gerätehandbuch)
- Innerhalb eines Datenbausteins steht Ihnen die gesamte Länge des Arrays zur Verfügung. Innerhalb eines Programmbausteins (OB, FB oder FC) verringert sich die mögliche Länge entsprechend der Speicherkapazität, die der Programmcode benötigt.

### Beispiel anhand einer CPU der Baureihe S7-1200

Die folgende Tabelle zeigt die maximale Elementanzahl innerhalb eines Bausteins mit der Bausteineigenschaft "mit optimiertem Zugriff":

| Datentypbreite (Bit) | Maximale Elementanzahl | Bemerkung  |
|----------------------|------------------------|--|
| 1                    | 524272                 | = 65534*8  |
| 8                    | 65534                  | Den Wert können Sie dem jeweiligen Gerätehandbuch der CPU entnehmen. |
| 16                   | 32767                  | = 65534/2 (Ganzzahldivision, Rest 0)                                 |
| 32                   | 16383                  | = 65534/4 (Ganzzahldivision, Rest 2)                                 |
| 64                   | 8191                   | = 65534/8 (Ganzzahldivision, Rest 6)                                 |

Durch verschiedene technische/interne Beschränkungen kann der tatsächlich nutzbare Speicherbereich um ca. 70 - 100 Byte kleiner sein. Zusätzlich kann z. B. durch die Bausteineigenschaft "Laden ohne Reinitialisierung", der Speicherbereich per Voreinstellung weiter eingeschränkt sein.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Indirektes Indizieren von ARRAY-Komponenten (Seite 1079)

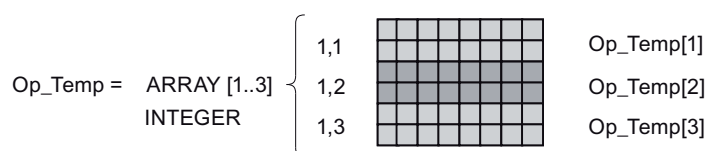
### Beispiel für ein eindimensionales Array

#### Deklaration

Die folgende Tabelle zeigt die Deklaration einer eindimensionalen Array-Variablen:

| Name    | Datentyp           | Kommentar   |
|---------|--------------------|---|
| Op_Temp | Array[1..3] of INT | Eindimensionale Array-Variable mit 3 Komponenten. |

Das folgende Bild zeigt die Struktur der deklarierten Array-Variablen:



### Zugriff auf die Feldkomponente

Auf die Werte der einzelnen Feldkomponenten greifen Sie über einen Index zu. Der Index der ersten Feldkomponente ist [1], der zweiten [2] und der dritten [3]. Um auf den Wert der zweiten

Feldkomponente zuzugreifen, ist in diesem Fall die Angabe "Op\_Temp[2]" im Programm erforderlich.

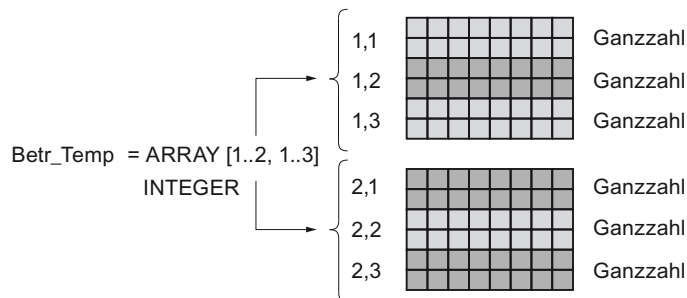
### Beispiel für ein mehrdimensionales Array

#### Deklaration

Die folgende Tabelle zeigt die Deklaration einer zweidimensionalen Array-Variablen:

| Name      | Datentyp                 | Wert     | Kommentar  |
|-----------|--------------------------|----------|--|
| Betr_Temp | Array[1..2, 1..3] of INT | 1,1,4(0) | Zweidimensionale Array-Variable mit 6 Komponenten. Die ersten zwei Komponenten werden mit dem Wert "1" belegt. Die restlichen vier Komponenten werden mit dem Wert "0" belegt. |

Das folgende Bild zeigt die Struktur der deklarierten Array-Variablen:



#### Zugriff auf die Feldkomponenten

Auf die Werte der einzelnen Feldkomponente greifen Sie über einen Index zu. Der Index der ersten Feldkomponente ist z. B. [1,1] und der Index der vierten Feldkomponente [2,1]. Im Programm ist z. B. die Angabe "Betr\_Temp[2,1]" erforderlich, um auf den Wert der vierten Feldkomponente zuzugreifen.

#### Weitere Zugriffsmöglichkeit

Die Variable "Betr\_Temp" können Sie auch als sechsdimensionales Feld deklarieren. Die folgende Tabelle zeigt ein Beispiel für die Deklaration einer sechsdimensionalen Array-Variablen:

| Name      | Datentyp   | Wert | Kommentar                        |
|-----------|--|------|----------------------------------|
| Betr_Temp | Array[1..3, 1..2, 1..3, 1..4, 1..3, 1..4] of INT | -    | Sechsdimensionale Array-Variable |

Der Index der ersten Feldkomponente ist in diesem Fall [1,1,1,1,1,1] und der Index der letzten Komponente ist [3,2,3,4,3,4]. Für die Werte dazwischen geben Sie für jede Dimension den entsprechenden Wert an.

## Strukturen

## STRUCT

### Beschreibung

Der Datentyp STRUCT repräsentiert eine Datenstruktur, die sich aus einer festen Anzahl von Komponenten unterschiedlicher Datentypen zusammensetzt. Auch Komponenten vom Datentyp STRUCT oder ARRAY können in einer Struktur geschachtelt werden. Die Schachtelungstiefe ist dabei auf acht Ebenen beschränkt. Mithilfe von Strukturen können Daten entsprechend der Prozesssteuerung gruppiert und Parameter als eine Dateneinheit übergeben werden.

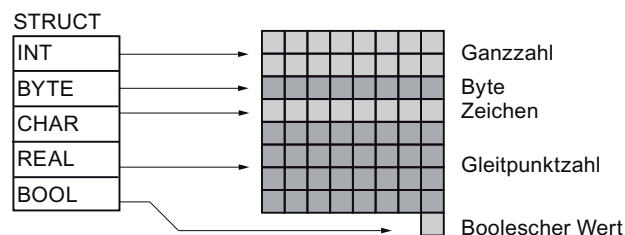
Eine Komponente vom Datentyp ARRAY beginnt immer an einer WORD-Grenze.

Die folgende Tabelle zeigt die Eigenschaften des Datentyps STRUCT:

| Länge   | Format | Wertebereich  | Beispiel für Werteingabe   |
|---|--------|---|--|
| Eine STRUCT-Variable beginnt an einem Byte mit gerader Adresse und belegt den Speicher bis zur nächsten Wortgrenze. | STRUCT | Es gelten die Wertebereiche der verwendeten Datentypen. | Es gelten die Regeln zur Werteingabe für die verwendeten Datentypen. |

### Beispiel

Das folgende Bild zeigt ein Beispiel für die Struktur einer STRUCT-Variablen:



### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Zeiger

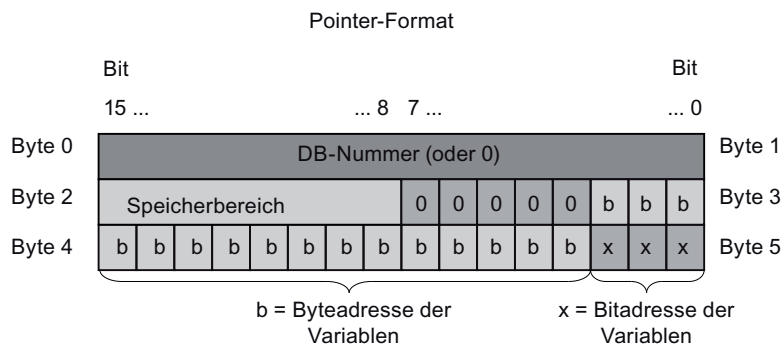
### POINTER

#### Beschreibung

Ein Parameter vom Typ POINTER ist ein Zeiger, der auf eine bestimmte Variable zeigen kann. Er belegt 6 Byte (48 Bit) im Speicher und kann die folgenden Informationen zu einer Variablen enthalten:

- DB-Nummer bzw. 0, wenn die Daten nicht in einem DB gespeichert werden
- Speicherbereich in der CPU
- Variablenadresse

Das folgende Bild zeigt die Struktur des Parametertyps POINTER:



#### Arten von Zeigern

Abhängig von der Angabe können Sie mit dem Parametertyp POINTER die folgenden vier Arten von Zeigern deklarieren:

- Bereichsinterner Zeiger:  
Ein bereichsinterner Zeiger enthält Angaben zur Adresse einer Variablen.
- Bereichsübergreifender Zeiger:  
Ein bereichsübergreifender Zeiger enthält Angaben zu dem Speicherbereich und der Adresse eines Operanden.
- DB-Zeiger:  
Mit einem DB-Zeiger können Sie auf eine Variable eines Datenbausteins zeigen. Ein DB-Zeiger enthält zusätzlich zu dem Speicherbereich und der Adresse einer Variablen auch eine Datenbausteinnummer.
- Null Zeiger:  
Mit einem Null Zeiger zeigen Sie das Fehlen eines Wertes an. Das Fehlen eines Wertes kann entweder bedeuten, dass kein Wert existiert oder dass der Wert noch nicht bekannt ist. Ein Nullwert steht für die Abwesenheit eines Wertes, ist aber gleichzeitig ein Wert.

Die folgende Tabelle zeigt die Formate zur Deklaration der verschiedenen Zeiger-Arten:

| P#ByteDarstellung | Format                       | Beispiel für Werteingabe | Beschreibung                  |
|-------------------|------------------------------|--------------------------|-------------------------------|
| Symbolisch        | P#Byte.Bit                   | "MyTag"                  | Bereichsinterner Zeiger       |
|                   | P#OperandenbereichByte.Bit   | "MyVariable"             | Bereichsübergreifender Zeiger |
|                   | P#Datenbaustein.Datenoperand | "MyDB"."MyTag"           | DB-Zeiger                     |
|                   | P#Nullwert                   | -                        | Null Zeiger                   |
| Absolut           | P#Byte.Bit                   | P#20.0                   | Bereichsinterner Zeiger       |
|                   | P#OperandenbereichByte.Bit   | P#M20.0                  | Bereichsübergreifender Zeiger |
|                   | P#Datenbaustein.Datenoperand | P#DB10.DBX20.0           | DB-Zeiger                     |
|                   | P#Nullwert                   | P#0.0, NULL              | Null Zeiger                   |

Sie können bei Versorgung eines Parameters vom Typ POINTER den Aktualwert ohne Präfix (P#) eingeben. Ihr Eintrag wird dann automatisch in das POINTER-Format umgewandelt.

#### Hinweis

Wenn Sie das Präfix P# verwenden, können Sie nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen.

## Speicherbereiche

Die folgende Tabelle zeigt die Hexadezimalcodes der Speicherbereiche für den Parametertyp POINTER:

| Hexadezimalcode       | Speicherbereich | Beschreibung                             |
|-----------------------|-----------------|--|
| B#16#80 <sup>1)</sup> | P               | Peripherie auf einer CPU S7-300/400      |
| 16#1                  | P               | Peripherieeingänge auf einer CPU S7-1500 |
| 16#2                  | P               | Peripherieausgänge auf einer CPU S7-1500 |
| B#16#81               | E               | Speicherbereich der Eingänge             |
| B#16#82               | A               | Speicherbereich der Ausgänge             |
| B#16#83               | M               | Speicherbereich der Merker               |
| B#16#84               | DBX             | Datenbaustein                            |
| B#16#85               | DIX             | Instanz-Datenbaustein                    |
| B#16#86               | L               | Lokaldaten                               |

| Hexadezimalcode   | Speicherbereich | Beschreibung         |
|---|-----------------|----------------------|
| B#16#87   | V               | Vorherige Lokaldaten |
| <sup>1)</sup> Diese Datentypen können für den POINTER-Zeiger nur auf einer CPU S7-300/400 verwendet werden. |                 |                      |

### Siehe auch

Grundlagen zur indirekten Adressierung (Seite 1077)

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

### ANY

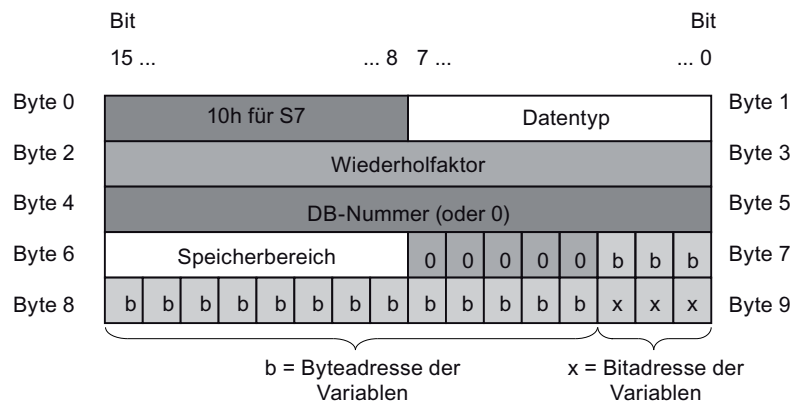
#### Beschreibung

Ein Parameter vom Typ ANY ist ein Zeiger, der auf den Anfang eines Datenbereichs zeigt und dessen Länge nennt. Ein ANY-Zeiger belegt 10 Byte im Speicher und kann die folgenden Informationen enthalten:

- Datentyp:  
Datentyp der Elemente des Datenbereichs
- Wiederholfaktor:  
Anzahl der Elemente des Datenbereichs
- DB-Nummer:  
Datenbaustein, in dem die Elemente des Datenbereichs deklariert sind.
- Speicherbereich:  
Speicherbereich der CPU, in dem die Elemente des Datenbereichs abgelegt sind.
- Anfangsadresse der Daten in Format "Byte.Bit":  
Anfang des Datenbereichs, auf den der ANY-Zeiger zeigt.
- Null Zeiger:  
Mit einem Null Zeiger zeigen Sie das Fehlen eines Wertes an. Das Fehlen eines Wertes kann entweder bedeuten, dass kein Wert existiert oder dass der Wert noch nicht bekannt ist. Ein Nullwert steht für die Abwesenheit eines Wertes, ist aber gleichzeitig ein Wert.

Das folgende Bild zeigt die Struktur des ANY-Zeigers:





Ein ANY-Zeiger kann keine Strukturen erkennen. Er kann nur lokalen Variablen zugewiesen werden.

Die folgende Tabelle zeigt die Formate zur Deklaration eines ANY-Zeigers:

| Darstellung | Format  | Beispiel für Werteingabe                  | Beschreibung   |
|-------------|---|---|--|
| Symbolisch  | P#Datenbaustein.Speicherbereich<br>Datenadresse Typ<br>Anzahl | "MyDB".StructVariable.<br>ErsteKomponente | Bereich mit 10 Wörtern<br>im globalen DB11 ab<br>DBB20.0 |
|             | P#Speicherbereich<br>Datenadresse Typ<br>Anzahl               | "MyMarkerTag"                             | Bereich mit 4 Bytes ab<br>MB20.0                         |
|             | P#Nullwert  | -   | Eingang E1.0   |
|             | P#Nullwert  | -   | Nullwert   |
| Absolut     | P#Datenbaustein.Speicherbereich<br>Datenadresse Typ<br>Anzahl | P#DB11.DBX20.0 INT<br>10                  | Bereich mit 10 Wörtern<br>im globalen DB11 ab<br>DBB20.0 |
|             | P#Speicherbereich<br>Datenadresse Typ<br>Anzahl               | P#M20.0 BYTE 10                           | Bereich mit 10 Bytes ab<br>MB20.0                        |
|             |   | P#E1.0 BOOL 1                             | Eingang E1.0   |
|             | P#Nullwert  | P#0.0 VOID 0, NULL                        | Nullwert   |

**Hinweis**

Mit dem Zeiger ANY können Sie nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen.

### Codierung der Datentypen

Die folgende Tabelle zeigt die Codierung der Datentypen für den ANY-Zeiger:

| Hexadezimalcode       | Datentyp  | Beschreibung            |
|-----------------------|-----------|-------------------------|
| B#16#00               | NIL       | Nullpointer             |
| B#16#01 <sup>1)</sup> | BOOL      | Bits                    |
| B#16#02               | BYTE      | Bytes, 8 Bits           |
| B#16#03               | CHAR      | 8-Bit-Zeichen           |
| B#16#04               | WORD      | 16-Bit-Wörter           |
| B#16#05               | INT       | 16-Bit-Ganzzahlen       |
| B#16#06               | DWORD     | 32-Bit-Wörter           |
| B#16#07               | DINT      | 32-Bit-Ganzzahlen       |
| B#16#08               | REAL      | 32-Bit-Gleitpunktzahlen |
| B#16#0B               | TIME      | Zeitdauer               |
| B#16#0C               | S5TIME    | Zeitdauer               |
| B#16#09               | DATE      | Datum                   |
| B#16#0A               | TOD       | Datum und Uhrzeit       |
| B#16#0E               | DT        | Datum und Uhrzeit       |
| B#16#13               | STRING    | Zeichenkette            |
| B#16#17 <sup>1)</sup> | BLOCK_FB  | Funktionsbaustein       |
| B#16#18 <sup>1)</sup> | BLOCK_FC  | Funktion                |
| B#16#19 <sup>1)</sup> | BLOCK_DB  | Datenbaustein           |
| B#16#1A <sup>1)</sup> | BLOCK_SDB | Systemdatenbaustein     |
| B#16#1C <sup>1)</sup> | COUNTER   | Zähler                  |
| B#16#1D <sup>1)</sup> | TIMER     | Zeit                    |

<sup>1)</sup> Diese Datentypen können für den ANY-Zeiger nur auf einer CPU S7-300/400 verwendet werden.

### Codierung der Speicherbereiche

Die folgende Tabelle zeigt die Codierung der Speicherbereiche für den ANY-Zeiger:

| Hexadezimalcode       | Bereich | Beschreibung                 |
|-----------------------|---------|------------------------------|
| B#16#80 <sup>1)</sup> | P       | Peripherie                   |
| B#16#81               | E       | Speicherbereich der Eingänge |
| B#16#82               | A       | Speicherbereich der Ausgänge |
| B#16#83               | M       | Speicherbereich der Merker   |
| B#16#84               | DBX     | Datenbaustein                |
| B#16#85               | DIX     | Instanz-Datenbaustein        |
| B#16#86               | L       | Lokaldaten                   |
| B#16#87               | V       | Vorherige Lokaldaten         |

<sup>1)</sup> Diese Speicherbereiche können für den ANY-Zeiger nur auf einer CPU S7-300/400 verwendet werden.

**Siehe auch**

Grundlagen zur indirekten Adressierung (Seite 1077)

Übersicht über die gültigen Datentypen (Seite 1087)

Konstanten (Seite 1058)

**VARIANT****Beschreibung**

Ein Parameter vom Typ VARIANT ist ein Zeiger, der auf Variablen verschiedener Datentypen, außer einer Instanz, zeigen kann. Der VARIANT-Zeiger kann ein Objekt eines elementaren Datentyps, wie z. B. INT oder REAL, sein. Es kann auch ein STRING, DTL, Array of STRUCT, UDT oder ein Array of UDT sein. Der VARIANT-Zeiger kann Strukturen erkennen und auf einzelne Strukturkomponenten zeigen. Ein Operand vom Datentyp VARIANT belegt keinen Platz im Instanz-Datenbaustein oder dem L-Stack. Innerhalb des Speicherbereichs der CPU wird jedoch Speicherplatz belegt.

Aber eine Variable vom Typ VARIANT ist kein Objekt, sondern ein Verweis auf ein anderes Objekt. Einzelne Elemente vom Typ VARIANT können nur innerhalb der Bausteinschnittstelle einer Funktion deklariert werden. Daher können sie z. B. nicht in einem Datenbaustein oder im Abschnitt Static der Bausteinschnittstelle eines Funktionsbausteins deklariert werden, da ihre Größe nicht bekannt ist. Die Größe der referenzierten Objekte kann sich ändern.

**Hinweis**

Auf einen ganzen Datenbaustein kann nur dann gezeigt werden, wenn dieser ursprünglich von einem anwenderdefinierten Datentyp (UDT) abgeleitet wurde.

Die folgende Tabelle zeigt die Eigenschaften des VARIANT-Zeigers:

| Länge (Byte) | Darstellung | Format   | Beispiel für Werteingabe                          |
|--------------|-------------|--|---|
| 0            | Symbolisch  | Operand  | "TagResult"                                       |
|              |             | NameDatenbaustein.NameOperand.Komponente   | "Daten_TIA_Portal".StructVariable.ErsteKomponente |
|              | Absolut     | Operand  | %MW10   |
|              |             | Datenbausteinnummer.Operand Typ Länge (gilt nur für Bausteine mit Standardzugriff) | P#DB10.DBX10.0 INT 12                             |
|              |             | P#Nullwert   | P#0.0 VOID 0, NULL                                |

**Hinweis**

Wenn Sie das Präfix P# verwenden, können Sie nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen.

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise von VARIANT anhand der AWL Anweisung "MOVE: Wert kopieren":

| AWL  | Erläuterung  |
|--|--|
| CALL MOVE  | // Die Anweisung "Wert kopieren" wird aufgerufen.  |
| VARIANT  | // Datentyp der Anweisung  |
| IN :=<br>"Daten_TIA_Portal".StructVariable.ErsteKomponente | // Die Inhalte des Operanden "ErsteKomponente" aus dem DB "Daten_TIA_Portal" werden kopiert. |
| OUT := "MotorDB".StructResult.TagResult                    | // Und in den Operanden "TagResult" aus dem DB "MotorDB" übertragen.                         |

**Siehe auch**

- Grundlagen zur indirekten Adressierung (Seite 1077)
- Übersicht über die gültigen Datentypen (Seite 1087)
- Konstanten (Seite 1058)
- Grundlagen zum Bausteinzugriff (Seite 1034)

**Parametertypen**

**Parametertypen**

**Beschreibung**

Die Parametertypen sind Datentypen für Formalparameter, die an aufgerufene Bausteine übergeben werden. Ein Parametertyp kann auch ein PLC-Datentyp sein.

Die folgende Tabelle zeigt verfügbare Parametertypen und deren Bedeutung:

| Parametertyp | Länge (Bits) | Beschreibung  |
|--------------|--------------|---|
| TIMER        | 16           | Dient zur Angabe einer Zeit, die im aufgerufenen Codebaustein verwendet wird. Wenn Sie einen Formalparameter vom Parametertyp TIMER versorgen, muss der zugehörige Aktualparameter eine Zeit sein.<br>Beispiel: T1        |
| COUNTER      | 16           | Dient zur Angabe eines Zählers, der im aufgerufenen Codebaustein verwendet wird. Wenn Sie einen Formalparameter vom Parametertyp COUNTER versorgen, muss der zugehörige Aktualparameter ein Zähler sein.<br>Beispiel: Z10 |
| BLOCK_FC     | 16           | Dient zur Angabe eines Bausteins, der im aufgerufenen Codebaustein als Eingang verwendet wird.<br>Die Deklaration des Parameters bestimmt die Bausteinart (z. B. FB, FC, DB), die verwendet werden soll.                  |
| BLOCK_FB     | 16           |   |
| BLOCK_DB     | 16           |   |
| BLOCK_SDB    | 16           |   |

| Parametertyp | Länge (Bits) | Beschreibung   |
|--------------|--------------|--|
| BLOCK_SFB    | 16           | Wenn Sie einen Formalparameter vom Parametertyp BLOCK versorgen, geben Sie eine Bausteinadresse als Aktualparameter an.<br>Beispiel: DB3   |
| BLOCK_SFC    | 16           |  |
| BLOCK_OB     | 16           |  |
| BLOCK_SDT    | -            |  |
| BLOCK_UDT    | -            |  |
| VOID         | -            | Der Parametertyp VOID speichert keine Werte. Dieser Parametertyp wird verwendet, wenn die Rückgabewerte eines Ausgangs nicht benötigt werden. Der Parametertyp VOID kann z. B. am Ausgang STATUS angegeben werden, wenn die Fehlerinformation nicht benötigt wird. |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zu PLC-Datentypen (Seite 1436)

**PLC-Datentypen****PLC-Datentypen****Beschreibung**

PLC-Datentypen sind von Ihnen definierte Datenstrukturen, die Sie im Programm mehrmals verwenden können. Die Struktur eines PLC-Datentyps setzt sich aus mehreren Komponenten zusammen, die verschiedene Datentypen aufweisen können. Die Art der Komponenten legen Sie bei der Deklaration des PLC-Datentyps fest.

Es gibt folgende Verwendungsmöglichkeiten für PLC-Datentypen:

- PLC-Datentypen können als Datentypen für Variablen in der Variablendeklaration von Codebausteinen oder in Datenbausteinen verwendet werden.
- PLC-Datentypen können als Vorlage für die Erstellung von globalen Datenbausteinen mit gleicher Datenstruktur dienen.

**Siehe auch**

Strukturierte Variablen adressieren (Seite 1070)

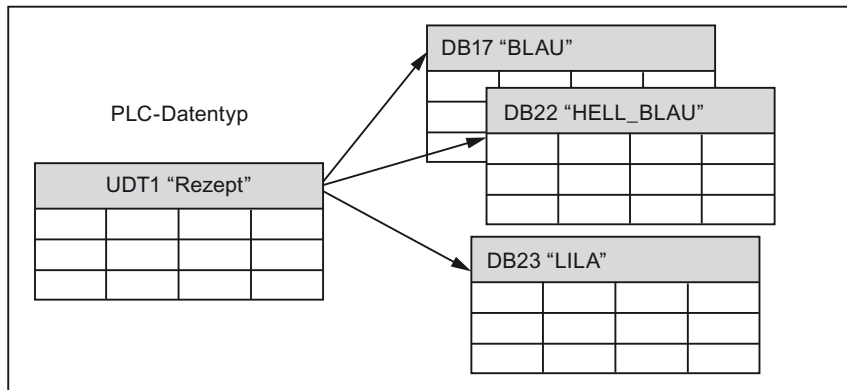
**Beispiel für einen PLC-Datentyp****Beispiel**

Sie können PLC-Datentypen bei der Erstellung von Datenbausteinen als Typ angeben. Basierend auf diesem Typ können Sie eine Vielzahl von Datenbausteinen erzeugen, die die

gleiche Datenstruktur haben. Diese Datenbausteine können durch die Eingabe von unterschiedlichen Aktualwerten für die entsprechende Aufgabe angepasst werden.

Erzeugen Sie z. B. einen PLC-Datentyp für ein Rezept zum Mischen von Farben. Sie können dann diesen Datentyp mehreren Datenbausteinen zuordnen, die jeweils andere Mengenangaben beinhalten.

Das folgende Bild zeigt diesen Anwendungsfall:



## Systemdatentypen

## Systemdatentypen

### Beschreibung

Die Systemdatentypen (SDT) werden vom System zur Verfügung gestellt und haben eine vordefinierte Struktur. Die Struktur eines Systemdatentyps besteht aus einer festen Anzahl von Komponenten, die unterschiedliche Datentypen aufweisen können. Eine Änderung der Struktur eines Systemdatentyps ist nicht möglich.

Die Systemdatentypen können nur für bestimmte Anweisungen verwendet werden. Die folgende Tabelle zeigt verfügbare Systemdatentypen und deren Bedeutung:

| Systemdatentyp | Länge (Byte) | Beschreibung  |
|----------------|--------------|---|
| IEC_TIMER      | 16           | Struktur eines Zeitgebers, dessen Zeitwerte vom Datentyp TIME sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "TP", "TOF", "TON", "TONR", "RT" und "PT" verwendet.  |
| IEC_LTIMER     | 32           | Struktur eines Zeitgebers, dessen Zeitwerte vom Datentyp LTIME sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "TP", "TOF", "TON", "TONR", "RT" und "PT" verwendet. |
| IEC_SCOUNTER   | 3            | Struktur eines Zählers, dessen Zählwerte vom Datentyp SINT sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.                       |

| Systemdatentyp | Länge (Byte) | Beschreibung   |
|----------------|--------------|--|
| IEC_USCOUNTER  | 3            | Struktur eines Zählers, dessen Zählwerte vom Datentyp USINT sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.                                       |
| IEC_COUNTER    | 6            | Struktur eines Zählers, dessen Zählwerte vom Datentyp INT sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.   |
| IEC_UCOUNTER   | 6            | Struktur eines Zählers, dessen Zählwerte vom Datentyp UINT sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.  |
| IEC_DCOUNT     | 12           | Struktur eines Zählers, dessen Zählwerte vom Datentyp DINT sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.  |
| IEC_UDCOUNT    | 12           | Struktur eines Zählers, dessen Zählwerte vom Datentyp UDINT sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.                                       |
| IEC_LCOUNT     | 24           | Struktur eines Zählers, dessen Zählwerte vom Datentyp LINT sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.  |
| IEC_ULCOUNT    | 24           | Struktur eines Zählers, dessen Zählwerte vom Datentyp ULINT sind.<br>Dieser Datentyp wird z. B. für die Anweisungen "CTU", "CTD" und "CTUD" verwendet.                                       |
| ERROR_STRUCT   | 28           | Struktur einer Fehlerinformation zu einem Programmier- oder Peripheriezugriffsfehler.<br>Dieser Datentyp wird z. B. für die Anweisung "GET_ERROR" verwendet.                                 |
| CREF           | 8            | Komponente des Datentyps ERROR_STRUCT, in der Informationen zur Adresse eines Bausteins gespeichert werden.  |
| NREF           | 8            | Komponente des Datentyps ERROR_STRUCT, in der Informationen zur Adresse eines Operanden gespeichert werden.  |
| VREF           | 12           | Dient zur Ablage eines VARIANT-Zeigers.<br>Dieser Datentyp wird z. B. für Anweisungen von S7-1200 Motion Control verwendet.  |
| STARTINFO      | 12           | Gibt die Datenstruktur an, in der Startinformationen gespeichert werden.<br>Dieser Datentyp wird z. B. für die Anweisung "RD_SINFO" verwendet.   |
| SSL_HEADER     | 4            | Gibt die Datenstruktur an, in der Informationen zu den Datensätzen beim Lesen von Systemzustandslisten gespeichert werden. Dieser Datentyp wird z. B. für die Anweisung "RDSYSST" verwendet. |

| Systemdatentyp | Länge (Byte) | Beschreibung  |
|----------------|--------------|---|
| CONDITIONS     | 52           | Definierte Datenstruktur, die Bedingungen für Anfang und Ende eines Datenempfangs definiert.<br>Dieser Datentyp wird z. B. für die Anweisung "RCV_CFG " verwendet.  |
| TADDR_Param    | 8            | Gibt die Struktur eines Datenbausteins vor, in dem Verbindungsbeschreibungen für die Open User Communication über UDP abgelegt werden.<br>Dieser Datentyp wird z. B. für die Anweisungen "TUSEND" und "TURSV" verwendet.                          |
| TCON_Param     | 64           | Gibt die Struktur eines Datenbausteins vor, in dem Verbindungsbeschreibungen für die Open User Communication über Industrial Ethernet (PROFINET) abgelegt werden.<br>Dieser Datentyp wird z. B. für die Anweisungen "TSEND" und "TRSV" verwendet. |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**Hardware-Datentypen**

**Hardware-Datentypen**

**Beschreibung**

Die Hardware-Datentypen werden von der CPU zur Verfügung gestellt. Die Anzahl der verfügbaren Hardware-Datentypen hängt dabei von der CPU ab.

Abhängig von den in der Hardware-Konfiguration parametrisierten Baugruppen, werden Konstanten von einem bestimmten Hardware-Datentyp angelegt. Wenn eine Anweisung zur Steuerung oder Aktivierung einer parametrisierten Baugruppe ins Anwenderprogramm eingefügt wird, können die verfügbaren Konstanten zur Versorgung der Parameter verwendet werden.

Die folgende Tabelle zeigt verfügbare Hardware-Datentypen und deren Bedeutung:

| Datentyp    | Basis-Datentyp | Beschreibung   |
|-------------|----------------|--|
| REMOTE      | ANY            | Dient zur Angabe der Adresse einer remoten CPU.<br>Dieser Datentyp wird z. B. für die Anweisungen "PUT" und "GET" verwendet. |
| GEOADDR     | HW_IOSYSTEM    | Geographische Adressangabe   |
| HW_ANY      | WORD           | Identifikation einer beliebigen Hardware-Komponente, z. B. einer Baugruppe.  |
| HW_DEVICE   | HW_ANY         | Identifikation eines DP-Slaves/PROFINET IO-Devices   |
| HW_DPMaster | HW_INTERFACE   | Identifikation eines DP-Masters  |



| Datentyp     | Basis-Datentyp | Beschreibung   |
|--------------|----------------|--|
| HW_DPSLAVE   | HW_DEVICE      | Identifikation eines DP-Slaves   |
| HW_IO        | HW_ANY         | Identifikationsnummer der CPU oder des Interfaces<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder des Interface in der Hardware-Konfiguration hinterlegt.        |
| HW_IOSYSTEM  | HW_ANY         | Identifikation eines PN/IO-Systems oder DP-Mastersystems   |
| HW_SUBMODULE | HW_IO          | Identifikation einer zentralen Hardware-Komponente   |
| HW_MODULE    | HW_IO          | Identifikation eines Moduls  |
| HW_INTERFACE | HW_SUBMODULE   | Identifikation einer Interface-Komponente  |
| HW_IEPORT    | HW_SUBMODULE   | Identifikation eines Ports (PN/IO)   |
| HW_HSC       | HW_SUBMODULE   | Identifikation eines schnellen Zählers<br>Dieser Datentyp wird z. B. für die Anweisung "CTRL_HSC" verwendet.   |
| HW_PWM       | HW_SUBMODULE   | Identifikation einer Pulsbreitenmodulation<br>Dieser Datentyp wird z. B. für die Anweisung "CTRL_PWM" verwendet.   |
| HW_PTO       | HW_SUBMODULE   | Identifikation eines Impulsgebers<br>Dieser Datentyp wird für Motion Control verwendet.  |
| AOM_AID      | DWORD          | Wird nur in Verbindung mit einem Systemfunktionsbaustein verwendet.  |
| AOM_IDENT    | DWORD          | Identifikation eines Objekts im Laufzeitsystem des AS  |
| EVENT_ANY    | AOM_IDENT      | Dient zur Angabe eines beliebigen Ereignisses  |
| EVENT_ATT    | EVENT_ANY      | Dient zur Angabe eines Ereignisses, das einem OB dynamisch zugewiesen werden kann<br>Dieser Datentyp wird z. B. für die Anweisungen "ATTACH" und "DETACH" verwendet.                               |
| EVENT_HWINT  | EVENT_ATT      | Dient zur Angabe eines Prozessalarm-Ereignisses  |
| OB_ANY       | INT            | Dient zur Angabe eines beliebigen Organisationsbausteins.  |
| OB_DELAY     | OB_ANY         | Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Verzögerungsalarms aufgerufen wird.<br>Dieser Datentyp wird z. B. für die Anweisungen "SRT_DINT" und "CAN_DINT" verwendet. |
| OB_TOD       | OB_ANY         | Gibt die Nummer eines Uhrzeitalarm-Organisationsbausteins an.<br>Dieser Datentyp wird z. B. für die Anweisungen "SET_TINT", "CAN_TINT", "ACT_TINT" und "QRY_TINT" verwendet.                       |
| OB_CYCLIC    | OB_ANY         | Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Weckalarms aufgerufen wird.  |
| OB_ATT       | OB_ANY         | Dient zur Angabe eines Organisationsbausteins, der einem Ereignis dynamisch zugewiesen werden kann.<br>Dieser Datentyp wird z. B. für die Anweisungen "ATTACH" und "DETACH" verwendet.             |

| Datentyp     | Basis-Datentyp | Beschreibung   |
|--------------|----------------|--|
| OB_PCYCLE    | OB_ANY         | Dient zur Angabe eines Organisationsbausteins, der einem Ereignis der Ereignisklasse "Zyklisches Programm" zugewiesen werden kann.       |
| OB_HWINT     | OB_ATT         | Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Prozessalarms aufgerufen wird.                                   |
| OB_DIAG      | OB_ANY         | Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Diagnosealarms aufgerufen wird.                                  |
| OB_TIMEERROR | OB_ANY         | Dient zur Angabe eines Organisationsbausteins, der beim Auftreten von Zeitfehlern aufgerufen wird.                                       |
| OB_STARTUP   | OB_ANY         | Dient zur Angabe eines Organisationsbausteins, der beim Auftreten eines Anlauf-Ereignisses aufgerufen wird.                              |
| PORT         | HW_SUBMODULE   | Dient zur Angabe eines Kommunikationsports. Dieser Datentyp wird für die Punkt-zu-Punkt-Kommunikation verwendet.                         |
| RTM          | UINT           | Dient zur Angabe der Nummer eines Betriebsstundenzählers. Dieser Datentyp wird z. B. für die Anweisung "RTM" verwendet.                  |
| PIP          | UINT           | Dient zum Anlegen und Verschalten eines OB "Synchronous Cycle". Dieser Datentyp wird bei den SFCs 26, 27, 126 und 127 verwendet.         |
| CONN_ANY     | WORD           | Dient zur Angabe einer beliebigen Verbindung.  |
| CONN_PRG     | CONN_ANY       | Dient zur Angabe einer Verbindung für die Offene Kommunikation über UDP.   |
| CONN_OUC     | CONN_ANY       | Dient zur Angabe einer Verbindung für die offene Kommunikation über Industrial Ethernet (PROFINET).                                      |
| CONN_R_ID    | DWORD          | Datentyp für den R_ID Parameter an den S7 Kommunikationsbausteinen.  |
| DB_ANY       | UINT           | Identifikation (Nummer) eines beliebigen DBs<br>Der Datentyp "DB_ANY" hat im Abschnitt "Temp" die Länge 0.                               |
| DB_WWW       | DB_ANY         | Nummer des DBs, der die Web Applikation beschreibt (z. B. Anweisung "WWW")<br>Der Datentyp "DB_WWW" hat im Abschnitt "Temp" die Länge 0. |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

## Datentypkonvertierung

## Datentypkonvertierung

## Übersicht über die Datentypkonvertierung

### Einführung

Wenn Sie mehrere Operanden in einer Anweisung verknüpfen, müssen Sie die Verträglichkeit ihrer Datentypen beachten. Dies gilt auch bei Zuweisungen oder beim Versorgen von Bausteinparametern. Sind die Operanden ungleichen Datentyps, muss eine Konvertierung durchgeführt werden.

Es gibt zwei Möglichkeiten der Konvertierung:

- Implizite Konvertierung  
Die Konvertierung erfolgt automatisch beim Ausführen der Anweisung.
- Explizite Konvertierung  
Sie verwenden eine explizite Konvertierungsanweisung, bevor die eigentliche Anweisung ausgeführt wird.

---

#### Hinweis

Die beschriebenen Möglichkeiten der Datentypkonvertierungen beziehen sich immer auf den neuesten Stand der CPU (V. 4). Es kann sein, dass als möglich gekennzeichnete Konvertierungen in den CPU Versionen 1 - 3 nicht zur Verfügung stehen.

---

#### Hinweis

##### Konvertieren von Bitfolgen in SCL

Alle Bitfolgen (BYTE, WORD, DWORD und LWORD) werden in Ausdrücken wie die entsprechenden Ganzzahlen ohne Vorzeichen (USINT, UINT, UDINT und ULINT) behandelt. Daher wird z. B. die implizite Konvertierung von DWORD nach REAL wie eine Konvertierung von UDINT nach REAL durchgeführt.

---

## Implizite Konvertierung

Eine implizite Konvertierung wird automatisch durchgeführt, wenn die Datentypen der Operanden kompatibel sind. Die Prüfung der Kompatibilität kann nach mehr oder weniger strengen Kriterien erfolgen:

- Mit IEC-Prüfung (Voreinstellung)  
Bei eingestellter IEC-Prüfung werden die folgenden Regeln angewandt:
  - Die implizite Konvertierung von BOOL in andere Datentypen ist nicht möglich.
  - Nur die Datentypen REAL, BYTE, WORD, DINT, INT, SINT, UDINT, UINT, USINT, TIME, DT, STRING und CHAR können implizit konvertiert werden.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp WORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp BYTE erwartet wird.
- Ohne IEC-Prüfung  
Bei nicht eingestellter IEC-Prüfung werden die folgenden Regeln angewandt:
  - Die implizite Konvertierung von BOOL in andere Datentypen ist nicht möglich.
  - Nur die Datentypen REAL, LREAL, BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, TIME, DTL, TOD, DATE, STRING und CHAR können implizit konvertiert werden.
  - Die Bitlänge des Quelldatentyps darf die Bitlänge des Zieldatentyps nicht überschreiten. Ein Operand vom Datentyp DWORD kann z. B. an einem Parameter nicht angegeben werden, wenn an diesem der Datentyp WORD erwartet wird.
  - An Durchgangsparametern (InOut) muss die Bitlänge eines angegebenen Operanden mit der programmierten Bitlänge des jeweiligen Parameters übereinstimmen.

---

### Hinweis

#### Implizite Konvertierung ohne IEC-Prüfung

Der Programmiereditor kennzeichnet Operanden, die implizit konvertiert werden, mit einem grauen Rechteck. Ein dunkelgraues Rechteck signalisiert, dass eine implizite Konvertierung ohne Genauigkeitsverlust möglich ist, z. B., wenn Sie den Datentyp SINT nach INT konvertieren. Ein hellgraues Rechteck signalisiert, dass eine implizite Konvertierung möglich ist, aber während der Laufzeit können Fehler auftreten. Wenn Sie z. B. den Datentyp DINT nach INT konvertieren und es zu einem Überlauf kommt, wird der Freigabeausgang ENO auf "0" gesetzt.

---

Weitere Informationen zu der Einstellung der IEC-Prüfung und der impliziten Konvertierung finden Sie unter "Siehe auch".

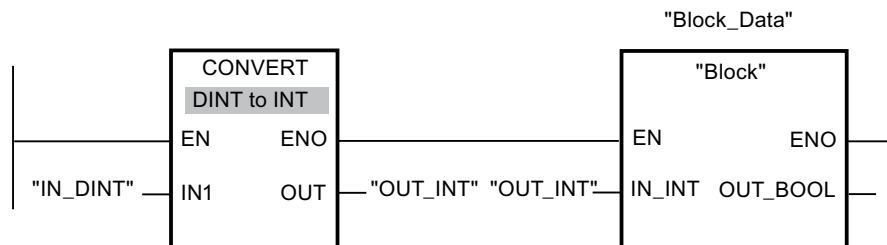
## Explizite Konvertierung

Wenn die Operanden nicht kompatibel sind und deshalb eine implizite Konvertierung nicht möglich ist, können Sie eine explizite Konvertierungsanweisung nutzen. Die Konvertierungsanweisungen finden Sie in der Task Card "Anweisungen".

Ein möglicher Überlauf wird am Freigabeausgang ENO angezeigt. Ein Überlauf entsteht, wenn z. B. der Wert des Quelldatentyps größer als der Wert des Zieldatentyps ist.

Weitere Informationen zur expliziten Konvertierung finden Sie unter "Siehe auch".

Das folgende Bild zeigt ein Beispiel, in dem eine explizite Datentypkonvertierung durchgeführt werden muss:



Der Funktionsbaustein "Block" erwartet am Eingangsparameter "IN\_INT" eine Variable vom Datentyp INT. Der Wert der Variablen "IN\_DINT" muss daher zunächst von DINT nach INT konvertiert werden. Befindet sich der Wert der Variablen "IN\_DINT" innerhalb des zulässigen Wertebereichs des Datentyps INT, dann findet eine Konvertierung statt. Andernfalls wird ein Überlauf gemeldet. Doch auch im Falle eines Überlaufs findet trotzdem eine Konvertierung statt, aber die Werte werden abgeschnitten und der Freigabeausgang ENO auf "0" gesetzt.

## Siehe auch

IEC-Prüfung einstellen oder aufheben (Seite 1135)

## Implizite Konvertierung

### IEC-Prüfung einstellen oder aufheben

Es wird geprüft, ob die Datentypen der verwendeten Operanden kompatibel sind. Diese Kompatibilitätsprüfung kann nach mehr oder weniger strengen Kriterien erfolgen. Bei aktivierter "IEC-Prüfung" werden strengere Kriterien angewandt.

Sie können die IEC-Prüfung zentral für alle neuen Bausteine des Projekts oder für einzelne Bausteine einstellen.

### IEC-Prüfung für neue Bausteine einstellen

Um die IEC-Prüfung für alle neuen Bausteine im Projekt einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > Allgemein".
3. Aktivieren oder deaktivieren Sie in der Gruppe "Voreinstellung für neue Bausteine" das Optionskästchen "IEC-Prüfung".  
Die IEC-Prüfung wird für alle neuen Bausteine im Programm aktiviert oder deaktiviert.

### IEC-Prüfung für einen Baustein einstellen

Um die IEC-Prüfung für einen Baustein einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Wählen Sie in der Bereichsnavigation die Gruppe "Attribute".
4. Aktivieren oder deaktivieren Sie das Optionskästchen "IEC-Prüfung".  
Die IEC-Prüfung wird für diesen Baustein aktiviert oder deaktiviert. Die Einstellung wird zusammen mit dem Projekt gespeichert.

### Binärzahlen

### Implizite Konvertierung von BOOL

#### Möglichkeiten zur impliziten Konvertierung

Die implizite Konvertierung vom Datentyp BOOL ist nicht möglich.

#### Siehe auch

BOOL (Bit) (Seite 1091)

### Bitfolgen

### Implizite Konvertierung von BYTE

#### Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp BYTE:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--------|-------|-----------------|------------------|---|
| BYTE   | BOOL  | -               | -                | Keine implizite Konvertierung<br><br>Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
|        | WORD  | x               | x                |   |
|        | DWORD | x               | x                |   |
|        | SINT  | -               | x                |   |
|        | USINT | -               | x                |   |
|        | INT   | -               | x                |   |
|        | UINT  | -               | x                |   |
|        | DINT  | -               | x                |   |
|        | UDINT | -               | x                |   |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--|--------|-----------------|------------------|---|
|  | REAL   | -               | -                | Keine implizite Konvertierung   |
|  | LREAL  | -               | -                |   |
|  | TIME   | -               | -                |   |
|  | DTL    | -               | -                |   |
|  | TOD    | -               | -                |   |
|  | DATE   | -               | -                |   |
|  | STRING | -               | -                |   |
|  | CHAR   | -               | x                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |   |

**Siehe auch**

- BYTE (Byte) (Seite 1091)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von BYTE (Seite 1153)

**Implizite Konvertierung von WORD**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp WORD:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--------|-------|-----------------|------------------|---|
| WORD   | BOOL  | -               | -                | Keine implizite Konvertierung   |
|        | BYTE  | -               | X                | Das niederwertige Byte wird in den Zieldatentyp übertragen, das höherwertige Byte wird ignoriert. |
|        | DWORD | X               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.      |
|        | SINT  | -               | X                | Das niederwertige Byte wird in den Zieldatentyp übertragen, das höherwertige Byte wird ignoriert. |
|        | USINT | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.      |
|        | INT   | -               | X                |   |
|        | UINT  | -               | X                |   |
|        | DINT  | -               | X                |   |
| UDINT  | -     | X               |                  |   |

| Quelle | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--------|--------|-----------------|------------------|---|
|        | REAL   | -               | -                | Keine implizite Konvertierung   |
|        | LREAL  | -               | -                |   |
|        | TIME   | -               | -                |   |
|        | DTL    | -               | -                |   |
|        | TOD    | -               | -                |   |
|        | DATE   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. |
|        | STRING | -               | -                | Keine implizite Konvertierung   |
|        | CHAR   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. |

x: Konvertierung möglich  
 -: Konvertierung nicht möglich

**Siehe auch**

WORD (Seite 1092)

IEC-Prüfung einstellen oder aufheben (Seite 1135)

Übersicht über die Datentypkonvertierung (Seite 1133)

Explizite Konvertierung von WORD (Seite 1155)

**Implizite Konvertierung von DWORD**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DWORD:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--------|-------|-----------------|------------------|---|
| DWORD  | BOOL  | -               | -                | Keine implizite Konvertierung   |
|        | BYTE  | -               | X                |   |
|        | WORD  | -               | X                |   |
|        | SINT  | -               | X                |   |
|        | USINT | -               | X                |   |
|        | INT   | -               | X                |   |
|        | UINT  | -               | X                |   |
|        | DINT  | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.                          |
|        | UDINT | -               | X                |   |
|        | REAL  | -               | X                | Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.) |



| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--|--------|-----------------|------------------|--|
|  | LREAL  | -               | -                | Keine implizite Konvertierung  |
|  | TIME   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
|  | DTL    | -               | -                | Keine implizite Konvertierung  |
|  | TOD    | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
|  | DATE   | -               | -                | Keine implizite Konvertierung  |
|  | STRING | -               | -                |  |
|  | CHAR   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |  |

### Siehe auch

- DWORD (Seite 1093)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von DWORD (Seite 1157)

### Ganzzahlen

### Implizite Konvertierung von SINT

### Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp SINT:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--------|-------|-----------------|------------------|--|
| SINT   | BOOL  | -               | -                | Keine implizite Konvertierung  |
|        | BYTE  | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden mit "0" aufgefüllt.                           |
|        | WORD  | -               | X                |  |
|        | DWORD | -               | X                |  |
|        | USINT | -               | X                | Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertübertragung von z. B. SINT #-1 -> INT #-1, es wird nicht mit "0" aufgefüllt.) |
|        | INT   | X               | X                |  |
|        | UINT  | -               | X                |  |
|        | DINT  | X               | X                |  |
| UDINT  | -     | X               |                  |  |

| Quelle | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--------|--------|-----------------|------------------|---|
|        | REAL   | X               | X                | Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.) |
|        | LREAL  | X               | X                |   |
|        | TIME   | -               | -                | Keine implizite Konvertierung   |
|        | DTL    | -               | -                |   |
|        | TOD    | -               | -                |   |
|        | DATE   | -               | -                |   |
|        | STRING | -               | -                |   |
|        | CHAR   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.                          |

x: Konvertierung möglich  
 -: Konvertierung nicht möglich

**Siehe auch**

- SINT (8-Bit-Ganzzahlen) (Seite 1095)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von SINT (Seite 1161)

**Implizite Konvertierung von USINT**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp USINT:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--------|-------|-----------------|------------------|--|
| USINT  | BOOL  | -               | -                | Keine implizite Konvertierung  |
|        | BYTE  | -               | X                |  |
|        | WORD  | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden mit "0" aufgefüllt.                           |
|        | DWORD | -               | X                |  |
|        | SINT  | -               | X                |  |
|        | INT   | X               | X                | Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. USINT #10 -> DINT #10, oder USINT #128 -> SINT #-128) |
|        | UINT  | X               | X                |  |
|        | DINT  | X               | X                |  |
|        | UDINT | X               | X                |  |
|        | REAL  | X               | X                |  |
|        | LREAL | X               | X                | Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. in den Wert "1.0" umgewandelt.)  |
|        | TIME  | -               | -                | Keine implizite Konvertierung  |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung |
|--|--------|-----------------|------------------|-------------|
|  | DTL    | -               | -                |             |
|  | TOD    | -               | -                |             |
|  | DATE   | -               | -                |             |
|  | STRING | -               | -                |             |
|  | CHAR   | -               | X                |             |
| Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |        |                 |                  |             |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich                                   |        |                 |                  |             |

### Siehe auch

- USINT (8-Bit-Ganzzahlen) (Seite 1095)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von USINT (Seite 1163)

### Implizite Konvertierung von INT

#### Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp INT:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--------|-------|-----------------|------------------|--|
| INT    | BOOL  | -               | -                | Keine implizite Konvertierung  |
|        | BYTE  | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.   |
|        | WORD  | -               | X                |  |
|        | DWORD | -               | X                |  |
|        | SINT  | -               | X                | Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. INT #-1 -> SINT #-1, oder INT #-32 767 -> UINT #32 769) |
|        | USINT | -               | X                |  |
|        | UINT  | -               | X                |  |
|        | DINT  | X               | X                |  |
|        | UDINT | -               | X                |  |
|        | REAL  | X               | X                | Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)  |
|        | LREAL | X               | X                |  |
|        | TIME  | -               | -                | Keine implizite Konvertierung  |
|        | DTL   | -               | -                |  |
|        | TOD   | -               | -                |  |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--|--------|-----------------|------------------|--|
|  | DATE   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
|  | STRING | -               | -                | Keine implizite Konvertierung  |
|  | CHAR   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |  |

**Siehe auch**

- INT (16-Bit-Ganzzahlen) (Seite 1096)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von INT (Seite 1164)

**Implizite Konvertierung von UINT**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp UINT:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--------|-------|-----------------|------------------|---|
| UINT   | BOOL  | -               | -                | Keine implizite Konvertierung   |
|        | BYTE  | -               | X                |   |
|        | WORD  | -               | X                |   |
|        | DWORD | -               | X                |   |
|        | SINT  | -               | X                | Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. UINT #100 -> DINT #100, oder UINT #60 000 -> INT #-5536) |
|        | USINT | -               | X                |   |
|        | INT   | -               | X                |   |
|        | DINT  | X               | X                |   |
|        | UDINT | X               | X                |   |
|        | REAL  | X               | X                | Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. in den Wert "1.0" umgewandelt.)   |
|        | LREAL | X               | X                |   |
|        | TIME  | -               | -                | Keine implizite Konvertierung   |
|        | DTL   | -               | -                |   |
|        | TOD   | -               | -                |   |
|        | DATE  | -               | -                | X   |
| STRING | -     | -               | -                | Keine implizite Konvertierung   |

| Quelle   | Ziel | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--|------|-----------------|------------------|--|
|  | CHAR | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |      |                 |                  |  |

### Siehe auch

- UINT (16-Bit-Ganzzahlen) (Seite 1097)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von UINT (Seite 1166)

### Implizite Konvertierung von DINT

#### Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DINT:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung              | Erläuterung   |
|--------|-------|-----------------|-------------------------------|---|
| DINT   | BOOL  | -               | -                             | Keine implizite Konvertierung   |
|        | BYTE  | -               | X                             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  |
|        | WORD  | -               | X                             |   |
|        | DWORD | -               | X                             |   |
|        | SINT  | -               | X                             | Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> SINT #-1, oder DINT #-1 -> USINT #255)   |
|        | USINT | -               | X                             |   |
|        | INT   | -               | X                             |   |
|        | UINT  | -               | X                             |   |
|        | UDINT | -               | X                             |   |
|        | REAL  | -               | X                             | Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 8 388 608) |
|        | LREAL | X               | X                             | Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "-1" wird z. B. in den Wert "-1.0" umgewandelt.)   |
|        | TIME  | -               | X                             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  |
|        | DTL   | -               | -                             | Keine implizite Konvertierung   |
|        | TOD   | -               | X                             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  |
| DATE   | -     | -               | Keine implizite Konvertierung |   |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--|--------|-----------------|------------------|--|
|  | STRING | -               | -                |  |
|  | CHAR   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |  |

**Siehe auch**

- DINT (32-Bit-Ganzzahlen) (Seite 1098)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von DINT (Seite 1168)

**Implizite Konvertierung von UDINT**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp UDINT:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--------|-------|-----------------|------------------|---|
| UDINT  | BOOL  | -               | -                | Keine implizite Konvertierung   |
|        | BYTE  | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  |
|        | WORD  | -               | X                |   |
|        | DWORD | -               | X                |   |
|        | SINT  | -               | X                | Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> SINT #-1, oder DINT #-1 -> USINT #255)   |
|        | USINT | -               | X                |   |
|        | INT   | -               | X                |   |
|        | UINT  | -               | X                |   |
|        | DINT  | -               | X                |   |
|        | REAL  | -               | X                | Das Bitmuster des Quellwerts wird konvertiert und in den Zieldatentyp übertragen. (Wertkonvertierung von z. B. DINT #-1 -> REAL #-1.0, aber es entsteht ein Genauigkeitsverlust bei Zahlen, deren Absolutwert größer ist als 8 388 608) |
|        | LREAL | X               | X                | Der Wert wird in das Format des Zieldatentyps konvertiert. (Der Wert "1" wird z. B. in den Wert "1.0" umgewandelt.)   |
|        | TIME  | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  |
|        | DTL   | -               | -                | Keine implizite Konvertierung   |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--|--------|-----------------|------------------|--|
|  | TOD    | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
|  | DATE   | -               | -                | Keine implizite Konvertierung  |
|  | STRING | -               | -                |  |
|  | CHAR   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |  |

### Siehe auch

- UDINT (32-Bit-Ganzzahlen) (Seite 1099)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von UDINT (Seite 1170)

### Gleitpunktzahlen

### Implizite Konvertierung von REAL

### Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp REAL:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung              | Erläuterung  |
|--------|-------|-----------------|-------------------------------|--|
| REAL   | BOOL  | -               | -                             | Keine implizite Konvertierung  |
|        | BYTE  | -               | -                             |  |
|        | WORD  | -               | -                             |  |
|        | DWORD | -               | X                             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.   |
|        | SINT  | -               | X                             | Das Bitmuster des Quellwerts wird gerundet und konvertiert und in den Zieldatentyp übertragen. (Rundung und Wertkonvertierung von z. B. REAL #2.5 -> INT #2, oder negative Zahlen REAL #-2.5 -> INT #-2 -> USINT #254. Bei einem Überlauf wird der Rest bestimmt REAL #305.5 -> INT #306 -> USINT #50) |
|        | USINT | -               | X                             |  |
|        | INT   | -               | X                             |  |
|        | UINT  | -               | X                             |  |
|        | DINT  | -               | X                             |  |
|        | UDINT | -               | X                             |  |
|        | LREAL | X               | X                             |  |
| TIME   | -     | -               | Keine implizite Konvertierung |  |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung |
|--|--------|-----------------|------------------|-------------|
|  | DTL    | -               | -                |             |
|  | TOD    | -               | -                |             |
|  | DATE   | -               | -                |             |
|  | STRING | -               | -                |             |
|  | CHAR   | -               | -                |             |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |             |

**Siehe auch**

- REAL (Seite 1102)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von REAL (Seite 1171)

**Implizite Konvertierung von LREAL**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp LREAL:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung              | Erläuterung  |
|--------|-------|-----------------|-------------------------------|--|
| LREAL  | BOOL  | -               | -                             | Keine implizite Konvertierung  |
|        | BYTE  | -               | -                             |  |
|        | WORD  | -               | -                             |  |
|        | DWORD | -               | -                             |  |
|        | SINT  | -               | X                             | Das Bitmuster des Quellwerts wird gerundet und konvertiert und in den Zieldatentyp übertragen. (Rundung und Wertkonvertierung von z. B. REAL #2.5 -> INT #2, oder negative Zahlen REAL #-2.5 -> INT #-2 -> USINT #254. Bei einem Überlauf wird der Rest bestimmt REAL #305.5 -> INT #306 -> USINT #50) |
|        | USINT | -               | X                             |  |
|        | INT   | -               | X                             |  |
|        | UINT  | -               | X                             |  |
|        | DINT  | -               | X                             |  |
|        | UDINT | -               | X                             |  |
| REAL   | -     | -               | Keine implizite Konvertierung |  |
| TIME   | -     | -               |                               |  |
| DTL    | -     | -               |                               |  |
| TOD    | -     | -               |                               |  |
| DATE   | -     | -               |                               |  |



| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung |
|--|--------|-----------------|------------------|-------------|
|  | STRING | -               | -                |             |
|  | CHAR   | -               | -                |             |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |             |

## Siehe auch

Explizite Konvertierung von LREAL (Seite 1173)

## Zeiten

### Implizite Konvertierung von TIME

#### Möglichkeiten zur impliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp TIME:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |  |
|--------|-------|-----------------|------------------|--|--|
| TIME   | BOOL  | -               | -                | Keine implizite Konvertierung  |  |
|        | BYTE  | -               | -                |  |  |
|        | WORD  | -               | -                |  |  |
|        | DWORD | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden. |  |
|        | SINT  | -               | -                | Keine implizite Konvertierung  |  |
|        | USINT | -               | -                |  |  |
|        | INT   | -               | -                |  |  |
|        | UINT  | -               | -                |  |  |
|        | DINT  | -               | X                |  |  |
|        | UDINT | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden. |  |
|        | REAL  | -               | -                | Keine implizite Konvertierung  |  |
|        | LREAL | -               | -                |  |  |
|        | DTL   | -               | -                |  |  |
|        | TOD   | -               | -                | X  | Wenn der Quellwert kleiner als 24 Stunden (86 400 00 ms) ist, dann wird das Bitmuster des Quellwerts ohne Änderung in den Zieldatentyp übertragen. Ansonsten wird der Zielwert nicht verändert. Das Ergebnis der Konvertierung zeigt die Zeit, die seit Mitternacht vergangen ist. |
|        | DATE  | -               | -                | -  | Keine implizite Konvertierung  |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung |
|--|--------|-----------------|------------------|-------------|
|  | STRING | -               | -                |             |
|  | CHAR   | -               | -                |             |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |             |

**Siehe auch**

- TIME (IEC-Zeit) (Seite 1107)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von TIME (Seite 1174)

**Datum und Uhrzeit**

**Implizite Konvertierung von DTL**

**Möglichkeiten zur impliziten Konvertierung**

Die implizite Konvertierung vom Datentyp DTL ist nicht möglich.

**Siehe auch**

- Explizite Konvertierung von DTL (Seite 1177)

**Implizite Konvertierung von TOD**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp TOD:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--------|-------|-----------------|------------------|---|
| TOD    | BOOL  | -               | -                | Keine implizite Konvertierung   |
|        | BYTE  | -               | -                |   |
|        | WORD  | -               | -                |   |
|        | DWORD | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr). |
|        | SINT  | -               | -                | Keine implizite Konvertierung   |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung   |
|--|--------|-----------------|------------------|---|
|  | USINT  | -               | -                |   |
|  | INT    | -               | -                |   |
|  | UINT   | -               | -                |   |
|  | DINT   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr). |
|  | UDINT  | -               | X                |   |
|  | REAL   | -               | -                | Keine implizite Konvertierung   |
|  | LREAL  | -               | -                |   |
|  | TIME   | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr). |
|  | DTL    | -               | -                |   |
|  | DATE   | -               | -                | Keine implizite Konvertierung   |
|  | STRING | -               | -                |   |
|  | CHAR   | -               | -                |   |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |   |

**Siehe auch**

- TOD (TIME\_OF\_DAY) (Seite 1108)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von TOD (Seite 1176)

**Implizite Konvertierung von DATE**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp DATE:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--------|-------|-----------------|------------------|--|
| DATE   | BOOL  | -               | -                | Keine implizite Konvertierung  |
|        | BYTE  | -               | -                |  |
|        | WORD  | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990. |
|        | DWORD | -               | -                | Keine implizite Konvertierung  |
|        | SINT  | -               | -                |  |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--|--------|-----------------|------------------|--|
|  | USINT  | -               | -                |  |
|  | INT    | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990. |
|  | UINT   | -               | X                |  |
|  | DINT   | -               | -                | Keine implizite Konvertierung  |
|  | UDINT  | -               | -                |  |
|  | REAL   | -               | -                |  |
|  | LREAL  | -               | -                |  |
|  | TIME   | -               | -                |  |
|  | DTL    | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung entspricht der Anzahl der Tage seit 01.01.1990. |
|  | TOD    | -               | -                | Keine implizite Konvertierung  |
|  | STRING | -               | -                |  |
|  | CHAR   | -               | -                |  |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |  |

**Siehe auch**

- DATE (Seite 1108)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von DATE (Seite 1175)

**Zeichenfolgen**

**Implizite Konvertierung von CHAR**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp CHAR:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--------|-------|-----------------|------------------|--|
| CHAR   | BOOL  | -               | -                | Keine implizite Konvertierung  |
|        | BYTE  | -               | X                | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Die restlichen Bit werden von links mit "0" aufgefüllt. |
|        | WORD  | -               | X                |  |
|        | DWORD | -               | X                |  |

| Quelle   | Ziel   | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung  |
|--|--------|-----------------|------------------|--|
|  | SINT   | -               | X                |  |
|  | USINT  | -               | X                |  |
|  | INT    | -               | X                |  |
|  | UINT   | -               | X                |  |
|  | DINT   | -               | X                |  |
|  | UDINT  | -               | X                |  |
|  | REAL   | -               | -                | Keine implizite Konvertierung                                    |
|  | LREAL  | -               | -                |  |
|  | TIME   | -               | -                |  |
|  | DTL    | -               | -                |  |
|  | TOD    | -               | -                |  |
|  | DATE   | -               | -                |  |
|  | STRING | X               | X                | Der STRING wird auf die Länge 1 gekürzt und enthält das Zeichen. |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |                 |                  |  |

**Siehe auch**

- CHAR (Zeichen) (Seite 1112)
- IEC-Prüfung einstellen oder aufheben (Seite 1135)
- Übersicht über die Datentypkonvertierung (Seite 1133)
- Explizite Konvertierung von CHAR (Seite 1178)

**Implizite Konvertierung von STRING**

**Möglichkeiten zur impliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten zur impliziten Konvertierung vom Datentyp STRING:

| Quelle | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung                   |
|--------|-------|-----------------|------------------|-------------------------------|
| STRING | BOOL  | -               | -                | Keine implizite Konvertierung |
|        | BYTE  | -               | -                |                               |
|        | WORD  | -               | -                |                               |
|        | DWORD | -               | -                |                               |
|        | SINT  | -               | -                |                               |
|        | USINT | -               | -                |                               |
|        | INT   | -               | -                |                               |

| Quelle   | Ziel  | Mit IEC-Prüfung | Ohne IEC-Prüfung | Erläuterung |
|--|-------|-----------------|------------------|-------------|
|  | UINT  | -               | -                |             |
|  | DINT  | -               | -                |             |
|  | UDINT | -               | -                |             |
|  | REAL  | -               | -                |             |
|  | LREAL | -               | -                |             |
|  | TIME  | -               | -                |             |
|  | DTL   | -               | -                |             |
|  | DATE  | -               | -                |             |
|  | TOD   | -               | -                |             |
|  | CHAR  | -               | X                |             |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |       |                 |                  |             |

**Siehe auch**

Explizite Konvertierung von STRING (Seite 1179)

**Explizite Konvertierung**

**Binärzahlen**

**Explizite Konvertierung von BOOL**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp BOOL:

| Quelle | Ziel  | Konvertierung | Erläuterung  | Mnemonik der Anweisung |
|--------|-------|---------------|--|------------------------|
| BOOL   | BYTE  | X             | Im Zieldatentyp wird nur das LSB (Least Significant Bit) gesetzt. Der Freigabeausgang ENO ist immer "1". | BOOL_TO_BYTE           |
|        | WORD  | X             |  | BOOL_TO_WORD           |
|        | DWORD | X             |  | BOOL_TO_DWORD          |
|        | SINT  | X             |  | BOOL_TO_SINT           |
|        | USINT | X             |  | BOOL_TO_USINT          |
|        | INT   | X             |  | BOOL_TO_INT            |
|        | UINT  | X             |  | BOOL_TO_UINT           |
|        | DINT  | X             |  | BOOL_TO_DINT           |
|        | UDINT | X             |  | BOOL_TO_UDINT          |

| Quelle   | Ziel   | Konvertierung | Erläuterung                   | Mnemonic der Anweisung |
|--|--------|---------------|-------------------------------|------------------------|
|  | REAL   | -             | Keine explizite Konvertierung | -                      |
|  | LREAL  | -             |                               | -                      |
|  | TIME   | -             |                               | -                      |
|  | DTL    | -             |                               | -                      |
|  | TOD    | -             |                               | -                      |
|  | DATE   | -             |                               | -                      |
|  | STRING | -             |                               | -                      |
|  | CHAR   | -             |                               | -                      |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |        |               |                               |                        |

**Siehe auch**

- BOOL (Bit) (Seite 1091)
- Implizite Konvertierung von BYTE (Seite 1136)
- Übersicht über die Datentypkonvertierung (Seite 1133)

**Bitfolgen**

**Explizite Konvertierung von BYTE**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp BYTE:

| Quelle             | Ziel                | Konvertierung | Erläuterung  | Mnemonic der Anweisung |
|--------------------|---------------------|---------------|--|------------------------|
| BYTE <sup>1)</sup> | BOOL                | X             | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bits ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | BYTE_TO_BOOL           |
|                    | WORD <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.   | BYTE_TO_WORD           |
|                    | DWORD <sup>1)</sup> | X             |  | BYTE_TO_DWORD          |
|                    | SINT                | X             |  | BYTE_TO_SINT           |

| Quelle | Ziel   | Konvertierung | Erläuterung  | Mnemonic der Anweisung |
|--------|--------|---------------|--|------------------------|
|        | USINT  | X             |  | BYTE_TO_USINT          |
|        | INT    | X             |  | BYTE_TO_INT            |
|        | UINT   | X             |  | BYTE_TO_UINT           |
|        | DINT   | X             |  | BYTE_TO_DINT           |
|        | UDINT  | X             |  | BYTE_TO_UDINT          |
|        | REAL   | X             |  | BYTE_TO_REAL           |
|        | LREAL  | X             |  | BYTE_TO_LREAL          |
|        | TIME   | X             |  | BYTE_TO_TIME           |
|        | DTL    | -             | Keine explizite Konvertierung  | -                      |
|        | TOD    | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. | BYTE_TO_TOD            |
|        | DATE   | X             |  | BYTE_TO_DATE           |
|        | STRING | -             | Keine explizite Konvertierung  | -                      |
|        | CHAR   | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. | BYTE_TO_CHAR           |

x: Konvertierung möglich

- : Konvertierung nicht möglich

<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) werden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT und DWORD als UDINT interpretiert.

**Siehe auch**

BYTE (Byte) (Seite 1091)

Implizite Konvertierung von BYTE (Seite 1136)

Übersicht über die Datentypkonvertierung (Seite 1133)



## Explizite Konvertierung von WORD

### Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp WORD:

| Quelle             | Ziel                | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--------------------|---------------------|---------------|---|------------------------|
| WORD <sup>1)</sup> | BOOL                | X             | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | WORD_TO_BOOL           |
|                    | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | WORD_TO_BYTE           |
|                    | DWORD <sup>1)</sup> | X             |   | WORD_TO_DWORD          |
|                    | SINT                | X             | <p>ENO = TRUE</p> <pre>#sint1 := WORD_TO_SINT(16#FFFF); // -1 bis #sint1 := WORD_TO_SINT(16#FF80); // -128 #sint1 := WORD_TO_SINT(16#0); // 0 bis #sint1 := WORD_TO_SINT(16#007F); // 127</pre> <p>ENO = FALSE</p> <pre>#sint1 := WORD_TO_SINT(16#FF7F); // -129 bis #sint1 := WORD_TO_SINT(16#8000); // -32768 #sint1 := WORD_TO_SINT(16#0080); // 128 bis #sint1 := WORD_TO_SINT(16#7FFF); // 32767</pre>   | WORD_TO_SINT           |
|                    | USINT               | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | WORD_TO_USINT          |
|                    | INT                 | X             |   | WORD_TO_INT            |
|                    | UINT                | X             |   | WORD_TO_UINT           |
|                    | DINT                | X             |   | WORD_TO_DINT           |
|                    | UDINT               | X             |   | WORD_TO_UDINT          |
|                    | REAL                | X             |   | WORD_TO_REAL           |
| LREAL              | X                   | WORD_TO_LREAL |   |                        |

| Quelle     | Ziel   | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|------------|--------|---------------|---|------------------------|
|            | TIME   | X             |   | WORD_TO_TIME           |
|            | DTL    | -             | Keine explizite Konvertierung   | -                      |
|            | TOD    | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | WORD_TO_TOD            |
|            | DATE   | X             |   | WORD_TO_DATE           |
|            | STRING | -             | Keine explizite Konvertierung   | -                      |
|            | CHAR   | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | WORD_TO_CHAR           |
| WORD_BCD16 | INT    | X             | Der zu konvertierende Wert hat den Datentyp WORD und wird als BCD-codierter Wert zwischen -999 und +999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ INT vor. Es findet eine echte Wandlung statt. Wenn das Bitmuster eine ungültige Tetrade enthält, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt. | WORD_BCD16_TO_INT      |
| BCD16      | INT    | X             |   | BCD16_TO_INT           |

x: Konvertierung möglich

- : Konvertierung nicht möglich

<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) werden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT und DWORD als UDINT interpretiert.

**Siehe auch**

WORD (Seite 1092)

Implizite Konvertierung von WORD (Seite 1137)

Übersicht über die Datentypkonvertierung (Seite 1133)

## Explizite Konvertierung von DWORD

### Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DWORD:

| Quelle              | Ziel               | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|---------------------|--------------------|---------------|---|------------------------|
| DWORD <sup>1)</sup> | BOOL               | X             | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | DWORD_TO_BOOL          |
|                     | BYTE <sup>1)</sup> | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | DWORD_TO_BYTE          |
|                     | WORD <sup>1)</sup> | X             |   | DWORD_TO_WORD          |

| Quelle | Ziel  | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--------|-------|---------------|---|------------------------|
|        | SINT  | X             | <pre> ENO = TRUE  #sint1 := DWORD_TO_SINT(16#FFFF_FFFF); // -1 bis #sint1 := DWORD_TO_SINT(16#FFFF_FF80); // -128 #sint1 := DWORD_TO_SINT(16#0); // 0 bis #sint1 := DWORD_TO_SINT(16#0000_007F); / / 127  ENO = FALSE  #sint1 := DWORD_TO_SINT(16#FFFF_FF7F); // -129 #sint1 := DWORD_TO_SINT(16#8000_0000); / / -2147483648 #sint1 := DWORD_TO_SINT(16#0000_0080); / / 128 bis #sint1 := DWORD_TO_SINT(16#7FFF_FFFF); // 2147483647                     </pre> | DWORD_TO_SINT          |
|        | USINT | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | DWORD_TO_USINT         |

| Quelle | Ziel   | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--------|--------|---------------|---|------------------------|
|        | INT    | X             | <p>ENO = TRUE</p> <pre>#int1 := DWORD_TO_INT(16#FFFF_FFFF); // -1 bis #int1 := DWORD_TO_INT(16#FFFF_8000); // -32768 #int1 := DWORD_TO_INT(16#0); // 0 bis #int1 := DWORD_TO_INT(16#0000_7FFF); // 32767</pre> <p>ENO = FALSE</p> <pre>#int1 := DWORD_TO_INT(16#FFFF_7FFF); // -32769 #int1 := DWORD_TO_INT(16#8000_0000); // -2147483648 #int1 := DWORD_TO_INT(16#8000); // 32768 bis #int1 := DWORD_TO_INT(16#7FFF_FFFF); // 2147483647</pre> | DWORD_TO_INT           |
|        | UINT   | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | DWORD_TO_UINT          |
|        | DINT   | X             |   | DWORD_TO_DINT          |
|        | UDINT  | X             |   | DWORD_TO_UDINT         |
|        | REAL   | X             | Der Datentyp DWORD wird nach REAL konvertiert. Wenn keine Fehler während der Konvertierung auftreten, ist der Signalzustand von ENO = 1; wenn ein Fehler während der Bearbeitung auftritt, ist der Signalzustand von ENO = 0.   | DWORD_TO_REAL          |
|        | LREAL  | X             | Der Datentyp DWORD wird nach LREAL konvertiert. Wenn keine Fehler während der Konvertierung auftreten, ist der Signalzustand von ENO = 1; wenn ein Fehler während der Bearbeitung auftritt, ist der Signalzustand von ENO = 0.  | DWORD_TO_LREAL         |
|        | TIME   | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | DWORD_TO_TIME          |
|        | DTL    | -             | Keine explizite Konvertierung   | -                      |
|        | TOD    | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | DWORD_TO_TOD           |
|        | DATE   | X             |   | DWORD_TO_DATE          |
|        | STRING | -             | Keine explizite Konvertierung   | -                      |

| Quelle  | Ziel | Konvertierung | Erläuterung   | Mnemonik der Anweisung |
|---|------|---------------|---|------------------------|
|   | CHAR | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | DWORD_TO_CHAR          |
| DWORD_BCD32   | DINT | X             | Der zu konvertierende Wert hat den Datentyp DWORD und wird als BCD-codierter Wert zwischen -9999999 und +9999999 angenommen. Das Ergebnis liegt nach der Konvertierung als Ganzzahl (Binärdarstellung) vom Typ DINT vor. Es findet eine echte Wandlung statt. Wenn das Bitmuster eine ungültige Tetrade enthält, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt. | DWORD_BCD32_TO_DINT    |
| BCD32   | DINT | X             |   | BCD32_TO_DINT          |
| <p>x: Konvertierung möglich<br/>                     - : Konvertierung nicht möglich<br/> <sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) werden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird als USINT, WORD als UINT und DWORD als UDINT interpretiert.</p> |      |               |   |                        |

**Siehe auch**

DWORD (Seite 1093)

Implizite Konvertierung von DWORD (Seite 1138)

Übersicht über die Datentypkonvertierung (Seite 1133)

## Ganzzahlen

### Explizite Konvertierung von SINT

#### Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp SINT:

| Quelle | Ziel                | Konvertierung | Erläuterung   | Mnemonic der Anweisung   |
|--------|---------------------|---------------|---|--|
| SINT   | BOOL                | X             | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | SINT_TO_BOOL   |
|        | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.  | SINT_TO_BYTE   |
|        | WORD <sup>1)</sup>  | X             |   | SINT_TO_WORD   |
|        | DWORD <sup>1)</sup> | X             |   | SINT_TO_DWORD  |
|        | USINT               | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFF) konvertiert. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.  | SINT_TO_USINT  |
|        | INT                 | X             |   | SINT_TO_INT  |
|        | UINT                | X             |   | SINT_TO_UINT   |
|        | DINT                | X             |   | SINT_TO_DINT   |
|        | UDINT               | X             |   | SINT_TO_UDINT  |
|        | REAL                | X             |   | Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "-1" wird z. B. mit Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt). |
|        | LREAL               | X             | SINT_TO_LREAL,<br>NORM_X  |  |
|        | TIME                | X             | Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.   | SINT_TO_TIME   |
|        | DTL                 | -             | Keine explizite Konvertierung   | -  |
|        | TOD                 | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFF) konvertiert. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0)   | SINT_TO_TOD  |

| Quelle | Ziel               | Konvertierung | Erläuterung   | Mnemonic der Anweisung           |
|--------|--------------------|---------------|---|----------------------------------|
|        | DATE               | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1) | SINT_TO_DATE                     |
|        | STRING             | X             | Der Wert wird in eine Zeichenkette konvertiert. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 4 Zeichen.  | SINT_TO_STRING, S_CONV, VAL_STRG |
|        | CHAR <sup>1)</sup> | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei der Konvertierung von negativen Werten, wird der Freigabeausgang ENO auf "0" gesetzt.   | SINT_TO_CHAR                     |

x: Konvertierung möglich  
 - : Konvertierung nicht möglich  
<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

**Siehe auch**

- SINT (8-Bit-Ganzzahlen) (Seite 1095)
- Implizite Konvertierung von SINT (Seite 1139)
- Übersicht über die Datentypkonvertierung (Seite 1133)



## Explizite Konvertierung von USINT

### Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp USINT:

| Quelle | Ziel                | Konvertierung  | Erläuterung   | Mnemonic der Anweisung |
|--------|---------------------|--|---|------------------------|
| USINT  | BOOL                | X  | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | USINT_TO_BOOL          |
|        | BYTE <sup>1)</sup>  | X  | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | USINT_TO_BYTE          |
|        | WORD <sup>1)</sup>  | X  |   | USINT_TO_WORD          |
|        | DWORD <sup>1)</sup> | X  |   | USINT_TO_DWORD         |
|        | SINT                | X  | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichen wechselt, wird der Freigabeausgang ENO auf "0" gesetzt.   | USINT_TO_SINT          |
|        | INT                 | X  | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | USINT_TO_INT           |
|        | UINT                | X  |   | USINT_TO_UINT          |
|        | DINT                | X  |   | USINT_TO_DINT          |
|        | UDINT               | X  |   | USINT_TO_UDINT         |
|        | REAL                | X  | Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "1" wird z. B. mit Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).  | USINT_TO_REAL, NORM_X  |
|        | LREAL               | X  |   | USINT_TO_LREAL, NORM_X |
|        | TIME                | X  | Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.   | USINT_TO_TIME          |
|        | DTL                 | -  | Keine Explizite Konvertierung   | -                      |
|        | TOD                 | X  | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | USINT_TO_TOD           |
|        | DATE                | X  |   | USINT_TO_DATE          |
| STRING | X                   | Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 4 Zeichen. | USINT_TO_STRING,<br>S_CONV, VAL_STRG  |                        |

| Quelle | Ziel               | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--------|--------------------|---------------|---|------------------------|
|        | CHAR <sup>1)</sup> | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. | USINT_TO_CHAR          |

x: Konvertierung möglich  
 - : Konvertierung nicht möglich  
<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

**Siehe auch**

- USINT (8-Bit-Ganzzahlen) (Seite 1095)
- Implizite Konvertierung von USINT (Seite 1140)
- Übersicht über die Datentypkonvertierung (Seite 1133)

**Explizite Konvertierung von INT**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp INT:

| Quelle | Ziel                | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--------|---------------------|---------------|---|------------------------|
| INT    | BOOL                | X             | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | INT_TO_BOOL            |
|        | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.  | INT_TO_BYTE            |
|        | WORD <sup>1)</sup>  | X             |   | INT_TO_WORD            |
|        | DWORD <sup>1)</sup> | X             |   | INT_TO_DWORD           |
|        | SINT                | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.   | INT_TO_SINT            |
|        | USINT               | X             |   | INT_TO_USINT           |
|        | UINT                | X             |   | INT_TO_UINT            |
|        | DINT                | X             |   | INT_TO_DINT            |
| UDINT  | X                   | INT_TO_UDINT  |   |                        |

| Quelle   | Ziel               | Konvertierung | Erläuterung  | Mnemonic der Anweisung                           |
|--|--------------------|---------------|--|--|
|  | REAL               | X             | Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "-1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).   | INT_TO_REAL, NORM_X                              |
|  | LREAL              | X             |  | INT_TO_LREAL, NORM_X                             |
|  | TIME               | X             | Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.  | INT_TO_TIME                                      |
|  | DTL                | -             | Keine explizite Konvertierung  | -  |
|  | TOD                | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)                           | INT_TO_TOD                                       |
|  | DATE               | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert)                    | INT_TO_DATE                                      |
|  | STRING             | X             | Der Wert wird in eine Zeichenkette konvertiert. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 6 Zeichen.   | INT_TO_STRING,<br>S_CONV, VAL_STRG <sup>1)</sup> |
|  | CHAR <sup>1)</sup> | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.  | INT_TO_CHAR                                      |
|  | BCD16              | X             | Der zu konvertierende Wert hat den Typ INT und wird als Ganzzahl mit einem Wert zwischen -999 und +999 angenommen. Das Ergebnis liegt nach der Konvertierung als BCD-codierte Zahl vom Typ WORD vor. Es findet eine echte Wandlung statt. Wenn der Wert außerhalb des Zielbereichs liegt, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt. | INT_TO_BCD16                                     |
|  | BCD16_WORD         | X             |  | INT_TO_BCD16_WORD                                |
| <p>x: Konvertierung möglich<br/> - : Konvertierung nicht möglich<br/> <sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.</p> |                    |               |  |  |

**Siehe auch**

- INT (16-Bit-Ganzzahlen) (Seite 1096)
- Implizite Konvertierung von INT (Seite 1141)
- Übersicht über die Datentypkonvertierung (Seite 1133)

**Explizite Konvertierung von UINT**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp UINT:

| Quelle | Ziel                | Konvertierung | Erläuterung   | Mnemonic der Anweisung        |
|--------|---------------------|---------------|---|-------------------------------|
| UINT   | BOOL                | X             | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | UINT_TO_BOOL                  |
|        | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn dabei Bit verloren gehen, dann wird der Freigabeausgang ENO auf "0" gesetzt.  | UINT_TO_BYTE                  |
|        | WORD <sup>1)</sup>  | X             |   | UINT_TO_WORD                  |
|        | DWORD <sup>1)</sup> | X             |   | UINT_TO_DWORD                 |
|        | SINT                | X             |   | UINT_TO_SINT                  |
|        | USINT               | X             |   | UINT_TO_USINT                 |
|        | INT                 | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichenbit geändert wird, wird der Freigabeausgang ENO auf "0" gesetzt.   | UINT_TO_INT                   |
|        | DINT                | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.  | UINT_TO_DINT                  |
|        | UDINT               | X             |   | UINT_TO_UDINT                 |
|        | REAL                | X             | Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).  | UINT_TO_REAL, NORM_X          |
|        | LREAL               | X             |   | UINT_TO_LREAL, NORM_X         |
|        | TIME                | X             | Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.   | UINT_TO_TIME                  |
|        | DTL                 | -             | -   | Keine Explizite Konvertierung |

| Quelle | Ziel               | Konvertierung | Erläuterung   | Mnemonic der Anweisung           |
|--------|--------------------|---------------|---|----------------------------------|
|        | TOD                | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)        | UINT_TO_TOD                      |
|        | DATE               | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert) | UINT_TO_DATE, T_CONV             |
|        | STRING             | X             | Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 6 Zeichen.  | UINT_TO_STRING, S_CONV, VAL_STRG |
|        | CHAR <sup>1)</sup> | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.  | UINT_TO_CHAR                     |

x: Konvertierung möglich

- : Konvertierung nicht möglich

<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD, LWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

### Siehe auch

UINT (16-Bit-Ganzzahlen) (Seite 1097)

Implizite Konvertierung von UINT (Seite 1142)

Übersicht über die Datentypkonvertierung (Seite 1133)

## Explizite Konvertierung von DINT

### Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DINT:

| Quelle | Ziel                | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--------|---------------------|---------------|---|------------------------|
| DINT   | BOOL                | X             | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | DINT_TO_BOOL           |
|        | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.  | DINT_TO_BYTE           |
|        | WORD <sup>1)</sup>  | X             |   | DINT_TO_WORD           |
|        | DWORD <sup>1)</sup> | X             |   | DINT_TO_DWORD          |
|        | SINT                | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.   | DINT_TO_SINT           |
|        | USINT               | X             |   | DINT_TO_USINT          |
|        | INT                 | X             |   | DINT_TO_INT            |
|        | UINT                | X             |   | DINT_TO_UINT           |
|        | UDINT               | X             |   | DINT_TO_UDINT          |
|        | REAL                | X             | Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "-1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "-1.0" umgewandelt).  | DINT_TO_REAL, NORM_X   |
|        | LREAL               | X             |   | DINT_TO_LREAL, NORM_X  |
|        | TIME                | X             | Der Wert wird in den Zieldatentyp übertragen und als Millisekunden interpretiert.   | DINT_TO_TIME, T_CONV   |
|        | DTL                 | -             | Keine explizite Konvertierung   | -                      |
|        | TOD                 | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0)  | DINT_TO_TOD            |

| Quelle | Ziel               | Konvertierung | Erläuterung  | Mnemonik der Anweisung           |
|--------|--------------------|---------------|--|----------------------------------|
|        | DATE               | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1)  | DINT_TO_DATE                     |
|        | STRING             | X             | Der Wert wird in eine Zeichenkette konvertiert. Die Zeichenkette wird mit einem führenden Vorzeichen dargestellt. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 11 Zeichen.  | DINT_TO_STRING, S_CONV, VAL_STRG |
|        | CHAR <sup>1)</sup> | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt.  | DINT_TO_CHAR                     |
|        | BCD32              | X             | Der zu konvertierende Wert hat den Typ DINT und wird als Ganzzahl mit einem Wert zwischen -999999 und +9999999 angenommen. Das Ergebnis liegt nach der Konvertierung als BCD-codierte Zahl vom Typ DWORD vor. Bei einem Überlauf wird der Freigabeausgang auf "0" gesetzt. Es findet eine echte Wandlung statt. Wenn der Wert außerhalb des Zielbereichs liegt, dann wird kein Synchronfehler ausgelöst, sondern nur das Statusbit OV gesetzt. | DINT_TO_BCD32                    |
|        | BCD32_DWORD        | X             |  | DINT_TO_BCD32_DWORD              |

x: Konvertierung möglich

- : Konvertierung nicht möglich

<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

## Siehe auch

DINT (32-Bit-Ganzzahlen) (Seite 1098)

Implizite Konvertierung von DINT (Seite 1143)

Übersicht über die Datentypkonvertierung (Seite 1133)

## Explizite Konvertierung von UDINT

### Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp UDINT:

| Quelle | Ziel                | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--------|---------------------|---------------|---|------------------------|
| UDINT  | BOOL                | X             | Die folgenden Möglichkeiten können auftreten: <ul style="list-style-type: none"> <li>• Wenn der Quellwert "0" ist, dann ist der Zieldatentyp ebenfalls "0" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert nur das LSB (Least Significant Bit) "1" ist, dann ist der Zieldatentyp ebenfalls "1" und der Freigabeausgang ENO "1".</li> <li>• Wenn im Quellwert Bit ungleich LSB sind, dann wird der Zieldatentyp gemäß LSB gesetzt und der Freigabeausgang ENO ist "0".</li> </ul> | UDINT_TO_BOOL          |
|        | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Wenn dabei Bit verloren gehen, dann wird der Freigabeausgang ENO auf "0" gesetzt.   | UDINT_TO_BYTE          |
|        | WORD <sup>1)</sup>  | X             |   | UDINT_TO_WORD          |
|        | DWORD <sup>1)</sup> | X             |   | UDINT_TO_DWORD         |
|        | SINT                | X             |   | UDINT_TO_SINT          |
|        | USINT               | X             |   | UDINT_TO_USINT         |
|        | INT                 | X             |   | UDINT_TO_INT           |
|        | UINT                | X             |   | UDINT_TO_UINT          |
|        | DINT                | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Wenn bei der Konvertierung das Vorzeichenbit geändert wird, wird der Freigabeausgang ENO auf "0" gesetzt.   | UDINT_TO_DINT          |
|        | REAL                | X             | Der Wert wird in das Format des Zieldatentyps konvertiert (Der Wert "1" wird z. B. mit der Anweisung "Wert konvertieren" (CONVERT) in den Wert "1.0" umgewandelt).  | UDINT_TO_REAL, NORM_X  |
|        | LREAL               | X             |   | UDINT_TO_LREAL, NORM_X |
|        | TIME                | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Millisekunden interpretiert, in den Zieldatentyp übertragen.  | UDINT_TO_TIME          |
|        | DTL                 | -             | Keine explizite Konvertierung   | -                      |
|        | TOD                 | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Millisekunden seit 0:0; Prüfung auf 24h Grenze)  | UDINT_TO_TOD, T_CONV   |



| Quelle | Ziel               | Konvertierung | Erläuterung   | Mnemonic der Anweisung            |
|--------|--------------------|---------------|---|-----------------------------------|
|        | DATE               | X             | Das Bitmuster des Quellwerts wird umgewandelt und in den Zieldatentyp übertragen. (Aus dem Wert "-1" (16#FF) wird der Wert "-1" (16#FFFFFF) Wenn ein negativer Wert in einen vorzeichenlosen Zieldatentyp konvertiert wird, wird der Freigabeausgang ENO auf "0" gesetzt. (Interpretation in Tagen seit 1990-1-1; Prüfung auf einen negativen Wert) | UDINT_TO_DATE                     |
|        | STRING             | X             | Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 11 Zeichen.   | UDINT_TO_STRING, S_CONV, VAL_STRG |
|        | CHAR <sup>1)</sup> | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt.  | UDINT_TO_CHAR                     |

x: Konvertierung möglich  
 - : Konvertierung nicht möglich

<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD, LWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert (das nicht vorhandene Vorzeichen wird durch Nullen ersetzt), anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

**Siehe auch**

- UDINT (32-Bit-Ganzzahlen) (Seite 1099)
- Implizite Konvertierung von UDINT (Seite 1144)
- Übersicht über die Datentypkonvertierung (Seite 1133)

**Gleitpunktzahlen**

**Explizite Konvertierung von REAL**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp REAL:

| Quelle | Ziel                | Konvertierung | Erläuterung                   | Mnemonic der Anweisung |
|--------|---------------------|---------------|-------------------------------|------------------------|
| REAL   | BOOL                | -             | Keine explizite Konvertierung | -                      |
|        | BYTE <sup>1)</sup>  | X             |                               | REAL_TO_BYTE           |
|        | WORD <sup>1)</sup>  | X             |                               | REAL_TO_WORD           |
|        | DWORD <sup>1)</sup> | X             |                               | REAL_TO_DWORD          |

| Quelle | Ziel   | Konvertierung | Erläuterung  | Mnemonic der Anweisung                                    |
|--------|--------|---------------|--|---|
|        | SINT   | X             | Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab. Wenn bei der Konvertierung der erlaubte Wertebereich des Zieldatentyps überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. | REAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X  |
|        | USINT  | X             |  | REAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X |
|        | INT    | X             |  | REAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X   |
|        | UINT   | X             |  | REAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X  |
|        | DINT   | X             |  | REAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X  |
|        | UDINT  | X             |  | REAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X |
|        | LREAL  | X             | Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab, z. B. TRUNC(2.5) = 2.0; CEIL(2.5) = 3.0  | REAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X |
|        | TIME   | -             | Keine explizite Konvertierung  | -   |
|        | DTL    | -             |  | -   |
|        | TOD    | -             |  | -   |
|        | DATE   | -             |  | -   |
|        | STRING | X             | Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 14 Zeichen.  | REAL_TO_STRING, S_CONV, VAL_STRG                          |
|        | CHAR   | -             | Keine explizite Konvertierung  | -   |

x: Konvertierung möglich

-: Konvertierung nicht möglich

<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

**Siehe auch**

REAL (Seite 1102)

Implizite Konvertierung von REAL (Seite 1145)

Übersicht über die Datentypkonvertierung (Seite 1133)

## Explizite Konvertierung von LREAL

### Möglichkeiten zur expliziten Konvertierung

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp LREAL:

| Quelle | Ziel  | Konvertierung   | Erläuterung  | Mnemonic der Anweisung  |   |
|--------|-------|---|--|---|---|
| LREAL  | BOOL  | -   | Keine explizite Konvertierung  | -   |   |
|        | BYTE  | X   | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen.   | LREAL_TO_BYTE   |   |
|        | WORD  | X   |  | LREAL_TO_WORD   |   |
|        | DWORD | X   |  | LREAL_TO_DWORD  |   |
|        | SINT  | X   | Der Wert wird in den Zieldatentyp konvertiert. Das Ergebnis der Konvertierung hängt von der verwendeten Anweisung ab. Wenn bei der Konvertierung der erlaubte Wertebereich überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. | LREAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X   |   |
|        | USINT | X   |  | LREAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X  |   |
|        | INT   | X   |  | LREAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X  |   |
|        | UINT  | X   |  | LREAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X   |   |
|        | DINT  | X   |  | LREAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X   |   |
|        | UDINT | X   |  | LREAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X  |   |
|        | REAL  | X   |  | Der Wert wird in den Zieldatentyp konvertiert. Wenn bei der Konvertierung der erlaubte Wertebereich überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. Ein Genauigkeitsverlust wird toleriert. | LREAL_TO_REAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X |
|        | TIME  | -   |  | Keine explizite Konvertierung   | -   |
|        | DTL   | -   | -  |   |   |
|        | TOD   | -   | -  |   |   |
|        | DATE  | -   | -  |   |   |
| STRING | X     | Der Wert wird in eine Zeichenkette konvertiert. Wenn die Länge der Zeichenkette überschritten wird oder der zu konvertierende Wert eine ungültige Gleitpunktzahl ist, wird der Freigabeausgang ENO auf "0" gesetzt. Die min. Länge der Zeichenkette beträgt 21 Zeichen. | REAL_TO_STRING, S_CONV, VAL_STRG   |   |   |

| Quelle   | Ziel | Konvertierung | Erläuterung                   | Mnemonic der Anweisung |
|--|------|---------------|-------------------------------|------------------------|
|  | CHAR | -             | Keine explizite Konvertierung | -                      |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |      |               |                               |                        |

**Siehe auch**

LREAL (Seite 1103)

Übersicht über die Datentypkonvertierung (Seite 1133)

**Zeiten**

**Explizite Konvertierung von TIME**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp TIME:

| Quelle | Ziel                | Konvertierung | Erläuterung  | Mnemonic der Anweisung |
|--------|---------------------|---------------|--|------------------------|
| TIME   | BOOL                | -             | Keine explizite Konvertierung  | -                      |
|        | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.  | TIME_TO_BYTE           |
|        | WORD <sup>1)</sup>  | X             |  | TIME_TO_WORD           |
|        | DWORD <sup>1)</sup> | X             |  | TIME_TO_DWORD          |
|        | SINT                | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Millisekunden interpretiert, in den Zieldatentyp übertragen.   | TIME_TO_SINT           |
|        | USINT               | X             |  | TIME_TO_USINT          |
|        | INT                 | X             |  | TIME_TO_INT            |
|        | UINT                | X             |  | TIME_TO_UINT           |
|        | DINT                | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen. Das Ergebnis der Konvertierung zeigt die Zeitdauer in Millisekunden.   | TIME_TO_DINT, T_CONV   |
|        | UDINT               | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig, als Millisekunden interpretiert, in den Zieldatentyp übertragen. Ein Vorzeichenwechsel führt dazu, dass der Freigabeausgang ENO "0" ist. | TIME_TO_UDINT          |
|        | REAL                | -             | Keine explizite Konvertierung  | -                      |
|        | LREAL               | -             |  | -                      |
|        | DTL                 | -             |  | -                      |

| Quelle | Ziel   | Konvertierung | Erläuterung  | Mnemonic der Anweisung |
|--------|--------|---------------|--|------------------------|
|        | TOD    | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. Wenn der Quellwert außerhalb des Wertebereichs von TOD liegt, dann wird der Zieldatentyp nicht verändert. | TIME_TO_TOD            |
|        | DATE   | -             | Keine explizite Konvertierung  | -                      |
|        | STRING | -             |  | -                      |
|        | CHAR   | -             |  | -                      |

x: Konvertierung möglich  
 -: Konvertierung nicht möglich  
<sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

**Siehe auch**

TIME (IEC-Zeit) (Seite 1107)

Implizite Konvertierung von TIME (Seite 1147)

Übersicht über die Datentypkonvertierung (Seite 1133)

**Datum und Uhrzeit**

**Explizite Konvertierung von DATE**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DATE:

| Quelle | Ziel                | Konvertierung | Erläuterung  | Mnemonic der Anweisung        |
|--------|---------------------|---------------|--|-------------------------------|
| DATE   | BOOL                | -             | Keine explizite Konvertierung  | -                             |
|        | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. | DATE_TO_BYTE                  |
|        | WORD <sup>1)</sup>  | X             |  | DATE_TO_WORD                  |
|        | DWORD <sup>1)</sup> | X             |  | DATE_TO_DWORD                 |
|        | SINT                | X             | Als Ergebnis wird die Anzahl der Tage seit dem 1.1.1990 geliefert.                           | DATE_TO_SINT                  |
|        | USINT               | X             |  | DATE_TO_USINT                 |
|        | INT                 | X             |  | DATE_TO_INT                   |
|        | UINT                | X             |  | DATE_TO_UINT                  |
|        | DINT                | X             |  | DATE_TO_DINT                  |
|        | UDINT               | X             |  | DATE_TO_UDINT                 |
|        | REAL                | -             |  | Keine explizite Konvertierung |
|        | LREAL               | -             | -  |                               |
|        | TIME                | -             | -  |                               |

| Quelle | Ziel   | Konvertierung | Erläuterung  | Mnemonic der Anweisung |
|--------|--------|---------------|--|------------------------|
|        | DTL    | X             | Das Bitmuster des Quellwerts wird ohne Änderung rechtsbündig in den Zieldatentyp übertragen. | DATE_TO_DTL            |
|        | TOD    | -             | Keine explizite Konvertierung  | -                      |
|        | STRING | -             |  | -                      |
|        | CHAR   | -             |  | -                      |

x: Konvertierung möglich  
 - : Konvertierung nicht möglich  
 1) Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation.

**Siehe auch**

DATE (Seite 1108)

Implizite Konvertierung von DATE (Seite 1149)

Übersicht über die Datentypkonvertierung (Seite 1133)

**Explizite Konvertierung von TOD**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp TOD:

| Quelle | Ziel                | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--------|---------------------|---------------|---|------------------------|
| TOD    | BOOL                | -             | Keine explizite Konvertierung   | -                      |
|        | BYTE <sup>1)</sup>  | X             | Das Bitmuster des Quellwerts wird ohne Änderung in den Zieldatentyp übertragen.                     | TOD_TO_BYTE            |
|        | WORD <sup>1)</sup>  | X             |   | TOD_TO_WORD            |
|        | DWORD <sup>1)</sup> | X             |   | TOD_TO_DWORD           |
|        | SINT                | X             | Als Ergebnis wird die Anzahl der Millisekunden seit Mitternacht geliefert.                          | TOD_TO_SINT            |
|        | USINT               | X             |   | TOD_TO_USINT           |
|        | INT                 | X             |   | TOD_TO_INT             |
|        | UINT                | X             |   | TOD_TO_UINT            |
|        | DINT                | X             |   | TOD_TO_DINT            |
|        | UDINT               | X             | Das Ergebnis der Konvertierung entspricht der Anzahl der Millisekunden seit Tagesbeginn (0:00 Uhr). | TOD_TO_UDINT, T_CONV   |
|        | REAL                | -             | Keine explizite Konvertierung   | -                      |
|        | LREAL               | -             |   | -                      |
|        | TIME                | X             | Als Ergebnis wird die Zeitdauer seit Mitternacht geliefert.   | TOD_TO_TIME            |
|        | DTL                 | X             | Als Ergebnis wird der Tag auf den 1.1.1970 gesetzt.   | TOD_TO_DTL             |

| Quelle   | Ziel   | Konvertierung | Erläuterung                   | Mnemonic der Anweisung |
|--|--------|---------------|-------------------------------|------------------------|
|  | DATE   | -             | Keine explizite Konvertierung | -                      |
|  | STRING | -             |                               | -                      |
|  | CHAR   | -             |                               | -                      |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich<br><sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst inklusive Vorzeichen auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation. |        |               |                               |                        |

**Siehe auch**

- TOD (TIME\_OF\_DAY) (Seite 1108)
- Implizite Konvertierung von TOD (Seite 1148)
- Übersicht über die Datentypkonvertierung (Seite 1133)

**Explizite Konvertierung von DTL**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp DTL:

| Quelle | Ziel  | Konvertierung                 | Erläuterung                   | Mnemonic der Anweisung   |                     |
|--------|-------|-------------------------------|-------------------------------|--|---------------------|
| DTL    | BYTE  | -                             | Keine explizite Konvertierung | -  |                     |
|        | WORD  | -                             |                               | -  |                     |
|        | DWORD | -                             |                               | -  |                     |
|        | SINT  | -                             |                               | -  |                     |
|        | USINT | -                             |                               | -  |                     |
|        | INT   | -                             |                               | -  |                     |
|        | UINT  | -                             |                               | -  |                     |
|        | DINT  | -                             |                               | -  |                     |
|        | UDINT | -                             |                               | -  |                     |
|        | REAL  | -                             |                               | -  |                     |
|        | LREAL | -                             |                               | -  |                     |
|        | TIME  | -                             |                               | -  |                     |
|        | TOD   | X                             |                               | Bei der Konvertierung wird die Tageszeit aus dem Format DTL extrahiert und in den Zieldatentyp geschrieben.  | DTL_TO_TOD, T_CONV  |
|        | DATE  | X                             |                               | Bei der Konvertierung wird das Datum aus dem Format DTL extrahiert und in den Zieldatentyp geschrieben. Bei einem Überlauf wird der Freigabeausgang ENO auf "0" gesetzt. | DTL_TO_DATE, T_CONV |
| STRING | -     | Keine explizite Konvertierung | -                             |  |                     |

| Quelle   | Ziel | Konvertierung | Erläuterung | Mnemonic der Anweisung |
|--|------|---------------|-------------|------------------------|
|  | CHAR | -             |             | -                      |
| x: Konvertierung möglich<br>-: Konvertierung nicht möglich |      |               |             |                        |

**Siehe auch**

DTL (Seite 1111)

Übersicht über die Datentypkonvertierung (Seite 1133)

**Zeichenfolgen**

**Explizite Konvertierung von CHAR**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp CHAR:

| Quelle | Ziel                | Konvertierung | Erläuterung                   | Mnemonic der Anweisung |
|--------|---------------------|---------------|-------------------------------|------------------------|
| CHAR   | BOOL                | -             | Keine explizite Konvertierung | -                      |
|        | BYTE <sup>1)</sup>  | X             |                               | CHAR_TO_BYTE           |
|        | WORD <sup>1)</sup>  | X             |                               | CHAR_TO_WORD           |
|        | DWORD <sup>1)</sup> | X             |                               | CHAR_TO_DWORD          |
|        | SINT                | X             |                               | CHAR_TO_SINT           |
|        | USINT               | X             |                               | CHAR_TO_USINT          |
|        | INT                 | X             |                               | CHAR_TO_INT            |
|        | UINT                | X             |                               | CHAR_TO_UINT           |
|        | DINT                | X             |                               | CHAR_TO_DINT           |
|        | UDINT               | X             |                               | CHAR_TO_UDINT          |
|        | REAL                | -             | Keine explizite Konvertierung | -                      |
|        | LREAL               | -             |                               | -                      |
|        | TIME                | -             |                               | -                      |
|        | DTL                 | -             |                               | -                      |
|        | TOD                 | -             |                               | -                      |
|        | DATE                | -             |                               | -                      |



| Quelle   | Ziel   | Konvertierung | Erläuterung   | Mnemonic der Anweisung |
|--|--------|---------------|---|------------------------|
|  | STRING | X             | Der Wert wird in das erste Zeichen der Zeichenkette (STRING) konvertiert. Wenn die Länge der Zeichenkette nicht definiert ist, wird nach der Konvertierung die Länge "1" eingestellt. Wenn die Länge der Zeichenkette definiert ist, bleibt diese nach der Konvertierung unverändert. | CHAR_TO_STRING         |
| x: Konvertierung möglich<br>- : Konvertierung nicht möglich<br><sup>1)</sup> Bitfolgen (BYTE, WORD, DWORD) und der Datentyp CHAR werden zuerst auf die notwendige Breite erweitert, anschließend werden die Bit kopiert. Der Quelltyp entscheidet über die Interpretation. |        |               |   |                        |

**Siehe auch**

CHAR (Zeichen) (Seite 1112)

Implizite Konvertierung von CHAR (Seite 1150)

Übersicht über die Datentypkonvertierung (Seite 1133)

**Explizite Konvertierung von STRING**

**Möglichkeiten zur expliziten Konvertierung**

Die folgende Tabelle zeigt die Möglichkeiten und Anweisungen zur expliziten Konvertierung vom Datentyp STRING:

| Quelle | Ziel  | Konvertierung | Erläuterung  | Mnemonic der Anweisung            |
|--------|-------|---------------|--|-----------------------------------|
| STRING | BOOL  | -             | Keine explizite Konvertierung  | -                                 |
|        | BYTE  | -             |  | -                                 |
|        | WORD  | -             |  | -                                 |
|        | DWORD | -             |  | -                                 |
|        | SINT  | X             | Die Konvertierung fängt mit dem ersten Zeichen der Zeichenkette (STRING) an und endet am Kettenende oder am ersten Zeichen, das nicht zulässig ist. Zulässig für die Konvertierung sind die folgenden Zeichen:   | STRING_TO_SINT, S_CONV, STRG_VAL  |
|        | USINT | X             |  | STRING_TO_USINT, S_CONV, STRG_VAL |
|        | INT   | X             |  | STRING_TO_INT, S_CONV, STRG_VAL   |
|        | UINT  | X             | <ul style="list-style-type: none"> <li>• Ziffer</li> <li>• Vorzeichen</li> <li>• Punkt</li> </ul>  | STRING_TO_UINT, S_CONV, STRG_VAL  |
|        | DINT  | X             | Das erste Zeichen der Zeichenkette darf ein Vorzeichen (+, -) oder eine Ziffer sein. Führende Leerzeichen werden ignoriert. Bei der Konvertierung von Gleitpunktzahlen dient der Punkt als Trennung. Die exponentielle Notation "e" oder "E" ist nicht erlaubt. Das Komma als Tausender-Trennzeichen links vom Dezimalpunkt ist erlaubt und wird | STRING_TO_DINT, S_CONV, STRG_VAL  |
|        | UDINT | X             |  | STRING_TO_UDINT, S_CONV, STRG_VAL |
| REAL   | X     |               | STRING_TO_REAL, S_CONV, STRG_VAL   |                                   |

| Quelle  | Ziel               | Konvertierung | Erläuterung   | Mnemonik der Anweisung            |
|---|--------------------|---------------|---|-----------------------------------|
|   | LREAL              | X             | ignoriert. Wenn der Aufbau der Zeichenkette für die Konvertierung ungültig ist oder ein Überlauf auftritt, wird der Freigabeausgang ENO auf "0" gesetzt.            | STRING_TO_LREAL, S_CONV, STRG_VAL |
|   | TIME               | -             | Keine explizite Konvertierung   | -                                 |
|   | DTL                | -             |   | -                                 |
|   | TOD                | -             |   | -                                 |
|   | DATE               | -             |   | -                                 |
|   | CHAR <sup>1)</sup> | X             | Das erste Zeichen der Zeichenkette (STRING) wird in den Zieldatentyp übertragen. Wenn die Zeichenkette leer ist, wird den Wert "0" in den Zieldatentyp geschrieben. | STRING_TO_CHAR, S_CONV            |
| x: Konvertierung möglich<br>- : Konvertierung nicht möglich |                    |               |   |                                   |

**Siehe auch**

STRING (Seite 1112)

Übersicht über die Datentypkonvertierung (Seite 1133)

**9.1.1.6 Programmflusssteuerung**

**EN-/ENO-Mechanismus**

**Grundlagen zum EN-/ENO-Mechanismus**

**Einführung**

Bei der Bearbeitung von Anweisungen können Laufzeitfehler auftreten, die einen Programmabbruch zur Folge haben. Um solche Programmabbrüche zu vermeiden, können Sie den EN-/ENO-Mechanismus nutzen. Dieser Mechanismus kann auf zwei Ebenen eingesetzt werden:

- EN-/ENO-Mechanismus für einzelne Anweisungen
- EN-/ENO-Mechanismus für Bausteinaufrufe

**EN-/ENO-Mechanismus für Anweisungen in KOP/FUP**

In KOP und FUP haben bestimmte Anweisungen einen Freigabeeingang EN (enable) und einen Freigabeausgang ENO (enable output).

Mit dem Freigabeeingang EN können Sie die Bearbeitung der Anweisung von Bedingungen abhängig machen. Die Anweisungen werden nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Mit dem Freigabeausgang ENO können Sie Laufzeitfehler in Anweisungen abfragen und darauf reagieren.

Der Freigabeausgang ENO führt den Signalzustand "1", wenn folgende Bedingungen zutreffen:

- Während der Bearbeitung ist kein Fehler aufgetreten.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Eingang EN liefert den Signalzustand "0".
- Während der Bearbeitung ist ein Fehler aufgetreten.

Der EN-/ENO-Mechanismus wird für folgende einfache Anweisungen verwendet:

- Mathematische Funktionen
- Verschieben
- Umwandler
- Wortverknüpfungen
- Schieben + Rotieren

In KOP und FUP können Sie die Auswertung des Freigabeausgangs ENO über das Kontextmenü gezielt für einzelne Anweisungen ein- und ausschalten.

### EN-/ENO-Mechanismus für Bausteinaufrufe in KOP/FUP

Alle Bausteine, die Sie in KOP oder FUP aufrufen, werden beim Aufruf mit einem Freigabeeingang EN und einem Freigabeausgang ENO versehen. Dies gilt für alle aufgerufenen Bausteine, unabhängig von der Programmiersprache, in der sie erstellt wurden.

Den Freigabeeingang EN können Sie nutzen, um den Baustein abhängig von Bedingungen aufzurufen. Der Baustein wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

Mit dem Freigabeausgang ENO können Sie den Fehlerstatus des Bausteins abfragen. Er führt Signal "1", sobald die Bausteinbearbeitung beginnt. Wenn Sie den Ausgang ENO im Programmcode nicht explizit auf "0" setzen, behält er Signal "1".

Sie können ihn jedoch explizit auf "0" setzen, um eine Fehleraussage an den aufgerufenen Baustein zurückzumelden. In KOP oder FUP wird der Ausgang ENO mit der Anweisung "RET: Zurück springen" gesetzt.

Siehe auch:

Beispiel für den EN-/ENO-Mechanismus in KOP (Seite 1183)

Beispiel für den EN-/ENO-Mechanismus in FUP (Seite 1183)

### EN-/ENO-Mechanismus für AWL

In AWL ist der EN-/ENO-Mechanismus für einzelne Anweisungen nicht notwendig. Diese Funktion wird durch sprachspezifische Anweisungsfolgen abgebildet.

Bausteine, die Sie aus einem AWL-Baustein heraus aufrufen, werden nicht mit den Parametern EN und ENO versehen. Unabhängig von der Programmiersprache, in der sie

erstellt wurden, können sie jedoch eine Fehleraussage an den aufrufenden AWL-Baustein über das BIE-Bit des Statusworts übergeben.

In AWL können Sie den Fehlerstatus des aufgerufenen Bausteins auswerten, indem Sie das BIE-Bit des Statusworts mit dem VKE verknüpfen. Es führt Signal "1", sobald die Bausteinbearbeitung beginnt. Wenn Sie es im Programmcode nicht explizit auf "0" setzen, behält es Signal "1".

Sie können es jedoch explizit auf "0" setzen, um eine Fehleraussage an den aufrufenden Baustein zurückzumelden. In AWL wird die Fehleraussage mit den Anweisungen "SAVE" oder "SPBNB" gesetzt.

Siehe auch: Beispiel für den Nachbau des EN-/ENO-Mechanismus in AWL (Seite 1185)

## EN-/ENO-Mechanismus in SCL

Bei SCL ist die Nutzung des EN-/ENO-Mechanismus für Anweisungen optional. Sie können sie durch die Bausteineigenschaft "ENO automatisch setzen" aktivieren. Wenn die Eigenschaft aktiv ist, erhalten alle Bausteine implizit eine Fehlerbehandlung.

Mit dem Freigabeeingang EN können Sie einen bedingten Bausteinanruf realisieren. Sie verwenden den Freigabeeingang EN in der Parameterliste wie einen normalen Eingangsparameter. Wenn EN Signal "1" führt oder wenn EN nicht verwendet wird, wird der Baustein aufgerufen. Wenn EN Signal "0" führt, wird der Baustein nicht aufgerufen.

---

### Hinweis

Beim Aufruf von Funktionen in SCL können Sie den Freigabemechanismus über EN nicht nutzen. Verwenden Sie stattdessen eine IF-Anweisung, um Funktionen bedingt aufzurufen.

---

Mit dem Freigabeausgang ENO fragen Sie den Fehlerstatus des Bausteins ab. Führt ENO Signal "1", wurde der Baustein ohne Fehler bearbeitet. Führt ENO Signal "0", trat während der Bearbeitung ein Fehler auf. Um den Zustand des Freigabeausgangs abzufragen, fügen Sie beim Bausteinanruf einen zusätzlichen Ausgangsparameter mit Namen ENO in die Parameterliste ein.

Siehe auch: Beispiel für den EN-/ENO-Mechanismus in SCL (Seite 1184)

## EN-/ENO-Mechanismus bei Speicher- und Peripheriezugriffsfehler

Speicher- und Peripheriezugriffsfehler können Sie nicht über den EN-/ENO-Mechanismus auswerten. Nutzen Sie hierzu entweder die globale Fehlerbehandlung über OBs (S7-300/400 und S7-1200/1500) oder die lokale Fehlerbehandlung über die Anweisung "GetError" (nur S7-1200/1500). Falls für eine Anweisung kein Speicherzugriffsfehler aufgetreten ist, können Sie das zugehörige ENO auswerten.

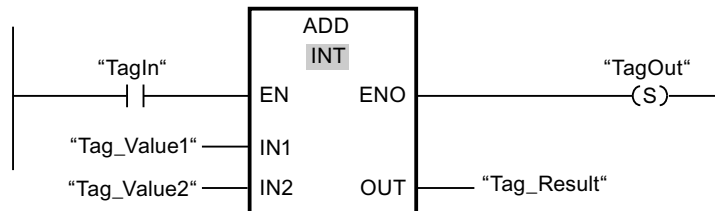
### Siehe auch

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

## Beispiel für den EN-/ENO-Mechanismus in KOP

### Beschreibung

Das folgende Bild zeigt einen Addierer mit EN- und ENO-Beschaltung:



Nach dem Schließkontakt enthält das VKE das Ergebnis der Vorverknüpfung:

- Führt "TagIn" Signal "0", dann wird die Addition nicht ausgeführt. EN und ENO führen beide den Signalzustand "0".
- Führt "TagIn" Signal "1", dann ist auch EN "1" und die Addition wird ausgeführt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Ausgang ENO ebenfalls den Signalzustand "1" und Ausgang "TagOut" wird gesetzt.

### Siehe auch

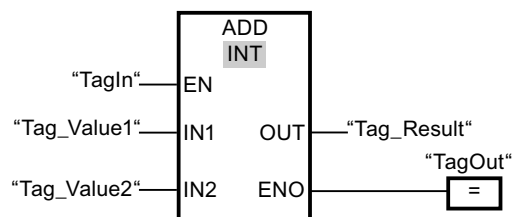
Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

ADD: Addieren (Seite 1727)

## Beispiel für den EN-/ENO-Mechanismus in FUP

### Beschreibung

Das folgende Bild zeigt einen Addierer mit EN- und ENO-Beschaltung:



- Führt "TagIn" Signal "1", ist auch EN "1" und die Addition wird ausgeführt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Ausgang ENO ebenfalls den Signalzustand "1" und Ausgang "TagOut" wird gesetzt.
- Führt "TagIn" Signal "0", dann wird die Addition nicht ausgeführt. EN und ENO führen beide den Signalzustand "0".

**Siehe auch**

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**Beispiel für den EN-/ENO-Mechanismus in SCL**

**Beispiel für den EN-/ENO-Mechanismus für einfache Anweisungen**

Um den EN-/ENO-Mechanismus für Anweisungen in SCL zu nutzen, müssen Sie die Bausteineigenschaft "ENO automatisch setzen" aktivieren. Das folgende Beispiel zeigt die Nutzung des Freigabeausgangs ENO für die Anweisung "a/b".

```
SCL
-----
"MyoutputREAL" :=a/b;
IF ENO
    THEN "MyOutputBool" :=1;
    ELSE "MyOutputBool" :=0;
END_IF;
```

Falls die Anweisung "a/b" fehlerfrei ausgeführt wird, führt MyOutputBool Signal "1".

**Beispiel für den EN-/ENO-Mechanismus für Bausteinaufrufe**

Das folgende Beispiel zeigt die Nutzung des Freigabeausgangs ENO für einen Bausteinaufruf.

```
SCL
-----
"MyDB"."MyFB" (EN:="MyTag1">"MyTag2",
              in1:="MyInputBool1",
              in2:="MyInputBool1",
              ENO=>"MyOutputBool");
```

Falls MyTag1 nicht größer als MyTag2 ist, wird der Bausteinaufruf nicht bearbeitet. EN und ENO führen beide den Signalzustand "0".

Falls MyTag1 größer als MyTag2 ist, führt EN Signal "1" und der Bausteinaufruf wird ausgeführt.

Wenn alle Anweisungen innerhalb von MyFB fehlerfrei ausgeführt werden, führt MyOutputBool Signal "1".

**Siehe auch**

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## Beispiel für den Nachbau des EN-/ENO-Mechanismus in AWL

### Beschreibung

Das folgende Beispiel zeigt einen Programmabschnitt zum Addieren von Werten mit EN- und ENO-Beschaltung:

| AWL             | Erläuterung   |
|-----------------|---|
| U "Tag_Input_1" | // Signalzustand des Operanden auf "1" abfragen<br>und mit dem aktuellen VKE nach UND verknüpfen  |
| SPBNB MyLABEL   | // Auswertung des EN-Eingangs<br>// Bei VKE=0 zur Sprungmarke "MyLABEL" springen<br>und das aktuelle VKE im BIE speichern<br>// Bei VKE="1" nächstfolgende Anweisung bearbeiten |
| L"Tag_Input_2"  | // Ersten Wert der Addition laden   |
| L"Tag_Input_3"  | // Zweiten Wert der Addition laden  |
| +I              | // Werte addieren   |
| T "Tag_Result"  | // Summe in den Operanden "Tag_Result" übertragen   |
| UN OV           | // Abfragen, ob Fehler aufgetreten sind   |
| SAVE            | // Signalzustand des VKE in das BIE-Bit übertragen  |
| CLR             | /// VKE auf "0" rücksetzen  |
| MyLABEL: U BIE  | // Sprungmarke "MyLABEL"<br>// BIE abfragen und mit dem VKE nach UND verknüpfen   |
| = "Tag_Output"  | //Signalzustand des VKE dem Operanden "Tag_Output"<br>zuweisen  |

Die Abfrage des Operanden "U "Tag\_Input\_1"" ergibt das Ergebnis der Vorverknüpfung (VKE). Die Anweisung "Springen bei VKE = 0 und VKE speichern (SPBNB)" speichert das VKE im BIE. Zudem wertet die Anweisung "Springen bei VKE = 0 und VKE speichern" das VKE aus und führt abhängig davon eine der folgende Aktionen aus:

- Ist das VKE "0", wird die Programmbearbeitung bei der Sprungmarke "MyLABEL" mit der Abfrage des BIE fortgesetzt. Die Addition wird nicht ausgeführt. Das aktuelle VKE wird dem Operanden "Tag\_Output" zugewiesen.
- Ist das VKE "1", wird die Addition ausgeführt. Mit der Abfrage des Überlaufbits (OV) wird ermittelt, ob bei der Addition ein Fehler auftrat. Das Abfrageergebnis wird im BIE gespeichert. Die Operation "CLR" setzt das VKE auf "0" zurück. Anschließend wird das BIE auf "1" abgefragt und mit dem aktuellen VKE nach UND verknüpft. Das Ergebnis wird dem Operanden "Tag\_Output" zugewiesen. Der Signalzustand des BIE sowie des Operanden "Tag\_Output" zeigt, ob die Addition fehlerfrei durchgeführt wurde

### Siehe auch

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## 9.1.2 PLC-Variablen deklarieren

### 9.1.2.1 Übersicht über PLC-Variablen tabellen

#### Einführung

PLC-Variablen tabellen enthalten die Definitionen der CPU-weit gültigen PLC-Variablen und symbolischen Konstanten. Für jede im Projekt angelegte CPU wird automatisch eine PLC-Variablen tabelle angelegt. Sie können selbst weitere Variablen tabellen anlegen, um Variablen und Konstanten darin zu sortieren und zu gruppieren.

In der Projektnavigation gibt es für jede CPU des Projekts einen Ordner "PLC-Variablen". Darin sind folgende Tabellen enthalten:

- Tabelle "Alle Variablen"
- Standard-Variablen tabelle
- Optional: Weitere anwenderdefinierte Variablen tabellen

#### Alle Variablen

Die Tabelle "Alle Variablen" zeigt eine Übersicht über alle PLC-Variablen, Anwenderkonstanten und Systemkonstanten der CPU. Diese Tabelle kann nicht gelöscht oder verschoben werden.

#### Standard-Variablen tabelle

Für jede CPU des Projekts gibt es eine Standard-Variablen tabelle. Sie kann nicht gelöscht, umbenannt oder verschoben werden. Die Standard-Variablen tabelle enthält PLC-Variablen, Anwenderkonstanten und Systemkonstanten. Sie können alle PLC-Variablen in der Standard-Variablen tabelle deklarieren, oder nach Bedarf weitere anwenderdefinierte Variablen tabellen anlegen.

#### Anwenderdefinierte Variablen tabellen

Sie können mehrere anwenderdefinierte Variablen tabellen für jede CPU anlegen, um Variablen nach Ihren Bedürfnissen zu gruppieren. Anwenderdefinierte Variablen tabellen können Sie umbenennen, in Gruppen zusammenfassen oder löschen. Anwenderdefinierte Variablen tabellen können PLC-Variablen und Anwenderkonstanten enthalten.

#### Siehe auch

Aufbau der PLC-Variablen tabellen (Seite 1187)

Variablen (Seite 1056)

Konstanten (Seite 1058)



### 9.1.2.2 Aufbau der PLC-Variablentabellen

#### Einführung


Jede PLC-Variablentabelle enthält je ein Register für Variablen und Anwenderkonstanten. Die Standard-Variablentabelle und die Tabelle "Alle Variablen" enthalten zusätzlich ein Register "Systemkonstanten".

#### Aufbau des Registers "Variablen"

Im Register "Variablen" deklarieren Sie die globalen PLC-Variablen, die Sie im Programm benötigen. Das folgende Bild zeigt den Aufbau des Registers. Die Anzahl der angezeigten Spalten kann variieren.

| Name  | Datentyp | Adresse | Remanenz                            | Erreichbar aus HMI                  | Sichtbar in HMI                     | Kommentar |
|---|----------|---------|-------------------------------------|-------------------------------------|-------------------------------------|-----------|
|  Motor1  | Bool     | %Q3.1   | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |           |
|  Motor2  | Bool     | %M3.2   | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |           |
|  Control | Bool     | %I3.3   | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |           |





Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Anzahl der angezeigten Spalten kann variieren. Die Spalten können Sie bei Bedarf ein- oder ausblenden.

| Spalte  | Erläuterung   |
|---|---|
|  | Symbol, das Sie anklicken können, um die Variable per Drag & Drop in ein Programm zu ziehen und dort als Operand zu verwenden.                                    |
| Name  | CPU-weit eindeutiger Name der Variablen.  |
| Datentyp  | Datentyp der Variablen.   |
| Adresse   | Adresse der Variablen.  |
| Remanenz  | Kennzeichnet die Variable als remanent.<br>Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten.                         |
| Erreichbar aus HMI  | Zeigt an, ob HMI zur Laufzeit auf diese Variable zugreifen kann.  |
| Sichtbar in HMI   | Zeigt an, ob die Variable in der Operandenauswahl von HMI per Voreinstellung sichtbar ist.  |
| Beobachtungswert  | Aktueller Datenwert in der CPU.<br>Die Spalte ist nur dann eingeblendet, wenn eine Online-Verbindung besteht und Sie die Schaltfläche "Alle beobachten" anwählen. |
| Variablentabelle  | Zeigt an, in welcher Variablentabelle die Variablendeklaration enthalten ist.<br>Diese Spalte ist nur in der Tabelle "Alle Variablen" enthalten.                  |
| Kommentar   | Kommentar zur Dokumentation der Variablen.  |


#### Aufbau der Register "Anwenderkonstanten" und "Systemkonstanten"

Im Register "Anwenderkonstanten" definieren Sie CPU-weit gültige symbolische Konstanten. Im Register "Systemkonstanten" werden vom System benötigte Konstanten angezeigt. Das

folgende Bild zeigt den Aufbau der beiden Register. Die Anzahl der gezeigten Spalten kann variieren.

|   | Name    | Datentyp | Wert  | Kommentar |
|---|---------|----------|-------|-----------|
|  | Const_1 | Bool     | true  |           |
|  | Const_2 | Byte     | 12    |           |
|  | Const_3 | Bool     | false |           |
|  | Const_4 | Real     | 1.0   |           |

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Spalten können Sie bei Bedarf ein- oder ausblenden.

| Spalte  | Erläuterung   |
|---|---|
|  | Symbol, das Sie anklicken können, um die Variable per Drag & Drop in ein Netzwerk zu ziehen und dort als Operand zu verwenden.                    |
| Name  | CPU-weit eindeutiger Name der Konstanten.   |
| Datentyp  | Datentyp der Konstanten.  |
| Wert  | Wert der Konstanten.  |
| Variablentabelle  | Zeigt an, in welcher Variablentabelle die Konstantendeklaration enthalten ist.<br>Diese Spalte ist nur in der Tabelle "Alle Variablen" enthalten. |
| Kommentar   | Kommentar zur Dokumentation der Konstanten.   |

**Siehe auch**

- Variablen (Seite 1056)
- Konstanten (Seite 1058)
- Übersicht über PLC-Variablentabellen (Seite 1186)
- Tabellenspalten ein- und ausblenden (Seite 1210)
- Bearbeiten von Tabellen (Seite 216)

**9.1.2.3 Regeln für PLC-Variablen**

**Zulässige Namen von PLC-Variablen**

**Zulässige Zeichen**

Für die Namen von PLC-Variablen gelten folgende Regeln:

- Buchstaben, Ziffern, Sonderzeichen sind zulässig.
- Anführungszeichen sind nicht zulässig.

### Eindeutige Variablennamen

Die Namen der PLC-Variablen müssen CPU-weit eindeutig sein, auch wenn sich die Variablen in unterschiedlichen Variablentabellen einer CPU befinden. Einen Namen, der bereits für einen Baustein, eine andere PLC-Variable oder eine Konstante innerhalb der CPU vergeben ist, können Sie nicht für eine neue PLC-Variable verwenden. Groß- und Kleinschreibung wird bei der Prüfung der Eindeutigkeit nicht unterschieden.

Wenn Sie einen bereits vergebenen Namen erneut eintragen, wird automatisch an den zweiten Namen eine fortlaufende Nummer angehängt. Wenn Sie z. B. zwei Mal den Namen "Motor" eingeben, wird der zweite Eintrag in "Motor(1)" umgewandelt.

### Eindeutige Tabellennamen

Auch die Namen der PLC-Variablentabellen müssen CPU-weit eindeutig sein. Beim Anlegen von anwenderdefinierten PLC-Variablentabellen wird automatisch ein eindeutiger Name vorgeschlagen.

### Siehe auch

Variablen (Seite 1056)

Zulässige Adressen und Datentypen von PLC-Variablen (Seite 1189)

Schlüsselwörter (Seite 1059)

### Zulässige Adressen und Datentypen von PLC-Variablen

Die Adressen von PLC-Variablen bestehen aus der Angabe des Operandenbereichs und der Adresse innerhalb dieses Bereichs.

Die Adressen müssen CPU-weit eindeutig sein. Wenn Sie eine Adresse eintragen, die bereits einer anderen Variablen zugewiesen ist, wird die Adresse an beiden Stellen gelb markiert und eine Fehlermeldung ausgegeben.

### Operandenbereiche

Die folgende Tabelle zeigt die möglichen Operandenbereiche. Die verfügbaren Datentypen sind abhängig von der verwendeten CPU:

| Operandenbereich        |                   | Erläuterung        | Datentyp  | Format         | Adressbereich: |              |              |
|-------------------------|-------------------|--------------------|---|----------------|----------------|--------------|--------------|
| Internationale Mnemonik | Deutsche Mnemonik |                    |   |                | S7-1200        | S7-300/400   | S7-1500      |
| I                       | E                 | Eingangsbitt       | BOOL  | I x.y<br>E x.y | 0.0..1023.7    | 0.0..65535.7 | 0.0..32766.7 |
| I                       | E                 | Eingang (64 Bit)   | LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL, PLC-Datentyp | I x.0<br>E x.0 | -              | -            | 0.0..32761.0 |
| IB                      | EB                | Eingangsbyte       | BYTE, CHAR, SINT, USINT, PLC-Datentyp                     | IB x<br>EB y   | 0..1023        | 0..65535     | 0..32767     |
| IW                      | EW                | Eingangswort       | WORD, INT, UINT, DATE, S5TIME, PLC-Datentyp               | IW x<br>EW y   | 0..1022        | 0..65534     | 0..32765     |
| ID                      | ED                | Eingangsdoppelwort | DWORD, DINT, UDINT, REAL, TIME, TOD, PLC-Datentyp         | ID x<br>ED y   | 0..1020        | 0..65532     | 0..32763     |
| Q                       | A                 | Ausgangsbitt       | BOOL  | Q x.y<br>A x.y | 0.0..1023.7    | 0.0..65535.7 | 0.0..32766.7 |
| Q                       | A                 | Ausgang (64 Bit)   | LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL, PLC-Datentyp | Q x.0<br>A x.0 | -              | -            | 0.0..32761.0 |
| QB                      | AB                | Ausgangsbyte       | BYTE, CHAR, SINT, USINT, PLC-Datentyp                     | QB x<br>AB y   | 0..1023        | 0..65535     | 0..32767     |
| QW                      | AW                | Ausgangswort       | WORD, INT, UINT, DATE, S5TIME, PLC-Datentyp               | QW x<br>AW y   | 0..1022        | 0..65534     | 0..32765     |

| Operandenbereich        |                   | Erläuterung                       | Datentyp  | Format       | Adressbereich: |              |              |
|-------------------------|-------------------|-----------------------------------|---|--------------|----------------|--------------|--------------|
| Internationale Mnemonik | Deutsche Mnemonik |                                   |   |              | S7-1200        | S7-300/400   | S7-1500      |
| QD                      | AD                | Ausgangsdoppelwort                | DWORD, DINT, UDINT, REAL, TIME, TOD, PLC-Datentyp | QD x<br>AD y | 0..1020        | 0..65532     | 0..32763     |
| M                       | M                 | Merkerbit                         | BOOL  | M x.y        | 0.0..8191.7    | 0.0..65535.7 | 0.0..16383.7 |
| M                       | M                 | Merker (64 Bit)                   | LREAL   | M x.0        | 0.0..8190.0    | -            | 0.0..16378.0 |
| M                       | M                 | Merker (64 Bit)                   | LWORD, LINT, ULINT, LTIME, LTOD, LDT              | M x.0        | -              | -            | 0.0..16378.0 |
| MB                      | MB                | Merkerbyte                        | BYTE, CHAR, SINT, USINT                           | MB x         | 0..8191        | 0..65535     | 0..16383     |
| MW                      | MW                | Merkerwort                        | WORD, INT, UINT, DATE, S5TIME                     | MW x         | 0..8190        | 0..65534     | 0..16382     |
| MD                      | MD                | Merkerdoppelwort                  | DWORD, DINT, UDINT, REAL, TIME, TOD               | MD x         | 0..8188        | 0..65532     | 0..16380     |
| T                       | T                 | Zeitfunktion (nur bei S7-300/400) | Timer   | T n          | -              | 0..65535     | 0..2047      |
| C                       | Z                 | Zählfunktion (nur bei S7-300/400) | Counter   | Z n<br>C n   | -              | 0..65535     | 0..2047      |

## Adressen

Die folgende Tabelle zeigt die möglichen Adressen der Variablen:

| Datentyp                      | Adresse  | Beispiel |
|-------------------------------|--|----------|
| BOOL                          | Variablen mit Datentyp BOOL werden mit einer Bytenummer und einer Bitnummer adressiert. Die Nummerierung der Bytes beginnt für jeden Operandenbereich bei 0. Die Nummerierung der Bits geht von 0 bis 7. | A 1.0    |
| BYTE, CHAR, SINT, USINT       | Variablen mit Datentyp BYTE, CHAR, SINT, USINT werden mit einer Bytenummer adressiert.   | MB 1     |
| WORD, INT, UINT, DATE, S5TIME | Variablen mit Datentyp WORD, INT, UINT, DATE, S5TIME bestehen aus zwei Bytes. Sie werden mit der Nummer des niedrigeren Bytes adressiert.  | EW 1     |

| Datentyp  | Adresse   | Beispiel |
|---|---|----------|
| DWORD,<br>DINT,<br>UDINT,<br>REAL, TIME,<br>TOD         | Variablen mit Datentyp DWORD, DINT, UDINT, REAL, TIME, TOD bestehen aus vier Bytes. Sie werden mit der Nummer des niedrigeren Bytes adressiert.                             | AD 1     |
| LWORD,<br>LINT, ULINT,<br>LTIME,<br>LTOD, LDT,<br>LREAL | Variablen mit Datentyp LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL bestehen aus acht Bytes. Sie werden mit der Nummer des niedrigeren Bytes und der Bitnummer 0 adressiert. | E 1.0    |

### Verwendete Mnemonik

Die Adressen, die Sie in der PLC-Variablen-tabelle eingeben, werden automatisch an die eingestellte Mnemonik angepasst.

### Siehe auch

- Einstellen der Mnemonik (Seite 1246)
- Variablen (Seite 1056)
- Zulässige Namen von PLC-Variablen (Seite 1188)
- Übersicht über die gültigen Datentypen (Seite 1087)

### 9.1.2.4 PLC-Variablen-tabellen anlegen und verwalten

#### PLC-Variablen-tabelle anlegen

Sie können mehrere anwenderdefinierten PLC-Variablen-tabellen in einer CPU anlegen. Jede Variablen-tabelle muss einen CPU-weit eindeutigen Namen haben.

#### Voraussetzung

Die Projektansicht ist geöffnet.

#### Vorgehen

Um eine neue PLC-Variablen-tabelle anzulegen, gehen Sie folgendermaßen vor:

- Öffnen Sie in der Projektnavigation den Ordner "PLC-Variablen" unterhalb der CPU.
- Doppelklicken Sie auf den Eintrag "Neue Variablen-tabelle hinzufügen".  
Eine neue PLC-Variablen-tabelle mit dem Standardnamen "Variablen-tabelle\_x" wird angelegt.
- Selektieren Sie die PLC-Variablen-tabelle in der Projektnavigation.

4. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
5. Geben Sie einen CPU-weit eindeutigen Namen ein.

## Ergebnis

Eine neue PLC-Variablentabelle wird angelegt. In dieser können Sie Variablen und Konstanten deklarieren.

## Siehe auch

- Übersicht über PLC-Variablentabellen (Seite 1186)
- Aufbau der PLC-Variablentabellen (Seite 1187)
- Importieren und Exportieren (Seite 1586)

## PLC-Variablentabellen gruppieren

Sie können anwenderdefinierte Variablentabellen einer CPU in Gruppen zusammenfassen. Die Standard-Variablentabelle und die Tabelle "Alle Variablen" können Sie jedoch nicht in eine Gruppe verschieben.

## Voraussetzung

Im Ordner "PLC-Variablen" der CPU sind mehrere anwenderdefinierte Variablentabellen enthalten.

## Vorgehen

Um mehrere PLC-Variablentabellen in einer Gruppe zusammenzufassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation den Ordner "PLC-Variablen" unterhalb der CPU.
2. Wählen Sie den Menübefehl "Einfügen > Gruppe".  
Eine neue Gruppe mit dem Standardnamen "Gruppe\_x" wird eingefügt.
3. Selektieren Sie die neu eingefügte Gruppe in der Projektnavigation.
4. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
5. Geben Sie der neuen Gruppe einen CPU-weit eindeutigen Namen.
6. Ziehen Sie die zu gruppierenden Tabellen per Drag & Drop in die neue Gruppe.

## Ergebnis

Die Variablentabellen werden in der neuen Gruppe zusammengefasst.

## Siehe auch

Übersicht über PLC-Variablen tabellen (Seite 1186)

Aufbau der PLC-Variablen tabellen (Seite 1187)

## PLC-Variablen tabelle öffnen

### Vorgehen

Um die PLC-Variablen tabelle einer CPU zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Variablen" unterhalb der CPU.
2. Doppelklicken Sie auf die PLC-Variablen tabelle im Ordner.
3. Wählen Sie in der rechten oberen Ecke das gewünschte Register.

### Ergebnis

Die zur CPU gehörende PLC-Variablen tabelle wird geöffnet. Sie können die benötigten Variablen und Konstanten deklarieren.

## Siehe auch

Übersicht über PLC-Variablen tabellen (Seite 1186)

Aufbau der PLC-Variablen tabellen (Seite 1187)

### 9.1.2.5 PLC-Variablen deklarieren

#### PLC-Variablen deklaration eingeben

#### Variablen in der PLC-Variablen tabelle deklarieren

#### Voraussetzung

Das Register "Variablen" einer PLC-Variablen tabelle ist geöffnet.

#### Vorgehen

Um PLC-Variablen zu definieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablen namen ein.
2. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.  
Eine dem Datentyp entsprechende Adresse wird automatisch ergänzt.



3. Optional: Klicken Sie in der Spalte "Adresse" auf die Pfeiltaste und geben Sie im nachfolgenden Dialog ein Operandenkennzeichen, einen Operandentyp, eine Adresse und eine Bitnummer ein.
4. Optional: Tragen Sie in der Spalte "Kommentar" einen Kommentar ein.
5. Wiederholen Sie die Schritte 1 - 4 für alle erforderlichen Variablen.

Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 1189)

## Syntaxprüfung

Nach jeder Eingabe wird automatisch eine Syntaxprüfung durchgeführt, die vorhandene Fehler rot anzeigt. Sie müssen diese Fehler nicht sofort beheben, sondern können mit der weiteren Bearbeitung fortfahren und die Korrekturen zu einem späteren Zeitpunkt durchführen. Solange die Variablendeklaration Syntaxfehler enthält und die Variable im Programm verwendet wird, lässt sich das Programm allerdings nicht übersetzen.

## Siehe auch

Importieren und Exportieren (Seite 1586)

Zulässige Namen von PLC-Variablen (Seite 1188)

PLC-Variablen im Programmiereditor deklarieren (Seite 1195)

Aufbau der PLC-Variablentabellen (Seite 1187)

Bearbeiten von Tabellen (Seite 216)

## PLC-Variablen im Programmiereditor deklarieren

### Voraussetzung

- Der Programmiereditor ist geöffnet.

### Vorgehen

Um Operanden als globale PLC-Variablen zu deklarieren, gehen Sie folgendermaßen vor:

1. Fügen Sie eine Anweisung in Ihr Programm ein.  
Die Zeichenfolgen "<???", "<??.>" oder "..." stellen Operandenplatzhalter dar.
2. Ersetzen Sie einen Operandenplatzhalter mit dem Namen der zu erstellenden PLC-Variablen.
3. Selektieren Sie den Variablennamen.  
Wenn Sie mehrere PLC-Variablen deklarieren wollen, selektieren Sie die Namen aller zu deklarierenden Variablen.
4. Wählen Sie im Kontextmenü den Befehl "Variable definieren".  
Der Dialog "Variable definieren" wird geöffnet. Er zeigt eine Deklarationstabelle, in die bereits der Variablenname eingetragen ist.

5. Klicken Sie in der Spalte "Abschnitt" auf die Pfeiltaste und wählen Sie einen der folgenden Einträge:
  - Global Memory
  - Global Input
  - Global Output
6. Geben Sie in den weiteren Spalten Adresse, Datentyp und Kommentar ein.  
Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 1189)
7. Falls die CPU mehrere PLC-Variablen enthält, können Sie in der Spalte "PLC-Variablen" eingeben, in welche Tabelle die Variable eingefügt werden soll. Wenn Sie in der Spalte keine Angabe machen, wird die neue Variable in die Standard-Variablen-Tabelle eingefügt.
8. Klicken Sie auf die Schaltfläche "Definieren", um die Eingabe abzuschließen.

## Ergebnis

Die Variablendeklaration wird in die PLC-Variablen-Tabelle geschrieben und ist für alle Bausteine in der CPU gültig.

## Siehe auch

Zulässige Namen von PLC-Variablen (Seite 1188)

Bearbeiten von Tabellen (Seite 216)

Variablen in der PLC-Variablen-Tabelle deklarieren (Seite 1194)

## Remanenz von PLC-Variablen einstellen

### Remanenzverhalten von PLC-Variablen

#### Remanente PLC-Variablen

Jede CPU verfügt über einen Speicherbereich, dessen Inhalt auch nach dem Ausschalten der Versorgungsspannung erhalten bleibt. Dieser Bereich wird als remanenter Speicherbereich bezeichnet.

Um Datenverluste bei Spannungsausfall zu vermeiden, können Sie bestimmte PLC-Variablen in diesem Speicherbereich ablegen. Die Remanenzeinstellung von PLC-Variablen legen Sie in der PLC-Variablen-Tabelle fest.

Je nach CPU-Familie kann der remanente Speicherbereich unterschiedliche Arten von PLC-Variablen aufnehmen. Die folgende Tabelle gibt eine Übersicht über die Möglichkeiten der unterschiedlichen CPUs:

| CPU-Typ             | Remanente Merker | Remanente SIMATIC-Zeiten | Remanente SIMATIC-Zähler |
|---------------------|------------------|--------------------------|--------------------------|
| Baureihe S7-300/400 | ✓                | -                        | -                        |
| Baureihe S7-1200    | ✓                | -                        | -                        |
| Baureihe S7-1500    | ✓                | ✓                        | ✓                        |

## Siehe auch

Remanenzverhalten von PLC-Variablen einstellen (Seite 1197)

## Remanenzverhalten von PLC-Variablen einstellen

### Einführung

In der PLC-Variablen-tabelle können Sie die Breite des remanenten Speicherbereichs für PLC-Variablen festlegen. Alle Variablen, deren Adresse in diesem Speicherbereich liegen, werden dann als remanent gekennzeichnet. Die Remanenzeinstellung einer Variablen erkennen Sie am gesetzten Häkchen in der Spalte "Remanenz" der PLC-Variablen-tabelle.

### Voraussetzung

Das Register "Variablen" der PLC-Variablen-tabelle ist geöffnet.

### Vorgehen

Um die Breite des remanenten Speicherbereichs für PLC-Variablen zu definieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Remanenz".  
Der Dialog "Remanenter Speicher" wird geöffnet.
2. Legen Sie die Breite des remanenten Speicherbereichs fest, indem Sie die Anzahl der remanenten Bytes, Zeiten oder Zähler in das Eingabefeld eingeben.
3. Klicken Sie auf die Schaltfläche "OK".

### Ergebnis

Die Breite des remanenten Speicherbereichs wird definiert. In der Spalte "Remanenz" der Variablen-tabelle wird automatisch ein Häkchen für alle Variablen gesetzt, die im remanenten Speicherbereich liegen.

## Siehe auch

Remanenzverhalten von PLC-Variablen (Seite 1196)

Bearbeiten von Tabellen (Seite 216)

### 9.1.2.6 PLC-Variablen für Ein- und Ausgänge in Strukturen zusammenfassen

#### Wissenswertes zu strukturierten PLC-Variablen

##### Nutzen von strukturierten PLC-Variablen (S7-1200 ab V4/S7-1500)

Um Ihr Programm übersichtlicher zu gestalten, können Sie mehrere Ein- oder Ausgangsadressen in einer übergeordneten PLC-Variable zusammenfassen. Die übergeordnete PLC-Variable stellt eine Struktur dar, die mehrere logisch zusammengehörige Ein- oder Ausgänge enthält. Beim Bausteinaufruf übergeben Sie die übergeordnete Variable und benötigen somit nur einen Ein- oder Ausgangsparameter für alle zusammengehörenden Ein- oder Ausgänge.

#### Funktionsweise

Um eine strukturierte PLC-Variable zu erstellen, definieren Sie zunächst einen PLC-Datentypen (UDT). Darin deklarieren Sie die benötigten Datenelemente und legen deren Namen und Datentypen fest.

Anschließend wechseln Sie in die PLC-Variablen-Tabelle und legen dort die übergeordnete PLC-Variable an. Als Datentyp für die Variable wählen Sie Ihren PLC-Datentypen aus. Das System reserviert nun ab der Startadresse der übergeordneten Variable eine bestimmte Anzahl von Ein- oder Ausgangsadressen. Die Anzahl der reservierten Adressen hängt von der Länge Ihres PLC-Datentyps ab.

Wenn Sie einen Baustein aufrufen, der die reservierten Ein- oder Ausgänge zur Programmbearbeitung benötigt, übergeben Sie die übergeordnete Variable als Bausteinparameter.

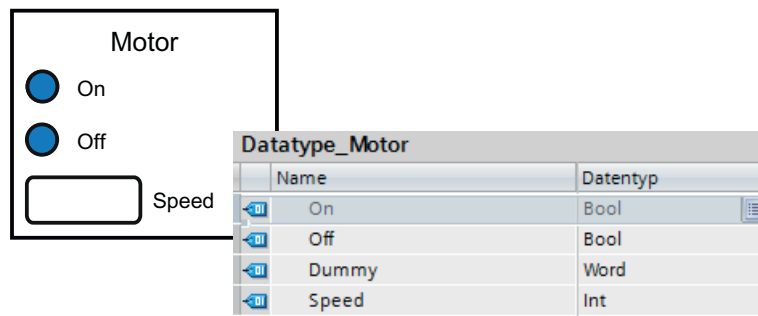
Im Programmcode können Sie die einzelnen PLC-Variablen wie Strukturelemente adressieren.

Eine detaillierte Beschreibung der einzelnen Handlungsschritte finden Sie in den nachfolgenden Kapiteln.

#### Anwendungsbeispiel

Sie können strukturierte PLC-Variablen einsetzen, um die Ein- oder Ausgänge eines Funktionsmoduls zu gruppieren. Das folgende Bild zeigt die schematische Darstellung eines Motors. Für jeden der drei Eingänge wurde eine Komponente im PLC-Datentyp "Datatype\_Motor" angelegt.

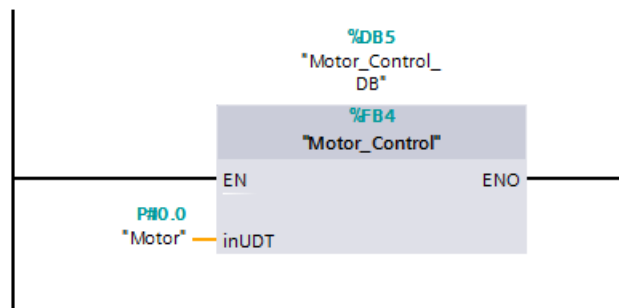
Die Speicherbereiche der deklarierten Variablen dürfen sich nicht überlappen. Im Beispiel sehen Sie, dass die Komponente "Speed" vom Datentyp "Integer" ist und deshalb an einer Wortadresse beginnen muss. Aus diesem Grund wurde das erste Eingangswort mit der Füll-Variable "Dummy" aufgefüllt. "Speed" liegt somit auf dem zweiten Eingangswort.



Das folgende Bild zeigt die übergeordnete PLC-Variable "Motor", die auf dem Datentyp "Datatype\_Motor" beruht. Durch die Deklaration von "Motor" werden die Adressen EW0 und EW1 auf der Eingangsbaugruppe reserviert.

| MyTagTable |              |                  |         |
|------------|--------------|------------------|---------|
|            | Name         | Datentyp         | Adresse |
| 1          | Motor        | *Datatype_Motor* | %I0.0   |
| 2          | <Hinzufügen> |                  |         |

Das folgende Bild zeigt die Übergabe der PLC-Variable "Motor" als Eingangsparameter des Bausteins "Motor\_Control".



Im Baustein "Motor\_Control" können Sie die einzelnen Komponenten der Variable adressieren.

| Adressierung | Erläuterung  |
|--------------|--|
| "Motor"      | Adressierung der übergeordneten PLC-Variable.                      |
| "Motor".On   | Adressierung einer Komponente einer strukturierten PLC-Variable.   |
| "Motor".On:P | Adressierung eines Peripherieeingangs oder -ausgangs (PE oder PA). |

## Regeln zur Verwendung strukturierter PLC-Variablen

Beachten Sie bei der Erstellung und Verwendung strukturierter PLC-Variablen folgende Regeln:

- Strukturierte PLC-Variablen sind in den Operandenbereichen "Eingänge" und "Ausgänge" verwendbar.
- Im Merkerbereich sind strukturierte PLC-Variablen nicht zulässig.
- Aus HMI können strukturierte PLC-Variablen nicht adressiert werden.

Beachten Sie bei der Erstellung des PLC-Datentyps, der als Basis für eine PLC-Variable dienen soll, folgende Regeln:

- Die Speicherbereiche der einzelnen Elemente dürfen sich nicht überlappen.  
Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 1189)
- Fassen Sie Ein- und Ausgänge nicht in einem PLC-Datentypen zusammen, sondern erzeugen Sie verschiedene PLC-Datentypen für Ein- und Ausgänge.
- Fassen Sie Ein- oder Ausgänge aus verschiedenen Baugruppen nicht in einem PLC-Datentyp zusammen, da nicht gewährleistet ist, dass die Prozessabbilder der Baugruppen synchron aktualisiert werden.
- In den unterlagerten PLC-Datentypen sind alle Datentypen mit Ausnahme des Datentyps "STRING" zulässig.

### Siehe auch

Strukturierte PLC-Variablen anlegen (Seite 1200)

## Strukturierte PLC-Variablen anlegen

### Regeln

Beachten Sie bei der Erstellung strukturierter PLC-Variablen folgende Regeln:

- Verwenden Sie getrennte PLC-Datentypen für die Operandenbereiche "Eingänge" und "Ausgänge".
- Im Merkerbereich sind strukturierte PLC-Variablen nicht zulässig.
- Fassen Sie Ein- oder Ausgänge aus verschiedenen Baugruppen nicht in einem PLC-Datentyp zusammen, da nicht gewährleistet ist, dass die Prozessabbilder der Baugruppen synchron aktualisiert werden.

## Vorgehen

Um eine strukturierte PLC-Variable zu erstellen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".  
Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Deklarieren Sie in dem PLC-Datentypen alle benötigten Komponenten. Alle Datentypen mit Ausnahme des Datentyps "STRING" sind zulässig.
3. Selektieren Sie in der Projektnavigation den PLC-Datentypen und wählen Sie aus dem Kontextmenü den Befehl "Übersetzen > Software (nur Änderungen)".  
Der PLC-Datentyp wird übersetzt und kann anschließend in der PLC-Variablentabelle verwendet werden.  
Auch wenn Sie Änderungen an bestehenden PLC-Datentypen durchführen, müssen Sie das Programm erneut übersetzen. Dadurch werden alle Verwendungsstellen des PLC-Datentyps aktualisiert.
4. Öffnen Sie eine PLC-Variablentabelle innerhalb derselben CPU.
5. Deklarieren Sie eine neue Variable oder wählen Sie eine bestehende Variable aus.
6. Wählen Sie in der Spalte "Datentyp" den PLC-Datentypen aus und weisen Sie ihn der PLC-Variable zu.  
Die PLC-Variable erhält die Struktur des PLC-Datentyps. Eine passende Adresse wird automatisch vergeben. Strukturierte PLC-Variablen beginnen immer an Wortadressen. In der Tabelle wird nur das oberste Strukturelement ohne seine Unterelemente dargestellt.

---

### Hinweis

#### Belegungsregeln und Defaultwerte

- Beachten Sie bei der Deklaration des PLC-Datentyps, dass die Speicherbereiche der einzelnen Variablen sich nicht überlappen dürfen. Zum Beispiel müssen Variablen vom Datentyp "Integer" an einer Wortgrenze beginnen. Fügen Sie, wenn nötig, "Füll-Variablen" ein, um Überlappungen zu vermeiden.  
Siehe auch: Zulässige Adressen und Datentypen von PLC-Variablen (Seite 1189)
- Es ist nicht möglich, Defaultwerte für die einzelnen Komponenten zu vergeben. Werte, die Sie in die Spalte "Defaultwert" eintragen, werden nicht ausgewertet. Variablen der Datentypen "DT" und "DTL" können deshalb möglicherweise ungültige Werte enthalten.

## Siehe auch

Wissenswertes zu strukturierten PLC-Variablen (Seite 1198)

### 9.1.2.7 Symbolische Konstanten deklarieren

#### Regeln für symbolische Konstanten

##### Zulässige Zeichen

Namen von symbolischen Konstanten dürfen aus folgenden Zeichen bestehen:

- Buchstaben, Ziffern, Sonderzeichen sind zulässig.
- Anführungszeichen sind nicht zulässig.

##### Eindeutige Konstantennamen

Die Namen der symbolischen Konstanten müssen CPU-weit eindeutig sein, auch wenn sich die Konstanten in unterschiedlichen Variablen Tabellen einer CPU befinden. Einen Namen, der bereits für einen Baustein, eine PLC-Variable oder eine andere Konstante innerhalb der CPU vergeben ist, können Sie nicht für eine neue Konstante verwenden. Groß- und Kleinschreibung wird bei der Prüfung der Eindeutigkeit nicht unterschieden.

Wenn Sie einen bereits vergebenen Namen erneut eintragen, wird automatisch an den zweiten Namen eine fortlaufende Nummer angehängt. Wenn Sie z. B. zwei Mal den Namen "Motor" eingeben, wird der zweite Eintrag in "Motor(1)" umgewandelt.

##### Zulässige Datentypen

Für Konstanten sind alle von der CPU unterstützten Datentypen zulässig, mit Ausnahme der strukturierten Datentypen.

##### Zulässige Werte

Als Konstantenwert können Sie einen beliebigen Wert aus dem Wertebereich des angegebenen Datentyps wählen. Angaben zu den Wertebereichen finden Sie im Kapitel "Datentypen".

Siehe auch: Auto-Hotspot

##### Siehe auch

Konstanten (Seite 1058)

Konstanten deklarieren (Seite 1202)

#### Konstanten deklarieren

##### Einführung

Konstanten deklarieren Sie im Register "Anwenderkonstanten" einer PLC-Variablen Tabelle. Bei der Deklaration ist für jede Konstante die Eingabe eines symbolischen Namens, eines



Datentyps und eines festen Werts erforderlich. Vom Datentyp der Konstanten hängen das Eingabeformat und der Wertebereich des Konstantenwerts ab.

Siehe auch: Auto-Hotspot

## Vorgehen

Um Konstanten zu deklarieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine PLC-Variablen-tabelle.
2. Öffnen Sie das Register "Anwenderkonstanten".  
Die Konstantentabelle wird geöffnet.
3. Tragen Sie in der Spalte "Name" einen Konstantennamen ein.
4. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
5. Tragen Sie in der Spalte "Wert" einen Konstantenwert ein, der für den gewählten Datentyp zulässig ist.
6. Tragen Sie in der Spalte "Kommentar" einen Kommentar ein, falls Sie Anmerkungen zu der Konstante haben. Die Eingabe eines Kommentars ist optional.
7. Falls Sie weitere Konstanten deklarieren wollen, setzen Sie die Einfügemarke in die nächste Zeile und wiederholen Sie die Schritte 3 bis 6.

## Syntaxprüfung

Nach jeder Eingabe wird automatisch eine Syntaxprüfung durchgeführt, die vorhandene Fehler rot anzeigt. Sie müssen diese Fehler nicht sofort beheben, sondern können mit der weiteren Bearbeitung fortfahren und die Korrekturen zu einem späteren Zeitpunkt durchführen. Solange die Variablendeklaration Syntaxfehler enthält und die Konstante im Programm verwendet wird, lässt sich das Programm allerdings nicht übersetzen.

## Siehe auch

PLC-Variablen-tabelle öffnen (Seite 1194)

Tabellenzeilen in der PLC-Variablen-tabelle einfügen (Seite 1207)

Aufbau der PLC-Variablen-tabellen (Seite 1187)

Regeln für symbolische Konstanten (Seite 1202)

Bearbeiten von Tabellen (Seite 216)

### 9.1.2.8 Eigenschaften bearbeiten

#### Eigenschaften von PLC-Variablen bearbeiten

#### Eigenschaften von PLC-Variablen

#### Überblick

Die folgende Tabelle zeigt eine Übersicht über die Eigenschaften von PLC-Variablen. Die Darstellung der Eigenschaften kann abhängig vom CPU-Typ variieren.

| Gruppe     | Eigenschaft         | Beschreibung   |
|------------|---------------------|--|
| Allgemein  | Name                | Innerhalb der CPU eindeutiger Name.  |
|            | Datentyp            | Datentyp der Variablen.  |
|            | Adresse             | Adresse der Variablen.   |
|            | Remanent            | Zeigt an, ob die Variable im remanenten Speicherbereich liegt.                         |
|            | Kommentar           | Kommentar zur Variablen.   |
| Verlauf    | Erstellt am         | Erstellzeitpunkt der Variablen (nicht änderbar).                                       |
|            | Zuletzt geändert am | Änderungszeitpunkt der Variablen (nicht änderbar).                                     |
| Verwendung | Sichtbar in HMI     | Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist. |
|            | Erreichbar aus HMI  | Zeigt an, ob HMI zur Laufzeit auf diese Variable zugreifen kann.                       |

#### Siehe auch

Eigenschaften von PLC-Variablen bearbeiten (Seite 1204)

#### Eigenschaften von PLC-Variablen bearbeiten

#### Eigenschaften in einer PLC-Variablen-tabelle bearbeiten

Um die Eigenschaften einer oder mehrerer Variablen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf die PLC-Variablen-tabelle, in der sich die Variablen befinden.  
Die PLC-Variablen-tabelle wird geöffnet.
2. Ändern Sie die Einträge in den Spalten.

## Adressen im Programmiereditor bearbeiten

Um die Adresse einer Variable im Programmiereditor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Variablennamen.
2. Wählen Sie im Kontextmenü den Befehl "Variable umverdrahten".  
Der Dialog "Variable umverdrahten" wird geöffnet. Er zeigt eine Deklarationstabelle.
3. Tragen Sie in der Spalte "Adresse" die neue Adresse ein.
4. Klicken Sie auf die Schaltfläche "Ändern", um die Eingabe zu bestätigen.

## Namen im Programmiereditor bearbeiten

Um den Namen einer Variablen im Programmiereditor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Variablennamen.
2. Wählen Sie im Kontextmenü den Befehl "Variable umbenennen".  
Der Dialog "Variable umbenennen" wird geöffnet. Er zeigt eine Deklarationstabelle.
3. Tragen Sie in der Spalte "Name" den neuen Namen ein.
4. Klicken Sie auf die Schaltfläche "Ändern", um die Eingabe zu bestätigen.

## Auswirkung im Programm

Bei einer Änderung des Namens, des Datentyps oder der Adresse einer Variablen wird automatisch jede Verwendungsstelle der Variablen im Programm aktualisiert.

## Siehe auch

Eigenschaften von PLC-Variablen (Seite 1204)

## Eigenschaften von symbolischen Konstanten bearbeiten

## Eigenschaften von Konstanten

## Überblick

Folgende Tabelle gibt eine Übersicht über die Eigenschaften von Konstanten:

| Gruppe    | Eigenschaft | Beschreibung                           |
|-----------|-------------|--|
| Allgemein | Name        | Innerhalb der Tabelle eindeutiger Name |
|           | Datentyp    | Datentyp der Konstanten                |

| Gruppe  | Eigenschaft         | Beschreibung   |
|---------|---------------------|--|
|         | Wert                | Wert, den Sie für die Konstante definiert haben.<br>Dieser Wert muss kompatibel zum angegebenen Datentyp sein.<br>Siehe auch: Auto-Hotspot |
|         | Kommentar           | Kommentar zur Konstanten   |
| Verlauf | Erstellt am         | Erstellzeitpunkt der Konstanten (nicht änderbar)   |
|         | Zuletzt geändert am | Änderungszeitpunkt der Konstanten (nicht änderbar)   |

### Eigenschaften von Konstanten bearbeiten

#### Eigenschaften in einer PLC-Variablen-tabelle bearbeiten

Um die Eigenschaften einer oder mehrerer Konstanten zu bearbeiten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf die PLC-Variablen-tabelle, in der sich die Konstanten befinden.  
Die PLC-Variablen-tabelle wird geöffnet.
2. Öffnen Sie das Register "Anwenderkonstanten".
3. Ändern Sie die Einträge in den Spalten "Name", "Datentyp", "Wert" oder "Kommentar".

#### Auswirkung im Programm

Bei einer Änderung des Namens, Datentyps oder Werts einer Konstanten wird automatisch jede Verwendungsstelle der Konstanten im Programm aktualisiert.

#### Siehe auch

Bearbeiten von Tabellen (Seite 216)

### 9.1.2.9 PLC-Variablen beobachten

#### PLC-Variablen beobachten

Sie können die Datenwerte, die die Variablen aktuell in der CPU annehmen, direkt in der PLC-Variablen-tabelle beobachten.

#### Voraussetzung

Eine Online-Verbindung zur CPU besteht.

## Vorgehen

Um die Datenwerte zu beobachten, gehen Sie folgendermaßen vor:

1. Öffnen Sie eine PLC-Variablentabelle.
2. Starten Sie die Beobachtung, indem Sie auf die Schaltfläche "Alle beobachten" klicken. Die zusätzliche Spalte "Beobachtungswert" wird in der Tabelle eingeblendet. Sie zeigt die aktuellen Datenwerte an.
3. Beenden Sie die Beobachtung, indem Sie erneut auf die Schaltfläche "Alle beobachten" klicken.

---

### Hinweis

Zusätzlich haben Sie die Möglichkeit, PLC-Variablen in eine Beobachtungs- oder Forcetable zu kopieren und sie dort zu beobachten, zu steuern oder zu forcen.

---

## Siehe auch

Aufbau der PLC-Variablentabellen (Seite 1187)

Einführung zum Testen mit der Beobachtungstabelle (Seite 1533)

Einführung zum Testen mit der Forcetable (Seite 1560)

Einträge in der PLC-Variablentabelle kopieren (Seite 1207)

## 9.1.2.10 PLC-Variablentabellen bearbeiten

### Tabellenzeilen in der PLC-Variablentabelle einfügen

#### Vorgehen

Um eine Zeile vor der selektierten Zeile einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile einfügen".

#### Ergebnis

Vor der markierten Zeile wird eine neue Zeile eingefügt.

#### Siehe auch

Bearbeiten von Tabellen (Seite 216)

### Einträge in der PLC-Variablentabelle kopieren

Sie können PLC-Variablen innerhalb einer Tabelle oder in andere Tabellen kopieren.

## Vorgehen

Um eine Variable zu kopieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Variable, die Sie kopieren möchten.  
Sie können auch mehrere Variablen selektieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Variable anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Positionieren Sie die Einfügemarke an der Stelle, an der Sie die Variable einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

Oder

1. Markieren Sie die Variable.
2. Halten Sie die linke Maustaste gedrückt.
3. Drücken Sie gleichzeitig die <Strg>-Taste.
4. Ziehen Sie die Variable an das Ziel.

## Ergebnis

- Die Variable wird an das Ziel kopiert.
- Bei einem Namenskonflikt wird der Variablenname automatisch um eine Nummer erweitert.  
Aus "Variable" wird zum Beispiel "Variable(1)".
- Alle weiteren Eigenschaften der Variablen bleiben gleich.

## Siehe auch

Bearbeiten von Tabellen (Seite 216)

## Einträge in der PLC-Variablen-tabelle löschen

### Vorgehen

Um eine Variable zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Variablen. Sie können auch mehrere Zeilen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

## Siehe auch

Bearbeiten von Tabellen (Seite 216)

## Zeilen in PLC-Variablen tabellen sortieren

Sie können die Zeilen in Tabellen alphanumerisch nach Name, Datentyp oder Adresse sortieren.

### Vorgehen

Um Tabellenzeilen zu sortieren, gehen Sie folgendermaßen vor:

1. Wählen Sie die Spalte aus, nach der Sie sortieren möchten.
2. Klicken Sie in den Spaltenkopf.  
Die Spalte wird aufsteigend sortiert.  
Ein Pfeil-nach-oben zeigt die Sortierreihenfolge an.
3. Um die Sortierreihenfolge zu ändern, klicken Sie auf den Pfeil.  
Die Spalte wird absteigend sortiert. Ein Pfeil-nach-unten zeigt die Sortierreihenfolge an.
4. Um die ursprüngliche Anordnung wiederherzustellen, klicken Sie ein drittes Mal auf den Spaltenkopf.

### Siehe auch

Bearbeiten von Tabellen (Seite 216)

## Zellen in der PLC-Variablen tabelle automatisch ausfüllen

Sie können den Inhalt einer oder mehrerer Tabellenzellen in die darunter liegenden Zellen übertragen und so aufeinander folgende Zellen automatisch ausfüllen.

Wenn Sie Zellen in der Spalte "Name" automatisch ausfüllen, wird jeder Name um eine fortlaufende Nummer erweitert. Aus dem Namen "Motor" wird z. B. der Name "Motor\_1".

Wenn Sie Zellen in der Spalte "Adresse" automatisch ausfüllen, werden die Adressen abhängig vom angegebenen Datentyp erhöht.

### Vorgehen

Um aufeinander folgende Zellen automatisch auszufüllen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu übertragenden Zellen.
2. Klicken Sie auf das Ausfüllzeichen in der rechten unteren Ecke der Zelle.  
Der Mauszeiger wird zu einem Fadenkreuz.
3. Halten Sie die Maustaste gedrückt und ziehen Sie den Mauszeiger nach unten über die Zellen, die Sie automatisch ausfüllen wollen.
4. Lassen Sie die Maustaste los.  
Die Zellen werden automatisch ausgefüllt. Wenn in den automatisch auszufüllenden Zellen bereits Einträge vorhanden sind, erscheint ein Dialog, in dem Sie angeben können, ob Sie die bestehenden Einträge überschreiben wollen, oder ob Sie für die neuen Variablen neue Zeilen einfügen möchten.

## Siehe auch

Bearbeiten von Tabellen (Seite 216)

## Tabellenspalten ein- und ausblenden

Sie können die Spalten in Tabellen bei Bedarf ein- und ausblenden.

## Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".  
Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.

## Siehe auch

Bearbeiten von Tabellen (Seite 216)

## PLC-Variablen mit externen Editoren bearbeiten

Um einzelne PLC-Variablen in externen Editoren außerhalb des TIA-Portals zu bearbeiten, können Sie sie per Copy & Paste exportieren und wieder importieren. Strukturierte Variablen können Sie allerdings nicht in einen Editor kopieren.

## Voraussetzung

Eine PLC-Variablen-tabelle und ein externer Editor sind geöffnet.

## Vorgehen

Um einzelne PLC-Variablen zu exportieren und wieder zu importieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere PLC-Variablen.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wechseln Sie in den externen Editor und fügen Sie die kopierten Variablen dort ein.
4. Bearbeiten Sie die Variablen nach Bedarf.
5. Kopieren Sie die Variablen im externen Editor.
6. Wechseln Sie wieder in die PLC-Variablen-tabelle.
7. Wählen Sie im Kontextmenü den Befehl "Einfügen".



**Hinweis**

Sie haben zusätzlich die Möglichkeit, PLC-Variablen als Massendaten zu exportieren oder zu importieren.

Siehe auch: Importieren und Exportieren (Seite 1586)

---

## 9.1.3 Bausteine anlegen und verwalten

### 9.1.3.1 Bausteine anlegen

#### Bausteinordner

#### Funktion

In der Projektnavigation finden Sie einen Ordner "Programmbausteine", in dem Sie folgende Bausteine anlegen und verwalten können:

- Organisationsbausteine (OB)
- Funktionsbausteine (FB)
- Funktionen (FC)
- Datenbausteine (DB)

Zusätzlich wird innerhalb des Ordners "Programmbausteine" ein Unterordner "Systembausteine" mit einem weiteren Unterordner "Programmressourcen" erstellt, wenn Sie zum ersten Mal eine Anweisung in Ihr Programm ziehen, die systemintern ein Funktionsbaustein ist. Dabei wird der Instanz-Datenbaustein des systeminternen Funktionsbausteins in den Ordner "Programmressourcen" eingefügt. Sie können solche Instanz-Datenbausteine aus dem Ordner "Programmressourcen" in jeden beliebigen Ordner verschieben, kopieren, umbenennen und löschen. Umgekehrt können Sie Ihre Bausteine auch in den Ordner "Programmressourcen" verschieben. Bausteine im Ordner "Programmressourcen", die für den Ablauf des Anwenderprogramms nicht erforderlich sind, werden beim nächsten Übersetzungsvorgang entfernt. Enthält der Ordner "Programmressourcen" dann keine Bausteine mehr, wird er zusammen mit dem Ordner "Systembausteine" ebenfalls gelöscht.

Für jedes Gerät wird automatisch ein Zyklus-OB erzeugt und in den Ordner "Programmbausteine" eingefügt.

## Siehe auch

Funktionen und Funktionsbausteine anlegen (Seite 1213)

Datenbausteine anlegen (Seite 1214)

Organisationsbausteine anlegen (Seite 1212)

Bausteine aus Bibliotheken verwenden (Seite 1215)

## Organisationsbausteine anlegen

### Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

### Vorgehen

Um einen Organisationsbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".  
Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Organisationsbaustein (OB)".
3. Wählen Sie den Typ des neuen Organisationsbausteins aus.
4. Geben Sie einen Namen für den neuen Organisationsbaustein ein.
5. Geben Sie die Eigenschaften des neuen Organisationsbausteins ein.
6. Um weitere Eigenschaften des neuen Organisationsbausteins einzugeben, klicken Sie auf "Weitere Informationen".  
Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
7. Geben Sie alle gewünschten Eigenschaften ein.
8. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Organisationsbaustein nach dem Erstellen gleich geöffnet werden soll.
9. Bestätigen Sie Ihre Eingaben mit "OK".

### Ergebnis

Der neue Organisationsbaustein wird erzeugt. Sie finden den Organisationsbaustein in der Projektnavigation im Ordner "Programmbausteine". Einige Organisationsbausteine können Sie nach dem Anlegen im Inspektorfenster oder in der Gerätesicht weiter parametrieren. Ob der neu angelegte Organisationsbaustein zusätzliche Parameter besitzt, können Sie in der Beschreibung des Organisationsbausteins nachlesen.

## Siehe auch

- Organisationsbausteine (OB) (Seite 1028)
- Bausteinordner (Seite 1211)
- Funktionen und Funktionsbausteine anlegen (Seite 1213)
- Datenbausteine anlegen (Seite 1214)
- Bausteine aus Bibliotheken verwenden (Seite 1215)
- Bausteintitel eingeben (Seite 1221)
- Bausteinkommentar eingeben (Seite 1222)

## Funktionen und Funktionsbausteine anlegen

### Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

### Vorgehen

Um eine Funktion (FC) oder einen Funktionsbaustein (FB) anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".  
Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Funktionsbaustein (FB)" oder "Funktion (FC)".
3. Geben Sie einen Namen für den neuen Baustein ein.
4. Geben Sie die Eigenschaften des neuen Bausteins ein.
5. Um weitere Eigenschaften des neuen Bausteins einzugeben, klicken Sie auf "Weitere Informationen".  
Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
6. Geben Sie alle gewünschten Eigenschaften ein.
7. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Baustein nach dem Erstellen gleich geöffnet werden soll.
8. Bestätigen Sie Ihre Eingaben mit "OK".

### Ergebnis

Der neue Baustein wird erzeugt. Sie finden den Baustein in der Projektnavigation im Ordner "Programmbausteine".

## Siehe auch

- Funktionsbausteine (FB) (Seite 1029)
- Funktionen (FC) (Seite 1028)
- Grundlagen zum Bausteinzugriff (Seite 1034)
- Bausteinordner (Seite 1211)
- Organisationsbausteine anlegen (Seite 1212)
- Datenbausteine anlegen (Seite 1214)
- Bausteine aus Bibliotheken verwenden (Seite 1215)
- Bausteintitel eingeben (Seite 1221)
- Bausteinkommentar eingeben (Seite 1222)

## Datenbausteine anlegen

### Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

### Vorgehen

Um einen Datenbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".  
Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".
3. Wählen Sie den Typ des Datenbausteins aus. Folgende Möglichkeiten stehen zur Verfügung:
  - Um einen globalen Datenbaustein anzulegen, wählen Sie den Listeneintrag "Global-DB".
  - Um einen ARRAY-Datenbaustein anzulegen, wählen Sie den Listeneintrag "ARRAY-DB".
  - Um einen Instanz-Datenbaustein anzulegen, wählen Sie aus der Liste den Funktionsbaustein aus, dem Sie den Instanz-Datenbaustein zuordnen möchten. In der Liste werden nur Funktionsbausteine angeboten, die zuvor für die CPU erstellt wurden.
  - Um einen Datenbaustein basierend auf einem PLC-Datentyp anzulegen, wählen Sie aus der Liste den PLC-Datentyp aus. In der Liste werden nur PLC-Datentypen angeboten, die zuvor für die CPU erstellt wurden.
  - Um einen Datenbaustein basierend auf einem Systemdatentyp anzulegen, wählen Sie aus der Liste den Systemdatentyp aus. In der Liste werden nur Systemdatentypen angeboten, die zuvor in Programmbausteinen der CPU eingefügt wurden.
4. Geben Sie einen Namen für den Datenbaustein ein.
5. Geben Sie die Eigenschaften des neuen Datenbausteins ein.

6. Wenn Sie als "Typ" einen ARRAY-DB ausgewählt haben, geben Sie den ARRAY-Datentyp und die obere ARRAY-Grenze ein.  
Die obere ARRAY-Grenze können Sie jederzeit im Eigenschaftsfenster des erzeugten Bausteins ändern. Der ARRAY-Datentyp ist nachträglich nicht mehr änderbar.
7. Um weitere Eigenschaften des neuen Datenbausteins einzugeben, klicken Sie auf "Weitere Informationen".  
Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
8. Geben Sie alle gewünschten Eigenschaften ein.
9. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Baustein nach dem Erstellen gleich geöffnet werden soll.
10. Bestätigen Sie Ihre Eingabe mit "OK".

## Ergebnis

Der neue Datenbaustein wird erzeugt. Sie finden den Datenbaustein in der Projektnavigation im Ordner "Programmbausteine".

## Siehe auch

- Globale Datenbausteine (DB) (Seite 1030)
- Instanz-Datenbausteine (Seite 1032)
- Bausteinordner (Seite 1211)
- Organisationsbausteine anlegen (Seite 1212)
- Funktionen und Funktionsbausteine anlegen (Seite 1213)
- Bausteine aus Bibliotheken verwenden (Seite 1215)
- Grundlagen zum Bausteinzugriff (Seite 1034)
- Systemdatentypen (Seite 1128)

## Bausteine aus Bibliotheken verwenden

Sie haben die Möglichkeit, Bausteine in der Projektbibliothek oder in einer globalen Bibliothek abzulegen, um sie in Ihrem Anwenderprogramm mehrfach zu verwenden. Dabei können Sie die Bausteine als Kopiervorlagen oder als Typen einfügen.

Siehe auch: Grundlagen zu Bibliotheken (Seite 313)

## Voraussetzung

- Die Task Card "Bibliotheken" ist eingeblendet.
- Für globale Bibliotheken ist kein Schreibschutz festgelegt.

### **Bausteine als Kopiervorlagen zur Projektbibliothek oder einer globalen Bibliothek hinzufügen**

Um der Projektbibliothek oder einer globalen Bibliothek neue Bausteine als Kopiervorlagen hinzuzufügen, gehen Sie folgendermaßen vor:

1. Maximieren Sie die Projektbibliothek oder die globale Bibliothek.
2. Ziehen Sie den Baustein, den Sie der Bibliothek hinzufügen möchten, per Drag & Drop in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen" der Projektbibliothek oder einer globalen Bibliothek. Lassen Sie die linke Maustaste erst los, wenn unter dem Mauszeiger ein kleines Plusymbol erscheint.

Oder:

1. Kopieren Sie das Element, das Sie als Kopiervorlage hinzufügen möchten.
2. Maximieren Sie die Projektbibliothek oder die globale Bibliothek.
3. Klicken Sie mit der rechten Maustaste auf den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen".
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

### **Bausteine als Typen zur Projektbibliothek oder einer globalen Bibliothek hinzufügen**

Um der Projektbibliothek oder einer globalen Bibliothek neue Bausteine als Typen hinzuzufügen, gehen Sie folgendermaßen vor:

1. Maximieren Sie die Projektbibliothek oder die globale Bibliothek.
2. Ziehen Sie das Element, das Sie als Typ hinzufügen möchten, mit Drag & Drop in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen" der Projektbibliothek oder einer globalen Bibliothek. Lassen Sie die linke Maustaste erst los, wenn unter dem Mauszeiger ein kleines Plusymbol erscheint.

Oder:

1. Kopieren Sie das Element, das Sie als Typ hinzufügen möchten.
2. Maximieren Sie die Projektbibliothek oder die globale Bibliothek.
3. Klicken Sie mit der rechten Maustaste auf den Ordner "Typen" oder einen beliebigen Unterordner von "Typen".
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

### **Bausteine der Projektbibliothek oder einer globalen Bibliothek verwenden**

Um einen Baustein aus der Projektbibliothek oder einer globalen Bibliothek in Ihrem Projekt zu verwenden, gehen Sie folgendermaßen vor:

1. Maximieren Sie die Projektbibliothek oder die globale Bibliothek, sodass Sie den Baustein sehen können, den Sie verwenden möchten.
2. Ziehen Sie den Baustein per Drag & Drop in den Bausteinordner der CPU. Nicht erlaubte Einfügestellen werden dadurch gekennzeichnet, dass sich der Mauszeiger in einen durchgestrichenen Kreis verwandelt.

**Hinweis**

Wenn Sie von einem Typ aus einer globalen Bibliothek eine Instanz ableiten, wird der Typ zusätzlich in die Projektbibliothek eingefügt. Die Instanz ist anschließend nur mit dem Typ in der Projektbibliothek verbunden.

---

**Siehe auch**

Bibliotheken verwenden (Seite 313)

**Bausteine kopieren und einfügen****Grundlagen zum Kopieren und Einfügen von Bausteinen****Funktion**

Sie können neue Bausteine auch dadurch anlegen, dass Sie bestehende Bausteine kopieren und die Kopie einfügen. Beachten Sie dabei folgende Prinzipien beim Kopieren in CPUs der gleichen Gerätefamilie:

- Organisationsbausteine (OBs), Funktionen (FCs), Funktionsbausteine (FBs) und globale Datenbausteine (DBs) können Sie uneingeschränkt kopieren.
- Instanz-Datenbausteine können Sie nur für den gleichen Funktionsbaustein kopieren, da Sie die Zuordnung zum Funktionsbaustein nicht nachträglich ändern können. Die Zuordnung wird jedoch aufgehoben, wenn Sie den Instanz-Datenbaustein in eine andere CPU kopieren. Falls es dort einen Funktionsbaustein mit dem gleichen Namen gibt, wird der Instanz-Datenbaustein diesem Funktionsbaustein zugeordnet. Wenn Sie den Instanz-Datenbaustein zusammen mit dem Funktionsbaustein in die andere CPU kopieren, wird der Instanz-Datenbaustein der Kopie des Funktionsbausteins zugeordnet.

Die unterschiedlichen Gerätefamilien unterstützen zum Teil unterschiedliche Bausteine, vor allem bei den Organisationsbausteinen. Aber auch Funktionsbausteine und Funktionen können auf den verschiedenen Geräten mit unterschiedlichen Zugriffsarten programmiert werden. Daher werden nicht auf allen Geräten alle Bausteine unterstützt. Beachten Sie beim Kopieren in eine andere Gerätefamilie folgende Prinzipien:

- Kopieren in eine S7-1200-CPU:
  - Organisationsbausteine mit der Zugriffsart "optimiert" können in eine S7-1200 kopiert werden. Falls der kopierte OB-Typ von der S7-1200-CPU unterstützt wird, behält der kopierte OB die Eigenschaften seines Ereignisses. Sie müssen ihn jedoch neu übersetzen.
  - Organisationsbausteine mit der Zugriffsart "standard" können Sie zwar auf eine S7-1200 kopieren, sie werden von der CPU jedoch nicht unterstützt.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) mit der Zugriffsart "optimiert" können in eine S7-1200 kopiert werden. Sie müssen anschließend neu übersetzt werden.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) mit der Zugriffsart "standard" können Sie zwar in eine S7-1200 kopieren, sie werden von der CPU jedoch nicht unterstützt.
  - Instanz-Datenbausteine: Falls es in der Ziel-CPU einen Funktionsbaustein mit dem Namen gibt, dem der Instanz-Datenbaustein in der Quell-CPU zugeordnet war, wird der Instanz-Datenbaustein dem Funktionsbaustein in der Ziel-CPU zugeordnet. Wenn Sie den Instanz-Datenbaustein zusammen mit dem Funktionsbaustein, dem er in der Quell-CPU zugeordnet war, in die Ziel-CPU kopieren, wird der Instanz-Datenbaustein der Kopie des Funktionsbausteins zugeordnet.
- Kopieren in eine S7-1500-CPU:
  - Organisationsbausteine mit der Zugriffsart "optimiert" können in eine S7-1500 kopiert werden. Falls der kopierte OB-Typ von der S7-1500-CPU unterstützt wird, behält der kopierte OB die Eigenschaften seines Ereignisses. Sie müssen ihn jedoch neu übersetzen. Nicht unterstützte OB-Typen erhalten ein Parkverbotssymbol.
  - Organisationsbausteine mit der Zugriffsart "standard" können in eine S7-1500 kopiert werden. Wenn der OB aus einer S7-300/400-CPU stammt, erhält er das Standardereignis des entsprechenden OB-Typs. Wenn der OB aus einer S7-1200/1500-CPU stammt, behält er die Eigenschaften seines Ereignisses, er muss jedoch neu übersetzt werden.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) mit der Zugriffsart "optimiert" können in eine S7-1500 kopiert werden. Sie müssen anschließend neu übersetzt werden.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) mit der Zugriffsart "standard" können Sie zwar in eine S7-1500 kopieren, sie werden von der CPU jedoch nicht unterstützt.
  - Instanz-Datenbausteine: Falls es in der Ziel-CPU einen Funktionsbaustein mit dem Namen gibt, dem der Instanz-Datenbaustein in der Quell-CPU zugeordnet war, wird der Instanz-Datenbaustein dem Funktionsbaustein in der Ziel-CPU zugeordnet. Wenn Sie den Instanz-Datenbaustein zusammen mit dem Funktionsbaustein, dem er in der Quell-



CPU zugeordnet war, in die Ziel-CPU kopieren, wird der Instanz-Datenbaustein der Kopie des Funktionsbausteins zugeordnet.

- Kopieren in S7-300/400-CPU:
  - Organisationsbausteine können zwischen S7-300 und S7-400 beliebig kopiert werden.
  - Organisationsbausteine aus S7-1200/1500-CPU können Sie zwar in S7-300/400-CPU kopieren, sie werden von der Ziel-CPU jedoch nicht unterstützt.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) können zwischen S7-300 und S7-400 beliebig kopiert werden.
  - Funktionsbausteine (FBs), Funktionen (FCs) und globale Datenbausteine (DBs) aus S7-1200/1500-CPU können Sie zwar in S7-300/400-CPU kopieren, sie werden von der Ziel-CPU jedoch nicht unterstützt.
  - Instanz-Datenbausteine: Falls es in der Ziel-CPU einen Funktionsbaustein mit dem Namen gibt, dem der Instanz-Datenbaustein in der Quell-CPU zugeordnet war, wird der Instanz-Datenbaustein dem Funktionsbaustein in der Ziel-CPU zugeordnet. Wenn Sie den Instanz-Datenbaustein zusammen mit dem Funktionsbaustein, dem er in der Quell-CPU zugeordnet war, in die Ziel-CPU kopieren, wird der Instanz-Datenbaustein der Kopie des Funktionsbausteins zugeordnet.

Nicht unterstützte Bausteine erkennen Sie in der Projektnavigation am Parkverbotssymbol. Bausteine mit einem Parkverbotssymbol können Sie nicht bearbeiten, sondern nur erneut als Kopierquellen nutzen.

## Kopierte Daten

Beim Einfügen werden alle Daten des Bausteins kopiert und an die Kopie weiter gegeben. Zu diesen Daten gehören:

- Variablen der Bausteinschnittstelle
- Alle Netzwerke
- Kommentare in allen vorhandenen Übersetzungen
- Im Baustein definierte Meldungen
- Der gesamte Programmcode des kopierten Bausteins mit darin befindlichen Aufrufanweisungen.  
Aufgerufene Bausteine und zugehörige Instanz-Datenbausteine werden jedoch nicht kopiert.

## Vermeidung von Namenskonflikten beim Einfügen

Beim Einfügen von kopierten Bausteinen mit identischen Namen wie bereits bestehende Bausteine werden folgende Mechanismen angewendet, um Namenskonflikte zu vermeiden:

- Einfügen des kopierten Bausteins in die gleiche CPU:  
Die Kopie des Bausteins erhält einen Namen, der um eine Nummer erweitert ist. Wenn z. B. Baustein "A" kopiert wird, ist ein möglicher Name der Kopie "A\_1". Dabei wird als Nummer kein fortlaufendes Nummernband verwendet, sondern immer die kleinste freie Nummer. Die Kopie von Baustein "A" kann also auch den Namen "A\_25" erhalten, falls keine niedrigere Nummer frei ist.
- Einfügen des kopierten Bausteins in eine andere CPU:  
Ein Dialog wird geöffnet, in dem Sie auswählen können, ob der Baustein mit dem gleichen Namen ersetzt oder ob der kopierte Baustein mit einer Duplikatskennung (Name\_Nummer) eingefügt werden soll.

---

### Hinweis

#### Nummernkonflikte

Es können jedoch Nummernkonflikte auftreten, wenn der eingefügte Baustein die gleiche Bausteinnummer wie ein bereits vorhandener Baustein hat. Die Bausteinnummer wird beim Einfügen nicht automatisch verändert. Dies kann z. B. Auswirkungen auf Bausteinaufrufe haben. Prüfen Sie daher beim Kopieren von Bausteinen die Bausteinnummer sorgfältig und korrigieren Sie doppelte Bausteinnummern manuell über die Eigenschaften des Bausteins. Doppelte Bausteinnummern führen zu einem Übersetzungsfehler.

---

## Siehe auch

Bausteine kopieren (Seite 1220)

Bausteine einfügen (Seite 1221)

## Bausteine kopieren

### Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

### Vorgehen

Um einen Baustein zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

## Ergebnis

Eine Kopie des Bausteins befindet sich nun in der Zwischenablage und kann entweder in die gleiche oder eine andere CPU eingefügt werden.

## Siehe auch

Grundlagen zum Kopieren und Einfügen von Bausteinen (Seite 1217)

Bausteine einfügen (Seite 1221)

## Bausteine einfügen

### Voraussetzung

Sie haben einen Baustein kopiert.

### Vorgehen

Um einen kopierten Baustein und seine Daten in eine CPU einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation die Ordnerstruktur für die CPU, in die Sie den kopierten Baustein einfügen möchten.

---

#### Hinweis

Beachten Sie, dass Sie den kopierten Baustein nur in eine CPU einfügen können, die die Programmiersprache und den Typ des Bausteins unterstützt.

---

2. Klicken Sie mit der rechten Maustaste auf den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".
  - Falls Sie den Baustein in die CPU des Originalbausteins einfügen, wird die Kopie mit der Namensweiterung "\_<fortlaufende Nummer>" eingefügt.
  - Falls Sie den Baustein in eine andere CPU einfügen und es einen Baustein mit diesem Namen schon gibt, wird der Dialog "Einfügen" geöffnet. Wählen Sie dann die gewünschte Option und bestätigen Sie Ihre Auswahl mit "OK".

## Siehe auch

Grundlagen zum Kopieren und Einfügen von Bausteinen (Seite 1217)

Bausteine kopieren (Seite 1220)

## Bausteintitel eingeben

Der Bausteintitel ist die Überschrift des Bausteins. Er ist nicht identisch mit dem Bausteinnamen, der beim Anlegen des Bausteins vergeben wurde. Die Länge des

Bausteintitels ist auf eine Zeile beschränkt. Sie können den Bausteintitel für geöffnete und geschlossene Bausteine eingeben.

### Voraussetzung

Ein Codebaustein ist vorhanden.

### Bausteintitel für geöffnete Bausteine eingeben

Um in einem geöffneten Baustein den Bausteintitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie im Programmiereditor in die Titelzeile des Bausteins.
2. Geben Sie den Bausteintitel ein.

### Bausteintitel für geschlossene Bausteine eingeben

Um in einem geschlossenen Baustein den Bausteintitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den Baustein mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Ein Dialog mit den Eigenschaften des Bausteins wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Information".
4. Geben Sie im Eingabefeld "Titel" den Bausteintitel ein.
5. Bestätigen Sie Ihre Eingabe mit "OK".

### Siehe auch

Organisationsbausteine anlegen (Seite 1212)

Funktionen und Funktionsbausteine anlegen (Seite 1213)

Bausteinkommentar eingeben (Seite 1222)

### Bausteinkommentar eingeben

Mithilfe des Bausteinkommentars können Sie den gesamten Codebaustein dokumentieren. Sie können z. B. den Zweck des Bausteins angeben oder auf Besonderheiten hinweisen. Sie können den Bausteinkommentar für geöffnete und geschlossene Bausteine eingeben.

### Voraussetzung

Ein Codebaustein ist vorhanden.

### Bausteinkommentar für geöffnete Bausteine eingeben

Um in einem geöffneten Baustein einen Bausteinkommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den kleinen Pfeil vor dem Bausteintitel.  
Aus dem Pfeil-nach-rechts wird ein Pfeil-nach-unten und der Kommentarbereich wird angezeigt.
2. Klicken Sie im Kommentarbereich auf "Kommentar".  
Die Textstelle "Kommentar" wird markiert.
3. Geben Sie den Bausteinkommentar ein.

### Bausteinkommentare für geschlossene Bausteine eingeben

Um für einen geschlossenen Baustein einen Bausteinkommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den Baustein mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Ein Dialog mit den Eigenschaften des Bausteins wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Information".
4. Geben Sie im Eingabefeld "Kommentar" den Bausteinkommentar ein.
5. Bestätigen Sie Ihre Eingabe mit "OK".

### Siehe auch

Organisationsbausteine anlegen (Seite 1212)

Funktionen und Funktionsbausteine anlegen (Seite 1213)

Bausteintitel eingeben (Seite 1221)

## 9.1.3.2 Bausteineigenschaften festlegen

### Grundlagen zu Bausteineigenschaften

#### Bausteineigenschaften

Jeder Baustein besitzt bestimmte Eigenschaften, die Sie anzeigen und bearbeiten können. Diese Eigenschaften haben z. B. folgende Aufgaben:

- Identifikation des Bausteins
- Anzeigen des Speicherbedarfs und des Übersetzungsstatus des Bausteins
- Anzeigen der Zeitstempel
- Anzeigen der Referenzinformationen
- Einstellen des Zugriffsschutzes

**Siehe auch**

- Übersicht über die Bausteineigenschaften (Seite 1224)
- Zeitstempel von Bausteinen (Seite 1227)
- Bausteineigenschaften anzeigen und bearbeiten (Seite 1229)
- Einstellen der Mnemonik (Seite 1246)

**Übersicht über die Bausteineigenschaften**

**Übersicht**

Die Eigenschaften der Bausteine sind baustein- und CPU-spezifisch. Daher sind nicht alle Eigenschaften für alle Bausteine und in allen CPU-Familien verfügbar. Die folgende Tabelle zeigt eine Übersicht der Bausteineigenschaften:

| Gruppe      | Eigenschaft             | Beschreibung  |
|-------------|-------------------------|---|
| Allgemein   | Name                    | Innerhalb der Station eindeutiger Bausteinname  |
|             | Konstantenname          | Name der Konstante, die für den OB in die PLC-Variablen-tabelle eingefügt wird  |
|             | Typ                     | Bausteintyp (nicht änderbar)  |
|             | Nummer                  | Bausteinnummer  |
|             | Ereignisklasse          | Ereignisklasse eines OBs (nicht änderbar)   |
|             | Sprache                 | Programmiersprache des Bausteins  |
|             | Sprache in Netzwerken   | Sprache, die zur Programmierung von Bedingungen in GRAPH-Bausteinen verwendet wird.   |
|             | Teilprozessabbildnummer | Anzeige der Teilprozessabbilder, die dem Organisationsbaustein zugeordnet sind (nicht änderbar)   |
|             | ARRAY-Datentyp          | Datentyp eines ARRAY-Datenbausteins (nicht änderbar)  |
|             | ARRAY-Grenze            | Obere Grenze eines ARRAY-Datenbausteins<br>Möglichkeiten zur Adressierung von ARRAY-Datenbausteins bietet die Task Card "Anweisungen" im Abschnitt "Verschieben". |
| Information | Titel                   | Bausteintitel   |
|             | Kommentar               | Bausteinkommentar   |
|             | Version                 | Versionsnummer des Bausteins  |
|             | Familie                 | Name der Bausteinfamilie  |
|             | Autor                   | Name des Autors, Firmenname, Abteilungsname oder andere Namen   |
|             | Anwenderdefinierte ID   | Vom Anwender erzeugte ID  |
| Zeitstempel | Baustein                | Erstell- und Änderungszeitpunkt des Bausteins (nicht änderbar)  |
|             | Schnittstelle           | Änderungszeitpunkt für die Schnittstelle des Bausteins (nicht änderbar)   |
|             | Code                    | Änderungszeitpunkt für Code (nicht änderbar)  |
|             | Daten                   | Änderungszeitpunkt für Daten (nicht änderbar)   |
| Übersetzung | Status                  | Details zum letzten Kompilervorgang (nicht änderbar)  |
|             | Längen                  | Details zu den Bausteinlängen (nicht änderbar)  |

| Gruppe    | Eigenschaft  | Beschreibung  |
|-----------|--|---|
| Schutz    | Schutz   | Know-how-Schutz und Kopierschutz für den Baustein einstellen<br>Siehe auch: Bausteine schützen  |
| Attribute | Optimierter Bausteinzugriff                            | Bei Bausteinen mit optimiertem Zugriff enthält die Variablendeklaration nur die symbolischen Namen der Datenelemente. Die Adressen werden vom System automatisch optimiert und verwaltet. Die Performance der CPU wird gesteigert, Zugriffsfehler, z. B. aus SIMATIC-HMI, können nicht auftreten.<br>Siehe auch: Auto-Hotspot |
|           | IEC-Prüfung  | Die Kompatibilität von Operanden in Vergleichsoperationen und arithmetischen Operationen wird gemäß IEC 61131 geprüft. Nicht kompatible Operanden müssen Sie explizit konvertieren.<br>Siehe auch: Übersicht über die Datentypkonvertierung (Seite 1133)  |
|           | Lokale Fehlerbehandlung im Baustein                    | Fehlerbehandlung innerhalb des Bausteins mit der Anweisung GET_ERROR bzw. GET_ERR_ID (nicht änderbar).<br>Siehe auch: Auto-Hotspot  |
|           | Erweiterte Statusinformation erstellen                 | Ermöglicht das Beobachten aller Variablen in einem SCL-Baustein. Der Speicherbedarf des Programms und die Ablaufzeiten erhöhen sich jedoch durch diese Option.  |
|           | ARRAY-Grenzen prüfen                                   | Prüft zur Laufzeit eines SCL-Bausteins, ob Feldindizes in dem für ein ARRAY deklarierten Bereich liegen. Wenn ein Feldindex außerhalb des zulässigen Bereichs liegt, wird der Freigabeausgang ENO des Bausteins auf "0" gesetzt.  |
|           | ENO automatisch setzen                                 | Prüft zur Laufzeit eines SCL-Bausteins, ob Fehler bei der Bearbeitung bestimmter Anweisungen auftreten. Wenn ein Laufzeitfehler auftritt, wird der Freigabeausgang ENO des Bausteins auf "0" gesetzt.   |
|           | Speicherplatzminimierten DB erzeugen                   | Erzeugt Instanz-Datenbausteine für GRAPH-Bausteine der S7-300 und S7-400 in minimiertem Format. Diese Option reduziert den Speicherbedarf des GRAPH-FBs erheblich, allerdings erhalten Sie nur eingeschränkte Programmstatus-Informationen.   |
|           | Schritte überspringen                                  | Wenn in einem GRAPH-Baustein die Transitionen vor und nach einem Schritt gleichzeitig gültig werden, wird der Schritt nicht aktiv und damit übersprungen.   |
|           | Quittierpflicht bei Supervisionsfehlern                | Tritt während des Betriebs eines GRAPH-Bausteins ein Supervisionsfehler auf, muss er quittiert werden, bevor das Programm weiter ausgeführt wird.   |
|           | Permanente Bearbeitung aller Interlocks im Handbetrieb | Bewirkt, dass in einem GRAPH-Baustein Interlock-Bedingungen im Handbetrieb permanent überprüft werden.  |
|           | Sperre der Betriebsartenwahl                           | Verhindert die Auswahl einer Betriebsart eines GRAPH-Bausteins.   |
|           | Datenbaustein im Gerät schreibgeschützt                | Zeigt an, ob der Datenbaustein im Zielsystem schreibgeschützt ist und während des Programmablaufs nicht überschrieben werden kann (nur für Datenbausteine)  |

9.1 Anwenderprogramm erstellen

| Gruppe                       | Eigenschaft   | Beschreibung  |
|------------------------------|---|---|
|                              | Nur im Ladespeicher ablegen                                     | Bei Aktivierung wird der Datenbaustein nur im Ladespeicher abgelegt, belegt keinen Platz im Arbeitsspeicher und wird nicht in das Programm eingebunden. Möglichkeiten zum Transferieren der Datenbausteine in den Arbeitsspeicher bietet die Task Card "Anweisungen" im Abschnitt "Verschieben" (nur für Datenbausteine). |
|                              | Startinformation  | Bei S7-1500-CPU's legen Sie hier die Struktur der Startinformation des Organisationsbausteins fest: entweder wie bei S7-300 und S7-400-CPU's oder optimierte Startinformation.  |
|                              | Priorität   | Zeigt bei Organisationsbausteinen die eingestellte Priorität an. Ob Sie die Priorität ändern können, hängt von der verwendeten CPU-Familie und von der Art des Organisationsbausteins ab.   |
|                              | Parameterversorgung über Registers                              | Ermöglicht in einem AWL-Baustein der S7-1500 die Parameterversorgung über Register. Dadurch können die Anweisungen "Bausteine bedingt aufrufen" (CC) und "Bausteine unbedingt aufrufen" (UC) im Baustein verwendet werden.  |
|                              | Baustein als know-how-geschütztes Bibliothekselement verwendbar | Zeigt an, ob der Baustein in der Bibliothek mit Know-how-Schutz verwendet werden kann.  |
|                              | Rücklesbarkeit aktivieren                                       | Ermöglicht es ihnen, einzelne Parameter des Bausteins als "rücklesbar" zu kennzeichnen. Die Funktion "Variablen zurücklesen" ist relevant, wenn der Baustein in einem CFC-Plan genutzt wird.  |
|                              | Bausteindarstellung   | Legt fest, wie der Baustein in einem CFC-Plan dargestellt wird.   |
| Uhrzeitalarm                 | Uhrzeitalarm  | Parameter des Uhrzeitalarm-OB: aktiv (ja oder nein), Ausführung, Startdatum und Uhrzeit, Lokalzeit bzw. Systemzeit  |
| Weckalarm                    | Weckalarm   | Zykluszeit und Phasenverschiebung des Weckalarm-OB  |
| Startereignisse              | Startereignisse   | Anzeige der Startereignisse des Prozessalarm-OB   |
| Taktsynchronität             | Taktsynchronität  | Parameter des Taktsynchronalarm-OB ein: Applikationszyklus, Automatische Einstellung (ja oder nein), Verzögerungszeit.<br>Außerdem wird das PROFINET IO-System bzw. DP-Mastersystem angezeigt, dessen IO-Devices bzw. DP-Slaves dem Taktsynchronalarm-OB zugeordnet sind.   |
| Laden ohne Reinitialisierung | Reserve im Standardspeicher                                     | Größe der Reserve im Standardspeicher, die für Schnittstellenerweiterungen genutzt werden kann.<br>Die Anzahl der aktuell verfügbaren Bytes wird in Klammern angegeben. Diese Angabe wird bei jedem Übersetzungsvorgang aktualisiert.   |
|                              | Laden ohne Reinitialisierung für remanente Variablen aktivieren | Ermöglicht die Definition einer Reserve im remanenten Speicher.   |
|                              | Reserve im remanenten Speicher                                  | Größe der Reserve im remanenten Speicher, die für Schnittstellenerweiterungen genutzt werden kann.<br>Die Anzahl der aktuell verfügbaren Bytes wird in Klammern angegeben. Diese Angabe wird bei jedem Übersetzungsvorgang aktualisiert.  |



## Siehe auch

- Grundlagen zu Bausteineigenschaften (Seite 1223)
- Zeitstempel von Bausteinen (Seite 1227)
- Bausteineigenschaften anzeigen und bearbeiten (Seite 1229)
- Grundlagen zum Bausteinzugriff (Seite 1034)

## Zeitstempel von Bausteinen

### Einführung

Bausteine erhalten verschiedene Zeitstempel, an denen Sie erkennen können, wann der Baustein erstellt und wann er zuletzt geändert wurde. Diese Zeitstempel werden auch für die automatische Konsistenzprüfung vor einem Übersetzungsvorgang verwendet.

### Zeitstempel für Codebausteine

Für Codebausteine (OBs, FBs, FCs) werden folgende Zeitstempel erzeugt:

- Baustein: Erstellungsdatum, Datum der letzten Änderung
- Schnittstelle: Datum der letzten Änderung
- Code/Daten: Datum der letzten Änderung

Beim Übersetzen wird ein Zeitstempelkonflikt angezeigt, wenn der Zeitstempel für den aufrufenden Baustein älter ist als der Zeitstempel für die Schnittstelle des aufgerufenen Bausteins.

Die Zeitstempel für Codebausteine werden folgendermaßen aktualisiert:

- Baustein: Der Zeitstempel für die letzte Änderung des Bausteins entspricht immer dem Zeitstempel der Schnittstelle oder des Codes, abhängig davon, welcher Bereich zuletzt geändert wurde.
- Schnittstelle: Der Zeitstempel der Schnittstelle wird bei jeder Änderung der Schnittstelle aktualisiert. Auch wenn Sie die Änderung der Schnittstelle manuell rückgängig machen, z. B. den Namen erneut ändern, ist das eine Änderung, die den Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, wird der Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.
- Code/Daten: Der Zeitstempel des Codes wird bei jeder Änderung des Baustein-codes aktualisiert. Auch wenn Sie die Änderung des Codes manuell rückgängig machen, z. B. eine Anweisung wieder entfernen, ist das eine Änderung, die den Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, wird der Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.

### Zeitstempel für globale Datenbausteine

Für globale Datenbausteine werden folgende Zeitstempel erzeugt:

- Baustein: Erstellungsdatum, Datum der letzten Änderung
- Schnittstelle: Datum der letzten Änderung
- Daten: Datum der letzten Änderung

Beim Übersetzen eines globalen Datenbausteins, der auf einem PLC-Datentyp basiert, wird ein Zeitstempelkonflikt angezeigt, wenn der Zeitstempel des globalen Datenbausteins älter als der Zeitstempel für den verwendeten PLC-Datentyp ist.

Die Zeitstempel für globale Datenbausteine werden folgendermaßen aktualisiert:

- Baustein: Der Zeitstempel für die letzte Änderung eines globalen Datenbausteins entspricht immer des Zeitstempels für die Schnittstelle und der Daten.
- Schnittstelle und Daten: Die Zeitstempel der Schnittstelle und der Daten werden bei jeder Änderung des globalen Datenbausteins aktualisiert. Auch wenn Sie die Änderung manuell rückgängig machen, z. B. eine Variable wieder entfernen, ist das eine Änderung, die die Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, werden die Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.

### Zeitstempel für Instanz-Datenbausteine

Für Instanz-Datenbausteine werden folgende Zeitstempel erzeugt:

- Baustein: Erstellungsdatum, Datum der letzten Änderung
- Schnittstelle: Datum der letzten Änderung
- Daten: Datum der letzten Änderung

Beim Übersetzen eines Instanz-Datenbausteins wird ein Zeitstempelkonflikt angezeigt, wenn die Zeitstempel für die Schnittstellen des Instanz-Datenbausteins und des Funktionsbausteins nicht identisch sind.

Die Zeitstempel für Instanz-Datenbausteine werden folgendermaßen aktualisiert:

- Baustein: Der Zeitstempel für die letzte Änderung eines Instanz-Datenbausteins entspricht immer des Zeitstempels für die Schnittstelle und der Daten.
- Schnittstelle und Daten: Die Zeitstempel der Schnittstelle und der Daten werden bei jeder Änderung des Instanz-Datenbausteins aktualisiert. Auch wenn Sie die Änderung manuell rückgängig machen, z. B. die Remanenz einer Variablen wieder aufheben, ist das eine Änderung, die die Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, werden die Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.

### Zeitstempel für PLC-Datentypen

Für PLC-Datentypen werden folgende Zeitstempel erzeugt:

- Baustein: Erstellungsdatum, Datum der letzten Änderung
- Schnittstelle: Datum der letzten Änderung

Die Zeitstempel für PLC-Datentypen werden folgendermaßen aktualisiert:

- Baustein: Der Zeitstempel für die letzte Änderung eines PLC-Datentyps entspricht immer dem Zeitstempel der Schnittstelle.
- Schnittstelle: Der Zeitstempel der Schnittstelle wird bei jeder Änderung des PLC-Datentyps aktualisiert. Auch wenn Sie die Änderung manuell rückgängig machen, z. B. Inhalte des PLC-Datentyps wieder löschen, ist das eine Änderung, die den Zeitstempel aktualisiert. Wenn Sie die Änderung jedoch über die Funktion "Rückgängig" rückgängig machen, wird der Zeitstempel auf den Wert vor der rückgängig gemachten Änderung zurückgesetzt.

## Siehe auch

Grundlagen zu Bausteineigenschaften (Seite 1223)

Übersicht über die Bausteineigenschaften (Seite 1224)

Bausteineigenschaften anzeigen und bearbeiten (Seite 1229)

Grundlagen zum Übersetzen von Bausteinen (Seite 1468)

## Bausteineigenschaften anzeigen und bearbeiten

Die Eigenschaften der Bausteine sind baustein- und CPU-spezifisch. Daher sind nicht alle Eigenschaften für alle Bausteine und in allen CPU-Familien verfügbar. Eigenschaften, die Sie nur anzeigen können, sind schreibgeschützt.

## Eigenschaften eines geschlossenen Bausteins anzeigen und bearbeiten

Um die Eigenschaften eines geschlossenen Bausteins anzuzeigen oder zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Klicken Sie mit der rechten Maustaste auf den Baustein, dessen Eigenschaften Sie anzeigen oder bearbeiten möchten.
3. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Der Eigenschaftsdialog des Bausteins wird geöffnet.
4. Klicken Sie in der Bereichsnavigation auf eine Gruppe, deren Eigenschaften Sie anzeigen oder bearbeiten möchten.
5. Ändern Sie die entsprechende Eigenschaft.
6. Bestätigen Sie Ihre Eingaben mit "OK".

### Eigenschaften eines geöffneten Bausteins anzeigen und bearbeiten

Um die Eigenschaften eines geöffneten Bausteins anzuzeigen oder zu bearbeiten, gehen Sie folgendermaßen vor:

1. Aktivieren Sie im Menü "Ansicht" die Option "Inspektorfenster".  
Das Inspektorfenster wird geöffnet.
2. Klicken Sie auf das Register "Eigenschaften".  
Die Eigenschaften des Bausteins werden im Register "Eigenschaften" des Inspektorfensters angezeigt.
3. Klicken Sie in der Bereichsnavigation auf eine Gruppe, deren Eigenschaften Sie anzeigen oder bearbeiten möchten.
4. Ändern Sie die entsprechende Eigenschaft.

### Ergebnis

Die Eigenschaften des Bausteins werden geändert. Die Änderungen werden jedoch erst zusammen mit dem Projekt gespeichert.

### Siehe auch

Grundlagen zu Bausteineigenschaften (Seite 1223)

Übersicht über die Bausteineigenschaften (Seite 1224)

Zeitstempel von Bausteinen (Seite 1227)

### 9.1.3.3 Bausteine verwalten

#### Bausteine öffnen

Um einen Baustein zu öffnen, haben Sie folgende Möglichkeiten:

- Baustein direkt öffnen  
Sie können einen Baustein direkt öffnen, wenn Sie den entsprechenden Bausteinordner in der Projektnavigation oder im Übersichtsfenster geöffnet haben.
- Baustein suchen und öffnen  
Sie können innerhalb eines Projekts, eines Geräts und des Ordners "Programmbausteine" nach Bausteinen suchen und diese dann öffnen.

#### Baustein direkt öffnen

Um einen Baustein direkt zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner mit dem Baustein, den Sie öffnen möchten, in der Projektnavigation oder im Übersichtsfenster.
2. Doppelklicken Sie auf den Baustein, den Sie öffnen möchten.

## Baustein suchen und öffnen

Um nach einem Baustein zu suchen und ihn dann zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation mit der rechten Maustaste auf das Projekt, ein Gerät, den Ordner "Programmbaustein" oder einen unterlagerten Ordner von "Programmbausteine".
2. Wählen Sie im Kontextmenü den Befehl "In PLC suchen und öffnen".  
Der Dialog "In PLC suchen und öffnen" wird geöffnet.
3. Geben Sie den Namen, die Adresse oder den Typ des Bausteins ein, den Sie suchen.  
Mit jedem eingegebenen Buchstaben wird die Bausteinliste weiter gefiltert. Wenn es keinen Baustein gibt, der mit Ihrer Eingabe übereinstimmt, wird die Bausteinliste geschlossen. Sie können die vollständige Bausteinliste jederzeit mit einem Klick auf die Schaltfläche rechts neben dem Textfeld einblenden. Beachten Sie jedoch, dass dann keine Filterung erfolgt.  
Möchten Sie wieder nach Ihren Eingaben filtern, klicken Sie erneut auf die Schaltfläche.
4. Klicken Sie in der Bausteinliste auf den Baustein, den Sie öffnen möchten.

## Ergebnis

Der Baustein wird im Programmiereditor geöffnet.

## Siehe auch

Bausteine speichern (Seite 1231)

Bausteine schließen (Seite 1232)

Bausteine umbenennen (Seite 1232)

Bausteine offline löschen (Seite 1233)

Bausteine online löschen (Seite 1234)

Know-how-geschützte Bausteine öffnen (Seite 1489)

## Bausteine speichern

Bausteine werden immer zusammen mit dem Projekt gespeichert. Auch fehlerhafte Bausteine können gespeichert werden. Dadurch kann die Fehlerbehebung zu jedem beliebigen Zeitpunkt erfolgen.

## Vorgehen

Um einen Baustein zu speichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Projekt" den Befehl "Speichern" oder "Speichern unter".  
Siehe auch: Projekte speichern (Seite 237)

## Siehe auch

Bausteine öffnen (Seite 1230)

Bausteine schließen (Seite 1232)

Bausteine umbenennen (Seite 1232)

Bausteine offline löschen (Seite 1233)

Bausteine online löschen (Seite 1234)

## Bausteine schließen

### Vorgehen

Um einen Baustein zu schließen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Titelleiste des Programmiereditors auf die Schaltfläche "Schließen".

---

### Hinweis

Beachten Sie, dass der Baustein beim Schließen nicht gespeichert wird.

---

## Siehe auch

Bausteine öffnen (Seite 1230)

Bausteine speichern (Seite 1231)

Bausteine umbenennen (Seite 1232)

Bausteine offline löschen (Seite 1233)

Bausteine online löschen (Seite 1234)

## Bausteine umbenennen

### Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

## Vorgehen

Um den Namen eines Bausteins zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den Baustein, den Sie umbenennen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".  
In der Projektnavigation wird der Bausteinname in ein Eingabefeld geändert.
3. Geben Sie den neuen Namen des Bausteins ein.
4. Bestätigen Sie Ihre Eingabe mit der Eingabetaste.

## Ergebnis

Der Name des Bausteins wird an allen Verwendungsstellen im Programm geändert.

## Siehe auch

Bausteine öffnen (Seite 1230)

Bausteine speichern (Seite 1231)

Bausteine schließen (Seite 1232)

Bausteine offline löschen (Seite 1233)

Bausteine online löschen (Seite 1234)

## Bausteine offline löschen

### Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

### Vorgehen

Um einen Baustein zu löschen, der offline vorhanden ist, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation im Ordner "Programmbausteine" mit der rechten Maustaste auf den Baustein, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Bestätigen Sie die Sicherheitsabfrage mit "Ja".  
Der Baustein wird offline aus dem Projekt gelöscht.

---

### Hinweis

Beachten Sie beim Löschen von Organisationsbausteinen, dass ihnen Ereignisse zugeordnet sein können. Wenn Sie solche Organisationsbausteine löschen, kann das Programm nicht auf parametrisierte Ereignisse reagieren.

---

## Siehe auch

Bausteine öffnen (Seite 1230)

Bausteine speichern (Seite 1231)

Bausteine schließen (Seite 1232)

Bausteine umbenennen (Seite 1232)

Bausteine online löschen (Seite 1234)

## Bausteine online löschen

### Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

### Vorgehen

Um einen Baustein zu löschen, der online vorhanden ist, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation im Ordner "Programmbausteine" mit der rechten Maustaste auf den Baustein, den Sie im Gerät löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".  
Der Dialog "Löschen" wird geöffnet.
3. Aktivieren Sie das Optionsfeld "Löschen aus Gerät".
4. Klicken Sie auf "Ja".  
Der Baustein wird online aus dem Gerät gelöscht.

## Siehe auch

Bausteine öffnen (Seite 1230)

Bausteine speichern (Seite 1231)

Bausteine schließen (Seite 1232)

Bausteine umbenennen (Seite 1232)

Bausteine offline löschen (Seite 1233)

## CPU-Datenbausteine löschen

Sie können CPU-Datenbausteine sowohl offline als auch online löschen.



### CPU-Datenbausteine offline löschen

Um einen CPU-Datenbaustein zu löschen, der offline im Projekt vorhanden ist, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation im Ordner "Programmbausteine" mit der rechten Maustaste auf den CPU-Datenbaustein, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Bestätigen Sie die Sicherheitsabfrage mit "Ja".  
Der CPU-Datenbaustein wird offline aus dem Projekt gelöscht.

### CPU-Datenbaustein online löschen

Um einen CPU-Datenbaustein zu löschen, der online vorhanden ist, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zum Gerät her, das den CPU-Datenbaustein enthält, den Sie löschen möchten.
2. Klicken Sie in der Projektnavigation im Ordner "Programmbausteine" mit der rechten Maustaste auf den CPU-Datenbaustein, den Sie im Gerät löschen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".  
Der Dialog "Löschen" wird geöffnet.
4. Aktivieren Sie das Optionsfeld "Löschen aus Gerät".
5. Klicken Sie auf "Ja".  
Der CPU-Datenbaustein wird online aus dem Gerät gelöscht.

### Siehe auch

CPU-Datenbausteine (Seite 1032)

## 9.1.4 Bausteine programmieren

### 9.1.4.1 Programmiereditor

#### Übersicht über den Programmiereditor

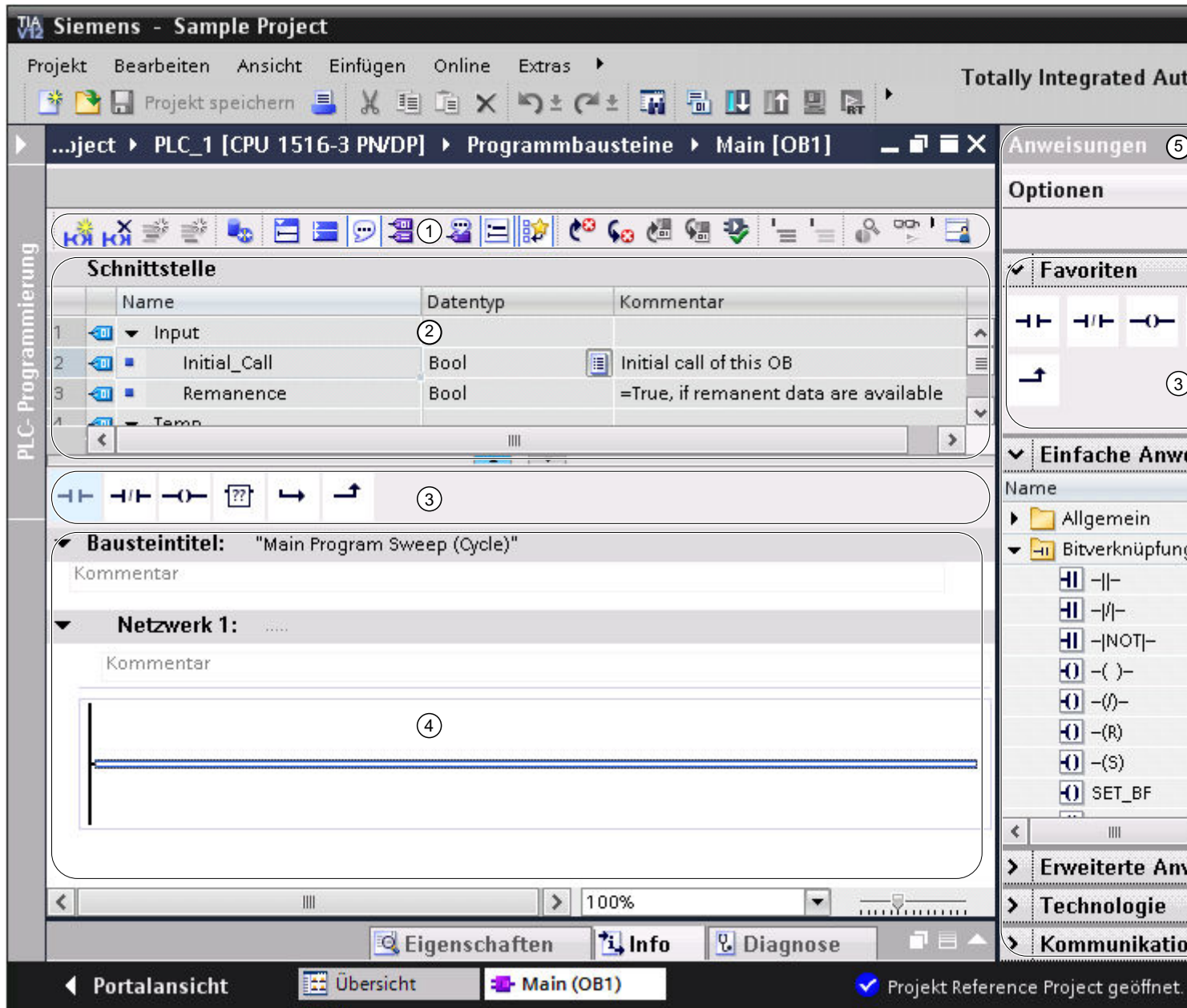
#### Funktion des Programmiereditors

Der Programmiereditor ist die integrierte Entwicklungsumgebung für das Programmieren von Funktionen, Funktionsbausteinen und Organisationsbausteinen. Er bietet maximale Unterstützung bei der Programmerstellung und bei der Fehlersuche.

Je nach verwendeter CPU, Programmiersprache und Bausteintyp können Aussehen und Funktionalität des Programmiereditors variieren.

### Aufbau des Programmiereditors

Das folgende Bild zeigt die Komponenten des Programmiereditors am Beispiel von KOP:



- ① Funktionsleiste
- ② Bausteinschnittstelle
- ③ Palette "Favoriten" in der Task Card "Anweisungen" und Favoriten im Programmiereditor
- ④ Programmierfenster
- ⑤ Task Card "Anweisungen"
- ⑥ Task Card "Testen"

## Funktionsleiste

Über die Funktionsleiste haben Sie Zugriff auf die wichtigsten Funktionen des Programmiereditors, z. B.:

- Absolute Operanden ein- und ausblenden
- Favoriten ein- und ausblenden
- Zu Syntaxfehlern springen
- Bausteinaufrufe aktualisieren
- Programmstatus ein- und ausblenden

Je nach verwendeter Programmiersprache können die zur Verfügung stehenden Funktionen der Funktionsleiste variieren.

## Bausteinschnittstelle

Die Bausteinschnittstelle enthält die Deklarationen lokaler Variablen, die nur innerhalb des Bausteins verwendet werden. Die verfügbaren Abschnitte variieren abhängig vom Bausteintyp.

## Favoriten

Sie haben die Möglichkeit, für häufig verwendete Anweisungen Favoriten anzulegen. Diese Favoriten werden in der Task Card "Anweisungen" in der Palette "Favoriten" angezeigt. Zusätzlich können Sie die Favoriten über die Funktionsleiste des Programmiereditors auch im Programmiereditor anzeigen. Dadurch können Sie auch dann auf Ihre Favoriten zugreifen, wenn die Task Card "Anweisungen" nicht sichtbar ist.

## Programmierfenster

Das Programmierfenster ist der Arbeitsbereich des Programmiereditors. Hier können Sie den Programmcode eingeben. Je nach verwendeter Programmiersprache kann das Programmierfenster in Aussehen und Funktionalität variieren.

## Task Card "Anweisungen"

Die Task Card "Anweisungen" ermöglicht Ihnen einen einfachen Zugriff auf alle Anweisungen, die Sie für die Erstellung Ihres Programms verwenden können. Die Anweisungen sind thematisch in unterschiedliche Paletten gegliedert. Über die Schaltfläche "Spaltenüberschriften und zusätzliche Spalten einblenden" in der Funktionsleiste der Task Card können Sie weitere Informationen zu den Anweisungen einblenden. Die Anordnung der Spalten können Sie ändern, indem Sie auf eine Spaltenüberschrift klicken und die Spalte per Drag & Drop verschieben.

Wenn ein Anweisungsprofil aktiv ist, können die angebotenen Anweisungen variieren. Siehe auch: Anweisungsprofile verwenden

## Task Card "Testen"

In der Task Card "Testen" können Sie Einstellungen vornehmen, die sich auf die Fehlersuche mithilfe des Programmstatus auswirken. Die Funktionen der Task Card "Testen" stehen Ihnen

nur im Online-Modus zur Verfügung. Sie enthält folgende Paletten, die abhängig von der ausgewählten CPU und der eingestellten Programmiersprache des Bausteins angezeigt werden:

- CPU-Bedienpanel  
Im CPU-Bedienpanel können Sie den Betriebszustand der CPU umschalten.
- Haltepunkte  
Bausteine, die Sie in einer der textuellen Programmiersprachen AWL oder SCL erstellt haben, können Sie im Einzelschrittmodus testen. Dazu setzen Sie Haltepunkte in den Programmcode.  
In der Palette "Haltepunkte" finden Sie alle Haltepunkte, die Sie gesetzt haben, und können diese aktivieren, löschen, zu einzelnen Haltepunkten navigieren oder die Aufrufumgebung für den Haltepunkt festlegen.
- AS-Register  
In dieser Palette können Sie die Werte für die AS-Register und die Akkumulatoren ablesen.
- Kettensteuerung  
In dieser Palette legen Sie die Betriebsart für das Testen von Ablaufketten für GRAPH-Bausteine fest.
- Testeinstellungen  
In dieser Palette legen Sie die Testeinstellungen für GRAPH-Bausteine fest.
- Aufrufumgebung  
In dieser Palette legen Sie die Aufrufumgebung für den Baustein fest.
- Aufrufhierarchie  
In dieser Palette können Sie die Aufrufhierarchie der Bausteine verfolgen. Sie sehen die Aufrufhierarchie nur während des Beobachtens von Bausteinen.

**Siehe auch**

Aufbau der Bausteinschnittstelle (Seite 1257)

Fläche des Programmierfensters vergrößern (Seite 1243)

**Tastaturbedienung im Programmiereditor**

**Im Editor navigieren**

| Funktion  | Tastenkombination |
|---|-------------------|
| Task Card "Anweisungen" öffnen  | <Strg+Shift+C>    |
| Task Card "Testen" öffnen   | <Strg+Shift+O>    |
| Neuen Baustein hinzufügen   | <Strg+N>          |
| Alle Netzwerke erweitern  | <Alt+F11>         |
| Alle Netzwerke reduzieren   | <Alt+F12>         |
| Zur nächsten Verwendungsstelle des selektierten Bausteins oder Operanden navigieren | <Strg+Shift+F>    |

| Funktion   | Tastenkombination |
|--|-------------------|
| Zur vorangehenden Verwendungsstelle des selektierten Bausteins oder Operanden navigieren | <Strg+Shift+G>    |
| Zum nächsten Lese-/Schreibzugriff navigieren   | <Alt+F8>          |
| Zum vorangehenden Lese-/Schreibzugriff navigieren  | <Alt+F9>          |

### Im Programmcode navigieren (KOP/FUP)

| Funktion  | Selektiertes Objekt | Tastenkombination |
|---|---------------------|-------------------|
| Zwischen den Objekten im Netzwerk navigieren    | Objekt im Netzwerk  | Pfeiltasten       |
| Zum ersten Element des Netzwerks navigieren     | Objekt im Netzwerk  | <Pos1>            |
| Zum letzten Element des Netzwerks navigieren    | Objekt im Netzwerk  | <Ende>            |
| Zum nächsten Element des Netzwerks navigieren   | Objekt im Netzwerk  | <Tab>             |
| Zum vorherigen Element des Netzwerks navigieren | Objekt im Netzwerk  | <Shift+Tab>       |
| Netzwerk einfügen                               | Beliebig            | <Strg+R>          |

### Im Programmcode navigieren (AWL/SCL)

| Funktion                          | Position der Einfügemarke | Tastenkombination   |
|-----------------------------------|---------------------------|---|
| Im Programmcode navigieren        | Zeile                     | Pfeiltasten   |
| Ein Wort nach rechts/links        | Zeile                     | <Strg+Pfeiltasten>  |
| Zum Zeilenanfang                  | Zeile                     | <Pos1>  |
| Zum Zeilenende                    | Zeile                     | <Ende>  |
| Zum Anfang eines Codeabschnitts   | Zeile                     | <Strg+Pos1>   |
| Zum Ende eines Codeabschnitts     | Zeile                     | <Strg+Ende>   |
| Zum nächsten Netzwerk (nur AWL)   | Netzwerktitle             | <Pfeil-nach-unten>  |
| Zum nächsten Netzwerk (nur AWL)   | Zeile                     | <Tab><br>Wiederholen Sie die Tastenkombination bis die Einfügemarke im nächsten Netzwerk steht.         |
| Zum vorherigen Netzwerk (nur AWL) | Netzwerktitle             | <Pfeil-nach-oben>   |
| Zum vorherigen Netzwerk (nur AWL) | Zeile                     | <Shift+Tab><br>Wiederholen Sie die Tastenkombination bis die Einfügemarke im vorherigen Netzwerk steht. |
| Netzwerk einfügen                 | Beliebig                  | <Strg+R>  |

**Anweisungen einfügen (KOP)**

| Funktion                         | Selektiertes Objekt | Tastenkombination |
|----------------------------------|---------------------|-------------------|
| Schließerkontakt einfügen        | Strompfad           | <Shift+F2>        |
| Öffnerkontakt einfügen           | Strompfad           | <Shift+F3>        |
| Leerbox einfügen                 | Strompfad           | <Shift+F5>        |
| Zuweisung einfügen               | Strompfad           | <Shift+F7>        |
| "Verzweigung öffnen" einfügen    | Strompfad           | <Shift+F8>        |
| "Verzweigung schließen" einfügen | Strompfad           | <Shift+F9>        |

**Anweisungen einfügen (FUP)**

| Funktion                      | Selektiertes Objekt                 | Tastenkombination |
|-------------------------------|-------------------------------------|-------------------|
| Zuweisung einfügen            | Netzwerk, Ein- oder Ausgang         | <Shift+F7>        |
| Leerbox einfügen              | Netzwerk                            | <Shift+F5>        |
| "Verzweigung öffnen" einfügen | Verbindungsline zwischen zwei Boxen | <Shift+F8>        |
| VKE invertieren               | Netzwerk, Ein- oder Ausgang         | <Strg+Shift+4>    |
| Eingang einfügen              | Netzwerk, Ein- oder Ausgang         | <Strg+Shift+3>    |

**Operanden eingeben (KOP/FUP/GRAPH)**

| Funktion  | Selektiertes Objekt       | Tastenkombination  |
|---|---------------------------|--|
| Eingabefeld für den ersten Operanden der Anweisung aktivieren | Anweisung                 | <Return><br>Oder<br><Beliebige Buchstaben/<br>Zahlen><br>Bei <Return> wird ein Eingabefeld geöffnet, bei Buchstaben oder Zahlen werden diese ins Eingabefeld übernommen. |
| Eingabefeld für den Operanden aktivieren                      | Operand                   | <F2>   |
| Operand löschen   | Operand                   | <Entf>   |
| Variable definieren   | Operand                   | <Strg+Shift+I>   |
| Variable umverdrahten   | Operand                   | <Strg+Shift+P>   |
| Variable umbenennen   | Operand                   | <Strg+Shift+T>   |
| Operanden eingeben  | Eingabefeld für Operanden | <Beliebige Buchstaben/<br>Zahlen>  |
| Eingabe des Operanden bestätigen                              | Eingabefeld für Operanden | <Return>   |

| Funktion                     | Selektiertes Objekt       | Tastenkombination  |
|------------------------------|---------------------------|--|
| Autovervollständigung öffnen | Eingabefeld für Operanden | <Strg+I>   |
| Aktuelle Änderung verwerfen  | Eingabefeld für Operanden | <ESC><br>Das Eingabefeld wird deaktiviert und der vorherige Inhalt wieder hergestellt. |

### Anweisungen bearbeiten (AWL/SCL)

| Funktion                                      | Selektiertes Objekt                       | Tastenkombination                      |
|---|---|--|
| Zeile einrücken (nur SCL)                     | Zeile                                     | <Tab> oder<br><Strg+R>                 |
| Zeile ausrücken (nur SCL)                     | Zeile                                     | <Shift+Tab> oder<br><Strg+Shift+R>     |
| Dialog "Aufrufoptionen" öffnen                | Einfügemarke hinter einem Bausteinaufruf  | <Return>                               |
| Variable definieren                           | Operand                                   | <Strg+Shift+I>                         |
| Variable umverdrahten                         | Operand                                   | <StrgShift+P>                          |
| Variable umbenennen                           | Operand                                   | <Strg+Shift+T>                         |
| Parameterliste erweitern/reduzieren (nur SCL) | Operand                                   | <Strg+Shift+Leertaste>                 |
| Codeabschnitt auf-/zuklappen                  | Einfügemarke innerhalb des Codeabschnitts | <Strg+Shift+Num+><br><Strg+Shift+Num-> |
| Alle Codeabschnitte auf-/zuklappen            | Beliebig                                  | <Strg+Shift+Num*><br><Strg+Shift+Num/> |
| Autovervollständigung öffnen                  | Beliebig                                  | <Strg+I> oder <Strg+Leertaste>         |
| Lesezeichen setzen/löschen                    |   | <Strg+Shift+M>                         |
| Zum nächsten Lesezeichen                      |   | <Strg+Shift+6>                         |
| Zum vorherigen Lesezeichen                    |   | <Strg+Shift+5>                         |
| Code deaktivieren                             | Zeile                                     | <Strg+Shift+Y>                         |
| Code aktivieren                               | Zeile                                     | <Strg+Shift+U>                         |

## Programmierfenster von GRAPH

| Funktion   | Bereich   | Tastenkombination  |
|--|---|--|
| Eine Seite nach oben/unten   | Navigation,<br>Einzelschrittansicht,<br>Kettenansicht,<br>permanente<br>Anweisungen | <Bild-nach-oben>/<br><Bild-nach-unten>                           |
| In der Navigation navigieren   | Navigation  | <Pfeil-nach-oben><br><Pfeil-nach-unten>                          |
| Objekt aufklappen/zuklappen  | Navigation  | <+> oder <Pfeil-nach-<br>rechts>/<br><-> oder <Pfeil-nach-links> |
| Zwischen Einzelschrittansicht und Kettenansicht<br>wechseln, wenn ein Schritt oder eine Transition<br>selektiert ist | Navigation  | <Return>   |
| Zwischen Navigation und Arbeitsbereich<br>wechseln   | Navigation,<br>Einzelschrittansicht,<br>Kettenansicht,<br>permanente<br>Anweisungen | <ALT+F6>   |
| Zum ersten Element in einem Netzwerk   | Einzelschrittansicht  | <Pos1>   |
| Zum letzten Element in einem Netzwerk  | Einzelschrittansicht  | <Ende>   |
| Zum Interlock wechseln   | Einzelschrittansicht  | <Strg+Pos1>  |
| Zur Transition wechseln  | Einzelschrittansicht  | <Strg+Ende>  |
| In der Struktur navigieren   | Kettenansicht   | Pfeiltasten  |
| Zum ersten Schritt   | Kettenansicht   | <Pos1> oder <Strg+Pos1>  |
| Zum letzten Schritt  | Kettenansicht   | <Ende> oder <Strg+Ende>  |
| Verzweigung öffnen   | Kettenansicht   | <Shift+F8>   |
| Verzweigung schließen  | Kettenansicht   | <Shift+F9>   |
| Kettenende einfügen  | Kettenansicht   | <Shift+F7>   |
| Sprung einfügen  | Kettenansicht   | <Shift+F12>  |
| Schritt und Transition einfügen  | Kettenansicht   | <Shift+F5>   |
| Element löschen  | Kettenansicht   | <Entf>   |
| Zum ersten editierbaren Element  | permanente<br>Anweisungen   | <Pos1>   |
| Zum nächsten editierbaren Element  | permanente<br>Anweisungen   | <Tab>  |
| Zum letzten editierbaren Element   | permanente<br>Anweisungen   | <Ende>   |
| Zum vorherigen editierbaren Element  | permanente<br>Anweisungen   | <Shift+Tab>  |
| Springe zum Anfang der Zelle "Aktion"  | Aktionen  | <Pos1>   |
| Springe zum Ende der Zelle "Aktion"  | Aktionen  | <Ende>   |
| Neue Aktion einfügen   | Aktionen  | <Return>   |



## Programm beobachten

| Funktion  | Tastenkombination |
|---|-------------------|
| Haltepunkt setzen/löschen (AWL, SCL)                              | <Strg+Shift+F9>   |
| Haltepunkt überspringen (AWL SCL)                                 | <Strg+Shift+F10>  |
| In einen unterlagerten Baustein hineinspringen (AWL SCL)          | <Strg+Shift+F11>  |
| In den aufrufenden Baustein zurückspringen (AWL, SCL)             | <Strg+Shift+F12>  |
| Programm bis zur Markierung (Cursor-Position) ausführen (AWL SCL) | <Strg+F3>         |
| Programmstatus anzeigen (AWL SCL)                                 | <Strg+T>          |
| Alle Haltepunkte aktivieren (AWL, SCL)                            | <Strg+Shift+F2>   |
| Alle Haltepunkte deaktivieren (AWL, SCL)                          | <Strg+Shift+F3>   |
| Steuern auf 0 (KOP, FUP)  | <Strg+Shift+9>    |
| Steuern auf 1 (KOP, FUP)  | <Strg+Shift+1>    |
| Operand steuern (KOP, FUP)  | <Strg+Shift+2>    |

## Siehe auch

Tastaturbedienung im TIA-Portal (Seite 207)  
 Nutzen projektbezogener Funktionen (Seite 209)  
 Anordnung von Fenstern (Seite 209)  
 Bearbeiten von Tabellen (Seite 216)  
 Textbearbeitung (Seite 214)

## Fläche des Programmierfensters vergrößern

### Einführung

Wenn alle Komponenten der Anwendung eingeblendet sind, ist die Fläche des Programmierfensters relativ klein. Daher kann es vorkommen, dass Sie den Arbeitsbereich bei umfangreichem Programmcode häufig verschieben müssen. Um dies zu vermeiden, können Sie die Anzeige folgender Komponenten der Anwendung und des Programmiereditors ausblenden oder minimieren:

- Projektnavigation
- Task Cards
- Bausteinschnittstelle
- Favoriten
- Kommentare
- Netzwerke

---

### Hinweis

Sie können für die Task Cards, die Projektnavigation und das Inspektorfenster zusätzlich die Option "Automatisch reduzieren" verwenden. Dadurch klappen diese Fenster automatisch zu, wenn Sie sie nicht benötigen.

Siehe auch: Arbeitsbereich maximieren und minimieren

---

## Projektnavigation aus- und einblenden

Über die Projektnavigation greifen Sie auf alle Bereiche des Projekts zu. Während der Programmerstellung können Sie die Projektnavigation ausblenden, um die Fläche des Programmierfensters zu vergrößern.

Um die Projektnavigation aus- oder einzublenden, gehen Sie folgendermaßen vor:

1. Deaktivieren Sie im Menü "Ansicht" das Optionskästchen "Projektnavigation" oder klicken Sie auf die Schaltfläche "Reduzieren" in der Titelleiste der Projektnavigation, um die Projektnavigation auszublenden.
2. Aktivieren Sie im Menü "Ansicht" das Optionskästchen "Projektnavigation" oder klicken Sie auf die Schaltfläche "Erweitern" in der Titelleiste der Projektnavigation, um die Projektnavigation einzublenden.

## Task Cards öffnen und schließen

Die Task Cards befinden sich am rechten Rand des Programmierfensters.

Um die Task Cards zu schließen oder zu öffnen, gehen Sie folgendermaßen vor:

1. Deaktivieren Sie im Menü "Ansicht" das Optionskästchen "Task Card" oder klicken Sie auf die Schaltfläche "Reduzieren" in der Titelleiste der Task Cards, um die Task Cards zu schließen.
2. Aktivieren Sie im Menü "Ansicht" das Optionskästchen "Task Card" oder klicken Sie auf die Schaltfläche "Erweitern" in der Titelleiste der Task Cards, um die Task Cards zu öffnen.

## Bausteinschnittstelle aus- und einblenden

Die Bausteinschnittstelle eines Bausteins befindet sich im oberen Bereich des Programmiereditors. Während der Programmerstellung kann sie aus- und bei Bedarf wieder eingeblendet werden.

Um die Bausteinschnittstelle aus- oder einzublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie im unteren Bereich der Schnittstelle innerhalb des Fensterteilers auf den Pfeil-nach-oben oder Pfeil-nach-unten.

### **Favoriten ein- und ausblenden**

Um die Favoriten im Programmiereditor ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

### **Kommentare ein- und ausblenden**

In einem Baustein können Sie einen Kommentar zum Baustein oder für jedes Netzwerk eingeben. Diese beiden Kommentararten werden unterschiedlich ein- und ausgeblendet.

Um einen Bausteinkommentar ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Zeile mit dem Bausteintitel auf das Dreieck am Anfang der Zeile.

Um Netzwerkkommentare ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Programmiereditors auf "Netzwerkkommentare ein/aus".

---

#### **Hinweis**

Je nach verwendeter Programmiersprache können die zur Verfügung stehenden Kommentare variieren.

---

### **Netzwerke auf- und zuklappen**

Einige Programmiersprachen verwenden Netzwerke. Diese Netzwerke können Sie je nach Bedarf auf- oder zuklappen.

Um einzelne Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Wenn Sie ein Netzwerk aufklappen möchten, klicken Sie auf den Pfeil-nach-rechts vor dem Netzwerktitel. Wenn Sie ein Netzwerk zuklappen möchten, klicken Sie auf den Pfeil-nach-unten vor dem Netzwerktitel.

Um alle Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Programmiereditors auf "Alle Netzwerke öffnen" oder auf "Alle Netzwerke schließen".

---

#### **Hinweis**

Netzwerke werden nicht in jeder Programmiersprache verwendet.

---

## Siehe auch

- Übersicht über den Programmiereditor (Seite 1235)
- Arbeitsbereich maximieren und minimieren (Seite 186)

## Einstellen der Mnemonik

Bausteine können Sie in deutscher oder internationaler Mnemonik programmieren. Wenn Sie das TIA-Portal zum ersten Mal öffnen, ist internationale Mnemonik voreingestellt. Sie können die Mnemonik jederzeit ändern.

## Vorgehen

Um die Mnemonik einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "Allgemein".
3. Wählen Sie in der Gruppe "Allgemeine Einstellungen" die gewünschte Mnemonik aus.  
Die Mnemonik wird in allen Bausteinen umgestellt.

## Symbolische und absolute Operanden anzeigen

Für die Darstellung der Operanden im Programmiereditor gibt es folgende Möglichkeiten:

- **Symbolische Darstellung**  
Im Programm werden die symbolischen Operanden angezeigt. Die dazugehörigen absoluten Adressen werden in Tooltips eingeblendet, wenn der Mauszeiger längere Zeit über dem Operanden verweilt.
- **Absolute Darstellung**  
Im Programm werden die absoluten Adressen angezeigt. Die dazugehörigen symbolischen Operanden werden in Tooltips eingeblendet.
- **Symbolische und absolute Darstellung**  
Im Programm werden symbolische Operanden und absolute Adressen angezeigt.

## Voraussetzung

Der Programmiereditor ist geöffnet.

## Vorgehen

Um die Darstellung der Operanden zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Absolute/symbolische Operanden".  
Mit jedem Klick auf die Schaltfläche erhalten Sie eine andere Darstellung und das Symbol der Schaltfläche wird geändert.

Oder:

1. Klicken Sie in der Funktionsleiste des Programmiereditors auf den kleinen Pfeil neben der Schaltfläche "Absolute/symbolische Operanden".  
Eine Klappliste wird geöffnet.
2. Wählen Sie die gewünschte Darstellung aus der Klappliste.  
Das Symbol der Schaltfläche wird geändert.

## Siehe auch

Grundlagen zu Operanden (Seite 1056)

## Anweisungsversionen verwenden

### Grundlagen zu Anweisungsversionen

Die Anweisungen, die Ihnen zur Programmierung des Anwenderprogramms zur Verfügung stehen, werden in Systembibliotheken verwaltet. Wenn durch eine Aktualisierung eine neue Version einer Systembibliothek installiert wird, werden dadurch eventuell auch neuere Versionen der Anweisungen dieser Systembibliothek installiert.

Wenn es für eine Anweisung mehrere Versionen gibt, werden diese in der Task Card "Anweisungen" hinter der jeweiligen Anweisung aufgelistet. Falls die Anweisungsversionen ausgeblendet sind, können Sie diese über die Schaltfläche "Spaltenüberschriften und zusätzliche Spalten einblenden" in der Funktionsleiste der Task Card "Anweisungen" einblenden. Anschließend können Sie für die Anweisungen die Versionen aus der Klappliste der Spalte "Version" auswählen, die im Programm verwendet werden sollen. Wenn Sie keine Versionen auswählen, werden die neuesten Versionen verwendet.

---

### Hinweis

Beachten Sie folgende Hinweise:

- Innerhalb eines Geräts können Sie eine Anweisung immer nur in der gleichen Version verwenden.
  - Wenn Sie die Version einer Anweisung ändern, von der andere Anweisungen abhängen, werden die Versionen der abhängigen Anweisungen ebenfalls geändert.
  - Falls Sie für eine Anweisung eine Version auswählen, die für die verwendete CPU nicht lauffähig ist, wird die Anweisung gegraut dargestellt. Das heißt, Sie können die Anweisung in dieser Version nicht mit Ihrer CPU verwenden.
  - Wenn Sie die Version einer Anweisung ändern, müssen Sie den Baustein übersetzen, bevor die neue Versionsnummer in den Eigenschaften der Anweisung angezeigt wird.
- 

## Änderungen in den Versionen

Neue Versionen können Haupt- oder Nebenversionen sein. Neue Hauptversionen, z. B. 2.0 oder 3.0, enthalten größere Änderungen. Daher können neue Hauptversionen Änderungen der Bausteinschnittstelle zur Folge haben. Neue Nebenversionen, z. B. 1.3 oder 1.4, enthalten kleinere Änderungen oder Fehlerbehebungen.

## Verwendung von Anweisungsversionen

Sie können innerhalb eines Geräts entscheiden, welche Version einer Anweisung Sie verwenden möchten. Wenn Sie für eine Anweisung eine andere Version auswählen, wird für alle Verwendungsstellen dieser Anweisung in Ihrem Programm die neue Version festgelegt. Im Programm werden diese Anweisungen mit einem roten Rahmen gekennzeichnet. Anschließend müssen Sie Ihr Programm in das Gerät laden, damit die neue Anweisungsversion verwendet wird.

## Anweisungsprofile verwenden

### Grundlagen zu Anweisungsprofilen

#### Einführung

Das TIA-Portal stellt Ihnen für die Programmierung Ihres Anwenderprogramms eine Vielzahl von Anweisungen zur Verfügung. Wenn Sie jedoch nicht alle Anweisungen verwenden möchten, haben Sie die Möglichkeit, die nicht benötigten Anweisungen auszufiltern. Zu diesem Zweck können Sie Anweisungsprofile erstellen und innerhalb eines solchen Profils genau festlegen, welche Anweisungen in der Task Card "Anweisungen" angeboten werden sollen. Sie können innerhalb eines Projekts mehrere Anweisungsprofile anlegen, jedoch kann immer nur ein Profil aktiv sein. Über globale Bibliotheken können Sie Anweisungsprofile mit anderen Anwendern austauschen.

---

#### Hinweis

Beachten Sie folgende Hinweise:

- Wenn in einem Baustein Anweisungen verwendet werden, die durch das aktive Profil nicht zugelassen sind, erhalten Sie einen Fehler beim Übersetzen des Bausteins. Dies könnte z. B. durch einen Baustein passieren, den Sie aus einer Bibliothek in Ihr Programm ziehen.
  - Anweisungen eines Profils, die durch die aktuell installierten Produkte nicht unterstützt werden, werden beim nächsten Bearbeiten des Profils aus dem Profil gelöscht. Wenn Sie dieses Profil in ein Engineering System übertragen, in dem diese Anweisungen durch die installierten Produkte unterstützt werden, sind die Anweisungen wieder im Profil vorhanden, jedoch deaktiviert. Sie können diese Anweisungen bei Bedarf jederzeit aktivieren.
  - Wenn Sie Änderungen am aktiven Profil vornehmen, müssen Sie die Bausteine im Projekt neu übersetzen. Dies ist auch beim Deaktivieren und Löschen des aktiven Profils oder beim Aktivieren eines Profils erforderlich.
- 

#### Siehe auch

Neues Anweisungsprofil erzeugen (Seite 1249)

Anweisungsprofil öffnen und bearbeiten (Seite 1250)

Anweisungsprofil aktivieren und deaktivieren (Seite 1251)

Anweisungsprofil löschen (Seite 1252)

## Neues Anweisungsprofil erzeugen

### Voraussetzung

Der Ordner "Gemeinsame Daten > Anweisungsprofile" in der Projektnavigation ist geöffnet.

### Vorgehen

Um ein neues Anweisungsprofil zu erzeugen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neues Anweisungsprofil hinzufügen".  
Der Anweisungsprofil-Editor wird geöffnet und das neue Anweisungsprofil wird angezeigt.  
Für das neue Anweisungsprofil sind alle Anweisungen aktiviert.
2. Bearbeiten Sie das neue Anweisungsprofil gemäß Ihren Anforderungen.

Sie können das neue Anweisungsprofil bei Bedarf umbenennen. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das neue Anweisungsprofil.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie einen Namen für das neue Anweisungsprofil ein.

---

#### Hinweis

Wenn Sie das erste Anweisungsprofil erzeugen, wird dieses als aktives Profil verwendet. Übersetzen Sie in diesem Fall alle Bausteine im Projekt. Falls bereits weitere Anweisungsprofile vorhanden sind, müssen Sie das neue Profil explizit aktivieren, damit es als aktives Profil verwendet wird. Das aktive Profil erkennen Sie in der Projektnavigation am Symbol.

---

### Siehe auch

- Grundlagen zu Anweisungsprofilen (Seite 1248)
- Anweisungsprofil öffnen und bearbeiten (Seite 1250)
- Anweisungsprofil aktivieren und deaktivieren (Seite 1251)
- Anweisungsprofil löschen (Seite 1252)

## Anweisungsprofil öffnen und bearbeiten

Nach dem Öffnen eines Anweisungsprofils können Sie es folgendermaßen bearbeiten:

- Anweisungen aktivieren und deaktivieren  
Sie können für jede Anweisung festlegen, ob sie innerhalb des Anweisungsprofils zugelassen ist.

---

### Hinweis

Beachten Sie, dass zwischen einigen Anweisungen Abhängigkeiten bestehen. Dadurch werden evtl. mit einer Aktion mehrere Anweisungen aktiviert oder deaktiviert. Sie können am Symbol des Optionskästchens erkennen, in welchen Ordnern Anweisungen deaktiviert sind.

---

- Anweisungsversionen aktivieren und deaktivieren  
Einige Anweisungen sind mit verschiedenen Versionen vorhanden. Wenn eine Anweisung mehrere Versionen besitzt, können Sie für jede Version festlegen, ob sie innerhalb des Anweisungsprofils zugelassen ist.
- Bausteinnummer ändern  
Wenn eine Anweisung systemintern ein Funktionsbaustein (FB) oder eine Funktion (FC) ist, erhält die Anweisung vom System eine bestimmte Bausteinnummer. Sie haben die Möglichkeit, diese Bausteinnummer durch eine eigene Bausteinnummer zu ersetzen. Für einige Anweisungen gibt es innerhalb einer Version mehrere Implementierungen. Bei solchen Anweisungen können Sie die Bausteinnummern nur für die einzelnen Implementierungen ändern.

---

### Hinweis

Wenn die Anweisung aus dem Anweisungsprofil im Programm verwendet wird und die festgelegte Bausteinnummer im Anwenderprogramm bereits durch einen anderen Baustein belegt ist, wird die festgelegte Bausteinnummer der Anweisung durch eine freie Bausteinnummer ersetzt.

---

## Voraussetzung

Der Ordner "Gemeinsame Daten > Anweisungsprofile" in der Projektnavigation ist geöffnet.



## Anweisungsprofile öffnen

Um ein Anweisungsprofil zu öffnen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf das Anweisungsprofil, das Sie bearbeiten möchten.  
Das Anweisungsprofil wird im Anweisungsprofil-Editor geöffnet.

## Anweisungsprofile bearbeiten

Um ein Anweisungsprofil im Anweisungsprofil-Editor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Gerätefamilie" das Gerät, für das Sie das Anweisungsprofil bearbeiten möchten.
2. Wählen Sie in der Klappliste "Sprache" die Programmiersprache, für die Sie das Anweisungsprofil bearbeiten möchten.
3. Deaktivieren Sie die Anweisungen bzw. Anweisungsversion, die Sie aus dem Anweisungsprofil ausschließen möchten. Wenn Sie einen Ordner deaktivieren, werden alle unterlagerten Anweisungen deaktiviert.
4. Aktivieren Sie die Anweisungen bzw. Anweisungsversionen, die Sie innerhalb des Anweisungsprofils erlauben möchten.
5. Vergeben Sie bei Bedarf eigene Bausteinnummern.

---

### Hinweis

Für CPUs der Baureihe S7-1200/1500 können Sie Nummern bis 65535 vergeben. Für CPUs der Baureihe S7300/400 finden Sie die Beschränkungen der Nummernbänder im jeweiligen Handbuch der CPU.

---

---

### Hinweis

Wenn Sie das aktive Profil ändern, ist ein neuer Übersetzungsvorgang für alle Bausteine im Projekt notwendig.

---

## Siehe auch

- Grundlagen zu Anweisungsprofilen (Seite 1248)
- Neues Anweisungsprofil erzeugen (Seite 1249)
- Anweisungsprofil aktivieren und deaktivieren (Seite 1251)
- Anweisungsprofil löschen (Seite 1252)
- Anweisungsversionen verwenden (Seite 1247)

## Anweisungsprofil aktivieren und deaktivieren

Damit die Filterung der Anweisungen eines Anweisungsprofils berücksichtigt werden kann, muss das Anweisungsprofil zuerst aktiviert werden. Sie können das aktive Anweisungsprofil

jederzeit wieder deaktivieren, anschließend enthält die Task Card "Anweisungen" wieder den Standardumfang an Anweisungen.

---

**Hinweis**

Es ist ein neuer Übersetzungsvorgang für alle Bausteine im Projekt notwendig.

---

**Voraussetzung**

Der Ordner "Gemeinsame Daten > Anweisungsprofile" in der Projektnavigation ist geöffnet.

**Anweisungsprofil aktivieren**

Um ein Anweisungsprofil zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Anweisungsprofil, das Sie aktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisungsprofil aktivieren".  
Das selektierte Anweisungsprofil ist jetzt das aktive Profil. Anweisungen können nur noch gemäß den Einstellungen dieses Profils verwendet werden.

**Anweisungsprofil deaktivieren**

Um das aktive Anweisungsprofil zu deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Anweisungsprofil, das Sie deaktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisungsprofil deaktivieren".  
Es ist kein Anweisungsprofil aktiv, in der Task Card "Anweisungen" werden wieder alle Anweisungen angezeigt und können verwendet werden.

**Siehe auch**

Grundlagen zu Anweisungsprofilen (Seite 1248)

Neues Anweisungsprofil erzeugen (Seite 1249)

Anweisungsprofil öffnen und bearbeiten (Seite 1250)

Anweisungsprofil löschen (Seite 1252)

**Anweisungsprofil löschen**

**Voraussetzung**

Der Ordner "Gemeinsame Daten > Anweisungsprofile" in der Projektnavigation ist geöffnet.

## Vorgehen

Um ein Anweisungsprofil zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Anweisungsprofil, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

---

### Hinweis

Wenn Sie das aktive Profil löschen, ist ein neuer Übersetzungsvorgang für alle Bausteine im Projekt notwendig.

---

## Ergebnis

Das selektierte Anweisungsprofil wird gelöscht. Wenn das gelöschte Anweisungsprofil das aktive Profil war, ist kein Profil mehr aktiv, sondern es wird in der Task Card "Anweisungen" wieder der Standardumfang an Anweisungen angeboten.

## Siehe auch

- Grundlagen zu Anweisungsprofilen (Seite 1248)
- Neues Anweisungsprofil erzeugen (Seite 1249)
- Anweisungsprofil öffnen und bearbeiten (Seite 1250)
- Anweisungsprofil aktivieren und deaktivieren (Seite 1251)

## Autovervollständigung verwenden

### Grundlagen zur Autovervollständigung

#### Funktion

Im Programmierfenster des Programmiereditors können Sie die Autovervollständigung verwenden, um während des Programmierens bequem auf vorhandene Variablen oder Anweisungen zuzugreifen. Autovervollständigung bedeutet dabei, dass Sie in einem Dialog eine kontextabhängige Liste erhalten, aus der Sie die gewünschten Variablen oder Anweisungen auswählen können.

## Siehe auch

- Autovervollständigung in grafischen Programmiersprachen verwenden (Seite 1254)
- Autovervollständigung in textuellen Programmiersprachen verwenden (Seite 1255)

## Autovervollständigung in grafischen Programmiersprachen verwenden

### Variablen mithilfe der Autovervollständigung einfügen

Um in grafischen Programmiersprachen Variablen mithilfe der Autovervollständigung einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie einen Operanden der Anweisung, dem Sie eine Variable zuweisen möchten.  
Das Eingabefeld für den Operanden wird geöffnet. Neben dem Eingabefeld erscheint die Schaltfläche für die Autovervollständigung.
2. Klicken Sie entweder auf die Schaltfläche für die Autovervollständigung oder geben Sie die Tastenkombination <Strg+I> ein.  
Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur diejenigen lokalen und globalen Variablen, Datenbausteine und Multiinstanzen, die für den Operanden zulässig sind. Mit <Esc> können Sie die Autovervollständigung jederzeit verlassen.
3. Wählen Sie die gewünschte Variable aus der Liste aus. Sie können die Liste bei Bedarf auch filtern:
  - Geben Sie z. B. die Anfangsbuchstaben der Variablen oder Anweisung ein, die Sie einfügen möchten. Mit jedem eingegebenen Buchstaben wird die Autovervollständigung weiter gefiltert. Wenn es keine Variable oder Anweisung gibt, die mit den eingegebenen Buchstaben beginnt, bleibt die Autovervollständigung bei der letzten Übereinstimmung stehen.
  - Geben Sie # ein, um auf die lokalen Variablen aus der Bausteinschnittstelle zuzugreifen.
  - Geben Sie " ein, um auf die globalen Variablen zuzugreifen.
  - Geben Sie % ein, um auf absolute Adressen zuzugreifen.

Falls es sich bei einer Variablen um eine strukturierte Variable, einen Datenbaustein oder eine Multiinstanz handelt, wird am Ende der Zeile ein Pfeil angezeigt. Klicken Sie auf den Pfeil, um die untergeordneten Elemente angezeigt zu bekommen. Auf diese Weise können Sie bis zur letzten Ebene navigieren. Falls für den Operanden eine Struktur als Datentyp zulässig ist, können Sie aus der Liste "Kein Eintrag" auswählen. Dadurch wird die komplette Struktur dem Operanden als Variable zugewiesen. Um zur vorherigen Ebene zurückzukehren, verwenden Sie die <Rücktaste>.

4. Drücken Sie die Taste <Return>, um die Variable zu übernehmen.

### Siehe auch

Grundlagen zur Autovervollständigung (Seite 1253)

Autovervollständigung in textuellen Programmiersprachen verwenden (Seite 1255)

## Autovervollständigung in textuellen Programmiersprachen verwenden

### Variablen und Anweisungen mithilfe der Autovervollständigung einfügen

Um in textuellen Programmiersprachen Variablen und Anweisungen mithilfe der Autovervollständigung einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie den Anfangsbuchstaben der Variable oder Anweisung ein, die Sie einfügen möchten. Sie können bei Bedarf auch direkt nach der Art der Variablen filtern:
  - Geben Sie # ein, um auf die lokalen Variablen aus der Bausteinschnittstelle zuzugreifen.
  - Geben Sie " ein, um auf die globalen Variablen zuzugreifen.
  - Geben Sie % ein, um auf absolute Adressen zuzugreifen.

Die Autovervollständigung wird geöffnet. Sie enthält kontextabhängig nur diejenigen lokalen und globalen Variablen, Datenbausteine, Multiinstanzen und Anweisungen, die an der aktuellen Position zulässig sind. Mit <Esc> können Sie die Autovervollständigung jederzeit verlassen.

2. Geben Sie weitere Buchstaben der Variablen oder Anweisung ein, die Sie einfügen möchten. Mit <Enter> oder <Tab> übernehmen Sie die Variable oder Anweisung und schließen die Autovervollständigung.  
Mit jedem eingegebenen Buchstaben wird die Autovervollständigung weiter gefiltert. Wenn es keine Variable oder Anweisung gibt, die mit den eingegebenen Buchstaben beginnt, enthält die Autovervollständigung lediglich die bisher passenden Treffer.
3. Wählen Sie die gewünschte Variable oder Anweisung aus der Liste aus.  
Falls es sich bei einer Variablen um eine strukturierte Variable, einen Datenbaustein oder eine Multiinstanz handelt, wählen Sie zunächst die Variable, den Datenbaustein oder die Multiinstanz aus der Autovervollständigung aus und übernehmen Sie die Auswahl mit <Enter>. Um die weiteren Bestandteile der Struktur, des Datenbausteins oder der Multiinstanz auszuwählen, geben Sie einen Punkt ein. Anschließend öffnet sich die Autovervollständigung erneut und Sie können den nächsten Bestandteil auswählen.
4. Drücken Sie die Taste <Return>, um die Variable zu übernehmen.

### Siehe auch

Grundlagen zur Autovervollständigung (Seite 1253)

Autovervollständigung in grafischen Programmiersprachen verwenden (Seite 1254)

## Allgemeine Einstellungen für die PLC-Programmierung

### Übersicht über die allgemeinen Einstellungen

#### Überblick

Die folgende Tabelle zeigt die allgemeinen Einstellungen, die Sie vornehmen können:

| Gruppe   | Einstellung   | Beschreibung   |
|--|---|--|
| Ansicht  | mit Kommentaren                                     | Netzwerkcommentare werden angezeigt.   |
|  | mit Variableninformationen                          | Im Programmiereditor werden zusätzliche Informationen für die verwendeten Variablen angezeigt. Diese Einstellung wirkt sich nur auf Bausteine aus, die mit KOP, FUP, AWL oder GRAPH programmiert wurden. |
| Übersetzung  | Aktualparameter löschen beim Schnittstellenabgleich | Aktualparameter werden gelöscht, wenn der zugehörige Formalparameter im aufgerufenen Baustein gelöscht wurde und Sie die Funktion "Bausteinaufruf aktualisieren" ausführen oder den Baustein übersetzen. |
| Voreinstellung für neue Bausteine                    | IEC-Prüfung   | Die Kompatibilität von Operanden in Vergleichsoperationen und arithmetischen Operationen wird nach den Regeln der IEC geprüft. Nicht kompatible Operanden müssen Sie explizit konvertieren.              |
| Speicherreserve für das Laden ohne Reinitialisierung | Speicherreserve für neue Bausteine                  | Definiert die Größe der Reserve im Standardspeicher, die für Schnittstellenerweiterungen genutzt werden kann.  |
| Weitere Einstellungen                                | Autovervollständigung anzeigen                      | Autovervollständigung wird angezeigt.  |
|  | Mnemonik  | Deutsche oder internationale Darstellung der Operationen und Operanden   |

#### Siehe auch

Einstellungen ändern (Seite 1257)

Zulässige Adressen und Datentypen von PLC-Variablen (Seite 1189)

Übersicht der Druckeinstellungen (Seite 174)

Grundlagen zum Bausteinzugriff (Seite 1034)

IEC-Prüfung einstellen oder aufheben (Seite 1135)

## Einstellungen ändern

### Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.

### Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

### Siehe auch

Übersicht über die allgemeinen Einstellungen (Seite 1256)

## 9.1.4.2 Codebausteine programmieren

### Bausteinschnittstelle deklarieren

### Aufbau der Bausteinschnittstelle

### Einführung

Die Schnittstelle enthält die Deklarationen lokaler Variablen, die innerhalb des Bausteins verwendet werden. Die Variablen unterteilen sich in zwei Gruppen:

- Bausteinparameter, die die Schnittstelle des Bausteins für den Aufruf im Programm bilden.
- Lokaldaten, die zum Speichern von Zwischenergebnissen dienen.

Durch die Variablendeklaration bestimmen Sie die Aufrufschnittstelle eines Bausteins im Programm sowie die Namen und Datentypen der Variablen, die Sie im Baustein verwenden möchten.

Die Schnittstelle von Funktionsbausteinen definiert darüber hinaus auch die Struktur der Instanzen, die dem Funktionsbaustein zugeordnet werden.

### Aufbau der Bausteinschnittstelle

Das folgende Bild zeigt den Aufbau der Bausteinschnittstelle. Die Anzahl der Spalten und Abschnitte variiert je nach Bausteinart.

| Name               | Datentyp | Defaultwert | Remanenz       | Sichtbar in HMI                     | Kommentar |
|--------------------|----------|-------------|----------------|-------------------------------------|-----------|
| ▼ Input            |          |             |                |                                     |           |
| ■ MyInput1         | Bool     | false       | Remanent       | <input checked="" type="checkbox"/> |           |
| ▼ Output           |          |             |                |                                     |           |
| ■ MyOutput1        | Byte     | 0           | Nicht remanent | <input checked="" type="checkbox"/> |           |
| ▼ InOut            |          |             |                |                                     |           |
| ■ <neu hinzufügen> |          |             |                | <input type="checkbox"/>            |           |
| ▼ Static           |          |             |                |                                     |           |
| ■ <neu hinzufügen> |          |             |                | <input type="checkbox"/>            |           |
| ▼ Temp             |          |             |                |                                     |           |

### Bausteinparameter

Die folgende Tabelle zeigt die Typen von Bausteinparametern:

| Typ                 | Abschnitt | Funktion  | Verfügbar in  |
|---------------------|-----------|---|---|
| Eingangsparameter   | Input     | Parameter, deren Werte der Baustein liest.  | Funktionen, Funktionsbausteinen und einigen Arten von Organisationsbausteinen |
| Ausgangsparameter   | Output    | Parameter, deren Werte der Baustein schreibt.   | Funktionen und Funktionsbausteinen  |
| Durchgangsparameter | InOut     | Parameter, deren Wert der Baustein beim Aufruf liest und nach der Bearbeitung wieder in denselben Parameter schreibt. | Funktionen und Funktionsbausteinen  |
| Rückgabewert        | Return    | Funktionswert, der an den aufrufenden Baustein zurückgeliefert wird.  | Funktionen  |

Je nach Art des geöffneten Bausteins können weitere Abschnitte angezeigt werden.




## Lokaldaten

Die folgende Tabelle zeigt die Typen von Lokaldaten:

| Typ                  | Abschnitt | Funktion   | Verfügbar in  |
|----------------------|-----------|--|---|
| Temporäre Lokaldaten | Temp      | Variablen, die zum Speichern von temporären Zwischenergebnissen dienen. Temporäre Daten bleiben nur für einen Zyklus erhalten. Wenn Sie temporäre Lokaldaten verwenden, müssen Sie sicherstellen, dass die Werte innerhalb des Zyklus geschrieben werden, in dem Sie sie lesen wollen. Ansonsten sind die Werte zufällig.                              | Funktionen, Funktionsbausteinen und Organisationsbausteinen |
| Statische Lokaldaten | Static    | Variablen, die zum Speichern von statischen Zwischenergebnissen im Instanz-Datenbaustein dienen. Statische Daten bleiben so lange erhalten, bis sie neu geschrieben werden, auch über mehrere Zyklen hinweg. Auch die Namen der Bausteine, die in einem Codebaustein als Multiinstanz aufgerufen werden, werden in den statischen Lokaldaten abgelegt. | Funktionsbausteinen   |

## Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Spalten können Sie bei Bedarf ein- oder ausblenden. Die Anzahl der angezeigten Spalten variiert abhängig von der CPU-Familie und der Art des geöffneten Objekts.

| Spalte  | Erläuterung   |
|---|---|
|  | Symbol, das Sie anklicken können, um die Variable per Drag & Drop in ein Programm zu ziehen und dort als Operand zu verwenden.  |
| Name  | Name der Variablen.   |
| Datentyp  | Datentyp der Variablen.   |
| Offset  | Relative Adresse der Variablen. Diese Spalte ist nur in Bausteinen mit Standardzugriff sichtbar.  |
| Defaultwert   | <p>Wert, mit dem Sie bestimmte Variablen in der Schnittstelle des Codebausteins vorbelegen können.</p> <p>Die Angabe des Defaultwerts ist optional. Wenn Sie keinen Wert festlegen, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert.</p> <p>Der Defaultwert wird im zugehörigen Instanz-Datenbaustein als Startwert übernommen. Diese übernommenen Werte können Sie im Instanz-Datenbaustein mit instanzspezifischen Startwerten ersetzen.</p> <p>Die Spalte ist nur in der Schnittstelle von Funktionsbausteinen vorhanden.</p> |

| Spalte             | Erläuterung  |
|--------------------|--|
| Remanenz           | Kennzeichnet die Variable als remanent.<br>Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten.<br>Diese Spalte ist nur in der Schnittstelle von Funktionsbausteinen mit optimiertem Zugriff sichtbar. |
| Sichtbar in HMI    | Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.   |
| Erreichbar aus HMI | Zeigt an, ob HMI zur Laufzeit auf diese Variable zugreifen kann.   |
| Einstellwert       | Kennzeichnet die Variable als Einstellwert. Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen.<br>Die Spalte ist nur in der Schnittstelle von Funktionsbausteinen vorhanden.                   |
| Kommentar          | Kommentar zur Dokumentation der Variablen.   |

**Siehe auch**

Variablen (Seite 1056)

Schlüsselwörter (Seite 1059)

Gültige Datentypen in der Bausteinschnittstelle (Seite 1261)

Remanenz von lokalen Variablen (Seite 1273)

**Regeln zur Deklaration der Bausteinschnittstelle**

**Allgemeine Regeln zur Deklaration der Bausteinschnittstelle**

**Verwenden von Bausteinparametern**

Für die Verwendung der Bausteinparameter innerhalb des Bausteins gelten die folgenden Regeln:

- Eingangsparameter dürfen nur gelesen werden.
- Ausgangsparameter dürfen nur geschrieben werden.
- Durchgangsparameter dürfen gelesen und geschrieben werden.

**Zuordnen von Defaultwerten zu Bausteinparametern**

Sie können bestimmten Parametern in der Schnittstelle von Funktionsbausteinen Defaultwerte zuordnen. Die Möglichkeit der Zuordnung ist abhängig vom Deklarationsabschnitt und Datentyp des Parameters.

Die folgende Tabelle zeigt, welchen Parametern ein Defaultwert zugeordnet werden kann:

| Parameterart         | Abschnitt | Zuordnung eines Defaultwerts möglich |                          |                |
|----------------------|-----------|--------------------------------------|--------------------------|----------------|
|                      |           | Elementare Datentypen                | Strukturierte Datentypen | Parametertypen |
| Eingangsparameter    | Input     | X                                    | X                        | -              |
| Ausgangsparameter    | Output    | X                                    | X                        | -              |
| Durchgangsparameter  | InOut     | X                                    | -                        | -              |
| Statische Lokaldaten | Static    | X                                    | X                        | -              |
| Temporäre Lokaldaten | Temp      | -                                    | -                        | -              |

**Siehe auch**

Variablen (Seite 1056)

Schlüsselwörter (Seite 1059)

**Gültige Datentypen in der Bausteinschnittstelle**

**Gültige Datentypen in der Bausteinschnittstelle in S7-1200**

Die folgende Tabelle zeigt, welche Datentypen Sie in den Parametern in den einzelnen Abschnitten der Schnittstelle zuweisen können.

| Abschnitt                    | Standard-Datentypen | ARRAY<br>STRUCT<br>STRING<br>DT | VOID | VARIANT |
|------------------------------|---------------------|---------------------------------|------|---------|
| <b>Organisationsbaustein</b> |                     |                                 |      |         |
| Temp                         | X                   | X                               | -    | X       |
| <b>Funktionsbaustein</b>     |                     |                                 |      |         |
| Input                        | X                   | X                               | -    | X       |
| Output                       | X                   | X                               | -    | -       |
| InOut                        | X                   | X <sup>(1)</sup>                | -    | X       |
| Static                       | X                   | X                               | -    | -       |
| Temp                         | X                   | X                               | -    | X       |
| <b>Funktion</b>              |                     |                                 |      |         |
| Input                        | X                   | X <sup>(1)</sup>                | -    | X       |
| Output                       | X                   | X <sup>(1)</sup>                | -    | X       |
| InOut                        | X                   | X <sup>(1)</sup>                | -    | X       |
| Temp                         | X                   | X                               | -    | X       |
| Return                       | X                   | X                               | X    | -       |

<sup>(1)</sup> STRING kann nur in der Standardlänge von 254 Zeichen definiert werden.

**Gültige Datentypen in der Bausteinschnittstelle in S7-1500**

Die folgende Tabelle zeigt, welche Datentypen Sie in den Parametern in den einzelnen Abschnitten der Schnittstelle zuweisen können.

| Abschnitt   | Standard-Datentypen | ARRAY<br>STRUCT<br>STRING<br>DT | Parametertypen | VOID | DB_ANY | POINTER | ANY  | VARIANT |
|---|---------------------|---------------------------------|----------------|------|--------|---------|------|---------|
| <b>Organisationsbaustein</b>  |                     |                                 |                |      |        |         |      |         |
| Temp  | X                   | X                               | _(4)           | -    | X      | -       | X(3) | X       |
| <b>Funktionsbaustein</b>  |                     |                                 |                |      |        |         |      |         |
| Input   | X                   | X                               | X              | -    | X      | X       | X    | X       |
| Output  | X                   | X                               | -              | -    | X      | -       | -    | -       |
| InOut   | X                   | X(1)                            | _(4)           | -    | X      | X       | X    | X       |
| Static  | X                   | X                               | -              | -    | X      | -       | -    | -       |
| Temp  | X                   | X                               | _(4)           | -    | -      | -       | X(3) | X       |
| <b>Funktion</b>   |                     |                                 |                |      |        |         |      |         |
| Input   | X                   | X(1)                            | X              | -    | X      | X       | X    | X       |
| Output  | X                   | X(1)                            | -              | -    | X      | X       | X    | X       |
| InOut   | X                   | X(1)                            | _(4)           | -    | X      | X       | X    | X       |
| Temp  | X                   | X                               | _(4)           | -    | X      | -       | X(3) | X       |
| Return  | X                   | x                               | -              | X    | X      | X       | x(2) | -       |
| (1) STRING kann nur in der Standardlänge von 254 Zeichen definiert werden.<br>(2) In SCL ist ANY nicht als Rückgabewert zulässig.<br>(3) ANY kann nur in Bausteinen mit Standardzugriff im Abschnitt "Temp" verwendet werden.<br>(4) Der Parametertyp "INSTANCE" ist als einzige Ausnahme in den Abschnitten "TEMP" und "InOut" zulässig. |                     |                                 |                |      |        |         |      |         |

**Lokale Variablen deklarieren**

**Lokale Variablen in der Bausteinschnittstelle deklarieren**

**Voraussetzung**

Die Bausteinschnittstelle ist geöffnet.

**Vorgehen**

Um eine Variable elementaren Datentyps zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Schnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Name" einen Variablennamen ein.

3. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
4. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten der Bausteinschnittstelle angezeigt werden.

## Ergebnis

Die Variable wird angelegt.

## Syntaxprüfung

Nach jeder Eingabe wird eine Syntaxprüfung durchgeführt, die vorhandene Fehler rot anzeigt. Sie müssen diese Fehler nicht sofort beheben, sondern können mit der weiteren Bearbeitung fortfahren und die Korrekturen zu einem späteren Zeitpunkt durchführen. Solange die Variablendeklaration Syntaxfehler enthält, lässt sich das Programm allerdings nicht übersetzen.

---

### Hinweis

Wenn Sie die Schnittstelle eines Bausteins ändern, werden die Aufrufe des Bausteins im Programm möglicherweise inkonsistent. Die Aufrufstellen werden, wenn möglich, automatisch aktualisiert.

Wenn eine automatische Aktualisierung nicht möglich ist, müssen Sie die inkonsistenten Bausteinaufrufe manuell aktualisieren.

Siehe auch:

Bausteinaufrufe in KOP aktualisieren (Seite 1300)

Bausteinaufrufe in FUP aktualisieren (Seite 1342)

---

## Siehe auch

Bearbeiten von Tabellen (Seite 216)

Grundlagen zu Startwerten (Seite 1419)

Variablen (Seite 1056)

Schlüsselwörter (Seite 1059)

Eigenschaften von lokalen Variablen (Seite 1272)

Remanenz von lokalen Variablen (Seite 1273)

## Lokale Variablen im Programmiereditor deklarieren

## Voraussetzung

Der Programmiereditor ist geöffnet.

## Vorgehen

Um eine lokale Variable zu deklarieren, gehen Sie folgendermaßen vor:

1. Fügen Sie eine Anweisung in Ihr Programm ein.  
Die Zeichenfolgen "<???">", "<???.?>" oder "..." stellen Operandenplatzhalter dar.
2. Ersetzen Sie einen Operandenplatzhalter mit dem Namen der zu erstellenden Variablen.
3. Selektieren Sie den Variablennamen.  
Wenn Sie mehrere Variablen deklarieren wollen, selektieren Sie die Namen aller zu deklarierenden Variablen.
4. Wählen Sie im Kontextmenü den Befehl "Variable definieren".  
Der Dialog "Variable definieren" wird geöffnet. Er zeigt eine Deklarationstabelle, in die bereits der Variablenname eingetragen ist.
5. Um eine lokale Variable zu deklarieren, wählen Sie einen der folgenden Abschnitte aus:
  - Local In
  - Local Out
  - Local InOut
  - Local Static
  - Local Temp
6. Geben Sie in den weiteren Spalten Datentyp und Kommentar ein.
7. Klicken Sie auf die Schaltfläche "Definieren", um die Eingabe abzuschließen.

## Ergebnis

Die Deklaration wird direkt in die Bausteinschnittstelle geschrieben und ist innerhalb des gesamten Bausteins gültig.

---

### Hinweis

Wenn Sie die Schnittstelle eines Bausteins ändern, werden die Aufrufe des Bausteins im Programm möglicherweise inkonsistent. Die Aufrufstellen werden, wenn möglich, automatisch aktualisiert.

Wenn eine automatische Aktualisierung nicht möglich ist, müssen Sie die inkonsistenten Bausteinaufrufe manuell aktualisieren.

Siehe auch:

Bausteinaufrufe in KOP aktualisieren (Seite 1300)

Bausteinaufrufe in FUP aktualisieren (Seite 1342)

---

## Siehe auch

- Bearbeiten von Tabellen (Seite 216)
- Variablen (Seite 1056)
- Schlüsselwörter (Seite 1059)
- Grundlagen zu Startwerten (Seite 1419)
- Eigenschaften von lokalen Variablen (Seite 1272)
- Remanenz von lokalen Variablen (Seite 1273)

## Variablen vom Datentyp ARRAY deklarieren

### Voraussetzung

Die Bausteinschnittstelle ist geöffnet.

### Vorgehen

Um eine Variable vom Datentyp ARRAY zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Schnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
3. Klicken Sie in der Spalte "Datentyp" auf die Schaltfläche für die Datentypauswahl. Eine Liste der zulässigen Datentypen wird geöffnet.
4. Wählen Sie den Datentyp "Array" aus. Der Dialog "Array" wird geöffnet.
5. Geben Sie im Eingabefeld "Datentyp" den Datentyp der Array-Elemente an.
6. Geben Sie anschließend im Eingabefeld "Array-Grenzen" für jede Dimension die Ober- und Untergrenze an.  
Beispiel für ein eindimensionales ARRAY:  
[0..3]  
Beispiel für ein dreidimensionales ARRAY:  
[0..3, 0..15, 0..33]
7. Bestätigen Sie die Eingabe.
8. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten der Bausteinschnittstelle angezeigt werden.

### Ergebnis

Die Variable vom Datentyp ARRAY wird angelegt.

---

#### Hinweis

Für ARRAY-Elemente können Sie keine spezifischen Defaultwerte definieren. Sie können ihnen jedoch in der Instanz Startwerte zuweisen.

---

## Siehe auch

- Variablen (Seite 1056)
- Schlüsselwörter (Seite 1059)
- Eigenschaften von lokalen Variablen (Seite 1272)
- Remanenz von lokalen Variablen (Seite 1273)
- Bearbeiten von Tabellen (Seite 216)

## Variablen vom Datentyp STRUCT deklarieren

### Voraussetzung

Die Bausteinschnittstelle ist geöffnet.

### Vorgehen

Um eine Variable vom Datentyp STRUCT zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Schnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
3. Tragen Sie in der Spalte "Datentyp" "Struct" ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.  
Nach der neuen Variablen wird eine leere, eingerückte Zeile eingefügt.
4. Fügen Sie in der leeren Zeile das erste Strukturelement ein.  
Nach dem Element wird erneut eine leere Zeile eingefügt.
5. Wählen Sie einen Datentyp für das Strukturelement aus.
6. Optional: Ändern Sie die Eigenschaften des Strukturelements, die in den weiteren Spalten der Bausteinschnittstelle angezeigt werden.
7. Wiederholen Sie Schritt 4 bis 7 für alle weiteren Strukturelemente.  
Ein explizites Beenden der Struktur ist nicht notwendig. Die Struktur endet mit dem letzten eingetragenen Element.
8. Um eine neue Variable nach der Struktur einzufügen, lassen Sie eine Zeile am Ende der Struktur leer und tragen Sie die neue Variable in die zweite leere Zeile ein.

### Ergebnis

Die Variable vom Datentyp STRUCT wird angelegt.



**Siehe auch**

- Variablen (Seite 1056)
- Schlüsselwörter (Seite 1059)
- Eigenschaften von lokalen Variablen (Seite 1272)
- Remanenz von lokalen Variablen (Seite 1273)
- Bearbeiten von Tabellen (Seite 216)

**Variablen basierend auf einem PLC-Datentyp deklarieren****Voraussetzung**

In der aktuellen CPU ist ein PLC-Datentyp deklariert.

**Vorgehen**

Um eine Variable basierend auf einem PLC-Datentyp zu deklarieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Schnittstelle den gewünschten Deklarationsabschnitt aus.
2. Tragen Sie in der Spalte "Datentyp" den PLC-Datentyp ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.

**Ergebnis**

Die Variable wird angelegt.

---

**Hinweis**

Die Defaultwerte der Variablen innerhalb eines PLC-Datentyps definieren Sie beim Anlegen des PLC-Datentyps. An der Verwendungsstelle des PLC-Datentyps können Sie diese Werte nicht ändern.

Wenn Sie PLC-Datentypen ändern oder löschen, die in der Bausteinschnittstelle verwendet werden, wird die Schnittstelle inkonsistent. Um diese Inkonsistenz zu beheben, müssen Sie die Schnittstelle aktualisieren.

Siehe auch: Bausteinschnittstelle aktualisieren (Seite 1269)

---

**Siehe auch**

- Bearbeiten von Tabellen (Seite 216)
- Grundlagen zu PLC-Datentypen (Seite 1436)

## Überlagerte Variablen deklarieren

### Einführung

Um auf Datenbereiche innerhalb einer deklarierten Variablen zuzugreifen, können Sie die deklarierten Variablen mit einer weiteren Deklaration überlagern. Sie haben so die Möglichkeit, eine bereits deklarierte Variable mit einem anderen Datentyp anzusprechen. Sie können z. B. die einzelnen Bits einer Variablen vom Datentyp WORD mit einem ARRAY of BOOL ansprechen.

### Variable überlagern

Um eine Variable mit einem neuen Datentyp zu überlagern, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Wählen Sie in der Schnittstelle die Variable aus, die Sie mit einem neuen Datentyp überlagern möchten.
3. Klicken Sie in der Funktionsleiste auf "Zeile hinzufügen".  
Nach der zu überlagernden Variable wird eine Zeile hinzugefügt. Die überlagernde Variable muss in der Zeile direkt nach der überlagerten Variable deklariert werden.
4. Geben Sie in der Spalte "Name" einen Variablennamen ein.
5. Tragen Sie in der Spalte "Datentyp" den Eintrag "AT" ein. Sie werden dabei von der Autovervollständigung unterstützt.  
Der Eintrag in der Spalte "Name" wird um folgenden Zusatz ergänzt:  
"AT<Name der überlagerten Variable>"
6. Klicken Sie erneut auf die Schaltfläche zur Datentypauswahl und wählen Sie den Datentyp für die neue Variable aus.  
Die Variable wird angelegt. Sie zeigt auf dieselben Daten wie die überlagerte Variable, interpretiert diese jedoch mit dem neuen Datentyp.

### Überlagerung entfernen

Um die Überlagerung einer Variablen wieder zu entfernen, gehen Sie folgendermaßen vor:

1. Wählen Sie die überlagernde Variable aus, die Sie entfernen wollen.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Die Überlagerung wird entfernt.

### Siehe auch

Bearbeiten von Tabellen (Seite 216)

Variablen mit AT überlagern (Seite 1074)

## Multiinstanzen deklarieren

### Voraussetzung

- Der aufzurufende Funktionsbaustein ist in der Projektnavigation vorhanden und multiinstanzfähig.
- Die Bausteinschnittstelle des aufrufenden Funktionsbausteins ist geöffnet.

### Vorgehen

Um einen aufzurufenden Funktionsbaustein als Multiinstanz zu deklarieren, gehen Sie folgendermaßen vor:

1. Geben Sie im Abschnitt "Static" in der Spalte "Name" eine Bezeichnung für den Bausteinaufruf ein.
2. Geben Sie in der Spalte "Datentyp" einen symbolischen Namen für den aufzurufenden Funktionsbaustein an.

---

### Hinweis

Der Programmiereditor deklariert die Multiinstanz selbst, wenn Sie in einem Netzwerk einen Bausteinaufruf programmieren und im nachfolgenden Dialog "Aufrufoptionen" angeben, dass Sie den Baustein als Multiinstanz aufrufen wollen.

---

### Siehe auch

Bausteinschnittstelle aktualisieren (Seite 1269)

## Bausteinschnittstelle aktualisieren

### Einführung

Wenn Sie PLC-Datentypen oder Multiinstanzen ändern oder löschen, die in der Bausteinschnittstelle verwendet werden, wird die Schnittstelle inkonsistent. Um diese Inkonsistenz zu beheben, müssen Sie die Schnittstelle aktualisieren.

Sie haben zwei Möglichkeiten, die Bausteinschnittstelle zu aktualisieren:

- Explizites Aktualisieren der Bausteinschnittstelle.  
Die verwendeten PLC-Datentypen und Multiinstanzen werden aktualisiert. Die zum Baustein gehörenden Instanz-Datenbausteine werden dabei nicht implizit aktualisiert.
- Implizites Aktualisieren während des Übersetzens.  
Alle verwendeten PLC-Datentypen und Multiinstanzen sowie die zugehörigen Instanz-Datenbausteine werden aktualisiert.

### Explizites Aktualisieren der Bausteinschnittstelle

Um die Bausteinschnittstelle explizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Wählen Sie im Kontextmenü den Befehl "Aktualisieren".

### Implizites Aktualisieren während des Übersetzens

Um alle Verwendungen von PLC-Datentypen und Multiinstanzen sowie die Instanz-Datenbausteine während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

### Siehe auch

Grundlagen zu PLC-Datentypen (Seite 1436)

Variablen basierend auf einem PLC-Datentyp deklarieren (Seite 1267)

Bearbeiten von Tabellen (Seite 216)

Grundlagen zu Startwerten (Seite 1419)

Variablen (Seite 1056)

Schlüsselwörter (Seite 1059)

Eigenschaften von lokalen Variablen (Seite 1272)

Remanenz von lokalen Variablen (Seite 1273)

Bausteinaufrufe in KOP aktualisieren (Seite 1300)

Multiinstanzen deklarieren (Seite 1269)

### Bausteinschnittstelle erweitern

#### Beschreibung

Um PLC-Programme, die bereits in Betrieb genommen wurden und fehlerfrei auf einer Anlage laufen, nachträglich zu ändern, bieten CPUs der Baureihe S7-1500 und die meisten CPUs der Baureihe S7-1200 V4 die Möglichkeit, die Schnittstellen von Funktionsbausteinen im laufenden Betrieb zu erweitern.

Die geänderten Bausteine können geladen werden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen.

Programmänderungen lassen sich so auf einfache Weise einbringen. Der gesteuerte Prozess wird durch diesen Ladevorgang (Laden ohne Reinitialisierung) nicht gestört.

## Funktionsweise

Grundsätzlich enthält jeder Funktionsbaustein per Voreinstellung eine Speicherreserve. Die Speicherreserve wird zunächst nicht genutzt. Wenn Sie den Baustein übersetzt und geladen haben und anschließend feststellen, dass sie Schnittstellenänderungen nachladen wollen, aktivieren Sie die Speicherreserve. Alle Variablen, die Sie daraufhin deklarieren, werden in die Speicherreserve gelegt. Beim anschließenden Laden werden die Werte bereits geladener Variablen nicht beeinflusst, der laufende Betrieb wird nicht gestört.

Um Ihr Programm zu einem späteren Zeitpunkt zu überarbeiten, während die Anlage gerade nicht in Betrieb ist, haben Sie zusätzlich die Möglichkeit, das Speicherlayout einzelner oder mehrerer Bausteine in einem Schritt zu überarbeiten. Dabei werden alle Variablen aus dem Reservebereich in den regulären Bereich verschoben. Die Speicherreserve wird wieder frei für weitere Schnittstellenerweiterungen.

## Voraussetzungen

Die Funktion "Laden ohne Reinitialisierung" steht zur Verfügung, wenn folgende Voraussetzungen erfüllt sind:

- Das Projekt liegt im Format "TIA-Portal V12" vor.
- Sie arbeiten mit einer CPU , die das "Laden ohne Reinitialisierung" unterstützt.
- Die Bausteine wurden in KOP, FUP, AWL oder SCL erstellt.
- Die Bausteine wurden vom Anwender erstellt, d.h. es sind keine mitgelieferten Bausteine.
- Es handelt sich um Bausteine mit optimiertem Zugriff.

## Grundsätzliche Schritte

Um die Schnittstelle eines Funktionsbausteins zu erweitern und den Baustein anschließend ohne Reinitialisierung zu laden, führen Sie folgende Schritte aus.

1. Alle Bausteine haben per Voreinstellung eine Speicherreserve von 100 Byte. Bei Bedarf können Sie die Größe der Speicherreserve anpassen.
2. Aktivieren Sie die Speicherreserve.
3. Erweitern Sie die Bausteinschnittstelle.
4. Übersetzen Sie den Baustein.
5. Laden Sie den Baustein wie gewohnt in die CPU.

Detaillierte Beschreibungen der einzelnen Schritte finden Sie im Kapitel "Bausteine laden (S7-1200/1500)".

**Hinweis**

Die Funktion "Laden ohne Reinitialisierung" steht Ihnen nur in CPUs der Baureihen S7-1500 und S7-1200 V4 in vollem Umfang zur Verfügung.

Grundsätzlich haben Sie jedoch in allen CPU-Familien folgende Möglichkeiten, die Schnittstelle von Funktionsbausteinen zu erweitern und neu deklarierte Variablen rückwirkungsfrei zu laden:

- Sie können im Abschnitt "Temp" neue Variablen hinzufügen und diese laden, ohne den Prozess zu beeinflussen.
- Sie können im Abschnitt "InOut" neue Variablen strukturierten Datentyps anlegen und diese laden, ohne den Prozess zu beeinflussen.

**Eigenschaften von lokalen Variablen bearbeiten**

**Eigenschaften von lokalen Variablen**

**Eigenschaften**

Die folgende Tabelle zeigt eine Übersicht über die Eigenschaften von lokalen Variablen:

| Gruppe    | Eigenschaft        | Beschreibung   |
|-----------|--------------------|--|
| Allgemein | Name               | Name der Variablen.  |
|           | Datentyp           | Datentyp der Variablen.  |
|           | Defaultwert        | Wert, mit dem Sie die Variable in der Schnittstelle des Codebausteins vorbelegen.<br>Die Angabe des Defaultwerts ist optional. Wenn Sie keinen Wert festlegen, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert.<br>Der Defaultwert wird in der zugehörigen Instanz als Startwert übernommen. Diese übernommenen Werte können Sie mit instanzspezifischen Startwerten ersetzen.<br>Diese Eigenschaft ist nur in der Schnittstelle von Funktionsbausteinen verfügbar. |
|           | Kommentar          | Kommentar zur Variablen.   |
| Attribute | Remanenz           | Kennzeichnet die Variable als remanent.<br>Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten.<br>Dieses Attribut ist nur in der Schnittstelle von Funktionsbausteinen mit optimiertem Zugriff verfügbar.   |
|           | Erreichbar aus HMI | Zeigt an, ob HMI zur Laufzeit auf diese Variable zugreifen kann.   |

| Gruppe | Eigenschaft                             | Beschreibung   |
|--------|---|--|
|        | Sichtbar in HMI                         | Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.   |
|        | Versteckter Parameter                   | Zeigt an, ob die Variable beim Bausteinaufruf verborgen werden soll. Dies ist nur möglich, wenn Sie einen gültigen vordefinierten Aktualparameter angegeben haben. |
|        | Vordefinierter Aktualparameter          | Definiert einen Parameter, der beim Bausteinaufruf als Aktualparameter verwendet werden soll.  |
|        | Parametrierbar                          | Kennzeichnet, ob ein Parameter in CFC parametrierbar ist.  |
|        | Für Test                                | Kennzeichnet, ob ein Parameter für den Testmodus von CFC angemeldet ist.   |
|        | Sichtbar                                | Kennzeichnet, ob ein Parameter im CFC sichtbar ist.  |
|        | Verschaltbar                            | Kennzeichnet, ob ein Parameter in CFC verschaltbar ist.  |
|        | Rücklesbarkeit für Variablen aktivieren | Kennzeichnet, ob ein Parameter für die Funktion "Plan rücklesen" in CFC relevant ist.  |
|        | Enumerationstexte                       | Ordnet einen Parameter einer Enumeration in CFC zu.  |
|        | Engineering Einheit                     | Ordnet einen Parameter einer Einheit in CFC zu.  |
|        | Untergrenze                             | Definiert den unteren Grenzwert für den Parameter in CFC:  |
|        | Obergrenze                              | Definiert den oberen Grenzwert für den Parameter in CFC:   |

## Siehe auch

Remanenz von lokalen Variablen (Seite 1273)

Eigenschaften von lokalen Variablen ändern (Seite 1274)

Schlüsselwörter (Seite 1059)

## Remanenz von lokalen Variablen

### Einführung

Funktionsbausteine speichern ihre Daten in einer Instanz. Um den Verlust dieser Daten bei Spannungsausfall zu vermeiden, können Sie Daten als remanent kennzeichnen. Diese werden in einem remanenten Speicherbereich abgelegt. Die Möglichkeiten zur Einstellung der Remanenz sind abhängig von der eingestellten Zugriffsart des Funktionsbausteins.

### Remanenzverhalten in Bausteinen mit Standardzugriff

In Bausteinen mit Standardzugriff können Sie das Remanenzverhalten einzelner Variablen nicht einstellen. Sie können Sie lediglich in der zugeordneten Instanz als remanent definieren. Alle darin enthaltenen Variablen gelten dann als remanent.

## Remanenz bei optimiertem Bausteinzugriff

In Datenbausteinen mit optimiertem Zugriff können Sie das Remanenzverhalten einzelner Variablen bestimmen.

Bei Variablen strukturierten Datentyps gilt die Remanenzeinstellung immer für die gesamte Struktur. Für einzelne Elemente innerhalb der Struktur können Sie keine individuelle Remanenzeinstellung vornehmen.

Im Abschnitt "InOut" können Sie keine remanenten Variablen strukturierten Datentyps anlegen. Durchgangsparameter mit strukturiertem Datentyp, z. B. ARRAY, STRUCT, oder STRING, sind immer nicht-remanent.

Folgende Einstellungen stehen zur Verfügung:

- Remanent  
Die Werte der Variablen oder der Struktur stehen auch nach Spannungsausfall zur Verfügung.
- Nicht remanent  
Die Werte der Variablen oder der Struktur gehen bei Spannungsausfall verloren.
- Im IDB setzen  
Die Remanenz kann im Instanz-Datenbaustein eingestellt werden. Die Einstellung, die im Instanz-Datenbaustein vorgenommen wird, gilt dann jedoch zentral für alle Variablen, die mit "Im IDB setzen" markiert sind.

## Siehe auch

Eigenschaften von lokalen Variablen (Seite 1272)

Grundlagen zum Bausteinzugriff (Seite 1034)

## Eigenschaften von lokalen Variablen ändern

### Eigenschaften in der Bausteinschnittstelle bearbeiten

Um die Eigenschaften einer oder mehrerer Variablen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Bausteinschnittstelle.
2. Ändern Sie die Einträge in den Spalten.

### Eigenschaften im Eigenschaftsfenster bearbeiten

Um Eigenschaften einer einzelnen Variable zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle eine Variable.  
Die Eigenschaften der Variablen werden im Inspektorfenster angezeigt.
2. Ändern Sie die Einträge im Inspektorfenster.



### Variablen direkt im Programmiereditor umbenennen

Um eine oder mehrere Variablen umzubenennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere Variablen im Programm.
2. Wählen Sie im Kontextmenü den Befehl "Variable umbenennen".  
Der Dialog "Variable umbenennen" wird geöffnet. Er zeigt eine Deklarationstabelle mit den selektierten Variablen.
3. Ändern Sie die Einträge in der Spalte "Name".
4. Bestätigen Sie die Eingabe mit der Schaltfläche "Ändern".

### Datentyp oder Kommentar im Programmiereditor bearbeiten

Um den Datentyp oder den Kommentar einer Variablen im Programmiereditor zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Variablennamen.
2. Wählen Sie im Kontextmenü den Befehl "Variable umverdrahten".  
Der Dialog "Variable umverdrahten" wird geöffnet. Er zeigt eine Deklarationstabelle.
3. Ändern Sie den Eintrag in den Spalten "Datentyp" oder "Kommentar".
4. Klicken Sie auf die Schaltfläche "Ändern", um die Eingabe zu bestätigen.

### Auswirkung im Programm

Bei einer Änderung des Namens, des Datentyps oder der Adresse einer Variablen wird automatisch jede Verwendungsstelle der Variablen im Programm aktualisiert.

---

#### Hinweis

Wenn Sie die Schnittstelle eines Bausteins ändern, wird das Programm möglicherweise inkonsistent. Die Inkonsistenzen werden, wenn möglich, automatisch aktualisiert.

Wenn eine automatische Aktualisierung nicht möglich ist, werden die inkonsistenten Aufrufe rot markiert. Sie müssen die Inkonsistenzen dann manuell aktualisieren.

Siehe auch:

Bausteinaufrufe in KOP aktualisieren (Seite 1300)

Bausteinaufrufe in FUP aktualisieren (Seite 1342)

---

## Siehe auch

- Aufbau der Bausteinschnittstelle (Seite 1257)
- Bearbeiten von Tabellen (Seite 216)
- Eigenschaften von lokalen Variablen (Seite 1272)
- Remanenz von lokalen Variablen (Seite 1273)
- Grundlagen zu Startwerten (Seite 1419)
- Variablen (Seite 1056)
- Schlüsselwörter (Seite 1059)
- Bausteinschnittstelle aktualisieren (Seite 1269)

## Bausteinschnittstelle bearbeiten

### Tabellenzeilen einfügen

#### Vorgehen

Um eine Zeile vor der selektierten Zeile einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile einfügen".

#### Ergebnis

Vor der markierten Zeile wird eine neue Zeile eingefügt.

## Siehe auch

- Bearbeiten von Tabellen (Seite 216)

### Tabellenzeilen hinzufügen

#### Vorgehen

Um eine neue Zeile nach der selektierten Zeile hinzuzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, nach der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile hinzufügen".

#### Ergebnis

Nach der selektierten Zeile wird eine neue, leere Zeile hinzugefügt.

**Siehe auch**

Bearbeiten von Tabellen (Seite 216)

**Variablen löschen****Vorgehen**

Um eine Variable zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Variablen. Sie können auch mehrere Zeilen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

**Siehe auch**

Bearbeiten von Tabellen (Seite 216)

**Aufeinander folgende Zellen automatisch ausfüllen**

Sie können den Inhalt einer oder mehrerer Tabellenzellen in die darunter liegenden Zellen übertragen und so aufeinander folgende Zellen automatisch ausfüllen.

Wenn Sie Zellen in der Spalte "Name" automatisch ausfüllen, wird jeder Name um eine fortlaufende Nummer erweitert. Aus dem Namen "Motor" wird z. B. der Name "Motor\_1".

Als Quellbereich können Sie eine einzelne oder mehrere Zellen und komplette Zeilen definieren.

Sollten in der geöffneten Tabelle weniger Zeilen vorhanden sein, als Sie füllen möchten, müssen Sie zunächst weitere Leerzeilen einfügen.

**Voraussetzung**

- Die Tabelle ist geöffnet.
- Es stehen genügend Deklarationszeilen zur Verfügung.

**Vorgehen**

Um aufeinander folgende Zellen automatisch auszufüllen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu übertragenden Zellen.
2. Klicken Sie auf das Ausfüllzeichen in der rechten unteren Ecke der Zelle. Der Mauszeiger wird zu einem Fadenkreuz.
3. Halten Sie die Maustaste gedrückt und ziehen Sie den Mauszeiger nach unten über die Zellen, die Sie automatisch ausfüllen wollen.

4. Lassen Sie die Maustaste los.  
Die Zellen werden automatisch ausgefüllt.
5. Wenn in den automatisch auszufüllenden Zellen bereits Einträge vorhanden sind, erscheint ein Dialog. In diesem Dialog können Sie angeben, ob Sie die bestehenden Einträge überschreiben wollen, oder ob Sie für die neuen Variablen neue Zeilen einfügen möchten.

### Siehe auch

Bearbeiten von Tabellen (Seite 216)

### Tabellenspalten ein- und ausblenden

Sie können die Spalten in Tabellen bei Bedarf ein- und ausblenden.

### Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".  
Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.

### Variablen mit externen Editoren bearbeiten

Um einzelne Variablen in externen Tabelleneditoren, z. B. in Excel, zu bearbeiten, können Sie sie per Copy & Paste exportieren oder importieren. Strukturierte Variablen können Sie allerdings nicht in einen Editor kopieren.

### Voraussetzung

Die Bausteinschnittstelle und ein externer Editor sind geöffnet.

### Vorgehen

Um einzelne Variablen in einen externen Editor zu exportieren und wieder zu importieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere Variablen.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wechseln Sie in den externen Editor und fügen Sie die kopierten Variablen dort ein.
4. Bearbeiten Sie die Variablen nach Bedarf.
5. Kopieren Sie die Variablen im externen Editor.
6. Selektieren Sie die Variablen im externen Editor.

7. Wechseln Sie wieder in die Bausteinschnittstelle.
8. Wählen Sie im Kontextmenü den Befehl "Einfügen".

## Programmcode erstellen

## KOP-Programme erstellen

## Grundlagen zu KOP

## Programmiersprache KOP

### Übersicht über die Programmiersprache KOP (Kontaktplan)

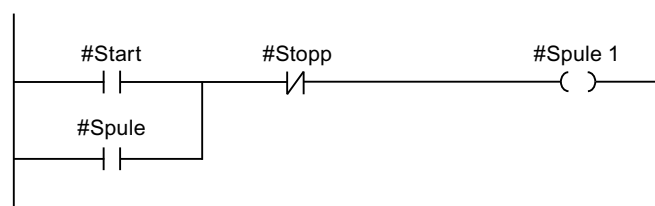
KOP ist eine grafische Programmiersprache. Die Darstellung ist Stromlaufplänen nachempfunden.

Das Programm wird in einem oder mehreren Netzwerken abgebildet. Ein Netzwerk enthält am linken Rand eine Stromschiene, von der die Strompfade ausgehen. Auf den Strompfaden werden die Abfragen der binären Signale in Form von Kontakten angeordnet. Durch serielle Anordnung der Elemente auf einem Strompfad entsteht eine Reihenschaltung, durch Anordnung auf parallelen Zweigen entsteht eine Parallelschaltung. Komplexe Funktionen werden durch Boxen dargestellt.

### Beispiel für Netzwerke in KOP

Das folgende Bild zeigt ein KOP-Netzwerk mit zwei Schließerkontakten, einem Öffnerkontakt und einer Spule:

Netzwerk 1



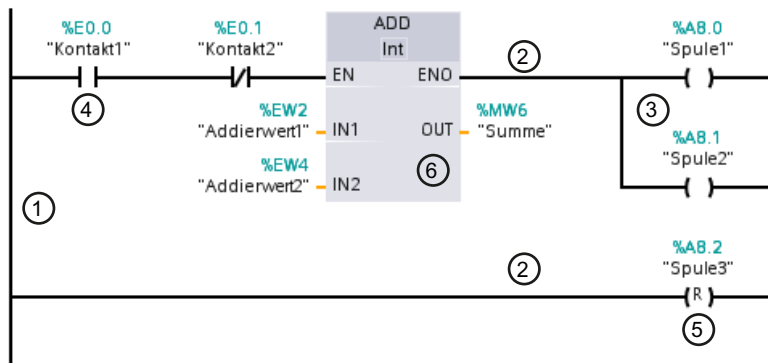
### Übersicht über die KOP-Elemente

#### KOP-Elemente

Ein KOP-Programm besteht aus einzelnen Elementen, die Sie an der Stromschiene eines Netzwerks in Reihe oder parallel zueinander anordnen können. Die meisten Programmelemente müssen mit Variablen versorgt werden.

Von der Stromschiene geht mindestens ein Strompfad aus. Die Programmierung des Netzwerks beginnt am linken Rand des Strompfads. Die Stromschiene können Sie durch mehrere Strompfade und Zweige erweitern.

Das folgende Bild zeigt beispielsweise Elemente eines KOP-Netzwerks:



- 1) Stromschiene
- 2) Strompfad
- 3) Zweig
- 4) Kontakt
- 5) Spule
- 6) Box

### Stromschiene

Jedes KOP-Netzwerk besteht aus einer Stromschiene, die mindestens einen Strompfad enthält. Ein Netzwerk kann durch Einfügen weiterer Strompfade erweitert werden. In den einzelnen Strompfaden können Sie mithilfe von Zweigen Parallelschaltungen programmieren.

### Kontakte

Kontakte dienen dazu, eine stromleitende Verbindung zwischen zwei Elementen herzustellen oder zu unterbrechen. Der Strom wird von links nach rechts weitergeleitet. Mit Kontakten können Sie den Signalzustand oder den Wert eines Operanden abfragen und abhängig vom Ergebnis den Stromfluss kontrollieren.

Die folgenden Arten von Kontakten stehen Ihnen in einem KOP-Programm zur Verfügung:

- **Schließerkontakt:**  
Schließerkontakte leiten den Strom weiter, wenn der Signalzustand eines angegebenen Binäroperanden "1" ist.
- **Öffnerkontakt:**  
Öffnerkontakte leiten den Strom weiter, wenn der Signalzustand eines angegebenen Binäroperanden "0" ist.
- **Kontakt mit Zusatzfunktion:**  
Kontakte mit Zusatzfunktion leiten den Strom weiter, wenn eine bestimmte Bedingung erfüllt ist. Mit diesen Kontakten können Sie auch eine zusätzliche Funktion, z.B. eine Flankenauswertung und einen Vergleich durchführen.

## Spulen

Mit Spulen können Sie Binäroperanden steuern. Spulen können einen Binäroperanden abhängig vom Signalzustand des Verknüpfungsergebnisses setzen oder rücksetzen.

Die folgenden Arten von Spulen stehen Ihnen in einem KOP-Programm zur Verfügung:

- **Einfache Spulen:**  
Einfache Spulen setzen einen Binäroperanden, wenn Strom in die Spule fließt. Beispiel für eine einfache Spule ist die Anweisung "Zuweisung".
- **Spulen mit Zusatzfunktion:**  
Diese Spulen weisen neben der Auswertung des Verknüpfungsergebnisses weitere Funktionalitäten auf. Beispiele für Spulen mit Zusatzfunktion sind die Spulen zur Flankenauswertung und Programmsteuerung.

## Boxen

Boxen sind KOP-Elemente mit komplexen Funktionen. Eine Ausnahme dabei ist die Leerbox. Die Leerbox dient als Platzhalter, in dem Sie die gewünschte Anweisung auswählen können.

Die folgenden Arten von Boxen stehen Ihnen in einem KOP-Programm zur Verfügung:

- **Boxen ohne EN-/ENO-Mechanismus:**  
Eine Box wird abhängig vom Signalzustand an den Box-Eingängen ausgeführt. Der Fehlerstatus der Bearbeitung kann nicht abgefragt werden.
- **Boxen mit EN-/ENO-Mechanismus:**  
Eine Box wird nur dann ausgeführt, wenn der Freigabeeingang "EN" den Signalzustand "1" führt. Bei ordnungsgemäßer Bearbeitung der Box führt der Freigabeausgang "ENO" den Signalzustand "1". Wenn während der Bearbeitung ein Fehler auftritt, wird der Freigabeausgang "ENO" zurückgesetzt.

Aufrufe von Codebausteinen werden im Netzwerk auch als Boxen mit EN-/ENO-Mechanismus dargestellt.

## Siehe auch

Regeln für die Verwendung von KOP-Elementen (Seite 1290)

## Einstellungen zu KOP

### Übersicht über die Einstellungen zu KOP

#### Überblick

Die folgende Tabelle zeigt die Einstellungen, die Sie vornehmen können:

| Gruppe        | Einstellung            | Beschreibung  |
|---------------|------------------------|---|
| Schriftart    | Schriftgröße           | Schriftgröße im Programmiereditor   |
| Ansicht       | Layout                 | Kompakt oder weit<br>Ändert den vertikalen Abstand zwischen Operanden und anderen Objekten (z. B. Operanden und Kontakt). Die Änderung wird erst nach dem erneuten Öffnen eines Bausteins sichtbar. |
|               | mit Absolutinformation | Zusätzliche Anzeige der Absolutadressen   |
| Operandenfeld | Maximale Breite        | Maximale Anzahl der Zeichen, die horizontal in das Operandenfeld eingegeben werden können. Mit dieser Einstellung wird das Layout der Netzwerke neu berechnet.                                      |
|               | Maximale Höhe          | Maximale Anzahl der Zeichen, die vertikal in das Operandenfeld eingegeben werden können. Mit dieser Einstellung wird das Layout der Netzwerke neu berechnet.  |

#### Siehe auch

Einstellungen ändern (Seite 1282)

### Einstellungen ändern

#### Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.



## Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

## Siehe auch

Übersicht über die Einstellungen zu KOP (Seite 1282)

## Arbeiten mit Netzwerken

### Netzwerke verwenden

## Funktion

Das Anwenderprogramm wird im Baustein innerhalb von Netzwerken erstellt. Damit ein Codebaustein programmiert werden kann, muss er mindestens ein Netzwerk enthalten. Um eine bessere Übersichtlichkeit des Anwenderprogramms zu erreichen, können Sie Ihr Programm auch in mehrere Netzwerke gliedern.

## Siehe auch

Netzwerktitel eingeben (Seite 1287)

Netzwerkcommentar eingeben (Seite 1288)

In Netzwerken navigieren (Seite 1289)

## Netzwerke einfügen

### Voraussetzung

Ein Baustein ist geöffnet.

### Vorgehen

Um ein neues Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, hinter dem Sie ein neues Netzwerk einfügen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Netzwerk einfügen".

## Ergebnis

Ein neues, leeres Netzwerk wird in den Baustein eingefügt.

## Siehe auch

- Netzwerke selektieren (Seite 1284)
- Netzwerke kopieren und einfügen (Seite 1285)
- Netzwerke löschen (Seite 1286)
- Netzwerke reduzieren und erweitern (Seite 1286)
- Netzwerktitel eingeben (Seite 1287)
- Netzwerkcommentar eingeben (Seite 1288)
- In Netzwerken navigieren (Seite 1289)

## Netzwerke selektieren

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Ein Netzwerk selektieren

Um ein Netzwerk zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelleiste des Netzwerks, das Sie selektieren möchten.

### Mehrere Netzwerke selektieren

Um mehrere einzelne Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie auf alle Netzwerke, die Sie selektieren möchten.

Um mehrere aufeinanderfolgende Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Shift>.
2. Klicken Sie auf das erste Netzwerk, das Sie selektieren möchten.
3. Klicken Sie auf das letzte Netzwerk, das Sie selektieren möchten.  
Das erste und das letzte Netzwerk und alle dazwischen liegenden Netzwerke werden selektiert.

**Siehe auch**

Netzwerke einfügen (Seite 1283)  
Netzwerke kopieren und einfügen (Seite 1285)  
Netzwerke löschen (Seite 1286)  
Netzwerke reduzieren und erweitern (Seite 1286)  
Netzwerktitle eingeben (Seite 1287)  
Netzwerkcommentar eingeben (Seite 1288)  
In Netzwerken navigieren (Seite 1289)

**Netzwerke kopieren und einfügen**

Kopierte Netzwerke können innerhalb des Bausteins oder in einen anderen Baustein eingefügt werden. Netzwerke, die in KOP oder FUP erstellt wurden, können auch in Bausteine der jeweils anderen Programmiersprache eingefügt werden.

**Voraussetzung**

Ein Netzwerk ist vorhanden.

**Vorgehen**

Um ein Netzwerk zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk oder die Netzwerke, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Selektieren Sie das Netzwerk, hinter dem Sie das kopierte Netzwerk einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

**Siehe auch**

Netzwerke einfügen (Seite 1283)  
Netzwerke selektieren (Seite 1284)  
Netzwerke löschen (Seite 1286)  
Netzwerke reduzieren und erweitern (Seite 1286)  
Netzwerktitle eingeben (Seite 1287)  
Netzwerkcommentar eingeben (Seite 1288)  
In Netzwerken navigieren (Seite 1289)

## Netzwerke löschen

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Vorgehen

Um ein Netzwerk zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Siehe auch

Netzwerke einfügen (Seite 1283)

Netzwerke selektieren (Seite 1284)

Netzwerke kopieren und einfügen (Seite 1285)

Netzwerke reduzieren und erweitern (Seite 1286)

Netzwerktitel eingeben (Seite 1287)

Netzwerkcommentar eingeben (Seite 1288)

In Netzwerken navigieren (Seite 1289)

## Netzwerke reduzieren und erweitern

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Ein Netzwerk auf- und zuklappen

Um ein Netzwerk aufzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts in der Titelleiste des Netzwerks.

Um ein Netzwerk zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten in der Titelleiste des Netzwerks.

### **Alle Netzwerke auf- und zuklappen**

Um alle Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Alle Netzwerke öffnen" bzw. "Alle Netzwerke schließen".

### **Siehe auch**

- Netzwerke einfügen (Seite 1283)
- Netzwerke selektieren (Seite 1284)
- Netzwerke kopieren und einfügen (Seite 1285)
- Netzwerke löschen (Seite 1286)
- Netzwerktitle eingeben (Seite 1287)
- Netzwerkcommentar eingeben (Seite 1288)
- In Netzwerken navigieren (Seite 1289)

### **Netzwerktitle eingeben**

Der Netzwerktitle ist die Überschrift eines Netzwerks. Die Länge des Netzwerktitles ist auf eine Zeile beschränkt.

### **Voraussetzung**

Ein Netzwerk ist vorhanden.

### **Vorgehen**

Um einen Netzwerktitle einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titlezeile des Netzwerks.
2. Geben Sie den Netzwerktitle ein.

## Siehe auch

- Netzwerke verwenden (Seite 1283)
- Netzwerke einfügen (Seite 1283)
- Netzwerke selektieren (Seite 1284)
- Netzwerke kopieren und einfügen (Seite 1285)
- Netzwerke löschen (Seite 1286)
- Netzwerke reduzieren und erweitern (Seite 1286)
- Netzwerkcommentar eingeben (Seite 1288)
- In Netzwerken navigieren (Seite 1289)

## Netzwerkcommentar eingeben

Mithilfe von Netzwerkcommentaren können Sie die Programminhalte der einzelnen Netzwerke erläutern. Sie können z. B. die Funktion des Netzwerks beschreiben oder auf Besonderheiten hinweisen.

## Voraussetzung

Ein Netzwerk ist vorhanden.

## Vorgehen

Um einen Netzwerkcommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts vor dem Netzwerktitel.
2. Wenn der Commentarbereich nicht sichtbar ist, klicken Sie in der Funktionsleiste auf die Schaltfläche "Netzwerkcommentare ein/aus".  
Der Commentarbereich wird angezeigt.
3. Klicken Sie im Commentarbereich auf "Commentar".  
Die Textstelle "Commentar" wird markiert.
4. Geben Sie den Netzwerkcommentar ein.

## Siehe auch

- Netzwerke verwenden (Seite 1283)
- Netzwerke einfügen (Seite 1283)
- Netzwerke selektieren (Seite 1284)
- Netzwerke kopieren und einfügen (Seite 1285)
- Netzwerke löschen (Seite 1286)
- Netzwerke reduzieren und erweitern (Seite 1286)
- Netzwerktitle eingeben (Seite 1287)
- In Netzwerken navigieren (Seite 1289)

## In Netzwerken navigieren

Sie haben die Möglichkeit, in einem Baustein direkt zu einer bestimmten Position zu navigieren.

## Vorgehen

Um zu einer bestimmten Position innerhalb eines Bausteins zu navigieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Codebereich des Programmierfensters.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu > Netzwerk/Zeile".  
Der Dialog "Gehe zu" wird geöffnet.
3. Geben Sie das Netzwerk ein, zu dem Sie navigieren möchten.
4. Geben Sie die Zeilennummer des Netzwerks ein, zu der Sie navigieren möchten.
5. Bestätigen Sie Ihre Eingabe mit "OK".

## Ergebnis

Wenn möglich wird die entsprechende Zeile angezeigt. Falls das gewünschte Netzwerk oder die gewünschte Zeile nicht existiert, wird das letzte existierende Netzwerk bzw. die letzte existierende Zeile im gewünschten Netzwerk angezeigt.

## Siehe auch

- Netzwerke verwenden (Seite 1283)
- Netzwerke einfügen (Seite 1283)
- Netzwerke selektieren (Seite 1284)
- Netzwerke kopieren und einfügen (Seite 1285)
- Netzwerke löschen (Seite 1286)
- Netzwerke reduzieren und erweitern (Seite 1286)
- Netzwerktitel eingeben (Seite 1287)
- Netzwerkcommentar eingeben (Seite 1288)

## KOP-Elemente einfügen

### Regeln für die Verwendung von KOP-Elementen

#### Regeln

Beachten Sie beim Einfügen von KOP-Elementen folgende Regeln:

- Jedes KOP-Netzwerk muss einen Abschluss in Form einer Spule oder einer Box haben. Folgende KOP-Elemente dürfen Sie jedoch nicht als Abschluss eines Netzwerks verwenden:
  - Vergleichboxen
  - Anweisungen für positive oder negative Flankenauswertung
- Der Ausgangspunkt des Zweigs für einen Boxanschluss muss immer die Stromschiene sein. Im Zweig vor der Box dürfen jedoch logische Verknüpfungen oder andere Boxen vorhanden sein.
- In Parallelzweigen mit Vorverknüpfungen können nur Kontakte eingefügt werden. Dabei macht der Kontakt zum Negieren des Verknüpfungsergebnisses (-|NOT|-) eine Ausnahme. Der Kontakt zum Negieren des Verknüpfungsergebnisses sowie Spulen und Boxen können in Parallelzweigen verwendet werden, wenn diese direkt von der Stromschiene ausgehen.
- Öffner- und Schließkontakte können nicht mit Konstanten belegt werden, z. B. mit TRUE oder FALSE. Verwenden Sie stattdessen Operanden vom Datentyp BOOL.
- Pro Netzwerk kann nur eine Sprunganweisung eingefügt werden.
- Pro Netzwerk kann nur eine Sprungmarke eingefügt werden.
- Anweisungen mit positiver oder negativer Flankenauswertung dürfen nicht direkt am linken Rand des Strompfads angeordnet werden, da diese eine Vorverknüpfung voraussetzen.



### Platzierungsregeln für S7-1200/1500 CPUs

Die folgende Tabelle zeigt die Anweisungen, die nur am Ende des Netzwerks positioniert werden können:

| Anweisung |                        | Vorverknüpfung notwendig |
|-----------|------------------------|--------------------------|
| Mnemonic  | Name                   |                          |
| SET_BF    | Bitfeld setzen         | Nein                     |
| RESET_BF  | Bitfeld rücksetzen     | Nein                     |
| JMP       | Springen bei VKE = 1   | Nein                     |
| JMPN      | Springen bei VKE = 0   | Ja                       |
| JMP_LIST  | Sprungliste definieren | Nein                     |
| SWITCH    | Sprungverteiler        | Nein                     |
| RET       | Zurück springen        | Nein                     |

### Platzierungsregeln für S7-300/400 CPUs

Die folgende Tabelle zeigt die Anweisungen, die nur am Ende des Netzwerks positioniert werden können:

| Anweisung |   | Vorverknüpfung notwendig |
|-----------|---|--------------------------|
| Mnemonic  | Name  |                          |
| S         | Ausgang setzen                                    | Ja                       |
| R         | Ausgang rücksetzen                                | Ja                       |
| SI        | Zeit als Impuls starten                           | Ja                       |
| SV        | Zeit als verlängerten Impuls starten              | Ja                       |
| SE        | Zeit als Einschaltverzögerung starten             | Ja                       |
| SS        | Zeit als speichernde Einschaltverzögerung starten | Ja                       |
| SA        | Zeit als Ausschaltverzögerung starten             | Ja                       |
| SZ        | Zähleranfangswert setzen                          | Ja                       |
| ZV        | Vorwärts zählen                                   | Ja                       |
| ZR        | Rückwärts zählen                                  | Ja                       |
| JMP       | Springen bei VKE = 1                              | Nein                     |
| JMPN      | Springen bei VKE = 0                              | Ja                       |
| RET       | Zurück springen                                   | Nein                     |
| OPN       | Globalen Datenbaustein öffnen                     | Nein                     |
| OPNI      | Instanz-Datenbaustein öffnen                      | Nein                     |
| CALL      | Baustein aufrufen                                 | Nein                     |
| SAVE      | VKE im BIE-Bit speichern                          | Nein                     |
| MCRA      | MCR-Bereich aktivieren                            | Nein                     |
| MCRD      | MCR-Bereich deaktivieren                          | Nein                     |
| MCR<      | MCR-Bereiche öffnen                               | Nein                     |
| MCR>      | MCR-Bereiche schließen                            | Nein                     |

**Siehe auch**

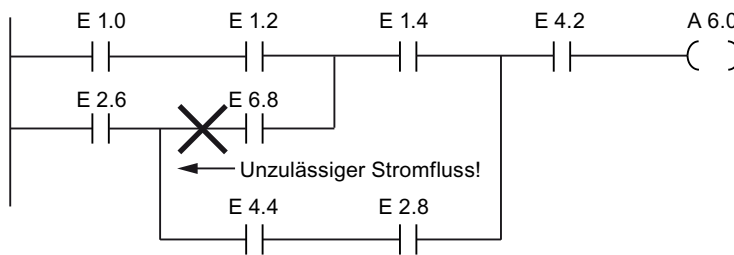
Unzulässige Verschaltungen in KOP (Seite 1292)

Übersicht über die KOP-Elemente (Seite 1279)

**Unzulässige Verschaltungen in KOP**

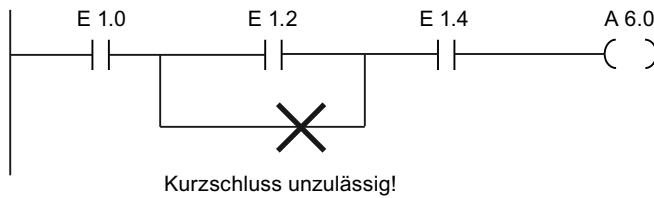
**Stromfluss von rechts nach links**

Es können keine Zweige programmiert werden, die einen Stromfluss in umgekehrter Richtung verursachen könnten.



**Kurzschluss**

Es können keine Zweige programmiert werden, die einen Kurzschluss verursachen.



**Verknüpfungen**

Für Verknüpfungen gelten folgende Regeln:

- Nur boolesche Eingänge sind vorverknüpfbar.
- Nur der erste boolesche Ausgang ist weiter verknüpfbar.
- Es existiert nur ein vollständiger Verknüpfungspfad pro Netzwerk. Nicht zusammenhängende Pfade können verbunden werden.

**Siehe auch**

Regeln für die Verwendung von KOP-Elementen (Seite 1290)

## KOP-Elemente über Task Card "Anweisungen" einfügen

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Vorgehen

Um ein KOP-Element mithilfe der Task Card "Anweisungen" in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie zu dem KOP-Element, das Sie einfügen möchten.
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk. Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie für den Funktionsbaustein einen Instanz-Datenbaustein als Einzel- oder Multiinstanz erstellen, in dem die Daten des eingefügten Elements gespeichert werden. Nach dem Erstellen finden Sie den neuen Instanz-Datenbaustein in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Wenn Sie "Multiinstanz" ausgewählt haben, finden Sie diese in der Bausteinschnittstelle im Abschnitt "Static".

Oder:

1. Markieren Sie die Stelle im Netzwerk, an der Sie das Element einfügen möchten.
2. Öffnen Sie die Task Card "Anweisungen".
3. Doppelklicken Sie auf das einzufügende Element. Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie für den Funktionsbaustein einen Instanz-Datenbaustein als Einzel- oder Multiinstanz erstellen, in dem die Daten des eingefügten Elements gespeichert werden. Nach dem Erstellen finden Sie den neuen Instanz-Datenbaustein in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Wenn Sie "Multiinstanz" ausgewählt haben, finden Sie diese in der Bausteinschnittstelle im Abschnitt "Static".

### Ergebnis

Das ausgewählte KOP-Element wird mit Platzhaltern für die Parameter eingefügt.

## KOP-Elemente mithilfe einer Leerbox einfügen

### Voraussetzung

Ein Netzwerk ist vorhanden.

## Vorgehen

Um ein KOP-Element mithilfe einer Leerbox in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Leerbox".
3. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
4. Bewegen Sie den Mauszeiger über das Dreieck in der oberen rechten Ecke der Leerbox. Eine Klappliste wird geöffnet.
5. Wählen Sie das gewünschte KOP-Element aus der Klappliste.  
Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie für den Funktionsbaustein einen Instanz-Datenbaustein als Einzel- oder Multiinstanz erstellen, in dem die Daten des eingefügten Elements gespeichert werden. Nach dem Erstellen finden Sie den neuen Instanz-Datenbaustein in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Wenn Sie "Multiinstanz" ausgewählt haben, finden Sie diese in der Bausteinschnittstelle im Abschnitt "Static".

## Ergebnis

Die Leerbox wird zum entsprechenden KOP-Element geändert. Für die Parameter werden Platzhalter eingefügt.

## Datentyp eines KOP-Elements auswählen

### Auswahl eines Datentyps

### Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie eine dieser Anweisungen im Programm verwenden, müssen Sie an der konkreten Programmstelle ein für die Anweisung zulässiger Datentyp festlegen. Bei einigen Anweisungen müssen Sie den Datentypen für die Eingänge und für die Ausgänge separat auswählen.

---

#### Hinweis

Der zulässige Datentyp (BOOL) für die Variablen am Freigabeeingang EN und am Freigabeausgang ENO ist vom System vorgegeben und kann nicht geändert werden.

---

Die für eine Anweisung zulässigen Datentypen sind in der Klappliste der Anweisung aufgeführt. Durch die Auswahl eines Eintrags aus der Klappliste bestimmen Sie den Datentyp der Anweisung. Wenn der Datentyp eines Operanden vom Datentyp der Anweisung abweicht und nicht implizit konvertiert werden kann, wird der Operand rot angezeigt und ein Rollout mit der entsprechenden Fehlermeldung eingeblendet.

## Datentypauswahl von mathematischen Anweisungen

Einige mathematische Anweisungen bieten Ihnen die Möglichkeit, den Datentyp automatisch passend zu den Datentypen der Operanden einstellen zu lassen. Diese Anweisungen besitzen in der Klappliste für die Datentypauswahl zusätzlich zu den eigentlichen Datentypen noch den Eintrag "Auto". Wenn Sie diesen Eintrag auswählen und den ersten Operanden beschalten, wird der Datentyp des Operanden als Datentyp für die Anweisung verwendet. Der Eintrag in der Klappliste ändert sich zu "Auto (<Datentyp>)", z. B. "Auto (Real)". Wenn Sie weitere Operanden beschalten, wird der automatisch eingestellte Datentyp der Anweisung nach folgenden Kriterien angepasst:

- Sie versorgen alle weiteren Operanden mit Variablen des gleichen Datentyps:  
Der Datentyp der Anweisung wird nicht geändert.
- Sie versorgen alle weiteren Operanden mit Variablen, deren Datentyp kleiner ist als der Datentyp der Anweisung:  
Der Datentyp der Anweisung wird nicht geändert. Für die Operanden mit dem kleineren Datentyp wird ggf. eine implizite Konvertierung durchgeführt.
- Sie versorgen einen weiteren Operanden mit einer Variablen, deren Datentyp größer ist als der Datentyp der Anweisung:  
Der Datentyp der Anweisung wird auf den größeren Datentyp geändert. Für Operanden, die von dem neu eingestellten Datentyp der Anweisung abweichen, erfolgt ggf. eine implizite Konvertierung.

Jede Datentypänderung eines Operanden kann eine Änderung des Datentyps der Anweisung nach sich ziehen. Dadurch werden möglicherweise auch weitere Operanden implizit konvertiert. Operanden, für die eine implizite Konvertierung durchgeführt wird, werden mit einem grauen Quadrat gekennzeichnet.

---

### Hinweis

Beachten Sie auch die Informationen zur Datentypkonvertierung für Ihr Gerät und hier insbesondere die Hinweise zur IEC-Prüfung.

Siehe auch: Datentypkonvertierung

---

## Siehe auch

Datentyp einer Anweisung festlegen (Seite 1295)

## Datentyp einer Anweisung festlegen

### Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie solche Anweisungen in Ihr Programm einfügen, müssen Sie für diese Anweisungen an der konkreten Programmstelle den Datentyp festlegen.

### Datentyp über die Klappliste festlegen

Um den Datentyp einer Anweisung über die Klappliste festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.  
In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Klicken Sie auf das Dreieck in der oberen Ecke der Klappliste.  
Die Klappliste wird geöffnet und die Datentypen angezeigt, die für die Anweisung zulässig sind.
3. Wählen Sie aus der Klappliste einen Datentyp aus.  
Der ausgewählte Datentyp wird angezeigt.
4. Falls die Anweisung über zwei Klapplisten verfügt, wählen Sie in der linken Klappliste den Datentyp für die Eingänge der Anweisung und in der rechten Klappliste den Datentyp für die Ausgänge der Anweisung.

### Datentyp durch Zuweisen von Variablen festlegen

Um den Datentyp einer Anweisung durch Zuweisen von Variablen festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.  
In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Geben Sie an einem Eingang oder Ausgang eine gültige Variable an, deren Datentyp als Datentyp der Anweisung übernommen werden soll.  
Der Datentyp der Variablen wird in der Klappliste angezeigt.
3. Falls sowohl für die Eingänge als auch für die Ausgänge der Anweisung Datentypen festgelegt werden müssen, geben Sie an einem Eingang und an einem Ausgang jeweils eine gültige Variable an. Die am Eingang angegebene Variable bestimmt den Datentyp der Eingänge und die Variable am Ausgang bestimmt den Datentyp der Ausgänge der Anweisung.

### Datentyp von mathematischen Anweisungen automatisch festlegen

Um für mathematische Anweisungen den Datentyp automatisch festlegen zu lassen, gehen Sie folgendermaßen vor:

1. Fügen Sie die mathematische Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.  
In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Wählen Sie in der Klappliste den Eintrag "Auto".
3. Geben Sie an einem Eingang oder Ausgang eine gültige Variable an.  
Der Datentyp der Variablen wird als Datentyp der Anweisung übernommen. Der Eintrag in der Klappliste ändert sich zu "Auto (<Datentyp>)".

Siehe auch: Auswahl eines Datentyps (Seite 1294)

### Siehe auch

Auswahl eines Datentyps (Seite 1294)

## Favoriten in KOP verwenden

### KOP-Elemente zu den Favoriten hinzufügen

#### Voraussetzung

- Ein Baustein ist geöffnet.
- Für die Task Card "Anweisungen" ist der Mehrpalettenmodus eingestellt oder die Favoriten werden auch im Editor angezeigt.

#### Vorgehen

Um Anweisungen zu den Favoriten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Maximieren Sie die Palette "Einfache Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu der Anweisung, die Sie den Favoriten hinzufügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop entweder in die Palette "Favoriten" oder in den Favoritenbereich im Programmiereditor.

---

#### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---

#### Siehe auch

KOP-Elemente aus den Favoriten entfernen (Seite 1298)

Übersicht über den Programmiereditor (Seite 1235)

### KOP-Elemente mithilfe von Favoriten einfügen

#### Voraussetzung

- Ein Baustein ist geöffnet.
- Favoriten sind vorhanden.

## Vorgehen

Um eine Anweisung mithilfe der Favoriten in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die gewünschte Anweisung per Drag & Drop aus den Favoriten an die gewünschte Position.

Oder:

1. Markieren Sie die Programmstelle, an der Sie die Anweisung einfügen möchten.
2. Klicken Sie auf die einzufügende Anweisung in den Favoriten.

---

### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---

## Siehe auch

KOP-Elemente aus den Favoriten entfernen (Seite 1298)

Übersicht über den Programmiereditor (Seite 1235)

## KOP-Elemente aus den Favoriten entfernen

### Voraussetzung

Ein Codebaustein ist geöffnet.

## Vorgehen

Um Anweisungen aus den Favoriten zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Anweisung, die Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisung entfernen".

---

### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---



## Siehe auch

- KOP-Elemente zu den Favoriten hinzufügen (Seite 1297)
- KOP-Elemente mithilfe von Favoriten einfügen (Seite 1297)
- Übersicht über den Programmiereditor (Seite 1235)

## Bausteinaufrufe in KOP einfügen

### Bausteinaufrufe per Drag & Drop einfügen

Sie können Aufrufe für bestehende Funktionen (FC) und Funktionsbausteine (FB) per Drag & Drop aus der Projektnavigation einfügen. Wenn Sie Funktionsbausteine aus anderen Funktionsbausteinen aufrufen, können Sie sie entweder als Einzelinstanz oder als Multiinstanz aufrufen. Wird ein Funktionsbaustein als Einzelinstanz aufgerufen, speichert er seine Daten in einem eigenen Datenbaustein. Wird ein Funktionsbaustein als Multiinstanz aufgerufen, speichert er seine Daten im Instanz-Datenbaustein des aufrufenden Funktionsbausteins.

### Voraussetzung

- Ein Netzwerk ist vorhanden.
- Der Baustein, der aufgerufen werden soll, ist vorhanden.

### Aufruf einer Funktion (FC) einfügen

Um den Aufruf einer Funktion (FC) per Drag & Drop in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die Funktion aus der Projektnavigation in das gewünschte Netzwerk.

### Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Funktionsbaustein aus der Projektnavigation in das gewünschte Netzwerk. Der Dialog "Aufrufoptionen" wird geöffnet.
2. Geben Sie im Dialog an, ob Sie den Baustein als Einzel- oder Multiinstanz aufrufen wollen.
  - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
  - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
3. Bestätigen Sie Ihre Eingaben mit "OK".

## Ergebnis

Die Funktion bzw. der Funktionsbaustein wird mit seinen Parametern eingefügt. Diese Parameter können Sie anschließend versorgen.

Siehe auch: Auto-Hotspot

---

### Hinweis

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt. Wenn Sie einen Funktionsbaustein als Multiinstanz aufgerufen haben, wird dieser als statische Variable in die Schnittstelle eingetragen.

---

## Siehe auch

Bausteinaufrufe in KOP aktualisieren (Seite 1300)

Instanztyp ändern (Seite 1301)

Einzelinstanzen (Seite 1041)

Multiinstanzen (Seite 1042)

## Bausteinaufrufe in KOP aktualisieren

Wenn sich Schnittstellenparameter eines aufgerufenen Bausteins ändern, kann der Bausteinaufruf nicht korrekt ausgeführt werden. Solche inkonsistenten Bausteinaufrufe vermeiden Sie, indem Sie die Bausteinaufrufe aktualisieren.

Sie haben zwei Möglichkeiten, Bausteinaufrufe zu aktualisieren:

- Explizites Aktualisieren im Programmiereditor.  
Die Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert.
- Implizites Aktualisieren während des Übersetzens.  
Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen werden aktualisiert.

## Bausteinaufrufe im Programmiereditor aktualisieren

Um in einem Baustein einen Bausteinaufruf zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Inkonsistente Bausteinaufrufe aktualisieren".

Oder:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf die Anweisung mit dem Bausteinaufruf.

3. Wählen Sie im Kontextmenü den Befehl "Bausteinanruf aktualisieren".  
Der Dialog "Schnittstellenabgleich" wird geöffnet. In diesem Dialog werden Ihnen die Unterschiede zwischen der verwendeten und der geänderten Bausteinschnittstelle des aufgerufenen Bausteins angezeigt.
4. Wenn Sie den Bausteinanruf aktualisieren möchten, klicken Sie auf "OK". Um die Aktualisierung abzubrechen, klicken Sie auf "Abbrechen".

### Bausteinanrufe während des Übersetzens aktualisieren

Um alle Bausteinanrufe und Verwendungen von PLC-Datentypen während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

### Siehe auch

Bausteinanrufe per Drag & Drop einfügen (Seite 1299)

Instanztyp ändern (Seite 1301)

### Instanztyp ändern

#### Instanztyp

Funktionsbausteine können Sie auf zwei verschiedene Arten aufrufen:

- Als Einzelinstanz
- Als Multiinstanz

Siehe auch: Auto-Hotspot

Sie können einen einmal definierten Instanztyp jederzeit ändern.

#### Voraussetzung

Das Anwenderprogramm enthält einen Bausteinanruf.

#### Vorgehen

Um den Instanztyp eines Funktionsbausteins zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Codebaustein und selektieren Sie den Bausteinanruf.
2. Wählen Sie im Kontextmenü den Befehl "Instanz ändern".  
Der Dialog "Aufrufoptionen" wird geöffnet.

3. Klicken Sie auf die Schaltfläche "Einzel-Instanz" oder "Multi-Instanz"
  - Wenn Sie als Instanztyp "Einzel-Instanz" wählen, geben Sie einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
  - Wenn Sie als Instanztyp "Multi-Instanz" wählen, geben Sie im Textfeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
4. Bestätigen Sie Ihre Eingaben mit "OK".

---

**Hinweis**

Die vorherigen Einzel- und Multiinstanzen werden nicht automatisch gelöscht.

---

**Siehe auch**

Bausteinaufrufe per Drag & Drop einfügen (Seite 1299)

Bausteinaufrufe in KOP aktualisieren (Seite 1300)

**Komplexe KOP-Anweisungen einfügen**

**Anweisung "Berechnen" verwenden**

**Voraussetzung**

Ein Netzwerk ist vorhanden.

**Vorgehen**

Um die Anweisung "Berechnen" zu verwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Mathematische Funktionen > CALCULATE".
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk. Die Anweisung "Berechnen" wird mit einem Platzhalteraussdruck und Fragezeichen für den Datentyp eingefügt.
4. Geben Sie den Datentyp für die Berechnung ein.
5. Geben Sie die Operanden für die Berechnung ein.

---

**Hinweis**

Die Berechnung wird mit den Eingängen der Anweisung "Berechnen" durchgeführt. Wenn Sie Konstanten verwenden möchten, müssen Sie daher auch für die Konstanten entsprechende Eingänge einfügen.

---

6. Klicken Sie auf die Schaltfläche "Anweisung 'Berechnen' bearbeiten", um den Platzhalteraussdruck durch den korrekten Ausdruck zu ersetzen.  
Der Dialog "Anweisung 'Berechnen' bearbeiten" wird geöffnet.
7. Geben Sie in das Textfeld "OUT:= " den gewünschten Ausdruck ein.

---

**Hinweis**

Im Bereich "Beispiel" finden Sie ein Beispiel für einen gültigen Ausdruck und mögliche Anweisungen, die Sie verwenden können.

Um z. B. mithilfe des Pythagoras einen Wert zu ermitteln, geben Sie "OUT := SQRT (SQR (IN1) + SQR (IN2))" ein.

---

8. Bestätigen Sie Ihre Eingabe mit "OK".

**Siehe auch**

CALCULATE: Berechnen (Seite 1723)

**Freie Kommentare verwenden****Grundlagen zur Verwendung von freien Kommentaren in KOP****Einführung**

Freie Kommentare ermöglichen das Kommentieren des Quellcodes für grafische Programmiersprachen ähnlich wie Zeilenkommentare für textuelle Sprachen.

Freie Kommentare können für folgende Elemente verwendet werden:

- Boxen
- Spulen

**Siehe auch**

Freie Kommentare einfügen (Seite 1303)

Freie Kommentare bearbeiten (Seite 1304)

Freie Kommentare löschen (Seite 1305)

**Freie Kommentare einfügen****Voraussetzung**

Ein Netzwerk mit Anweisungen ist vorhanden.

## Vorgehen

Um einen freien Kommentar zu einer Anweisung hinzuzufügen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie ggf. in der Funktionsleiste die Schaltfläche "Freie Kommentare ein/aus".
2. Klicken Sie mit der rechten Maustaste auf die Anweisung, für die Sie einen freien Kommentar einfügen möchten.
3. Wählen Sie den Befehl "Kommentar einfügen" aus dem Kontextmenü.  
Eine Kommentarbox mit einem Standardkommentar wird geöffnet. Die Kommentarbox ist über einen Pfeil mit der zugehörigen Anweisung verbunden.
4. Geben Sie in die Kommentarbox den gewünschten Kommentar ein.

## Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in KOP (Seite 1303)

Freie Kommentare bearbeiten (Seite 1304)

Freie Kommentare löschen (Seite 1305)

## Freie Kommentare bearbeiten

### Einführung

Freie Kommentare können folgendermaßen bearbeitet werden:

- Kommentartext ändern
- Platzierung oder Größe der Kommentarbox ändern
- Kommentar an ein anderes Element anhängen
- Freie Kommentare ein- und ausblenden

### Kommentartext ändern

Um den Text von freien Kommentaren zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Kommentarbox.
2. Geben Sie den gewünschten Text ein.

### Platzierung der Kommentarbox ändern

Um die Platzierung der Kommentarbox zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Kommentarbox und halten Sie die Maustaste gedrückt.
2. Ziehen Sie die Kommentarbox an die gewünschte Stelle.

### Größe der Kommentarbox ändern

Um die Größe der Kommentarbox zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Kommentarbox.
2. Ziehen Sie die Kommentarbox am Anfasser in der unteren rechten Ecke auf die gewünschte Größe.

### Kommentar an ein anderes Element anhängen

Um einen freien Kommentar an ein anderes Element zu hängen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Pfeilspitze, die die Kommentarbox mit der Anweisung verbindet, und halten Sie die Maustaste gedrückt.
2. Ziehen Sie den Pfeil auf das Element, an das Sie den Kommentar anhängen möchten. Mögliche Einfügestellen werden mit einem grünen Quadrat gekennzeichnet.
3. Lassen Sie die Maustaste los.

### Freie Kommentare ein- und ausblenden

Um freie Kommentare ein- und auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Freie Kommentare ein/aus".

### Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in KOP (Seite 1303)

Freie Kommentare einfügen (Seite 1303)

Freie Kommentare löschen (Seite 1305)

### Freie Kommentare löschen

#### Vorgehen

Um einen freien Kommentar zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den freien Kommentar, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in KOP (Seite 1303)

Freie Kommentare einfügen (Seite 1303)

Freie Kommentare bearbeiten (Seite 1304)

## KOP-Elemente bearbeiten

### KOP-Elemente selektieren

Sie können mehrere einzelne Elemente oder alle Elemente eines Netzwerks selektieren.

#### Voraussetzung

KOP-Elemente sind vorhanden.

### Mehrere einzelne KOP-Elemente selektieren

Um mehrere einzelne KOP-Elemente zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie die KOP-Elemente an, die Sie selektieren möchten.
3. Lassen Sie die Taste <Strg> los.

### Alle KOP-Elemente eines Netzwerks selektieren

Um alle KOP-Elemente eines Netzwerks zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in das Netzwerk, dessen Elemente Sie selektieren möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Alles auswählen" oder verwenden Sie die Tastenkombination <Strg+A>.

#### Siehe auch

KOP-Elemente kopieren (Seite 1307)

KOP-Elemente ausschneiden (Seite 1307)

KOP-Elemente aus der Zwischenablage einfügen (Seite 1308)

KOP-Elemente ersetzen (Seite 1309)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

KOP-Elemente löschen (Seite 1313)



## KOP-Elemente kopieren

### Voraussetzung

Ein KOP-Element ist vorhanden.

### Vorgehen

Um ein KOP-Element zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das KOP-Element, das Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

### Ergebnis

Das KOP-Element wird kopiert und in der Zwischenablage gespeichert.

### Siehe auch

KOP-Elemente selektieren (Seite 1306)

KOP-Elemente ausschneiden (Seite 1307)

KOP-Elemente aus der Zwischenablage einfügen (Seite 1308)

KOP-Elemente ersetzen (Seite 1309)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

KOP-Elemente löschen (Seite 1313)

## KOP-Elemente ausschneiden

### Voraussetzung

Ein KOP-Element ist vorhanden.

### Ausschneiden

Um ein KOP-Element auszuschneiden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das KOP-Element, das Sie ausschneiden möchten.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

## Ergebnis

Das KOP-Element wird ausgeschnitten und in der Zwischenablage gespeichert.

## Siehe auch

KOP-Elemente selektieren (Seite 1306)

KOP-Elemente kopieren (Seite 1307)

KOP-Elemente aus der Zwischenablage einfügen (Seite 1308)

KOP-Elemente ersetzen (Seite 1309)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

KOP-Elemente löschen (Seite 1313)

## KOP-Elemente aus der Zwischenablage einfügen

### Voraussetzung

Ein KOP-Element ist vorhanden.

### Vorgehen

Um ein KOP-Element aus der Zwischenablage einzufügen, gehen Sie folgendermaßen vor:

1. Kopieren Sie ein KOP-Element oder schneiden Sie ein KOP-Element aus.
2. Klicken Sie mit der rechten Maustaste auf die Stelle im Netzwerk, an der Sie das Element einfügen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

## Siehe auch

KOP-Elemente selektieren (Seite 1306)

KOP-Elemente kopieren (Seite 1307)

KOP-Elemente ausschneiden (Seite 1307)

KOP-Elemente ersetzen (Seite 1309)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

KOP-Elemente löschen (Seite 1313)

## KOP-Elemente ersetzen

Sie können KOP-Elemente ganz einfach gegen andere KOP-Elemente des gleichen Typs austauschen. Dies hat den Vorteil, dass die Parameter bestehen bleiben und nicht erneut eingegeben werden müssen. Sie können z. B. Schließer und Öffner oder RS-FlipFlop und SR-FlipFlop gegeneinander austauschen.

## Voraussetzung

Ein Netzwerk mit mindestens einem KOP-Element ist vorhanden.

## Vorgehen

Um ein KOP-Element durch ein anderes KOP-Element zu ersetzen, gehen Sie folgendermaßen vor:

1. Markieren Sie das KOP-Element, das Sie ersetzen möchten.
2. Bewegen Sie den Mauszeiger über das Dreieck in der oberen rechten Ecke des KOP-Elements.  
Eine Klappliste wird geöffnet.
3. Wählen Sie das KOP-Element aus der Klappliste, durch das Sie das aktuelle KOP-Element ersetzen möchten.

## Siehe auch

KOP-Elemente selektieren (Seite 1306)

KOP-Elemente kopieren (Seite 1307)

KOP-Elemente ausschneiden (Seite 1307)

KOP-Elemente aus der Zwischenablage einfügen (Seite 1308)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

KOP-Elemente löschen (Seite 1313)

## Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen

### Einführung

Durch zusätzliche Eingänge können Sie KOP-Elemente erweitern, die kommutative arithmetische Anweisungen durchführen. Solche Elemente sind z. B. die Anweisungen "Addieren" (ADD) und "Multiplizieren" (MUL). Durch zusätzliche Ausgänge können Sie die Boxen der Anweisungen "Wert kopieren" (MOVE) und "Demultiplexen" (DEMUX) erweitern.

## Voraussetzung

Ein KOP-Element ist vorhanden, das das Einfügen von zusätzlichen Eingängen bzw. Ausgängen erlaubt.

## Zusätzlichen Eingang einfügen

Um die Box eines KOP-Elements durch einen zusätzlichen Eingang zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Eingang des KOP-Elements.
2. Wählen Sie im Kontextmenü den Befehl "Eingang einfügen".  
Die Box des KOP-Elements wird durch einen zusätzlichen Eingang erweitert.

Oder:

1. Klicken Sie auf das gelbe Stern-Symbol neben dem letzten Eingang der Anweisungsbox.  
Die Box des KOP-Elements wird durch einen zusätzlichen Eingang erweitert.

## Zusätzlichen Ausgang einfügen

Um die Box eines KOP-Elements durch einen zusätzlichen Ausgang zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Ausgang des KOP-Elements.
2. Wählen Sie im Kontextmenü den Befehl "Ausgang einfügen".  
Die Box des KOP-Elements wird durch einen zusätzlichen Ausgang erweitert.

Oder:

1. Klicken Sie auf das gelbe Stern-Symbol neben dem letzten Eingang der Anweisungsbox.  
Die Box des KOP-Elements wird durch einen zusätzlichen Ausgang erweitert.

## Siehe auch

KOP-Elemente selektieren (Seite 1306)

KOP-Elemente kopieren (Seite 1307)

KOP-Elemente ausschneiden (Seite 1307)

KOP-Elemente aus der Zwischenablage einfügen (Seite 1308)

KOP-Elemente ersetzen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

KOP-Elemente löschen (Seite 1313)

## Eingänge und Ausgänge entfernen

### Einführung

Eingänge oder Ausgänge, die Sie für eine Anweisung zusätzlich eingefügt haben, können wieder entfernt werden.

### Voraussetzung

Ein KOP-Element ist vorhanden, das Sie durch zusätzliche Ein- oder Ausgänge erweitert haben.

### Eingang entfernen

Um einen Eingang zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Eingang, den Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".  
Der Eingang des KOP-Elements wird entfernt.

### Ausgang entfernen

Um einen Ausgang zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ausgang, den Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".  
Der Ausgang des KOP-Elements wird entfernt.

### Siehe auch

KOP-Elemente selektieren (Seite 1306)

KOP-Elemente kopieren (Seite 1307)

KOP-Elemente ausschneiden (Seite 1307)

KOP-Elemente aus der Zwischenablage einfügen (Seite 1308)

KOP-Elemente ersetzen (Seite 1309)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

KOP-Elemente löschen (Seite 1313)

### EN-/ENO-Mechanismus aktivieren und deaktivieren

In KOP und FUP haben bestimmte Anweisungen einen Freigabeausgang ENO (enable output) und nutzen somit den EN-/ENO-Mechanismus. Damit können Sie Laufzeitfehler in Anweisungen abfragen und darauf reagieren. Um die Performance der CPU zu steigern, ist

der EN-/ENO-Mechanismus in der Voreinstellung deaktiviert. Dadurch haben Sie zunächst nicht mehr die Möglichkeit auf Laufzeitfehler der Anweisung über den ENO-Wert zu reagieren. Sie können den EN-/ENO-Mechanismus jedoch bei Bedarf jederzeit wieder aktivieren.

Sie können für jede Anweisung den EN-/ENO-Mechanismus einzeln aktivieren, um den ENO zu generieren. Wenn Sie für eine Anweisung den EN-/ENO-Mechanismus aktivieren, werden weitere Anweisungen, die Sie danach in Ihr Programm ziehen, ebenfalls mit aktiviertem EN-/ENO-Mechanismus eingefügt. Falls Sie die Auswertung des ENO für eine Anweisung nicht nutzen möchten, können Sie den EN-/ENO-Mechanismus jederzeit wieder deaktivieren. Weitere Anweisungen, die Sie danach in Ihr Programm ziehen, werden dann ohne EN-/ENO-Mechanismus eingefügt.

Siehe auch: Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### EN-/ENO-Mechanismus aktivieren

Um den EN-/ENO-Mechanismus einer Anweisung zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus aktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO generieren".  
Für die Anweisung wird der ENO-Wert wieder generiert. Weitere Anweisungen werden mit dem Freigabeausgang eingefügt.

### EN-/ENO-Mechanismus deaktivieren

Um den EN-/ENO-Mechanismus einer Anweisung zu deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus deaktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO nicht generieren".  
Für die Anweisung wird der ENO-Wert nicht mehr generiert. Weitere Anweisungen werden ohne den Freigabeausgang eingefügt.

### Siehe auch

KOP-Elemente selektieren (Seite 1306)

KOP-Elemente kopieren (Seite 1307)

KOP-Elemente ausschneiden (Seite 1307)

KOP-Elemente aus der Zwischenablage einfügen (Seite 1308)

KOP-Elemente ersetzen (Seite 1309)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

KOP-Elemente löschen (Seite 1313)

## KOP-Elemente löschen

### Voraussetzung

Ein KOP-Element ist vorhanden.

### Vorgehen

Um ein KOP-Element zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das KOP-Element, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Siehe auch

KOP-Elemente selektieren (Seite 1306)

KOP-Elemente kopieren (Seite 1307)

KOP-Elemente ausschneiden (Seite 1307)

KOP-Elemente aus der Zwischenablage einfügen (Seite 1308)

KOP-Elemente ersetzen (Seite 1309)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1311)

## Operanden in KOP-Anweisungen beschalten

### Operanden beschalten

Beim Einfügen eines KOP-Elements werden die Zeichenfolgen "<???", "<??.>" und "..." als Platzhalter für die Parameter eingefügt. Die rot dargestellten Zeichenfolgen "<???", "<??.>" kennzeichnen Parameter, die beschaltet werden müssen. Die schwarz dargestellte Zeichenfolge "..." kennzeichnet Parameter, die beschaltet werden können. "<??.>" steht für boolesche Platzhalter.

---

#### Hinweis

Wenn Sie den Mauszeiger über den Platzhalter bewegen, wird der erwartete Datentyp angezeigt.

---

### Voraussetzung

Ein KOP-Element ist vorhanden.

## Vorgehen

Um die Parameter eines KOP-Elements zu beschalten, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Platzhalter des Parameters.  
Ein Eingabefeld wird geöffnet, der Platzhalter ist markiert.
2. Geben Sie den entsprechenden Parameter ein.

---

### Hinweis

Wenn Sie die absolute Adresse eines Parameters eingeben, der bereits definiert ist, wird diese Absolutadresse in den symbolischen Namen des Parameters geändert, sobald Sie die Eingabe bestätigen. Haben Sie den Parameter noch nicht definiert, wird in der PLC-Variablen-tabelle eine neue Variable mit dieser Absolutadresse und dem Standardnamen "Tag\_<n>" eingetragen. Bestätigen Sie Ihre Eingabe, wird die Absolutadresse gegen den symbolischen Namen "Tag\_<n>" ausgetauscht.

---

3. Bestätigen Sie den Parameter mit der Eingabetaste.
4. Wenn Sie den Parameter bisher nicht definiert haben, können Sie dies über das Kontextmenü direkt im Programmiereditor erledigen.  
Siehe auch:  
PLC-Variablen im Programmiereditor deklarieren (Seite 1195)  
Lokale Variablen im Programmiereditor deklarieren (Seite 1263)

Oder per Drag & Drop aus der PLC-Variablen-tabelle:

1. Selektieren Sie in der Projektnavigation den Ordner "PLC-Variablen" oder öffnen Sie die PLC-Variablen-tabelle.
2. Wenn Sie die PLC-Variablen-tabelle geöffnet haben, ziehen Sie das Symbol aus der ersten Spalte der gewünschten Variablen an die entsprechende Stelle in Ihrem Programm. Falls Sie die PLC-Variablen-tabelle nicht geöffnet haben, öffnen Sie die Detailansicht. Ziehen Sie die gewünschte Variable aus der Detailansicht an die entsprechende Stelle in Ihrem Programm.

Oder per Drag & Drop aus der Bausteinschnittstelle:

1. Öffnen Sie die Bausteinschnittstelle.
2. Ziehen Sie den gewünschten Operanden aus der Bausteinschnittstelle in das Anweisungsfenster.

## Ergebnis

- Wenn die Syntax fehlerfrei ist, wird der Parameter schwarz dargestellt. Der Editor springt zum nächsten Platzhalter.
- Wenn in der Syntax ein Fehler vorhanden ist, wird das Eingabefeld nicht verlassen und es wird eine entsprechende Fehlermeldung in der Statuszeile angezeigt. Wenn Sie die Eingabetaste erneut drücken, wird das Eingabefeld geschlossen und die fehlerhafte Eingabe wird rot und kursiv dargestellt.



## Versteckte Parameter beschalten

### Einführung

Abhängig von der verwendeten CPU können Sie in Ihrem Programm komplexe Anweisungen verwenden, die mit dem TIA-Portal ausgeliefert werden. Diese Anweisungen können Parameter enthalten, die als versteckt deklariert wurden.

Wenn eine Anweisung versteckte Parameter enthält, verfügt die Anweisungsbox über einen kleinen Pfeil am unteren Rand. Die versteckten Parameter erkennen Sie an der weißen Schriftfarbe.

Sie können die versteckten Parameter jederzeit einblenden und beschalten.

### Versteckte Parameter ein- oder ausblenden

Um versteckte Parameter ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten am unteren Rand der Anweisungsbox, um versteckte Parameter einzublenden.
2. Klicken Sie auf den Pfeil-nach-oben am unteren Rand der Anweisungsbox, um versteckte Parameter auszublenden.

### Versteckte Parameter beschalten

Um versteckte Parameter zu beschalten, gehen Sie folgendermaßen vor:

1. Beschalten Sie die versteckten Parameter wie normale sichtbare Parameter. Der versteckte Parameter wird zu einem sichtbaren Parameter.

### Siehe auch

Bibliotheken verwenden (Seite 313)

## Variableninformationen ein- oder ausblenden

### Einführung

Sie können die folgenden Informationen zu den verwendeten Variablen im Programmiereditor einblenden:

- Name der Variablen
- Adresse der Variablen
- Kommentar zur Dokumentation der Variablen

Die Informationen werden bei lokalen Variablen und DB-Variablen aus der Bausteinschnittstelle und bei CPU-weit gültigen Variablen aus der PLC-Variablen-tabelle entnommen.

Sie können die Variableninformationen entweder für alle Bausteine oder für einzelne geöffnete Bausteine einblenden. Wenn Sie die Variableninformationen für alle Bausteine einblenden,

werden die Variableninformationen für alle aktuell und zukünftig geöffneten Bausteine eingeblendet.

Sie können die Variableninformationen jederzeit wieder ausblenden. Wenn Sie die Variableninformationen für alle Bausteine ausgeblendet haben, können Sie sie für einzelne geöffnete wieder einblenden.

### Variableninformationen für alle Bausteine ein- oder ausblenden

Um die Variableninformationen für alle Bausteine ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Wenn Sie die Variableninformationen einblenden möchten, aktivieren Sie im Bereich "Ansicht" das Optionskästchen "mit Variableninformationen". Wenn Sie die Variableninformationen ausblenden möchten, deaktivieren Sie das Optionskästchen "mit Variableninformationen".  
Die Variableninformationen werden für alle geöffneten Bausteine ein- oder ausgeblendet. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen abhängig von der gewählten Einstellung angezeigt oder nicht angezeigt.

### Variableninformationen für einen geöffneten Baustein ein- oder ausblenden

Um die Variableninformationen für einen geöffneten Baustein ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Aktivieren oder deaktivieren Sie im Menü "Ansicht > Anzeigen mit" das Optionskästchen "Variableninformationen" oder klicken Sie in der Funktionsleiste auf die Schaltfläche "Variableninformationen ein/aus".  
Die Variableninformationen werden ein- oder ausgeblendet.

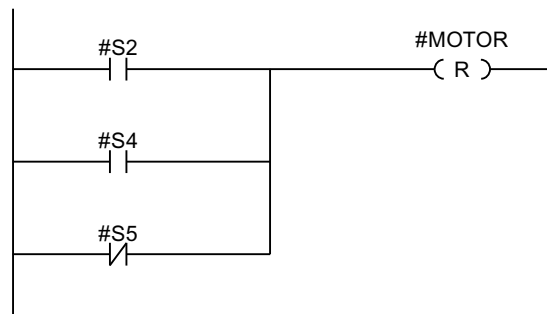
## Zweige in KOP

### Grundlagen zu Zweigen in KOP

#### Definition

Um mit der Programmiersprache Kontaktplan (KOP) Parallelschaltungen zu programmieren, verwenden Sie Zweige. Zweige werden in den Hauptstrompfad eingefügt. Sie können in den Zweig mehrere Kontakte einfügen und damit eine Parallelschaltung von Reihenschaltungen erreichen. Auf diese Weise können Sie komplexe Kontaktpläne programmieren.

Das folgende Bild zeigt ein Beispiel für die Verwendung von Zweigen:



MOTOR führt Signal 1, wenn eine der folgenden Bedingungen erfüllt ist:

- An S2 oder S4 steht Signal 1 an
- An S5 steht Signal 0 an.

### Siehe auch

Regeln für Zweige in KOP (Seite 1317)

Zweige in KOP-Netzwerke einfügen (Seite 1318)

Zweige in KOP-Netzwerke schließen (Seite 1318)

Zweige in KOP-Netzwerke löschen (Seite 1319)

## Regeln für Zweige in KOP

### Regeln

Für parallele Verzweigungen gelten folgende Regeln:

- Das Einfügen eines Parallelzweigs ist nur dann möglich, wenn im Hauptzweig bereits ein KOP-Element existiert.
- Parallelzweige werden nach unten geöffnet oder direkt an die Stromschiene angeschlossen und nach oben geschlossen.
- Parallelzweige werden hinter dem markierten KOP-Element geöffnet.
- Parallelzweige werden hinter dem markierten KOP-Element geschlossen.
- Um einen Parallelzweig zu löschen, müssen Sie alle KOP-Elemente dieses Zweiges löschen. Beim Entfernen des letzten KOP-Elements der Verzweigung wird der Rest der Verzweigung ebenfalls entfernt.

### Siehe auch

Grundlagen zu Zweigen in KOP (Seite 1316)

Zweige in KOP-Netzwerke einfügen (Seite 1318)

Zweige in KOP-Netzwerke löschen (Seite 1319)

Zweige in KOP-Netzwerke schließen (Seite 1318)

## Zweige in KOP-Netzwerke einfügen

Sie können in einem Netzwerk mehrere Zweige erstellen.

### Voraussetzung

- Ein Netzwerk ist vorhanden.
- Das Netzwerk enthält Elemente.

### Vorgehen

Um einen neuen Zweig in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Verzweigung öffnen".
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk.  
Falls Sie den neuen Zweig direkt an die Stromschiene setzen möchten, ziehen Sie das Element auf die Stromschiene.

### Siehe auch

Grundlagen zu Zweigen in KOP (Seite 1316)

Regeln für Zweige in KOP (Seite 1317)

Zweige in KOP-Netzwerke löschen (Seite 1319)

## Zweige in KOP-Netzwerke schließen

Zweige müssen an geeigneten Stellen wieder geschlossen werden. Wenn erforderlich, werden die Zweige so angeordnet, dass Überkreuzungen von Zweigen vermieden werden.

### Voraussetzung

Ein Zweig ist vorhanden.

### Vorgehen

Um einen offenen Zweig zu schließen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den offenen Zweig.
2. Drücken und halten Sie die linke Maustaste.  
Eine gestrichelte Linie wird eingeblendet, sobald Sie den Mauszeiger bewegen.
3. Ziehen Sie die gestrichelte Linie auf eine geeignete Stelle im Netzwerk. Erlaubte Verbindungen werden durch grüne Linien angezeigt.
4. Lassen Sie die linke Maustaste los.

## Siehe auch

Grundlagen zu Zweigen in KOP (Seite 1316)  
Regeln für Zweige in KOP (Seite 1317)

## Zweige in KOP-Netzwerke löschen

### Voraussetzung

Ein Zweig ist vorhanden.

### Vorgehen

Um einen Zweig zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Verbindungslinie, die den Zweig mit dem Hauptzweig verbindet.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

## Siehe auch

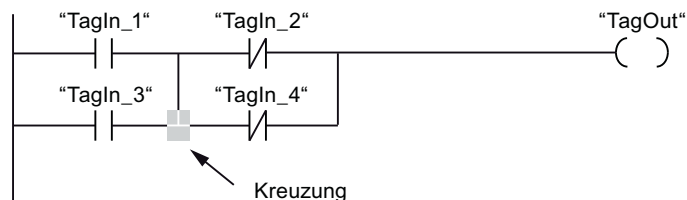
Grundlagen zu Zweigen in KOP (Seite 1316)  
Regeln für Zweige in KOP (Seite 1317)  
Zweige in KOP-Netzwerke einfügen (Seite 1318)

## Kreuzungen in KOP

### Grundlagen zu Kreuzungen in KOP

### Definition

Eine Kreuzung ist eine Stelle in einem KOP-Netzwerk, an der ein Zweig geschlossen und gleichzeitig ein weiterer Zweig geöffnet wird.



"TagOut" erhält Signal 1, wenn folgende beiden Bedingungen erfüllt sind:

- "TagIn\_1" oder "TagIn\_3" führt Signal 1
- "TagIn\_2" oder "TagIn\_4" führt Signal 0

## Kreuzungen einfügen

Sie können Kreuzungen in ein KOP-Netzwerk einfügen, indem Sie Verbindungen zwischen dem Hauptzweig und einem zusätzlichen Zweig oder zwischen verschiedenen Zweigen erstellen.

### Voraussetzung

Ein Zweig ist vorhanden.

### Vorgehen

Um in ein KOP-Netzwerk eine Kreuzung einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Verzweigung öffnen".
3. Ziehen Sie das Element per Drag & Drop hinter den vorhandenen Zweig.
4. Fügen Sie ein beliebiges Element in den offenen Zweig ein.
5. Klicken Sie hinter dem eingefügten Element auf den Pfeil des geöffneten Zweigs.
6. Halten Sie die linke Maustaste gedrückt und ziehen Sie die gestrichelte Verbindungslinie auf den Hauptzweig.
7. Lassen Sie die linke Maustaste los.

### Siehe auch

Kreuzungen umhängen (Seite 1320)

Kreuzungen löschen (Seite 1321)

Zweige in KOP-Netzwerke einfügen (Seite 1318)

## Kreuzungen umhängen

### Voraussetzung

Eine Kreuzung ist vorhanden.

### Vorgehen

Um eine Kreuzung umzuhängen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Verbindungslinie, die in den jeweiligen Zweigen die Kreuzungen definiert.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".
3. Öffnen Sie die Task Card "Anweisungen".
4. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Verzweigung öffnen".

5. Ziehen Sie das Element per Drag & Drop an die Stelle im Netzwerk, an der Sie die neue Kreuzung einfügen möchten.
6. Klicken Sie auf den Pfeil des geöffneten Zweigs.
7. Halten Sie die linke Maustaste gedrückt und ziehen Sie die gestrichelte Verbindungslinie auf den Nebenzweig, in den Sie die neue Kreuzung einfügen möchten.
8. Lassen Sie die linke Maustaste los.

### Siehe auch

Kreuzungen einfügen (Seite 1320)

Kreuzungen löschen (Seite 1321)

### Kreuzungen löschen

#### Voraussetzung

Eine Kreuzung ist vorhanden.

#### Vorgehen

Um eine Kreuzung zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Verbindungslinie, die in den jeweiligen Zweigen die Kreuzungen definiert.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Siehe auch

Kreuzungen einfügen (Seite 1320)

Kreuzungen umhängen (Seite 1320)

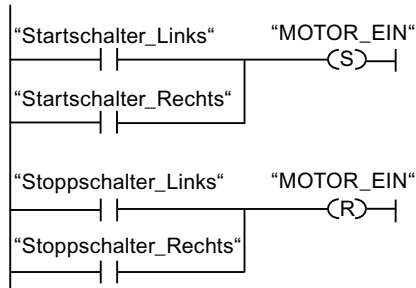
### Strompfade in KOP

#### Grundlagen zu Strompfaden in KOP

#### Verwendung von Strompfaden

Das Programm wird in einem oder mehreren Netzwerken abgebildet. Ein Netzwerk enthält am linken Rand eine Stromschiene, von der ein oder mehrere Strompfade ausgehen können. Auf den Strompfaden werden die Abfragen der binären Signale in Form von Kontakten angeordnet. Durch serielle Anordnung der Elemente auf einem Strompfad entsteht eine Reihenschaltung, durch Anordnung auf parallelen Zweigen entsteht eine Parallelschaltung. Ein Strompfad wird von einer Spule oder einer Box abgeschlossen, in die das Verknüpfungsergebnis geschrieben wird.

Das folgende Bild zeigt ein Beispiel für die Verwendung von mehreren Strompfaden innerhalb eines Netzwerks:



## Regeln

Beachten Sie folgende Regeln bei der Verwendung von mehreren Strompfaden:

- Verbindungen zwischen Strompfaden sind nicht erlaubt.
- Es ist nur eine Sprunganweisung pro Netzwerk zulässig. Die Platzierungsregeln für Sprunganweisungen bleiben dabei erhalten.

## Abarbeitung von Strompfaden

Die Abarbeitung von Strompfaden und Netzwerken erfolgt von oben nach unten und von links nach rechts. Das bedeutet, dass zuerst die erste Anweisung im ersten Strompfad des ersten Netzwerks bearbeitet wird. Anschließend werden alle Anweisungen dieses Strompfads bearbeitet. Danach kommen alle weiteren Strompfade des ersten Netzwerks an die Reihe. Erst wenn alle Strompfade abgearbeitet wurden, wird das nächste Netzwerk bearbeitet.

## Unterschiede zwischen Zweigen und Strompfaden

Der Unterschied zwischen Zweigen und Strompfaden besteht darin, dass die Strompfade eigenständige Zweige sind, die auch in einem anderen Netzwerk stehen können. Zweige hingegen erlauben die Programmierung einer Parallelschaltung.

## Siehe auch

Strompfad einfügen (Seite 1322)

Strompfad löschen (Seite 1323)

## Strompfad einfügen

### Voraussetzung

- Ein Baustein ist geöffnet.
- Ein Netzwerk ist vorhanden.



## Vorgehen

Um einen neuen Strompfad in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Fügen Sie eine beliebige Spule an der Stromschiene ein.  
Ein neuer Strompfad wird eingefügt und die Spule wird am Ende des Strompfads platziert.
2. Fügen Sie weitere Anweisungen in den neuen Strompfad ein.

## Siehe auch

Grundlagen zu Strompfaden in KOP (Seite 1321)

Strompfad löschen (Seite 1323)

## Strompfad löschen

### Voraussetzung

Ein Strompfad ist vorhanden.

## Vorgehen

Um einen Strompfad zu löschen, gehen Sie folgendermaßen vor:

1. Ziehen Sie mit gedrückter linker Maustaste einen Rahmen um den Strompfad. Achten Sie darauf, dass Sie dabei alle Anweisungen des Strompfads selektieren. Alternativ können Sie auch mit gedrückter <Shift>-Taste die erste und letzte Anweisung des Strompfads selektieren.
2. Klicken Sie mit der rechten Maustaste auf eine der Anweisungen im Strompfad.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".

## Siehe auch

Grundlagen zu Strompfaden in KOP (Seite 1321)

Strompfad einfügen (Seite 1322)

## FUP-Programme erstellen

### Grundlagen zu FUP

### Programmiersprache FUP

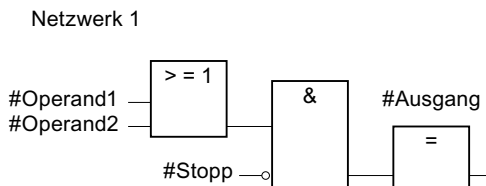
## Übersicht über die Programmiersprache FUP (Funktionsplan)

FUP ist eine grafische Programmiersprache. Die Darstellung ist elektronischen Schaltkreissystemen nachempfunden.

Das Programm wird in einem oder mehreren Netzwerken abgebildet. Ein Netzwerk enthält ein oder mehrere Verknüpfungspfade. Die Abfragen der binären Signale werden durch Boxen miteinander verknüpft. Zur Darstellung der Logik werden die von der booleschen Algebra bekannten grafischen Logiksymbole verwendet.

### Beispiel für Netzwerke in FUP

Das folgende Bild zeigt ein FUP-Netzwerk mit UND- und ODER-Boxen und einer Zuweisung:



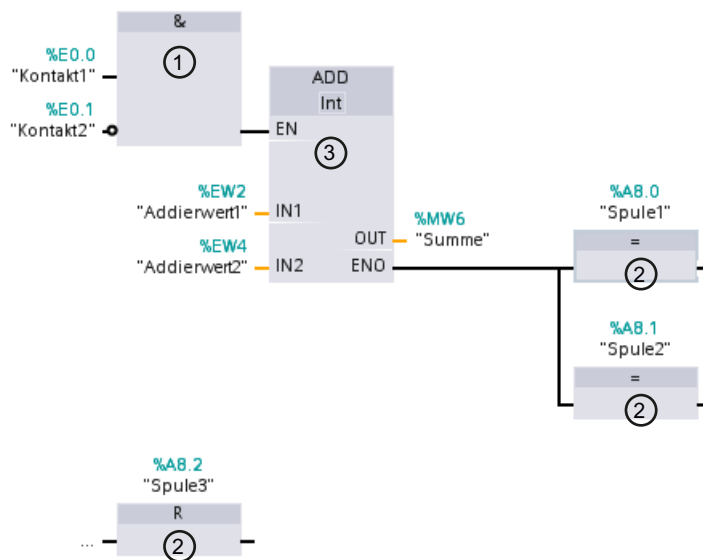
### Übersicht über die FUP-Elemente

#### FUP-Elemente

Ein FUP-Programm besteht aus einzelnen Elementen, die über den binären Signalfuss miteinander verbunden sind. Die meisten Programmelemente müssen mit Variablen versorgt werden.

Das Programmieren in einem FUP-Netzwerk verläuft von links nach rechts.

Das folgende Bild zeigt beispielsweise Elemente eines FUP-Netzwerks:



- 1) Binäre Funktion
- 2) Einfache Box

### 3) Komplexe Box

#### Binäre Funktionen

Mit binären Funktionen können Sie Binäroperanden abfragen und deren Signalzustände verknüpfen. Beispiele für binäre Funktionen sind die Anweisungen "UND-Verknüpfung", "ODER-Verknüpfung" und "EXKLUSIV ODER-Verknüpfung".

#### Einfache Boxen

Mit einfachen Boxen können Sie Binäroperanden steuern, Flanken auswerten oder Sprungfunktionen im Programm ausführen. Einfache Boxen haben im Allgemeinen nur einen einzigen Eingang.

#### Komplexe Boxen

Komplexe Boxen stellen Programmelemente mit komplexen Funktionen dar. Eine Ausnahme dabei ist die Leerbox. Die Leerbox dient als Platzhalter, in dem Sie die gewünschte Anweisung auswählen können.

Die folgenden Arten von Boxen stehen Ihnen in einem FUP-Programm zur Verfügung:

- Komplexe Boxen ohne EN-/ENO-Mechanismus:  
Eine Box wird unabhängig vom Signalzustand an den Box-Eingängen ausgeführt. Der Fehlerstatus der Bearbeitung kann nicht abgefragt werden.
- Komplexe Boxen mit EN-/ENO-Mechanismus:  
Eine Box wird nur dann ausgeführt, wenn der Freigabeeingang "EN" den Signalzustand "1" führt. Bei ordnungsgemäßer Bearbeitung der Box führt der Freigabeausgang "ENO" den Signalzustand "1". Wenn während der Bearbeitung ein Fehler auftritt, wird der Freigabeausgang "ENO" zurückgesetzt.  
Wenn der Freigabeeingang EN nicht verschaltet ist, wird die Box immer ausgeführt.

Aufrufe von Codebausteinen werden im Netzwerk auch als komplexe Boxen mit EN-/ENO-Mechanismus dargestellt.

## Einstellungen zu FUP

### Übersicht über die Einstellungen zu FUP

#### Überblick

Die folgende Tabelle zeigt die Einstellungen, die Sie vornehmen können:

| Gruppe        | Einstellung            | Beschreibung  |
|---------------|------------------------|---|
| Schriftart    | Schriftgröße           | Schriftgröße im Programmiereditor   |
| Ansicht       | Layout                 | Kompakt oder weit<br>Ändert den vertikalen Abstand zwischen Operanden und anderen Objekten (z. B. Operanden und Kontakt). Die Änderung wird erst nach dem erneuten Öffnen eines Bausteins sichtbar. |
|               | mit Absolutinformation | Zusätzliche Anzeige der Absolutadressen   |
| Operandenfeld | Maximale Breite        | Maximale Anzahl der Zeichen, die horizontal in das Operandenfeld eingegeben werden können. Mit dieser Einstellung wird das Layout der Netzwerke neu berechnet.                                      |
|               | Maximale Höhe          | Maximale Anzahl der Zeichen, die vertikal in das Operandenfeld eingegeben werden können. Mit dieser Einstellung wird das Layout der Netzwerke neu berechnet.  |

#### Siehe auch

Einstellungen ändern (Seite 1326)

### Einstellungen ändern

#### Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.

## Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

## Siehe auch

Übersicht über die Einstellungen zu FUP (Seite 1326)

## Arbeiten mit Netzwerken

### Netzwerke verwenden

## Funktion

Das Anwenderprogramm wird im Baustein innerhalb von Netzwerken erstellt. Damit ein Codebaustein programmiert werden kann, muss er mindestens ein Netzwerk enthalten. Um eine bessere Übersichtlichkeit des Anwenderprogramms zu erreichen, können Sie Ihr Programm auch in mehrere Netzwerke gliedern.

## Siehe auch

Netzwerktitel eingeben (Seite 1330)

Netzwerkcommentar eingeben (Seite 1331)

In Netzwerken navigieren (Seite 1332)

## Netzwerke einfügen

### Voraussetzung

Ein Baustein ist geöffnet.

### Vorgehen

Um ein neues Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, hinter dem Sie ein neues Netzwerk einfügen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Netzwerk einfügen".

## Ergebnis

Ein neues, leeres Netzwerk wird in den Baustein eingefügt.

## Siehe auch

- Netzwerktitel eingeben (Seite 1330)
- Netzwerkcommentar eingeben (Seite 1331)
- In Netzwerken navigieren (Seite 1332)

## Netzwerke selektieren

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Ein Netzwerk selektieren

Um ein Netzwerk zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelleiste des Netzwerks, das Sie selektieren möchten.

### Mehrere Netzwerke selektieren

Um mehrere einzelne Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie auf alle Netzwerke, die Sie selektieren möchten.

Um mehrere aufeinanderfolgende Netzwerke zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Shift>.
2. Klicken Sie auf das erste Netzwerk, das Sie selektieren möchten.
3. Klicken Sie auf das letzte Netzwerk, das Sie selektieren möchten.  
Das erste und das letzte Netzwerk und alle dazwischen liegenden Netzwerke werden selektiert.

## Siehe auch

- Netzwerke einfügen (Seite 1327)
- Netzwerktitel eingeben (Seite 1330)
- Netzwerkcommentar eingeben (Seite 1331)
- In Netzwerken navigieren (Seite 1332)

## Netzwerke kopieren und einfügen

Kopierte Netzwerke können innerhalb des Bausteins oder in einen anderen Baustein eingefügt werden. Netzwerke, die in KOP oder FUP erstellt wurden, können auch in Bausteine der jeweils anderen Programmiersprache eingefügt werden.

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Vorgehen

Um ein Netzwerk zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk oder die Netzwerke, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Selektieren Sie das Netzwerk, hinter dem Sie das kopierte Netzwerk einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

### Siehe auch

Netzwerke einfügen (Seite 1327)  
Netzwerke selektieren (Seite 1328)  
Netzwerktitle eingeben (Seite 1330)  
Netzwerkcommentar eingeben (Seite 1331)  
In Netzwerken navigieren (Seite 1332)

## Netzwerke löschen

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Vorgehen

Um ein Netzwerk zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Netzwerk, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Siehe auch

Netzwerke einfügen (Seite 1327)  
Netzwerke selektieren (Seite 1328)  
Netzwerke kopieren und einfügen (Seite 1328)  
Netzwerktitle eingeben (Seite 1330)  
Netzwerkcommentar eingeben (Seite 1331)  
In Netzwerken navigieren (Seite 1332)

## Netzwerke reduzieren und erweitern

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Ein Netzwerk auf- und zuklappen

Um ein Netzwerk aufzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts in der Titelleiste des Netzwerks.

Um ein Netzwerk zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten in der Titelleiste des Netzwerks.

### Alle Netzwerke auf- und zuklappen

Um alle Netzwerke auf- oder zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Alle Netzwerke öffnen" bzw. "Alle Netzwerke schließen".

### Siehe auch

Netzwerke einfügen (Seite 1327)

Netzwerke selektieren (Seite 1328)

Netzwerke kopieren und einfügen (Seite 1328)

Netzwerke löschen (Seite 1329)

Netzwerktitel eingeben (Seite 1330)

Netzwerkcommentar eingeben (Seite 1331)

In Netzwerken navigieren (Seite 1332)

### Netzwerktitel eingeben

Der Netzwerktitel ist die Überschrift eines Netzwerks. Die Länge des Netzwerktitels ist auf eine Zeile beschränkt.

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Vorgehen

Um einen Netzwerktitel einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Titelzeile des Netzwerks.
2. Geben Sie den Netzwerktitel ein.



**Siehe auch**

- Netzwerke verwenden (Seite 1327)
- Netzwerke einfügen (Seite 1327)
- Netzwerke selektieren (Seite 1328)
- Netzwerke kopieren und einfügen (Seite 1328)
- Netzwerke löschen (Seite 1329)
- Netzwerke reduzieren und erweitern (Seite 1330)
- Netzwerkcommentar eingeben (Seite 1331)
- In Netzwerken navigieren (Seite 1332)

**Netzwerkcommentar eingeben**

Mithilfe von Netzwerkcommentaren können Sie die Programminhalte der einzelnen Netzwerke erläutern. Sie können z. B. die Funktion des Netzwerks beschreiben oder auf Besonderheiten hinweisen.

**Voraussetzung**

Ein Netzwerk ist vorhanden.

**Vorgehen**

Um einen Netzwerkcommentar einzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-rechts vor dem Netzwerktitel.
2. Wenn der Commentarbereich nicht sichtbar ist, klicken Sie in der Funktionsleiste auf die Schaltfläche "Netzwerkcommentare ein/aus".  
Der Commentarbereich wird angezeigt.
3. Klicken Sie im Commentarbereich auf "Commentar".  
Die Textstelle "Commentar" wird markiert.
4. Geben Sie den Netzwerkcommentar ein.

## Siehe auch

- Netzwerke verwenden (Seite 1327)
- Netzwerke einfügen (Seite 1327)
- Netzwerke selektieren (Seite 1328)
- Netzwerke kopieren und einfügen (Seite 1328)
- Netzwerke löschen (Seite 1329)
- Netzwerke reduzieren und erweitern (Seite 1330)
- Netzwerktitel eingeben (Seite 1330)
- In Netzwerken navigieren (Seite 1332)

## In Netzwerken navigieren

Sie haben die Möglichkeit, in einem Baustein direkt zu einer bestimmten Position zu navigieren.

## Vorgehen

Um zu einer bestimmten Position innerhalb eines Bausteins zu navigieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Codebereich des Programmierfensters.
2. Wählen Sie im Kontextmenü den Befehl "Gehe zu > Netzwerk/Zeile".  
Der Dialog "Gehe zu" wird geöffnet.
3. Geben Sie das Netzwerk ein, zu dem Sie navigieren möchten.
4. Geben Sie die Zeilennummer des Netzwerks ein, zu der Sie navigieren möchten.
5. Bestätigen Sie Ihre Eingabe mit "OK".

## Ergebnis

Wenn möglich wird die entsprechende Zeile angezeigt. Falls das gewünschte Netzwerk oder die gewünschte Zeile nicht existiert, wird das letzte existierende Netzwerk bzw. die letzte existierende Zeile im gewünschten Netzwerk angezeigt.

## Siehe auch

- Netzwerke verwenden (Seite 1327)
- Netzwerke einfügen (Seite 1327)
- Netzwerke selektieren (Seite 1328)
- Netzwerke kopieren und einfügen (Seite 1328)
- Netzwerke löschen (Seite 1329)
- Netzwerke reduzieren und erweitern (Seite 1330)
- Netzwerktitle eingeben (Seite 1330)
- Netzwerkcommentar eingeben (Seite 1331)

## FUP-Elemente einfügen

### Regeln für die Verwendung von FUP-Elementen

#### Regeln

Beachten Sie beim Einfügen von FUP-Elementen folgende Regeln:

- Ein FUP-Netzwerk kann aus mehreren Elementen bestehen. Alle Elemente eines Verknüpfungspfad müssen nach IEC 61131-3 untereinander verbunden sein.
- An Boxen mit binären Verknüpfungen (z. B. UND, ODER) können Standard-Boxen (Flipflops, Zähler, Zeiten, Rechenoperationen usw.) als Ausgang angefügt werden. Ausgenommen von dieser Regelung sind Vergleichboxen.
- Nur boolesche Eingänge einer Anweisung sind vorverknüpfbar.
- Nur der unterste boolesche Ausgang einer Anweisung ist weiter verknüpfbar.
- Die Beschaltung des Freigabeeingangs EN bzw. des Freigabeausgangs ENO von Boxen kann erfolgen, ist aber nicht zwingend erforderlich.
- Binäre Verknüpfungen können nicht mit Konstanten belegt werden, z. B. mit TRUE oder FALSE. Verwenden Sie stattdessen Variablen mit dem Datentyp BOOL.
- Pro Netzwerk kann nur eine Sprunganweisung eingefügt werden.
- Pro Netzwerk kann nur eine Sprungmarke eingefügt werden.
- Anweisungen mit positiver oder negativer Flankenauswertung dürfen nicht direkt am linken Rand des Netzwerks angeordnet werden, da diese eine Vorverknüpfung voraussetzen.

### Platzierungsregeln für S7-1200/1500 CPUs

Die folgende Tabelle zeigt die Anweisungen, die nur am Ende des Netzwerks positioniert werden können:

| Anweisung |                        | Vorverknüpfung notwendig |
|-----------|------------------------|--------------------------|
| Mnemonic  | Name                   |                          |
| SET_BF    | Bitfeld setzen         | Nein                     |
| RESET_BF  | Bitfeld rücksetzen     | Nein                     |
| JMP       | Springen bei VKE = 1   | Nein                     |
| JMPN      | Springen bei VKE = 0   | Ja                       |
| JMP_LIST  | Sprungliste definieren | Nein                     |
| SWITCH    | Sprungverteiler        | Nein                     |
| RET       | Zurück springen        | Nein                     |

### Platzierungsregeln für S7-300/400 CPUs

Die folgende Tabelle zeigt die Anweisungen, die nur am Ende des Netzwerks positioniert werden können:

| Anweisung |   | Vorverknüpfung notwendig |
|-----------|---|--------------------------|
| Mnemonic  | Name  |                          |
| S         | Ausgang setzen                                    | Ja                       |
| R         | Ausgang rücksetzen                                | Ja                       |
| SI        | Zeit als Impuls starten                           | Ja                       |
| SV        | Zeit als verlängerten Impuls starten              | Ja                       |
| SE        | Zeit als Einschaltverzögerung starten             | Ja                       |
| SS        | Zeit als speichernde Einschaltverzögerung starten | Ja                       |
| SA        | Zeit als Ausschaltverzögerung starten             | Ja                       |
| SZ        | Zähleranfangswert setzen                          | Ja                       |
| ZV        | Vorwärts zählen                                   | Ja                       |
| ZR        | Rückwärts zählen                                  | Ja                       |
| JMP       | Springen bei VKE = 1                              | Nein                     |
| JMPN      | Springen bei VKE = 0                              | Ja                       |
| RET       | Zurück springen                                   | Nein                     |
| OPN       | Globalen Datenbaustein öffnen                     | Nein                     |
| OPNI      | Instanz-Datenbaustein öffnen                      | Nein                     |
| CALL      | Baustein aufrufen                                 | Nein                     |
| SAVE      | VKE im BIE-Bit speichern                          | Nein                     |
| MCRA      | MCR-Bereich aktivieren                            | Nein                     |
| MCRD      | MCR-Bereich deaktivieren                          | Nein                     |
| MCR<      | MCR-Bereiche öffnen                               | Nein                     |
| MCR>      | MCR-Bereiche schließen                            | Nein                     |

## FUP-Elemente über Task Card "Anweisungen" einfügen

### Voraussetzung

Ein Netzwerk ist vorhanden.

### Vorgehen

Um ein FUP-Element mithilfe der Task Card "Anweisungen" in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie zu dem FUP-Element, das Sie einfügen möchten.
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk. Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie für den Funktionsbaustein einen Instanz-Datenbaustein als Einzel- oder Multiinstanz erstellen, in dem die Daten des eingefügten Elements gespeichert werden. Nach dem Erstellen finden Sie den neuen Instanz-Datenbaustein in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Wenn Sie "Multiinstanz" ausgewählt haben, finden Sie diese in der Bausteinschnittstelle im Abschnitt "Static".

Oder:

1. Markieren Sie die Stelle im Netzwerk, an der Sie das Element einfügen möchten.
2. Öffnen Sie die Task Card "Anweisungen".
3. Doppelklicken Sie auf das einzufügende Element. Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie für den Funktionsbaustein einen Instanz-Datenbaustein als Einzel- oder Multiinstanz erstellen, in dem die Daten des eingefügten Elements gespeichert werden. Nach dem Erstellen finden Sie den neuen Instanz-Datenbaustein in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Wenn Sie "Multiinstanz" ausgewählt haben, finden Sie diese in der Bausteinschnittstelle im Abschnitt "Static".

### Ergebnis

Das ausgewählte FUP-Element wird mit Platzhaltern für die Parameter eingefügt.

### Siehe auch

Regeln für die Verwendung von FUP-Elementen (Seite 1333)

## FUP-Elemente mithilfe einer Leerbox einfügen

### Voraussetzung

Ein Netzwerk ist vorhanden.

## Vorgehen

Um ein FUP-Element mithilfe einer Leerbox in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Leerbox".
3. Ziehen Sie das Element "Leerbox" per Drag & Drop an die gewünschte Stelle im Netzwerk.
4. Bewegen Sie den Mauszeiger über das Dreieck in der oberen rechten Ecke der Leerbox. Eine Klappliste wird geöffnet.
5. Wählen Sie das gewünschte FUP-Element aus der Klappliste.  
Wenn das Element systemintern ein Funktionsbaustein (FB) ist, wird der Dialog "Aufrufoptionen" geöffnet. In diesem Dialog können Sie für den Funktionsbaustein einen Instanz-Datenbaustein als Einzel- oder Multiinstanz erstellen, in dem die Daten des eingefügten Elements gespeichert werden. Nach dem Erstellen finden Sie den neuen Instanz-Datenbaustein in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Wenn Sie "Multiinstanz" ausgewählt haben, finden Sie diese in der Bausteinschnittstelle im Abschnitt "Static".

## Ergebnis

Die Leerbox wird zum entsprechenden FUP-Element geändert. Für die Parameter werden Platzhalter eingefügt.

## Datentyp eines FUP-Elements auswählen

### Auswahl eines Datentyps

### Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie eine dieser Anweisungen im Programm verwenden, müssen Sie an der konkreten Programmstelle ein für die Anweisung zulässiger Datentyp festlegen. Bei einigen Anweisungen müssen Sie den Datentypen für die Eingänge und für die Ausgänge separat auswählen.

---

#### Hinweis

Der zulässige Datentyp (BOOL) für die Variablen am Freigabeeingang EN und am Freigabeausgang ENO ist vom System vorgegeben und kann nicht geändert werden.

---

Die für eine Anweisung zulässigen Datentypen sind in der Klappliste der Anweisung aufgeführt. Durch die Auswahl eines Eintrags aus der Klappliste bestimmen Sie den Datentyp der Anweisung. Wenn der Datentyp eines Operanden vom Datentyp der Anweisung abweicht und nicht implizit konvertiert werden kann, wird der Operand rot angezeigt und ein Rollout mit der entsprechenden Fehlermeldung eingeblendet.

## Datentypauswahl von mathematischen Anweisungen

Einige mathematische Anweisungen bieten Ihnen die Möglichkeit, den Datentyp automatisch passend zu den Datentypen der Operanden einstellen zu lassen. Diese Anweisungen besitzen in der Klappliste für die Datentypauswahl zusätzlich zu den eigentlichen Datentypen noch den Eintrag "Auto". Wenn Sie diesen Eintrag auswählen und den ersten Operanden beschalten, wird der Datentyp des Operanden als Datentyp für die Anweisung verwendet. Der Eintrag in der Klappliste ändert sich zu "Auto (<Datentyp>)", z. B. "Auto (Real)". Wenn Sie weitere Operanden beschalten, wird der automatisch eingestellte Datentyp der Anweisung nach folgenden Kriterien angepasst:

- Sie versorgen alle weiteren Operanden mit Variablen des gleichen Datentyps:  
Der Datentyp der Anweisung wird nicht geändert.
- Sie versorgen alle weiteren Operanden mit Variablen, deren Datentyp kleiner ist als der Datentyp der Anweisung:  
Der Datentyp der Anweisung wird nicht geändert. Für die Operanden mit dem kleineren Datentyp wird ggf. eine implizite Konvertierung durchgeführt.
- Sie versorgen einen weiteren Operanden mit einer Variablen, deren Datentyp größer ist als der Datentyp der Anweisung:  
Der Datentyp der Anweisung wird auf den größeren Datentyp geändert. Für Operanden, die von dem neu eingestellten Datentyp der Anweisung abweichen, erfolgt ggf. eine implizite Konvertierung.

Jede Datentypänderung eines Operanden kann eine Änderung des Datentyps der Anweisung nach sich ziehen. Dadurch werden möglicherweise auch weitere Operanden implizit konvertiert. Operanden, für die eine implizite Konvertierung durchgeführt wird, werden mit einem grauen Quadrat gekennzeichnet.

---

### Hinweis

Beachten Sie auch die Informationen zur Datentypkonvertierung für Ihr Gerät und hier insbesondere die Hinweise zur IEC-Prüfung.

Siehe auch: Datentypkonvertierung

---

## Siehe auch

Datentyp einer Anweisung festlegen (Seite 1337)

## Datentyp einer Anweisung festlegen

### Einführung

Manche Anweisungen können mit mehreren verschiedenen Datentypen ausgeführt werden. Wenn Sie solche Anweisungen in Ihr Programm einfügen, müssen Sie für diese Anweisungen an der konkreten Programmstelle den Datentyp festlegen.

### Datentyp über die Klappliste festlegen

Um den Datentyp einer Anweisung über die Klappliste festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.  
In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Klicken Sie auf das Dreieck in der oberen Ecke der Klappliste.  
Die Klappliste wird geöffnet und die Datentypen angezeigt, die für die Anweisung zulässig sind.
3. Wählen Sie aus der Klappliste einen Datentyp aus.  
Der ausgewählte Datentyp wird angezeigt.
4. Falls die Anweisung über zwei Klapplisten verfügt, wählen Sie in der linken Klappliste den Datentyp für die Eingänge der Anweisung und in der rechten Klappliste den Datentyp für die Ausgänge der Anweisung.

### Datentyp durch Zuweisen von Variablen festlegen

Um den Datentyp einer Anweisung durch Zuweisen von Variablen festzulegen, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.  
In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Geben Sie an einem Eingang oder Ausgang eine gültige Variable an, deren Datentyp als Datentyp der Anweisung übernommen werden soll.  
Der Datentyp der Variablen wird in der Klappliste angezeigt.
3. Falls sowohl für die Eingänge als auch für die Ausgänge der Anweisung Datentypen festgelegt werden müssen, geben Sie an einem Eingang und an einem Ausgang jeweils eine gültige Variable an. Die am Eingang angegebene Variable bestimmt den Datentyp der Eingänge und die Variable am Ausgang bestimmt den Datentyp der Ausgänge der Anweisung.

### Datentyp von mathematischen Anweisungen automatisch festlegen

Um für mathematische Anweisungen den Datentyp automatisch festlegen zu lassen, gehen Sie folgendermaßen vor:

1. Fügen Sie die mathematische Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.  
In der Klappliste der eingefügten Anweisung wird der Eintrag "???" (undefiniert) angezeigt.
2. Wählen Sie in der Klappliste den Eintrag "Auto".
3. Geben Sie an einem Eingang oder Ausgang eine gültige Variable an.  
Der Datentyp der Variablen wird als Datentyp der Anweisung übernommen. Der Eintrag in der Klappliste ändert sich zu "Auto (<Datentyp>)".

Siehe auch: Auswahl eines Datentyps (Seite 1336)

### Siehe auch

Auswahl eines Datentyps (Seite 1336)



## Favoriten in FUP verwenden

### FUP-Elemente zu den Favoriten hinzufügen

#### Voraussetzung

- Ein Baustein ist geöffnet.
- Für die Task Card "Anweisungen" ist der Mehrpalettenmodus eingestellt oder die Favoriten werden auch im Editor angezeigt.

#### Vorgehen

Um Anweisungen zu den Favoriten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Maximieren Sie die Palette "Einfache Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu der Anweisung, die Sie den Favoriten hinzufügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop entweder in die Palette "Favoriten" oder in den Favoritenbereich im Programmiereditor.

---

#### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---

#### Siehe auch

FUP-Elemente aus den Favoriten entfernen (Seite 1340)

Übersicht über den Programmiereditor (Seite 1235)

### FUP-Elemente mithilfe von Favoriten einfügen

#### Voraussetzung

- Ein Baustein ist geöffnet.
- Favoriten sind vorhanden.

## Vorgehen

Um eine Anweisung mithilfe der Favoriten in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die gewünschte Anweisung per Drag & Drop aus den Favoriten an die gewünschte Position.

Oder:

1. Markieren Sie die Programmstelle, an der Sie die Anweisung einfügen möchten.
2. Klicken Sie auf die einzufügende Anweisung in den Favoriten.

---

### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---

## Siehe auch

Übersicht über den Programmiereditor (Seite 1235)

FUP-Elemente aus den Favoriten entfernen (Seite 1340)

## FUP-Elemente aus den Favoriten entfernen

### Voraussetzung

Ein Codebaustein ist geöffnet.

### Vorgehen

Um Anweisungen aus den Favoriten zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Anweisung, die Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisung entfernen".

---

### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---

## Siehe auch

- FUP-Elemente zu den Favoriten hinzufügen (Seite 1339)
- FUP-Elemente mithilfe von Favoriten einfügen (Seite 1339)
- Übersicht über den Programmiereditor (Seite 1235)

## Bausteinaufrufe in FUP einfügen

### Bausteinaufrufe per Drag & Drop einfügen

Sie können Aufrufe für bestehende Funktionen (FC) und Funktionsbausteine (FB) per Drag & Drop aus der Projektnavigation einfügen. Wenn Sie Funktionsbausteine aus anderen Funktionsbausteinen aufrufen, können Sie sie entweder als Einzelinstanz oder als Multiinstanz aufrufen. Wird ein Funktionsbaustein als Einzelinstanz aufgerufen, speichert er seine Daten in einem eigenen Datenbaustein. Wird ein Funktionsbaustein als Multiinstanz aufgerufen, speichert er seine Daten im Instanz-Datenbaustein des aufrufenden Funktionsbausteins.

### Voraussetzung

- Ein Netzwerk ist vorhanden.
- Der Baustein, der aufgerufen werden soll, ist vorhanden.

### Aufruf einer Funktion (FC) einfügen

Um den Aufruf einer Funktion (FC) per Drag & Drop in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die Funktion aus der Projektnavigation in das gewünschte Netzwerk.

### Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Funktionsbaustein aus der Projektnavigation in das gewünschte Netzwerk. Der Dialog "Aufrufoptionen" wird geöffnet.
2. Geben Sie im Dialog an, ob Sie den Baustein als Einzel- oder Multiinstanz aufrufen wollen.
  - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
  - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
3. Bestätigen Sie Ihre Eingaben mit "OK".

## Ergebnis

Die Funktion bzw. der Funktionsbaustein wird mit seinen Parametern eingefügt. Diese Parameter können Sie anschließend versorgen.

Siehe auch: Auto-Hotspot

---

### Hinweis

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt. Wenn Sie einen Funktionsbaustein als Multiinstanz aufgerufen haben, wird dieser als statische Variable in die Schnittstelle eingetragen.

---

## Siehe auch

Bausteinaufrufe in FUP aktualisieren (Seite 1342)

Instanztyp ändern (Seite 1343)

Einzelinstanzen (Seite 1041)

Multiinstanzen (Seite 1042)

## Bausteinaufrufe in FUP aktualisieren

Wenn sich Schnittstellenparameter eines aufgerufenen Bausteins ändern, kann der Bausteinaufruf nicht korrekt ausgeführt werden. Solche inkonsistenten Bausteinaufrufe vermeiden Sie, indem Sie die Bausteinaufrufe aktualisieren.

Sie haben zwei Möglichkeiten, Bausteinaufrufe zu aktualisieren:

- Explizites Aktualisieren im Programmiereditor.  
Die Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert.
- Implizites Aktualisieren während des Übersetzens.  
Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen werden aktualisiert.

## Bausteinaufrufe im Programmiereditor aktualisieren

Um in einem Baustein einen Bausteinaufruf zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Inkonsistente Bausteinaufrufe aktualisieren".

Oder:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf die Anweisung mit dem Bausteinaufruf.

3. Wählen Sie im Kontextmenü den Befehl "Bausteinanruf aktualisieren".  
Der Dialog "Schnittstellenabgleich" wird geöffnet. In diesem Dialog werden Ihnen die Unterschiede zwischen der verwendeten und der geänderten Bausteinschnittstelle des aufgerufenen Bausteins angezeigt.
4. Wenn Sie den Bausteinanruf aktualisieren möchten, klicken Sie auf "OK". Um die Aktualisierung abzubrechen, klicken Sie auf "Abbrechen".

### Bausteinanrufe während des Übersetzens aktualisieren

Um alle Bausteinanrufe und Verwendungen von PLC-Datentypen während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

### Siehe auch

Bausteinanrufe per Drag & Drop einfügen (Seite 1341)

Instanztyp ändern (Seite 1343)

### Instanztyp ändern

#### Instanztyp

Funktionsbausteine können Sie auf zwei verschiedene Arten aufrufen:

- Als Einzelinstanz
- Als Multiinstanz

Siehe auch: Auto-Hotspot

Sie können einen einmal definierten Instanztyp jederzeit ändern.

### Voraussetzung

Das Anwenderprogramm enthält einen Bausteinanruf.

### Vorgehen

Um den Instanztyp eines Funktionsbausteins zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Codebaustein und selektieren Sie den Bausteinanruf.
2. Wählen Sie im Kontextmenü den Befehl "Instanz ändern".  
Der Dialog "Aufrufoptionen" wird geöffnet.

3. Klicken Sie auf die Schaltfläche "Einzel-Instanz" oder "Multi-Instanz"
  - Wenn Sie als Instanztyp "Einzel-Instanz" wählen, geben Sie einen Namen für den Datenbaustein ein, der dem Funktionsbaustein zugeordnet werden soll.
  - Wenn Sie als Instanztyp "Multi-Instanz" wählen, geben Sie im Textfeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen werden soll.
4. Bestätigen Sie Ihre Eingaben mit "OK".

---

**Hinweis**

Die vorherigen Einzel- und Multiinstanzen werden nicht automatisch gelöscht.

---

**Siehe auch**

Bausteinaufrufe per Drag & Drop einfügen (Seite 1341)

Bausteinaufrufe in FUP aktualisieren (Seite 1342)

**Komplexe FUP-Anweisungen einfügen**

**Anweisung "Berechnen" verwenden**

**Voraussetzung**

Ein Netzwerk ist vorhanden.

**Vorgehen**

Um die Anweisung "Berechnen" zu verwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Mathematische Funktionen > CALCULATE".
3. Ziehen Sie das Element per Drag & Drop an die gewünschte Stelle im Netzwerk. Die Anweisung "Berechnen" wird mit einem Platzhalteraussdruck und Fragezeichen für den Datentyp eingefügt.
4. Geben Sie den Datentyp für die Berechnung ein.
5. Geben Sie die Operanden für die Berechnung ein.

---

**Hinweis**

Die Berechnung wird mit den Eingängen der Anweisung "Berechnen" durchgeführt. Wenn Sie Konstanten verwenden möchten, müssen Sie daher auch für die Konstanten entsprechende Eingänge einfügen.

---

6. Klicken Sie auf die Schaltfläche "Anweisung 'Berechnen' bearbeiten", um den Platzhalterausdruck durch den korrekten Ausdruck zu ersetzen.  
Der Dialog "Anweisung 'Berechnen' bearbeiten" wird geöffnet.
7. Geben Sie in das Textfeld "OUT:= " den gewünschten Ausdruck ein.

---

**Hinweis**

Im Bereich "Beispiel" finden Sie ein Beispiel für einen gültigen Ausdruck und mögliche Anweisungen, die Sie verwenden können.

Um z. B. mithilfe des Pythagoras einen Wert zu ermitteln, geben Sie "OUT := SQRT (SQR (IN1) + SQR (IN2))" ein.

---

8. Bestätigen Sie Ihre Eingabe mit "OK".

**Siehe auch**

CALCULATE: Berechnen (Seite 1988)

**Freie Kommentare verwenden****Grundlagen zur Verwendung von freien Kommentaren in FUP****Einführung**

Freie Kommentare ermöglichen das Kommentieren des Quellcodes für grafische Programmiersprachen ähnlich wie Zeilenkommentare für textuelle Sprachen.

Freie Kommentare können für alle nicht-binären Boxen verwendet werden.

**Siehe auch**

Freie Kommentare einfügen (Seite 1345)

Freie Kommentare bearbeiten (Seite 1346)

Freie Kommentare löschen (Seite 1347)

**Freie Kommentare einfügen****Voraussetzung**

Ein Netzwerk mit Anweisungen ist vorhanden.

## Vorgehen

Um einen freien Kommentar zu einer Anweisung hinzuzufügen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie ggf. in der Funktionsleiste die Schaltfläche "Freie Kommentare ein/aus".
2. Klicken Sie mit der rechten Maustaste auf die Anweisung, für die Sie einen freien Kommentar einfügen möchten.
3. Wählen Sie den Befehl "Kommentar einfügen" aus dem Kontextmenü.  
Eine Kommentarbox mit einem Standardkommentar wird geöffnet. Die Kommentarbox ist über einen Pfeil mit der zugehörigen Anweisung verbunden.
4. Geben Sie in die Kommentarbox den gewünschten Kommentar ein.

## Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in FUP (Seite 1345)

Freie Kommentare bearbeiten (Seite 1346)

Freie Kommentare löschen (Seite 1347)

## Freie Kommentare bearbeiten

### Einführung

Freie Kommentare können folgendermaßen bearbeitet werden:

- Kommentartext ändern
- Platzierung oder Größe der Kommentarbox ändern
- Kommentar an ein anderes Element anhängen
- Freie Kommentare ein- und ausblenden

### Kommentartext ändern

Um den Text von freien Kommentaren zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in die Kommentarbox.
2. Geben Sie den gewünschten Text ein.

### Platzierung der Kommentarbox ändern

Um die Platzierung der Kommentarbox zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Kommentarbox und halten Sie die Maustaste gedrückt.
2. Ziehen Sie die Kommentarbox an die gewünschte Stelle.



### Größe der Kommentarbox ändern

Um die Größe der Kommentarbox zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Kommentarbox.
2. Ziehen Sie die Kommentarbox am Anfasser in der unteren rechten Ecke auf die gewünschte Größe.

### Kommentar an ein anderes Element anhängen

Um einen freien Kommentar an ein anderes Element zu hängen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Pfeilspitze, die die Kommentarbox mit der Anweisung verbindet, und halten Sie die Maustaste gedrückt.
2. Ziehen Sie den Pfeil auf das Element, an das Sie den Kommentar anhängen möchten. Mögliche Einfügestellen werden mit einem grünen Quadrat gekennzeichnet.
3. Lassen Sie die Maustaste los.

### Freie Kommentare ein- und ausblenden

Um freie Kommentare ein- und auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Freie Kommentare ein/aus".

### Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in FUP (Seite 1345)

Freie Kommentare einfügen (Seite 1345)

Freie Kommentare löschen (Seite 1347)

### Freie Kommentare löschen

#### Vorgehen

Um einen freien Kommentar zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf den freien Kommentar, den Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Siehe auch

Grundlagen zur Verwendung von freien Kommentaren in FUP (Seite 1345)

Freie Kommentare einfügen (Seite 1345)

Freie Kommentare bearbeiten (Seite 1346)

## FUP-Elemente bearbeiten

### FUP-Elemente selektieren

Sie können mehrere einzelne Elemente oder alle Elemente eines Netzwerks selektieren.

#### Voraussetzung

FUP-Elemente sind vorhanden.

### Mehrere einzelne FUP-Elemente selektieren

Um mehrere einzelne FUP-Elemente zu selektieren, gehen Sie folgendermaßen vor:

1. Drücken und halten Sie die Taste <Strg>.
2. Klicken Sie die FUP-Elemente an, die Sie selektieren möchten.
3. Lassen Sie die Taste <Strg> los.

### Alle FUP-Elemente eines Netzwerks selektieren

Um alle FUP-Elemente eines Netzwerks zu selektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in das Netzwerk, dessen Elemente Sie selektieren möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Alles auswählen" oder verwenden Sie die Tastenkombination <Strg+A>.

#### Siehe auch

FUP-Elemente kopieren (Seite 1348)

FUP-Elemente ausschneiden (Seite 1349)

FUP-Elemente aus der Zwischenablage einfügen (Seite 1350)

FUP-Elemente ersetzen (Seite 1350)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1353)

FUP-Elemente löschen (Seite 1355)

## FUP-Elemente kopieren

#### Voraussetzung

Ein FUP-Element ist vorhanden.

## Vorgehen

Um ein FUP-Element zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das FUP-Element, das Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

## Ergebnis

Das FUP-Element wird kopiert und in der Zwischenablage gespeichert.

## Siehe auch

FUP-Elemente selektieren (Seite 1348)

FUP-Elemente ausschneiden (Seite 1349)

FUP-Elemente aus der Zwischenablage einfügen (Seite 1350)

FUP-Elemente ersetzen (Seite 1350)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1353)

FUP-Elemente löschen (Seite 1355)

## FUP-Elemente ausschneiden

### Voraussetzung

Ein FUP-Element ist vorhanden.

### Ausschneiden

Um ein FUP-Element auszuschneiden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das FUP-Element, das Sie ausschneiden möchten.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

## Ergebnis

Das FUP-Element wird ausgeschnitten und in der Zwischenablage gespeichert.

### Siehe auch

- FUP-Elemente selektieren (Seite 1348)
- FUP-Elemente kopieren (Seite 1348)
- FUP-Elemente aus der Zwischenablage einfügen (Seite 1350)
- FUP-Elemente ersetzen (Seite 1350)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)
- Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)
- EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1353)
- FUP-Elemente löschen (Seite 1355)

### FUP-Elemente aus der Zwischenablage einfügen

#### Voraussetzung

Ein FUP-Element ist vorhanden.

#### Vorgehen

Um ein FUP-Element aus der Zwischenablage einzufügen, gehen Sie folgendermaßen vor:

1. Kopieren Sie ein FUP-Element oder schneiden Sie ein FUP-Element aus.
2. Klicken Sie mit der rechten Maustaste auf die Stelle im Netzwerk, an der Sie das Element einfügen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

### Siehe auch

- FUP-Elemente selektieren (Seite 1348)
- FUP-Elemente kopieren (Seite 1348)
- FUP-Elemente ausschneiden (Seite 1349)
- FUP-Elemente ersetzen (Seite 1350)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)
- Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)
- EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1353)
- FUP-Elemente löschen (Seite 1355)

### FUP-Elemente ersetzen

Sie können FUP-Elemente ganz einfach gegen andere FUP-Elemente des gleichen Typs austauschen. Dies hat den Vorteil, dass die Parameter bestehen bleiben und nicht erneut

eingegeben werden müssen. Sie können z. B. ODER und UND, RS-FlipFlop und SR-FlipFlop, Vergleichsfunktionen oder Sprunganweisungen gegeneinander austauschen.

## Voraussetzung

Ein Netzwerk mit mindestens einem FUP-Element ist vorhanden.

## Vorgehen

Um ein FUP-Element durch ein anderes FUP-Element zu ersetzen, gehen Sie folgendermaßen vor:

1. Markieren Sie das FUP-Element, das Sie ersetzen möchten.  
Wenn zu dem gewählten FUP-Element kompatible Elemente vorhanden sind, erscheint in der oberen rechten Ecke des Elements ein Dreieck.
2. Bewegen Sie den Mauszeiger über das Dreieck des FUP-Elements.  
Eine Klappliste wird geöffnet.
3. Wählen Sie das FUP-Element aus der Klappliste, durch das Sie das aktuelle FUP-Element ersetzen möchten.

## Siehe auch

FUP-Elemente selektieren (Seite 1348)

FUP-Elemente kopieren (Seite 1348)

FUP-Elemente ausschneiden (Seite 1349)

FUP-Elemente aus der Zwischenablage einfügen (Seite 1350)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1353)

FUP-Elemente löschen (Seite 1355)

## Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen

### Einführung

Sie können einige FUP-Elemente, die arithmetische oder binäre Operationen durchführen, durch zusätzliche Eingänge erweitern. Solche Elemente sind z. B. die Anweisungen "Addieren" (ADD), "Multiplizieren" (MUL), UND oder ODER. Durch zusätzliche Ausgänge können Sie die Boxen der Anweisungen "Wert kopieren" (MOVE) und "Demultiplexen" (DEMUX) erweitern.

Der Name der neuen Ein- und Ausgänge setzt sich aus der Art des eingefügten Elements und einer fortlaufenden Nummer zusammen. Der Name eines neuen Eingangs lautet z. B. "IN2", der Name eines neuen Ausgangs z. B. "OUT2".

## Voraussetzung

Ein FUP-Element ist vorhanden, das das Einfügen von zusätzlichen Eingängen bzw. Ausgängen erlaubt.

## Zusätzlichen Eingang einfügen

Um die Box eines FUP-Elements durch einen zusätzlichen Eingang zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Eingang des FUP-Elements.
2. Wählen Sie im Kontextmenü den Befehl "Eingang einfügen".  
Die Box des FUP-Elements wird durch einen zusätzlichen Eingang erweitert.

Oder:

1. Klicken Sie auf das gelbe Stern-Symbol neben dem letzten Eingang der Anweisungsbox.  
Die Box des FUP-Elements wird durch einen zusätzlichen Eingang erweitert.

## Zusätzlichen Ausgang einfügen

Um die Box eines FUP-Elements durch einen zusätzlichen Ausgang zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf einen vorhandenen Ausgang des FUP-Elements.
2. Wählen Sie im Kontextmenü den Befehl "Ausgang einfügen".  
Die Box des FUP-Elements wird durch einen zusätzlichen Ausgang erweitert.

Oder:

1. Klicken Sie auf das gelbe Stern-Symbol neben dem letzten Ausgang der Anweisungsbox.  
Die Box des FUP-Elements wird durch einen zusätzlichen Ausgang erweitert.

## Siehe auch

FUP-Elemente selektieren (Seite 1348)

FUP-Elemente kopieren (Seite 1348)

FUP-Elemente ausschneiden (Seite 1349)

FUP-Elemente aus der Zwischenablage einfügen (Seite 1350)

FUP-Elemente ersetzen (Seite 1350)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1353)

FUP-Elemente löschen (Seite 1355)

## Eingänge und Ausgänge einer Anweisung entfernen

### Einführung

Eingänge oder Ausgänge, die Sie für eine Anweisung zusätzlich eingefügt haben, können wieder entfernt werden.

### Voraussetzung

Ein FUP-Element ist vorhanden, das Sie durch zusätzliche Ein- oder Ausgänge erweitert haben.

### Eingang entfernen

Um einen Eingang zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Eingang, den Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".  
Der Eingang des FUP-Elements wird entfernt.

### Ausgang entfernen

Um einen Ausgang zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ausgang, den Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".  
Der Ausgang des FUP-Elements wird entfernt.

### Siehe auch

FUP-Elemente selektieren (Seite 1348)

FUP-Elemente kopieren (Seite 1348)

FUP-Elemente ausschneiden (Seite 1349)

FUP-Elemente aus der Zwischenablage einfügen (Seite 1350)

FUP-Elemente ersetzen (Seite 1350)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1353)

FUP-Elemente löschen (Seite 1355)

### EN-/ENO-Mechanismus aktivieren und deaktivieren

In KOP und FUP haben bestimmte Anweisungen einen Freigabeausgang ENO (enable output) und nutzen somit den EN-/ENO-Mechanismus. Damit können Sie Laufzeitfehler in Anweisungen abfragen und darauf reagieren. Um die Performance der CPU zu steigern, ist

der EN-/ENO-Mechanismus in der Voreinstellung deaktiviert. Dadurch haben Sie zunächst nicht mehr die Möglichkeit auf Laufzeitfehler der Anweisung über den ENO-Wert zu reagieren. Sie können den EN-/ENO-Mechanismus jedoch bei Bedarf jederzeit wieder aktivieren.

Sie können für jede Anweisung den EN-/ENO-Mechanismus einzeln aktivieren, um den ENO zu generieren. Wenn Sie für eine Anweisung den EN-/ENO-Mechanismus aktivieren, werden weitere Anweisungen, die Sie danach in Ihr Programm ziehen, ebenfalls mit aktiviertem EN-/ENO-Mechanismus eingefügt. Falls Sie die Auswertung des ENO für eine Anweisung nicht nutzen möchten, können Sie den EN-/ENO-Mechanismus jederzeit wieder deaktivieren. Weitere Anweisungen, die Sie danach in Ihr Programm ziehen, werden dann ohne EN-/ENO-Mechanismus eingefügt.

Siehe auch: Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### **EN-/ENO-Mechanismus aktivieren**

Um den EN-/ENO-Mechanismus einer Anweisung zu aktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus aktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO generieren".  
Für die Anweisung wird der ENO-Wert wieder generiert. Weitere Anweisungen werden mit dem Freigabeausgang eingefügt.

### **EN-/ENO-Mechanismus deaktivieren**

Um den EN-/ENO-Mechanismus einer Anweisung zu deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in Ihrem Programm mit der rechten Maustaste auf die Anweisung, für die Sie den EN-/ENO-Mechanismus deaktivieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "ENO nicht generieren".  
Für die Anweisung wird der ENO-Wert nicht mehr generiert. Weitere Anweisungen werden ohne den Freigabeausgang eingefügt.

### **Siehe auch**

FUP-Elemente selektieren (Seite 1348)

FUP-Elemente kopieren (Seite 1348)

FUP-Elemente ausschneiden (Seite 1349)

FUP-Elemente aus der Zwischenablage einfügen (Seite 1350)

FUP-Elemente ersetzen (Seite 1350)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

FUP-Elemente löschen (Seite 1355)



## FUP-Elemente löschen

### Voraussetzung

Ein FUP-Element ist vorhanden.

### Vorgehen

Um ein FUP-Element zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das FUP-Element, das Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Siehe auch

FUP-Elemente selektieren (Seite 1348)

FUP-Elemente kopieren (Seite 1348)

FUP-Elemente ausschneiden (Seite 1349)

FUP-Elemente aus der Zwischenablage einfügen (Seite 1350)

FUP-Elemente ersetzen (Seite 1350)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

EN-/ENO-Mechanismus aktivieren und deaktivieren (Seite 1353)

## Operanden in FUP-Anweisungen beschalten

### Operanden beschalten

Beim Einfügen eines FUP-Elements werden die Zeichenfolgen "<???", "<??.>" und "..." als Platzhalter für die Parameter eingefügt. Die rot dargestellten Zeichenfolgen "<???", "<??.>" kennzeichnen Parameter, die beschaltet werden müssen. Die schwarz dargestellte Zeichenfolge "..." kennzeichnet Parameter, die beschaltet werden können. "<??.>" steht für boolesche Platzhalter.

---

#### Hinweis

Wenn Sie den Mauszeiger über den Platzhalter bewegen, werden die möglichen Datentypen in einem Tooltip angezeigt.

---

### Voraussetzung

Ein FUP-Element ist vorhanden.

## Vorgehen

Um die Parameter eines FUP-Elements zu beschalten, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Platzhalter des Parameters.  
Ein Eingabefeld wird geöffnet.
2. Geben Sie den entsprechenden Parameter ein, z. B. eine PLC-Variable, eine lokale Variable oder eine Konstante.

---

### Hinweis

Wenn Sie die absolute Adresse eines Parameters eingeben, der bereits definiert ist, wird diese Absolutadresse in den symbolischen Namen des Parameters geändert, sobald Sie die Eingabe bestätigen. Haben Sie den Parameter noch nicht definiert, wird in der PLC-Variablen-tabelle eine neue Variable mit dieser Absolutadresse und dem Standardnamen "Tag\_1" eingetragen. Bestätigen Sie Ihre Eingabe, wird die Absolutadresse gegen den symbolischen Namen "Tag\_1" ausgetauscht.

---

3. Bestätigen Sie den Parameter mit der Eingabetaste.
4. Wenn Sie den Parameter bisher nicht definiert haben, können Sie dies über das Kontextmenü direkt im Programmiereditor erledigen.  
Siehe auch: "PLC-Variablen im Programmiereditor deklarieren (Seite 1195)".

Oder per Drag & Drop aus der PLC-Variablen-tabelle:

1. Selektieren Sie in der Projektnavigation den Ordner "PLC-Variablen" und öffnen Sie die PLC-Variablen-tabelle.
2. Wenn Sie die PLC-Variablen-tabelle geöffnet haben, ziehen Sie die gewünschte Variable an die entsprechende Stelle in Ihrem Programm. Falls Sie die PLC-Variablen-tabelle nicht geöffnet haben, öffnen Sie die Detailansicht. Ziehen Sie die gewünschte Variable aus der Detailansicht an die entsprechende Stelle in Ihrem Programm.

Oder per Drag & Drop aus der Bausteinschnittstelle:

1. Öffnen Sie die Bausteinschnittstelle.
2. Ziehen Sie den gewünschten Operanden aus der Bausteinschnittstelle an die entsprechende Stelle in Ihrem Programm.

## Ergebnis

- Wenn die Syntax fehlerfrei ist, wird der Parameter schwarz dargestellt.
- Wenn in der Syntax ein Fehler vorhanden ist, wird das Eingabefeld nicht verlassen und es wird im Inspektorfenster im Register "Info > Syntax" eine entsprechende Fehlermeldung angezeigt.

## Versteckte Parameter beschalten

### Einführung

Abhängig von der verwendeten CPU können Sie in Ihrem Programm komplexe Anweisungen verwenden, die mit dem TIA-Portal ausgeliefert werden. Diese Anweisungen können Parameter enthalten, die als versteckt deklariert wurden.

Wenn eine Anweisung versteckte Parameter enthält, verfügt die Anweisungsbox über einen kleinen Pfeil am unteren Rand. Die versteckten Parameter erkennen Sie an der weißen Schriftfarbe.

Sie können die versteckten Parameter jederzeit einblenden und beschalten.

### Versteckte Parameter ein- oder ausblenden

Um versteckte Parameter ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Pfeil-nach-unten am unteren Rand der Anweisungsbox, um versteckte Parameter einzublenden.
2. Klicken Sie auf den Pfeil-nach-oben am unteren Rand der Anweisungsbox, um versteckte Parameter auszublenden.

### Versteckte Parameter beschalten

Um versteckte Parameter zu beschalten, gehen Sie folgendermaßen vor:

1. Beschalten Sie die versteckten Parameter wie normale sichtbare Parameter. Der versteckte Parameter wird zu einem sichtbaren Parameter.

### Siehe auch

Bibliotheken verwenden (Seite 313)

### Variableninformationen ein- oder ausblenden

#### Einführung

Sie können die folgenden Informationen zu den verwendeten Variablen im Programmiereditor einblenden:

- Name der Variablen
- Adresse der Variablen
- Kommentar zur Dokumentation der Variablen

Die Informationen werden bei lokalen Variablen und DB-Variablen aus der Bausteinschnittstelle und bei CPU-weit gültigen Variablen aus der PLC-Variablen-tabelle entnommen.

Sie können die Variableninformationen entweder für alle Bausteine oder für einzelne geöffnete Bausteine einblenden. Wenn Sie die Variableninformationen für alle Bausteine einblenden,

werden die Variableninformationen für alle aktuell und zukünftig geöffneten Bausteine eingeblendet.

Sie können die Variableninformationen jederzeit wieder ausblenden. Wenn Sie die Variableninformationen für alle Bausteine ausgeblendet haben, können Sie sie für einzelne geöffnete wieder einblenden.

### Variableninformationen für alle Bausteine ein- oder ausblenden

Um die Variableninformationen für alle Bausteine ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Wenn Sie die Variableninformationen einblenden möchten, aktivieren Sie im Bereich "Ansicht" das Optionskästchen "mit Variableninformationen". Wenn Sie die Variableninformationen ausblenden möchten, deaktivieren Sie das Optionskästchen "mit Variableninformationen".  
Die Variableninformationen werden für alle geöffneten Bausteine ein- oder ausgeblendet. Wenn Sie weitere Bausteine öffnen, werden die Variableninformationen abhängig von der gewählten Einstellung angezeigt oder nicht angezeigt.

### Variableninformationen für einen geöffneten Baustein ein- oder ausblenden

Um die Variableninformationen für einen geöffneten Baustein ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Aktivieren oder deaktivieren Sie im Menü "Ansicht > Anzeigen mit" das Optionskästchen "Variableninformationen" oder klicken Sie in der Funktionsleiste auf die Schaltfläche "Variableninformationen ein/aus".  
Die Variableninformationen werden ein- oder ausgeblendet.

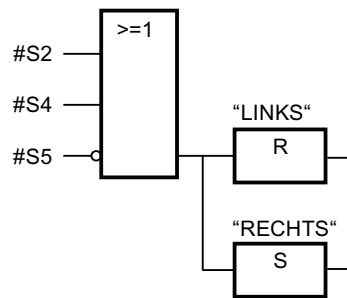
## Zweige in FUP

### Grundlagen zu Zweigen in FUP

#### Definition

Sie können mit der Programmiersprache Funktionsplan (FUP) Parallelzweige programmieren. Dazu verwenden Sie Zweige, die Sie zwischen den Boxen einfügen. Sie können in den Zweig weitere Boxen einfügen und auf diese Weise komplexe Funktionspläne programmieren.

Das folgende Bild zeigt ein Beispiel für die Verwendung von Zweigen:



## Siehe auch

Regeln für Zweige in FUP (Seite 1359)

Zweige in FUP-Netzwerke einfügen (Seite 1359)

Zweige in FUP-Netzwerke löschen (Seite 1360)

## Regeln für Zweige in FUP

### Regeln

Für die Verwendung von Zweigen in FUP gelten folgende Regeln:

- Zweige werden nach unten geöffnet.
- Zweige können nur zwischen zwei FUP-Elementen eingefügt werden.
- Um einen Zweig zu löschen, müssen Sie alle FUP-Elemente löschen, einschließlich des Zweigs selbst.
- Wenn Sie die Verbindung zwischen zwei Zweigen löschen, werden die FUP-Elemente des aufgebrochenen Zweiges frei im Netzwerk platziert.

## Siehe auch

Grundlagen zu Zweigen in FUP (Seite 1358)

Zweige in FUP-Netzwerke einfügen (Seite 1359)

Zweige in FUP-Netzwerke löschen (Seite 1360)

## Zweige in FUP-Netzwerke einfügen

### Voraussetzung

Ein Netzwerk ist vorhanden.

## Vorgehen

Um einen neuen Zweig in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Navigieren Sie in der Palette "Einfache Anweisungen" zu "Allgemein > Abzweig".
3. Ziehen Sie das Element per Drag & Drop an der gewünschten Stelle auf eine Verbindungslinie zwischen zwei Boxen.

## Siehe auch

Regeln für Zweige in FUP (Seite 1359)

Grundlagen zu Zweigen in FUP (Seite 1358)

Zweige in FUP-Netzwerke löschen (Seite 1360)

## Zweige in FUP-Netzwerke löschen

### Voraussetzung

Ein Zweig ist vorhanden.

### Vorgehen

Um einen Zweig zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Verbindungslinie, die den Zweig mit dem Hauptzweig verbindet.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Ergebnis

Der Zweig wird gelöscht. Am gelöschten Zweig angeschlossene Boxen werden frei im Netzwerk platziert.

## Siehe auch

Regeln für Zweige in FUP (Seite 1359)

Grundlagen zu Zweigen in FUP (Seite 1358)

Zweige in FUP-Netzwerke einfügen (Seite 1359)

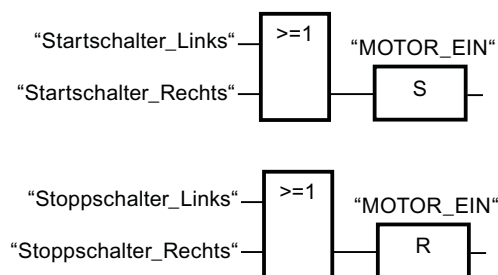
## Verknüpfungspfade in FUP

### Grundlagen zu Verknüpfungspfaden in FUP

#### Verwendung von Verknüpfungspfaden

Das Anwenderprogramm wird in einem oder mehreren Netzwerken abgebildet. Innerhalb der Netzwerke können sich ein oder mehrere Verknüpfungspfade befinden, auf denen die binären Signale in Form von Boxen angeordnet sind.

Das folgende Bild zeigt ein Beispiel für die Verwendung von mehreren Verknüpfungspfaden innerhalb eines Netzwerks:



#### Regeln

Beachten Sie folgende Regeln bei der Verwendung von Verknüpfungspfaden:

- Verbindungen zwischen Verknüpfungspfaden sind nicht erlaubt.
- Es ist nur eine Sprunganweisung pro Netzwerk zulässig. Die Platzierungsregeln für Sprunganweisungen bleiben dabei erhalten.

#### Abarbeitung von Verknüpfungspfaden

Die Abarbeitung von Verknüpfungspfaden und Netzwerken erfolgt von oben nach unten und von links nach rechts. Das bedeutet, dass zuerst die erste Anweisung im ersten Verknüpfungspfad im ersten Netzwerk bearbeitet wird. Anschließend werden alle Anweisungen dieses Verknüpfungspfads bearbeitet. Danach kommen alle weiteren Verknüpfungspfade des ersten Netzwerks an die Reihe. Erst wenn alle Verknüpfungspfade abgearbeitet wurden, wird das nächste Netzwerk bearbeitet.

Bei der Verwendung von Sprüngen wird die reguläre Abarbeitung der Verknüpfungspfade umgangen und es wird die Anweisung am Sprungziel bearbeitet.

#### Unterschiede zwischen Zweigen und Verknüpfungspfaden

Der Unterschied zwischen Zweigen und Verknüpfungspfaden besteht darin, dass die Verknüpfungspfade eigenständige Zweige sind, die auch in einem anderen Netzwerk stehen können. Zweige hingegen erlauben die Programmierung einer Parallelschaltung und haben eine gemeinsame Vorverknüpfung.

## Siehe auch

Verknüpfungspfad einfügen (Seite 1362)

Verknüpfungspfad löschen (Seite 1362)

## Verknüpfungspfad einfügen

### Voraussetzung

- Ein Baustein ist geöffnet.
- Ein Netzwerk ist vorhanden.

### Vorgehen

Um einen neuen Verknüpfungspfad in ein Netzwerk einzufügen, gehen Sie folgendermaßen vor:

1. Fügen Sie eine beliebige Anweisung so in ein Netzwerk ein, dass sie keine Verbindung zu vorhandenen Anweisungen besitzt.  
Ein neuer Verknüpfungspfad wird eingefügt.
2. Fügen Sie eine Zuweisung am Ende des neuen Verknüpfungspfads ein.
3. Fügen Sie weitere Anweisungen in den neuen Verknüpfungspfad ein.

## Siehe auch

Grundlagen zu Verknüpfungspfaden in FUP (Seite 1361)

Verknüpfungspfad löschen (Seite 1362)

## Verknüpfungspfad löschen

### Voraussetzung

Ein Verknüpfungspfad ist vorhanden.

### Vorgehen

Um einen Verknüpfungspfad zu löschen, gehen Sie folgendermaßen vor:

1. Ziehen Sie mit gedrückter linker Maustaste einen Rahmen um den Verknüpfungspfad.  
Achten Sie darauf, dass Sie dabei alle Anweisungen des Verknüpfungspfads selektieren.  
Alternativ können Sie auch mit gedrückter <Shift>-Taste die erste und letzte Anweisung des Verknüpfungspfads selektieren.
2. Klicken Sie mit der rechten Maustaste auf eine der Anweisungen im Verknüpfungspfad.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".



**Siehe auch**

Grundlagen zu Verknüpfungspfaden in FUP (Seite 1361)

Verknüpfungspfad einfügen (Seite 1362)

**SCL-Programme erstellen****Grundlagen zu SCL****Programmiersprache SCL****Programmiersprache SCL**

SCL (Structured Control Language) ist eine höhere Programmiersprache, die sich an PASCAL orientiert. Die Sprache basiert auf DIN EN-61131-3 (international IEC 1131-3).

Die Norm standardisiert die Programmiersprachen für speicherprogrammierbare Steuerungen. Die Programmiersprache SCL erfüllt den in dieser Norm definierten PLCopen Basis Level der Sprache ST (Strukturierter Text).

**Sprachelemente**

SCL enthält neben den typischen Elementen der SPS, wie Eingängen, Ausgängen, Zeiten, oder Merkern, auch Sprachelemente höherer Programmiersprachen:

- Ausdrücke
- Wertzuweisungen
- Operatoren

**Programmsteuerung**

SCL bietet komfortable Anweisungen zur Programmsteuerung, mit denen Sie z. B. Programmverzweigungen, -schleifen oder -sprünge realisieren können.

**Anwendungsbereich**

SCL ist damit für folgende Anwendungsbereiche besonders geeignet:

- Datenverwaltung
- Prozessoptimierung
- Rezepturverwaltung
- mathematische/statistische Aufgaben

## Ausdrücke

### Beschreibung

Ausdrücke werden zur Laufzeit des Programms berechnet und liefern einen Wert zurück. Ein Ausdruck besteht aus Operanden (z. B. aus Konstanten, Variablen oder Funktionsaufrufen) und optional aus Operatoren (z. B. \*, /, + oder -). Ausdrücke können durch Operatoren miteinander verknüpft oder ineinander geschachtelt sein.

### Auswertungsreihenfolge

Die Auswertung des Ausdrucks geschieht in einer bestimmten Reihenfolge, die durch folgende Faktoren festgelegt ist:

- Priorität der beteiligten Operatoren
- Links-Rechts-Reihenfolge
- Klammerung

### Arten von Ausdrücken

Abhängig vom Operator werden folgende Ausdrucksarten unterschieden:

- **Arithmetische Ausdrücke**  
Arithmetische Ausdrücke bestehen entweder aus einem numerischen Wert oder verknüpfen zwei Werte oder Ausdrücke mit arithmetischen Operatoren.
- **Vergleichsausdrücke**  
Vergleichsausdrücke vergleichen die Werte zweier Operanden und liefern einen booleschen Wert. Das Ergebnis ist TRUE, falls der Vergleich erfüllt ist, und FALSE, falls er nicht erfüllt ist.
- **Logische Ausdrücke**  
Logische Ausdrücke verknüpfen zwei Operanden durch logische Operatoren (AND, OR, XOR) oder negieren Operanden (NOT).

### Verwendung von Ausdrücken

Das Ergebnis eines Ausdrucks können Sie unterschiedlich verwenden:

- als Wertzuweisung zu einer Variablen
- als Bedingung für eine Kontrollanweisung
- als Parameter für den Aufruf eines Bausteins oder einer Anweisung

### Siehe auch

Operatoren und Operatorenrangfolge (Seite 1370)

Arithmetische Ausdrücke (Seite 1365)

Vergleichsausdrücke (Seite 1367)

Logische Ausdrücke (Seite 1369)

## Arithmetische Ausdrücke

### Beschreibung

Arithmetische Ausdrücke bestehen entweder aus einem numerischen Wert oder verknüpfen zwei Werte oder Ausdrücke mit arithmetischen Operatoren.

Arithmetische Operatoren können die Datentypen verarbeiten, die in der verwendeten CPU zulässig sind. Wenn an der Operation zwei Operanden beteiligt sind, dann wird der Datentyp des Ergebnisses nach folgenden Kriterien bestimmt:

- Wenn die beiden Operanden Ganzzahlen mit Vorzeichen sind und unterschiedliche Längen haben, erhält das Ergebnis den Datentyp der längeren Ganzzahl (z. B. INT + DINT = DINT).
- Wenn die beiden Operanden Ganzzahlen ohne Vorzeichen sind und unterschiedliche Längen haben, erhält das Ergebnis den Datentyp der längeren Ganzzahl (z. B. USINT + UDINT = UDINT).
- Wenn ein Operand eine Ganzzahl mit Vorzeichen und der andere Operand eine Ganzzahl ohne Vorzeichen ist, erhält das Ergebnis den nächst größeren Datentyp mit Vorzeichen, der die Ganzzahl ohne Vorzeichen abdeckt (z. B. SINT + USINT = INT).  
Eine Operation mit solchen Operanden lässt sich nur ausführen, wenn keine IEC-Prüfung eingestellt ist.
- Wenn ein Operand eine Ganzzahl und der andere Operand eine Gleitpunktzahl ist, erhält das Ergebnis den Datentyp der Gleitpunktzahl (z. B. INT + REAL = REAL).
- Wenn die beiden Operanden Gleitpunktzahlen sind und unterschiedliche Längen haben, erhält das Ergebnis den Datentyp der längeren Gleitpunktzahl (z. B. REAL + LREAL = LREAL).
- Den Datentyp des Ergebnisses einer Operation an der Operanden von den Datentypgruppen "Zeiten" und "Datum und Uhrzeit" beteiligt sind, können Sie der Tabelle im Abschnitt "Datentypen von arithmetischen Ausdrücken" entnehmen.  
Bei eingestellter IEC-Prüfung können Sie keine Datentypen aus den Datentypgruppen "Zeiten" und "Datum und Uhrzeit" verwenden.

### Datentypen von arithmetischen Ausdrücken

Die folgende Tabelle zeigt die Datentypen, die Sie in arithmetischen Ausdrücken verwenden können:

| Operation      | Operator | 1. Operand                                 | 2. Operand              | Ergebnis                                   |
|----------------|----------|--|-------------------------|--|
| Potenz         | **       | Ganzzahl/<br>Gleitpunktzahl                | Ganzzahl/Gleitpunktzahl | Ganzzahl/<br>Gleitpunktzahl                |
| Unäres Plus    | +        | Ganzzahl/<br>Gleitpunktzahl<br>TIME, LTIME | -                       | Ganzzahl/<br>Gleitpunktzahl<br>TIME, LTIME |
| Unäres Minus   | -        | Ganzzahl/<br>Gleitpunktzahl<br>TIME, LTIME | -                       | Ganzzahl/<br>Gleitpunktzahl<br>TIME, LTIME |
| Multiplikation | *        | Ganzzahl/<br>Gleitpunktzahl                | Ganzzahl/Gleitpunktzahl | Ganzzahl/<br>Gleitpunktzahl                |

| Operation  | Operator | 1. Operand                  | 2. Operand                                  | Ergebnis  |
|--|----------|-----------------------------|---|---|
|  |          | TIME, LTIME                 | Ganzzahl                                    | TIME, LTIME   |
| Division   | /        | Ganzzahl/<br>Gleitpunktzahl | Ganzzahl/<br>Gleitpunktzahl (ungleich<br>0) | Ganzzahl/<br>Gleitpunktzahl   |
|  |          | TIME, LTIME                 | Ganzzahl                                    | TIME, LTIME   |
| Modulo-<br>Funktion  | MOD      | Ganzzahl                    | Ganzzahl                                    | Ganzzahl  |
| Addition   | +        | Ganzzahl/<br>Gleitpunktzahl | Ganzzahl/Gleitpunktzahl                     | Ganzzahl/<br>Gleitpunktzahl   |
|  |          | TIME                        | TIME  | TIME  |
|  |          | TIME                        | DINT  | TIME  |
|  |          | LTIME                       | TIME, LTIME                                 | LTIME   |
|  |          | LTIME                       | LINT  | LTIME   |
|  |          | TOD                         | TIME  | TOD   |
|  |          | TOD                         | DINT  | TOD   |
|  |          | LTOD                        | TIME, LTIME                                 | LTOD  |
|  |          | LTOD                        | LINT  | LTOD  |
|  |          | DATE                        | LTOD  | DTL   |
|  |          | DATE                        | TOD   | <ul style="list-style-type: none"> <li>• S7-300/400: DT</li> <li>• S7-1200/1500:<br/>DTL</li> </ul>     |
|  |          | DT                          | TIME  | DT  |
|  |          | LDT                         | TIME, LTIME                                 | LDT   |
|  |          | DTL                         | TIME, LTIME                                 | DTL   |
| Subtraktion  | -        | Ganzzahl/<br>Gleitpunktzahl | Ganzzahl/Gleitpunktzahl                     | Ganzzahl/<br>Gleitpunktzahl   |
|  |          | TIME                        | TIME  | TIME  |
|  |          | TIME                        | DINT  | TIME  |
|  |          | LTIME <sup>1)</sup>         | TIME, LTIME                                 | LTIME   |
|  |          | LTIME                       | LINT  | LTIME   |
|  |          | TOD                         | TIME  | TOD   |
|  |          |                             | DINT  | TOD   |
|  |          | LTOD                        | TIME, LTIME                                 | LTOD  |
|  |          | LTOD                        | LINT  | LTOD  |
|  |          | DATE                        | DATE  | <ul style="list-style-type: none"> <li>• S7-300/400: TIME</li> <li>• S7-1200/1500:<br/>LTIME</li> </ul> |
|  |          | DT                          | TIME  | DT  |
|  |          | LDT                         | TIME, LTIME                                 | LDT   |
|  |          | DTL                         | TIME, LTIME                                 | DTL   |
|  |          | DTL                         | DTL   | LTIME   |
| <sup>1)</sup> Kombinationen zwischen Nanosekunden und Millisekunden sind innerhalb von Ausdrücken nicht möglich. |          |                             |   |   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt einen arithmetischen Ausdruck:

```
SCL  
"MyTag1" := "MyTag2" * "MyTag3";
```

## Siehe auch

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

## Vergleichsausdrücke

### Beschreibung

Vergleichsausdrücke vergleichen die Werte zweier Operanden und liefern einen booleschen Wert. Das Ergebnis ist TRUE, falls der Vergleich erfüllt ist, und FALSE, falls er nicht erfüllt ist.

Vergleichsoperatoren können die Datentypen verarbeiten, die in der verwendeten CPU zulässig sind. Der Datentyp des Ergebnisses ist immer BOOL.

Die folgenden Regeln müssen bei der Bildung von Vergleichsausdrücken beachtet werden:

- Innerhalb folgender Datentypgruppen sind alle Variablen vergleichbar:
  - Ganzzahlen/Gleitpunktzahlen
  - Binärzahlen
  - Zeichenfolgen
- Bei folgenden Datentypen/Datentypgruppen sind nur typgleiche Variablen vergleichbar:
  - TIME, LTIME
  - Datum und Uhrzeit
- Der Vergleich von Zeichenfolgen erfolgt nach dem ASCII-Zeichensatz. Für den Vergleich werden die Länge der Variablen und der numerische Wert jedes ASCII-Zeichens herangezogen.
- S5TIME-Variablen sind als Vergleichsoperanden nicht erlaubt. Eine explizite Konvertierung von S5TIME nach TIME oder LTIME ist notwendig.

### Datentypen von Vergleichsausdrücken

Die folgende Tabelle zeigt die Datentypen/Datentypgruppen, die Sie in Vergleichsausdrücken verwenden können:

| Operation  | Operator     | 1. Operand                      | 2. Operand                      | Ergebnis |
|--|--------------|---------------------------------|---------------------------------|----------|
| Vergleich auf gleich, ungleich                               | =, <>        | Ganzzahl/<br>Gleitpunktzahl     | Ganzzahl/<br>Gleitpunktzahl     | BOOL     |
|  |              | Binärzahl                       | Binärzahl                       | BOOL     |
|  |              | Zeichenfolge                    | Zeichenfolge                    | BOOL     |
|  |              | TIME, LTIME                     | TIME, LTIME                     | BOOL     |
|  |              | Datum und Uhrzeit               | Datum und Uhrzeit               | BOOL     |
| Vergleich auf kleiner, kleiner-gleich, größer, größer-gleich | <, <=, >, >= | Ganzzahl/<br>Gleitpunktzahl     | Ganzzahl/<br>Gleitpunktzahl     | BOOL     |
|  |              | Bitfolgen<br>(nur S7-1200/1500) | Bitfolgen<br>(nur S7-1200/1500) | BOOL     |
|  |              | Zeichenfolge                    | Zeichenfolge                    | BOOL     |
|  |              | TIME, LTIME                     | TIME, LTIME                     | BOOL     |
|  |              | Datum und Uhrzeit               | Datum und Uhrzeit               | BOOL     |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt einen Vergleichsausdruck:

```

SCL
IF a > b THEN c := a;
IF A > 20 AND B < 20 THEN C := TRUE;
IF A<>(B AND C) THEN C := FALSE;
    
```

#### Hinweis

Der Vergleich bei STRING und DT wird bei S7-300/400 intern durch erweiterte Anweisungen durchgeführt. Folgende Operanden sind für diese Funktionen nicht zulässig:

- Parameter eines FC
- Durchgangsparameter eines FB vom Typ STRUCT oder ARRAY

### Siehe auch

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

## Logische Ausdrücke

### Beschreibung

Logische Ausdrücke verknüpfen zwei Operanden durch logische Operatoren (AND, OR, XOR) oder negieren Operanden (NOT).

Logische Operatoren können die Datentypen verarbeiten, die in der verwendeten CPU zulässig sind. Das Ergebnis eines logischen Ausdrucks ist vom Datentyp BOOL, wenn beide Operanden vom Datentyp BOOL sind. Wenn mindestens einer der beiden Operanden eine Bitfolge ist, ist das Ergebnis ebenfalls eine Bitfolge und wird durch den Typ des höchstwertigen Operanden bestimmt. Wenn Sie zum Beispiel einen Operanden vom Typ BYTE mit einem Operanden vom Typ WORD verknüpfen, ist das Ergebnis vom Typ WORD.

Um einen Operanden vom Typ BOOL mit einer Bitfolge zu verknüpfen, müssen Sie ihn zunächst durch explizite Konvertierung in eine Bitfolge umwandeln.

### Datentypen von logischen Ausdrücken

Die folgende Tabelle zeigt die Datentypen, die Sie in logischen Ausdrücken verwenden können:

| Operation                 | Operator   | 1. Operand | 2. Operand | Ergebnis |
|---------------------------|------------|------------|------------|----------|
| Negation                  | NOT        | BOOL       | -          | BOOL     |
| Und-Verknüpfung           | AND oder & | BOOL       | BOOL       | BOOL     |
|                           |            | Bitfolge   | Bitfolge   | Bitfolge |
| Oder-Verknüpfung          | OR         | BOOL       | BOOL       | BOOL     |
|                           |            | Bitfolge   | Bitfolge   | Bitfolge |
| Exklusiv-Oder-Verknüpfung | XOR        | BOOL       | BOOL       | BOOL     |
|                           |            | Bitfolge   | Bitfolge   | Bitfolge |

### Beispiel

Das folgende Beispiel zeigt einen logischen Ausdruck:

```

SCL
IF "MyTag1" AND NOT "MyTag2" THEN c := a;
MyTag := ALPHA OR BETA;

```

### Siehe auch

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

## Operatoren und Operatorenrangfolge

### Operatoren und ihre Auswertungsreihenfolge

Ausdrücke können durch Operatoren miteinander verknüpft oder ineinander geschachtelt sein.

Die Reihenfolge der Auswertung von Ausdrücken hängt von der Rangfolge der Operatoren und der Klammerung ab. Grundsätzlich gelten folgende Regeln:

- Arithmetische Operatoren werden vor Vergleichsoperatoren und diese vor logischen Operatoren ausgewertet.
- Gleichrangige Operatoren werden nach ihrem Auftreten von links nach rechts ausgewertet.
- Operationen in Klammern werden zuerst ausgewertet.

Die folgende Tabelle gibt eine Übersicht über die Operatoren und ihre Rangfolge:

| Operator                       | Operation       | Rangfolge |
|--------------------------------|-----------------|-----------|
| <b>Arithmetische Ausdrücke</b> |                 |           |
| +                              | Unäres Plus     | 2         |
| -                              | Unäres Minus    | 2         |
| **                             | Potenz          | 3         |
| *                              | Multiplikation  | 4         |
| /                              | Division        | 4         |
| MOD                            | Modulo-Funktion | 4         |
| +                              | Addition        | 5         |
| -                              | Subtraktion     | 5         |
| <b>Vergleichsausdrücke</b>     |                 |           |
| <                              | Kleiner         | 6         |
| >                              | Größer          | 6         |
| <=                             | Kleiner gleich  | 6         |
| >=                             | Größer gleich   | 6         |
| =                              | Gleich          | 7         |
| <>                             | Ungleich        | 7         |
| <b>Logische Ausdrücke</b>      |                 |           |
| NOT                            | Negation        | 3         |
| AND oder &                     | Boolesches Und  | 8         |
| XOR                            | Exklusives Oder | 9         |
| OR                             | Boolesches Oder | 10        |
| <b>Weitere Operationen</b>     |                 |           |
| ( )                            | Klammerung      | 1         |
| :=                             | Zuweisung       | 11        |

### Siehe auch

Ausdrücke (Seite 1364)



## Wertzuweisungen

### Definition

Mit einer Wertzuweisung weisen Sie einer Variablen den Wert eines Ausdrucks zu. Auf der linken Seite der Zuweisung steht die Variable, die den Wert des rechts stehenden Ausdrucks übernimmt.

Als Ausdruck kann auch der Name einer Funktion angegeben werden. Die Funktion wird durch die Wertzuweisung aufgerufen und liefert ihren Rückgabewert an die links stehende Variable zurück.

Der Datentyp der Wertzuweisung wird durch den Datentyp der links stehenden Variablen definiert. Der Datentyp des rechts stehenden Ausdrucks muss zu diesem Datentyp passen.

Weitere Informationen zu Kompatibilität und Konvertierung von Datentypen finden Sie unter "Siehe auch".

### Wertzuweisungen für Datentyp STRUCT oder PLC-Datentypen

Eine gesamte Struktur ist einer anderen Struktur zuweisbar, wenn die Strukturen identisch aufgebaut sind und sowohl die Datentypen als auch die Namen der Strukturkomponenten übereinstimmen.

Sie können einem einzelnen Strukturelement eine Variable, einen Ausdruck oder ein anderes Strukturelement zuweisen.

### Wertzuweisungen für Datentyp ARRAY

Ein gesamtes ARRAY ist einem anderen ARRAY zuweisbar, wenn sowohl die Datentypen der ARRAY-Elemente als auch die ARRAY-Grenzen übereinstimmen.

Sie können einem einzelnen ARRAY-Element eine Variable, einen Ausdruck oder ein anderes ARRAY-Element zuweisen.

### Wertzuweisungen für Datentyp STRING

Ein gesamter STRING ist einem anderen STRING zuweisbar. Ist die zugewiesene Zeichenkette länger als die links stehende Zeichenkette, wird beim Übersetzen eine Warnung ausgegeben.

Sie können einem einzelnen STRING-Element ein anderes STRING-Element zuweisen.

### Wertzuweisungen für Datentyp ANY

Eine Variable mit dem Datentyp ANY können Sie nur folgenden Objekten zuweisen:

- Eingangparametern oder temporären Lokaldaten von FBs, die ebenfalls den Datentyp ANY haben.
- Temporären Lokaldaten von FCs, die ebenfalls den Datentyp ANY haben.

Beachten Sie, dass Sie mit dem Zeiger ANY nur auf Speicherbereiche mit der Zugriffsart "Standard" zeigen können.

## Wertzuweisungen für Datentyp POINTER

In SCL können Sie POINTER in Wertzuweisungen nicht verwenden.

### Beispiele

Die folgende Tabelle zeigt Beispiele für Wertzuweisungen:

| SCL                                   |  |
|---------------------------------------|--|
| "MyTag1" := "MyTag2";                 | (* Zuweisung einer Variablen*)   |
| "MyTag1" := "MyTag2" * "MyTag3";      | (* Zuweisung eines Ausdrucks*)   |
| "MyTag" := "MyFC" ();                 | (* Aufruf einer Funktion, die ihren Rückgabewert der Variablen "MyTag" zuweist*) |
| #MyStruct.MyStructElement := "MyTag"; | (* Zuweisung einer Variablen an ein Strukturelement*)                            |
| #MyArray[2] := "MyTag";               | (* Zuweisung einer Variablen an ein ARRAY-Element*)                              |
| "MyTag" := #MyArray[1,4];             | (* Zuweisung eines ARRAY-Elements an eine Variable*)                             |
| #MyString[2] := #MyOtherString[5];    | (* Zuweisung eines STRING-Elements an ein anderes STRING-Element*)               |

### Siehe auch

Operatoren und Operatorenrangfolge (Seite 1370)

### Einstellungen zu SCL

### Übersicht über die Einstellungen zu SCL

### Übersicht

Die folgenden Tabellen zeigen die Einstellungen, die Sie für SCL vornehmen können:

### Editoreinstellungen

| Gruppe  | Einstellung              | Beschreibung   |
|---------|--------------------------|--|
| Ansicht | Schlüsselwothervorhebung | Schreibweise, mit der Schlüsselwörter der Programmiersprache dargestellt werden. Sie können zwischen Groß- und Kleinschreibung oder einer den Konventionen der Programmiersprache Pascal entsprechenden Schreibweise wählen. |

## Voreinstellung für neue Bausteine

Wenn Sie Bausteine neu anlegen, werden die folgenden Einstellungen als Standardwerte gesetzt. Sie können diese zu einem späteren Zeitpunkt innerhalb der Bausteineigenschaften ändern.

| Gruppe  | Einstellung                            | Beschreibung   |
|---|--|--|
| Übersetzen  | Erweiterte Statusinformation erstellen | Ermöglicht das Beobachten aller Variablen in einem Baustein. Der Speicherbedarf des Programms und die Ablaufzeiten erhöhen sich jedoch durch diese Option.   |
|   | ARRAY-Grenzen prüfen <sup>1)</sup>     | Prüft zu Laufzeit, ob Feldindizes in dem für ein ARRAY deklarierten Bereich liegen. Wenn ein Feldindex den zulässigen Bereich überschreitet, wird der Freigabeausgang ENO des Bausteins auf "0" gesetzt. |
|   | ENO automatisch setzen                 | Prüft zur Laufzeit, ob Fehler bei der Bearbeitung bestimmter Anweisungen auftreten. Wenn ein Laufzeitfehler auftritt, wird der Freigabeausgang ENO des Bausteins auf "0" gesetzt.                        |
| <sup>1)</sup> Für CPUs der Baureihen S7-300/400 gilt: Bei Überschreitung der ARRAY-Grenzen wird der Freigabeausgang ENO auf FALSE gesetzt.<br>Für CPUs der Baureihen S7-1200/1500 gilt: Bei Überschreitung der ARRAY-Grenzen wird der Freigabeausgang ENO nicht auf FALSE gesetzt. Siehe "Auto-Hotspot" zu Möglichkeiten der Fehlerabfrage. |  |  |

## Siehe auch

Einstellungen ändern (Seite 1373)

## Einstellungen ändern

### Vorgehen

Um die Einstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung".
3. Ändern Sie die Einstellungen.

### Ergebnis

Die Änderung wird übernommen und muss nicht explizit gespeichert werden.

## Siehe auch

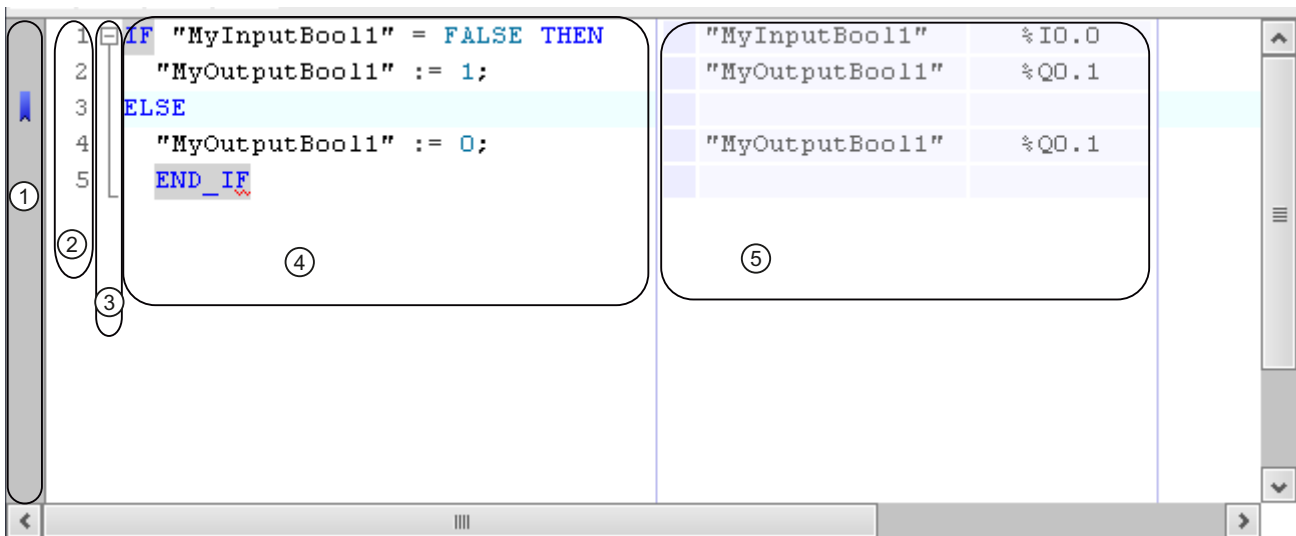
Übersicht über die Einstellungen zu SCL (Seite 1372)

## Das Programmierfenster von SCL

### Übersicht über das Programmierfenster

#### Funktion

Das Programmierfenster ist der Arbeitsbereich, in dem Sie das SCL-Programm eingeben.  
Das folgende Bild zeigt das Programmierfenster von SCL:



Das Programmierfenster besteht aus folgenden Bereichen:

| Bereich                           | Bedeutung   |
|-----------------------------------|---|
| ① Seitenleiste                    | In der Seitenleiste können Sie Lesezeichen und Haltepunkte setzen.                  |
| ② Zeilennummern                   | Links neben dem Programmcode werden die Zeilennummern angezeigt.                    |
| ③ Gliederungsanzeige              | Die Gliederungsanzeige markiert zusammenhängende Codeabschnitte.                    |
| ④ Codebereich                     | Im Codebereich bearbeiten Sie das SCL-Programm.                                     |
| ⑤ Anzeige der absoluten Operanden | Diese Tabelle zeigt die Zuordnung der symbolischen Operanden zu absoluten Adressen. |

#### Siehe auch

- Lesezeichen verwenden (Seite 1377)
- Anpassen des Programmierfensters (Seite 1375)
- Zeilen ein- und ausrücken (Seite 1376)
- Codeabschnitte auf- und zuklappen (Seite 1377)

## Anpassen des Programmierfensters

### Einführung

Die Darstellung des Programmierfensters und des Programmcodes können Sie auf folgende Weise anpassen:

- Einstellen der Schriftart, -größe und -farbe
- Einstellen der Tabulatorweite
- Anzeige der Zeilennummern
- Ein- oder Ausblenden der absoluten Operanden

### Schriftart, -größe und -farbe einstellen

Um Schriftart, -größe und -farbe einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie die Gruppe "Allgemein > Skript-/Texteditoren".
3. Stellen Sie die gewünschte Schriftart und -größe ein oder wählen Sie Schriftfarben für die einzelnen Sprachelemente aus.

### Tabulatorweite einstellen

Um das Programm übersichtlicher zu gestalten, werden Zeilen syntaxgerecht eingerückt. Die Breite der Einrückung definieren Sie über die Tabulatorweite.

Um die Tabulatorweite einzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie die Gruppe "Allgemein > Skript-/Texteditoren".
3. Stellen Sie die Tabulatorweite ein.

### Zeilennummern anzeigen

Um die Zeilennummern anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie die Gruppe "Allgemein > Skript-/Texteditoren".
3. Aktivieren Sie die Option "Zeilennummern anzeigen".

### Ein- und Ausblenden der absoluten Operanden

Die Zuordnung von symbolischen und absoluten Operanden können Sie bei Bedarf in einer Tabelle neben dem Programmcode einblenden.

Um die Anzeige der absoluten Operanden ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol "Absolute/symbolische Operanden". Die Anzeige der absoluten Operanden wird eingeblendet.
2. Um die Anzeige zu verschieben, klicken Sie auf die Tabelle und ziehen sie bei gedrückter Maustaste an die gewünschte Position.
3. Um die Breite der Tabelle zu verändern, klicken Sie auf den rechten oder linken Tabellenrand und ziehen Sie ihn bei gedrückter Maustaste nach rechts oder links.

### Siehe auch

Lesezeichen verwenden (Seite 1377)

Übersicht über das Programmierfenster (Seite 1374)

Zeilen ein- und ausrücken (Seite 1376)

Codeabschnitte auf- und zuklappen (Seite 1377)

### Zeilen ein- und ausrücken

#### Einführung

Um das Programm übersichtlicher zu gestalten, werden Zeilen syntaxgerecht eingerückt. Sie können jedoch auch einzelne Zeilen manuell einrücken.

#### Vorgehen

Um einzelne Zeilen ein- oder auszurücken, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Funktionsleiste des Programmiereditors die Schaltfläche "Text einrücken" oder "Text ausrücken".

---

#### Hinweis

Die Breite der Einrückung können Sie über "Extras > Einstellungen" definieren.

---

### Siehe auch

Lesezeichen verwenden (Seite 1377)

Übersicht über das Programmierfenster (Seite 1374)

Anpassen des Programmierfensters (Seite 1375)

Codeabschnitte auf- und zuklappen (Seite 1377)

## Codeabschnitte auf- und zuklappen

### Einführung

SCL-Anweisungen können sich über mehrere Zeilen erstrecken. Beispiele hierfür sind Programmsteuerungsanweisungen oder Bausteinaufrufe.

Diese Anweisungen werden wie folgt gekennzeichnet:

- Eine Gliederungsanzeige zwischen der Zeilennummer und dem Programmcode markiert den gesamten Codeabschnitt.
- Wenn Sie das öffnende Schlüsselwort markieren, wird automatisch auch das schließende Schlüsselwort markiert.

### Vorgehen

Um Codeabschnitte auf- und zuzuklappen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Minus-Zeichen in der Gliederungsanzeige.  
Der Codeabschnitt wird zugeklappt.
2. Klicken Sie auf das Plus-Zeichen in der Gliederungsanzeige.  
Der Codeabschnitt wird wieder aufgeklappt.

### Siehe auch

Lesezeichen verwenden (Seite 1377)

Übersicht über das Programmierfenster (Seite 1374)

Anpassen des Programmierfensters (Seite 1375)

Zeilen ein- und ausrücken (Seite 1376)

## Lesezeichen verwenden

### Grundlagen zu Lesezeichen

#### Funktion

Mit Lesezeichen können Sie Programmstellen in umfangreichen Programmen markieren, um sie für eine spätere Überarbeitung leichter aufzufinden. Lesezeichen werden in der Seitenleiste des Programmierfensters angezeigt. Sie können über Menübefehle zwischen mehreren Lesezeichen innerhalb eines Bausteins navigieren.

Lesezeichen werden mit dem Projekt gespeichert und stehen somit allen Bearbeitern des Bausteins zur Verfügung. Sie werden jedoch nicht mit in ein Gerät geladen.

Beim Bausteinvergleich werden Lesezeichen nicht ausgewertet.

## Siehe auch

- Lesezeichen setzen (Seite 1378)
- Zwischen Lesezeichen navigieren (Seite 1378)
- Lesezeichen löschen (Seite 1379)

## Lesezeichen setzen

### Voraussetzung

Der SCL-Baustein ist geöffnet.

### Vorgehen

Um ein Lesezeichen zu setzen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in der Seitenleiste in die gewünschte Zeile.
2. Wählen Sie im Kontextmenü den Befehl "Lesezeichen > Setzen".

Oder:

1. Klicken Sie in die Zeile, in der Sie das Lesezeichen setzen möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Lesezeichen setzen/löschen".

Oder:

1. Halten Sie die Taste <Strg> gedrückt.
2. Klicken Sie in der Seitenleiste in die Zeile, in der Sie das Lesezeichen setzen möchten.

### Ergebnis

Ein Lesezeichen wird in den Programmcode gesetzt.

## Siehe auch

- Grundlagen zu Lesezeichen (Seite 1377)
- Zwischen Lesezeichen navigieren (Seite 1378)
- Lesezeichen löschen (Seite 1379)

## Zwischen Lesezeichen navigieren

### Voraussetzung

In einem Baustein sind mehrere Lesezeichen gesetzt.



## Vorgehen

Um zwischen Lesezeichen zu navigieren, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke in den Programmcode.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Gehe zu > Nächstem Lesezeichen" oder "Gehe zu > Vorangehendem Lesezeichen".

Oder:

1. Setzen Sie die Einfügemarke in den Programmcode.
2. Klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Gehe zu nächstem Lesezeichen" oder "Gehe zu vorangehendem Lesezeichen".

Oder:

1. Klicken Sie in die Seitenleiste.
2. Wählen Sie im Kontextmenü den Befehl "Lesezeichen > Nächstes" oder "Lesezeichen > Vorangehendes".

## Ergebnis

Die Zeile mit dem Lesezeichen wird hervorgehoben.

## Siehe auch

Grundlagen zu Lesezeichen (Seite 1377)

Lesezeichen setzen (Seite 1378)

Lesezeichen löschen (Seite 1379)

## Lesezeichen löschen

Sie können einzelne Lesezeichen oder alle Lesezeichen aus dem Baustein bzw. der CPU löschen.

### Einzelne Lesezeichen löschen

Um ein einzelnes Lesezeichen zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Zeile, in der Sie das Lesezeichen löschen möchten, mit der rechten Maustaste in die Seitenleiste.
2. Wählen Sie im Kontextmenü den Befehl "Lesezeichen > Löschen".

Oder:

1. Klicken Sie in die Zeile, in der Sie das Lesezeichen löschen möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Lesezeichen > Löschen".

Oder:

1. Klicken Sie in die Zeile, in der Sie das Lesezeichen löschen möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Lesezeichen setzen/löschen".

## Alle Lesezeichen aus dem Baustein löschen

Um alle Lesezeichen aus dem Baustein zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in die Seitenleiste.
2. Wählen Sie im Kontextmenü den Befehl "Lesezeichen > Alle aus Baustein löschen".

Oder:

1. Wählen Sie im Menü "Bearbeiten" den Befehl "Lesezeichen > Alle aus Baustein löschen".

## Siehe auch

Grundlagen zu Lesezeichen (Seite 1377)

Lesezeichen setzen (Seite 1378)

Zwischen Lesezeichen navigieren (Seite 1378)

## SCL-Anweisungen eingeben

## Regeln zu SCL-Anweisungen

## Anweisungen in SCL

SCL kennt folgende Arten von Anweisungen:

- Wertzuweisungen  
Wertzuweisungen dienen dazu, einer Variablen einen konstanten Wert, das Ergebnis eines Ausdrucks, oder den Wert einer anderen Variablen zuzuweisen.
- Anweisungen zur Programmsteuerung  
Anweisungen zur Programmsteuerung dienen dazu, Programmverzweigungen, -schleifen oder -sprünge zu realisieren.
- Weitere Anweisungen aus der Task Card "Anweisungen"  
Die Task Card "Anweisungen" bietet eine breite Auswahl von Standardanweisungen, die Sie in Ihrem SCL-Programm verwenden können.
- Bausteinaufrufe  
Bausteinaufrufe dienen dazu, Unterprogramme, die in andere Bausteine ausgelagert sind, zu bearbeiten und deren Ergebnis weiter zu verarbeiten.

## Regeln

Bei der Eingabe von SCL-Anweisungen müssen Sie folgende Regeln beachten:

- Anweisungen können sich über mehrere Zeilen erstrecken.
- Jede Anweisung endet mit einem Semikolon (;).

- Groß- und Kleinschreibung wird nicht unterschieden.
- Kommentare dienen lediglich der Dokumentation des Programms. Sie beeinträchtigen den Programmablauf nicht.

## Beispiele

Die folgenden Beispiele zeigen die verschiedenen Anweisungsarten:

### SCL

```
// Beispiel für eine Wertzuweisung
"MyTag" := 0;

// Beispiel für einen Bausteinanruf
"MyDB"."MyFB" (ParamInput:= 10);

// Beispiel für eine Anweisung zur Programmsteuerung
WHILE "Counter" < 10 DO
    "MyTAG" := "MyTag" + 2;
END_WHILE;
```

## Siehe auch

Grundlagen zu SCL (Seite 1363)

## SCL-Anweisungen manuell eingeben

### Voraussetzung

Ein SCL-Baustein ist geöffnet.

### Vorgehen

Um SCL-Anweisungen einzugeben, gehen Sie folgendermaßen vor:

1. Geben Sie die Syntax der Anweisung über die Tastatur ein.  
Bei der Eingabe werden Sie durch die Autovervollständigung unterstützt. Sie bietet alle Anweisungen und Operanden an, die an der aktuellen Position zulässig sind.
2. Wählen Sie aus der Autovervollständigung die gewünschte Anweisung oder den gewünschten Operanden aus.  
Wenn Sie eine Anweisung auswählen, die die Angabe von Operanden erfordert, werden Platzhalter für die Operanden ins Programm eingefügt. Die Platzhalter für die Operanden sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
3. Ersetzen Sie den Platzhalter durch einen Operanden.
4. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Operanden.

---

### Hinweis

Sie können einen bereits definierten Operanden auch per Drag & Drop aus der PLC-Variablen-tabelle oder der Bausteinschnittstelle in das Programm ziehen.

---

### Ergebnis

Die Anweisung wird eingefügt.

Der Programmiereditor führt eine Syntaxprüfung durch. Dabei werden fehlerhafte Eingaben rot und kursiv angezeigt. Zusätzlich erhalten Sie eine detaillierte Fehlermeldung im Inspektorfenster.

### Siehe auch

Parameterliste erweitern und reduzieren (Seite 1396)

### SCL-Anweisungen über Task Card "Anweisungen" einfügen

Die Task Card "Anweisungen" bietet eine breite Auswahl von Anweisungen, die Sie in Ihrem SCL-Programm verwenden können. Auch die SCL-spezifischen Anweisungen zur Programmsteuerung sind über die Task Card "Anweisungen" verfügbar.

### Voraussetzung

Ein SCL-Baustein ist geöffnet.

### Vorgehen

Um SCL-Anweisungen mithilfe der Task Card "Anweisungen" in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Zum Einfügen der Anweisung wählen Sie einen der folgenden Schritte:
  - Navigieren Sie zu der SCL-Anweisung, die Sie einfügen möchten, und ziehen Sie sie per Drag & Drop an die gewünschte Zeile im Programmcode. Die Einfügestelle wird durch ein grünes Rechteck gekennzeichnet.
  - Markieren Sie die Stelle im Programmcode, an die Sie die Anweisung einfügen möchten, und doppelklicken Sie dann auf die einzufügende Anweisung.

Die Anweisung wird ins Programm eingefügt. Die Platzhalter für die Operanden sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
3. Ersetzen Sie den Platzhalter durch einen Operanden. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablen-tabelle per Drag & Drop auf den Platzhalter ziehen.
4. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Operanden.

## Ergebnis

Die Anweisung wird eingefügt.

Der Programmiereditor führt eine Syntaxprüfung durch. Dabei werden fehlerhafte Eingaben rot und kursiv angezeigt. Zusätzlich erhalten Sie eine detaillierte Fehlermeldung im Inspektorfenster.

## Siehe auch

Parameterliste erweitern und reduzieren (Seite 1396)

## Datentyp einer SCL-Anweisung festlegen

### Grundlagen zu den Datentypen von SCL-Anweisungen

## Einführung

Die SCL-Anweisungen, die Sie für die Programmierung Ihrer Bausteine einsetzen, nutzen bestimmte Datentypen für die Ermittlung des Funktionswerts. Dabei gibt es SCL-Anweisungen, für die nur ein ganz bestimmter Datentyp verwendet werden kann. Bei diesen Anweisungen können Sie den Datentyp nicht ändern. Die Mehrheit der SCL-Anweisungen kann jedoch mit unterschiedlichen Datentypen arbeiten. Bei diesen Anweisungen werden folgende beide Arten unterschieden:

- Anweisungen, bei denen der Datentyp des Funktionswerts durch den Datentyp der Eingangsparameter bestimmt wird. Dies ist bei den meisten Anweisungen der Fall.
- Anweisungen, bei denen ein Datentyp voreingestellt ist. Dies betrifft die Anweisungen, die in der unten stehenden Tabelle aufgelistet sind.

Bei der zweiten Gruppe müssen Sie den voreingestellten Datentyp ändern, falls dieser vom Datentyp des verwendeten Eingangsparameters abweicht. Grundsätzlich können Sie den Datentyp mit der folgenden Syntax ändern:

`_<Datentyp>`

## SCL-Anweisungen mit voreingestelltem Datentyp

Die folgende Tabelle zeigt die SCL-Anweisungen mit voreingestellten Datentypen:

| Anweisung | Voreingestellter Datentyp |
|-----------|---------------------------|
| CEIL      | DINT                      |
| DECO      | DWORD                     |
| FLOOR     | DINT                      |
| NORM_X    | REAL                      |
| PEEK      | BYTE                      |
| SCALE_X   | INT                       |
| TRUNC     | DINT                      |
| CONCAT    | STRING                    |

## Siehe auch

Datentyp einer SCL-Anweisung ändern (Seite 1384)

Beispiel zum Ändern des Datentyps einer SCL-Anweisung (Seite 1385)

## Datentyp einer SCL-Anweisung ändern

### Vorgehen

Um eine SCL-Anweisung einzufügen und anschließend ihren Datentyp zu ändern, gehen Sie folgendermaßen vor:

1. Fügen Sie die Anweisung per Drag & Drop an der gewünschten Stelle im Programm ein.
2. Geben Sie die Operanden für die Anweisung an.  
Der Datentyp des Funktionswerts wird entsprechend den Eingangsparametern festgelegt oder es wird der voreingestellte Datentyp der Anweisung verwendet.
3. Ergänzen Sie den Anweisungsnamen um "\_<Datentyp>".  
"<Datentyp>" ist dabei der gewünschte Datentyp der Anweisung.

## Siehe auch

Grundlagen zu den Datentypen von SCL-Anweisungen (Seite 1383)

Beispiel zum Ändern des Datentyps einer SCL-Anweisung (Seite 1385)

## Datentyp von IEC-Zeiten und IEC-Zählern ändern

IEC-Zeiten und IEC-Zähler sind systeminterne Funktionsbausteine, die einen Instanz-Datenbaustein benötigen. Den Instanz-Datenbaustein können Sie entweder als Einzel- oder Multiinstanz anlegen. Der Datentyp des Instanz-Datenbausteins richtet sich nach der zugehörigen Anweisung. Für CPUs der Baureihen S7-1200 und S7-1500 können Sie die Anweisungen jedoch auch mit unterschiedlichen Datentypen ausführen, um sie Ihren Bedürfnissen anzupassen.

Wenn der neu eingestellte Datentyp des Instanz-Datenbausteins nicht mit dem Datentyp der Eingangsparameter übereinstimmt, erfolgt eine implizite Konvertierung, falls dies möglich ist. Wenn die Konvertierung nicht möglich ist, erhalten Sie eine Fehlermeldung.

## Vorgehen

Um den Datentyp des Instanz-Datenbausteins von IEC-Zeiten oder IEC-Zählern zu ändern, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, in dem Sie die IEC-Zeit oder den IEC-Zähler aufrufen.  
Abhängig vom Instanztyp des Instanz-Datenbausteins befindet sich ein grün umrandetes Rechteck vor (Multiinstanz) oder nach (Einzelinstanz) dem Namen des Instanz-Datenbausteins.
2. Klicken Sie auf das grün umrandete Rechteck.  
Eine Klappliste mit den möglichen Datentypen für den Instanz-Datenbaustein wird geöffnet.
3. Wählen Sie den gewünschten Datentyp.

## Beispiel zum Ändern des Datentyps einer SCL-Anweisung

### Ändern des voreingestellten Datentyps der Anweisung "Decodieren" (DECO)

Wenn Sie die Anweisung "Decodieren" in das Programm einfügen, ist der Datentyp DINT bereits voreingestellt.

```
"Tag_Result" := DECO(IN := "Tag_Value");
```

Um den Datentyp anschließend von DINT zu BYTE zu ändern, ändern Sie den Programmcode folgendermaßen:

```
"Tag_Result" := DECO_BYTE(IN := "Tag_Value");
```

## Siehe auch

Grundlagen zu den Datentypen von SCL-Anweisungen (Seite 1383)

Datentyp einer SCL-Anweisung ändern (Seite 1384)

## Favoriten in SCL verwenden

## SCL-Anweisungen zu den Favoriten hinzufügen

## Voraussetzung

- Ein Baustein ist geöffnet.
- Für die Task Card "Anweisungen" ist der Mehrpalettenmodus eingestellt oder die Favoriten werden auch im Editor angezeigt.

## Vorgehen

Um Anweisungen zu den Favoriten hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Task Card "Anweisungen".
2. Maximieren Sie die Palette "Einfache Anweisungen".
3. Navigieren Sie in der Palette "Einfache Anweisungen" zu der Anweisung, die Sie den Favoriten hinzufügen möchten.
4. Ziehen Sie die Anweisung per Drag & Drop entweder in die Palette "Favoriten" oder in den Favoritenbereich im Programmiereditor.

---

### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---

## Siehe auch

SCL-Anweisungen mithilfe von Favoriten einfügen (Seite 1386)

SCL-Anweisungen aus den Favoriten entfernen (Seite 1387)

## SCL-Anweisungen mithilfe von Favoriten einfügen

### Voraussetzung

- Ein Baustein ist geöffnet.
- Favoriten sind vorhanden.

## Vorgehen

Um eine Anweisung mithilfe der Favoriten in ein Programm einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die gewünschte Anweisung per Drag & Drop aus den Favoriten an die gewünschte Position.

Oder:

1. Markieren Sie die Programmstelle, an der Sie die Anweisung einfügen möchten.
2. Klicken Sie auf die einzufügende Anweisung in den Favoriten.



---

### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---

### Siehe auch

SCL-Anweisungen zu den Favoriten hinzufügen (Seite 1385)

SCL-Anweisungen aus den Favoriten entfernen (Seite 1387)

### SCL-Anweisungen aus den Favoriten entfernen

### Voraussetzung

Ein Codebaustein ist geöffnet.

### Vorgehen

Um Anweisungen aus den Favoriten zu entfernen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Anweisung, die Sie entfernen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Anweisung entfernen".

---

### Hinweis

Um die Favoriten zusätzlich auch im Programmiereditor anzuzeigen, klicken Sie in der Funktionsleiste des Programmiereditors auf die Schaltfläche "Favoriten auch im Editor anzeigen".

---

### Siehe auch

SCL-Anweisungen zu den Favoriten hinzufügen (Seite 1385)

SCL-Anweisungen mithilfe von Favoriten einfügen (Seite 1386)

## Bausteinaufrufe in SCL einfügen

### Grundlagen zum Bausteinaufruf in SCL

#### Aufruf von Funktionsbausteinen

##### Syntax eines Aufrufs

Für den Aufruf eines Funktionsbausteins als Einzel- oder Multiinstanz wird die folgende Syntax verwendet:

- Einzelinstanz:
  - Wenn der Funktionsbaustein aus dem Projekt stammt:  
`<DBName> (Parameterliste)`
  - Wenn der Funktionsbaustein aus der Task Card "Anweisungen" stammt:  
`<DBName>.<Anweisungsname> (Parameterliste)`
- Multiinstanz  
`<#Instanz-Name> (Parameterliste)`

##### Aufruf als Einzel-/ oder Multiinstanz

Funktionsbausteine können Sie entweder als Einzelinstanz oder als Multiinstanz aufrufen.

- Aufruf als Einzelinstanz  
Der aufgerufene Funktionsbaustein speichert seine Daten in einem eigenen Datenbaustein.
- Aufruf als Multiinstanz  
Der aufgerufene Funktionsbaustein speichert seine Daten im Instanz-Datenbaustein des aufrufenden Funktionsbausteins.

Weitere Informationen zu den Aufrufarten finden Sie unter "Siehe auch".

##### Parameterliste

Wenn Sie aus einem SCL-Baustein heraus einen anderen Codebaustein aufrufen, können Sie die Formalparameter des aufgerufenen Bausteins mit Aktualparametern versorgen.

Die Angabe der Parameter hat die Form einer Wertzuweisung. Durch diese Wertzuweisung weisen Sie den Parametern, die Sie im aufgerufenen Baustein definiert haben, einen Wert (Aktualparameter) zu.

Die Formalparameter des aufgerufenen Codebausteins werden in einer Klammer direkt nach dem Aufruf aufgelistet. Eingangs- und Durchgangparameter haben die Zuweisungskennung ":", Ausgangsparameter haben die Zuweisungskennung "=>". Ein Platzhalter hinter dem Parameter zeigt den erforderlichen Datentyp und die Art des Parameters an.

## Regeln zur Parameterversorgung

Zur Parameterversorgung gelten folgende Regeln:

- Als Aktualparameter können Konstanten, Variablen und Ausdrücke verwendet werden.
- Die Reihenfolge der Zuweisungen ist beliebig.
- Datentyp von Formal- und Aktualparameter müssen übereinstimmen.
- Die einzelnen Zuweisungen sind durch Komma getrennt.
- Wenn der aufgerufene Baustein nur einen Parameter besitzt, genügt die Angabe des Aktualparameters in der Klammer. Der Formalparameter muss nicht mit angegeben werden.

## Siehe auch

Bausteinaufrufe manuell einfügen (Seite 1393)

Bausteinaufrufe per Drag & Drop einfügen (Seite 1394)

Beispiele zum Aufruf eines Funktionsbausteins in SCL (Seite 1391)

## Aufruf von Funktionen

### Syntax eines Aufrufs

Für den Aufruf einer Funktion wird die folgende Syntax verwendet:

```
<Funktionsname> (Parameterliste); //Standardaufruf  
<Operand>:=<Funktionsname> (Parameterliste); // Aufruf in einem  
Ausdruck
```

### Rückgabewert

Die Aufrufmöglichkeiten von Funktionen hängen davon ab, ob die Funktion einen Rückgabewert an den aufrufenden Baustein zurückgibt. Der Rückgabewert wird im Parameter RET\_VAL definiert. Ist der Parameter RET\_VAL vom Datentyp VOID, liefert die Funktion keinen Wert an den aufrufenden Baustein zurück. Wenn der Parameter RET\_VAL einen anderen Datentyp hat, liefert die Funktion einen Rückgabewert dieses Datentyps zurück.

Für den Parameter RET\_VAL sind in SCL alle Datentypen zulässig mit Ausnahme von ANY, ARRAY, STRUCT und VARIANT sowie der Parametertypen TIMER und COUNTER.

## Aufrufmöglichkeiten

Es gibt zwei Möglichkeiten des Funktionsaufrufs in SCL:

- Standardaufruf für Funktionen mit und ohne Rückgabewert  
Beim Standardaufruf stehen die Ergebnisse der Funktion als Ausgangs- und Durchgangparameter zur Verfügung.
- Aufruf in einem Ausdruck für Funktionen mit Rückgabewert  
Funktionen, die einen Rückgabewert liefern, können in jedem Ausdruck anstelle eines Operanden verwendet werden, z. B. in einer Wertzuweisung.  
Die Funktion berechnet den Rückgabewert, der den gleichen Namen trägt wie die Funktion, und gibt ihn an den aufrufenden Baustein zurück. Dort ersetzt der Wert den Funktionsaufruf. Nach dem Aufruf stehen die Ergebnisse der Funktion als Rückgabewert oder als Ausgangs- und Durchgangparameter zur Verfügung.

## Parameterliste

Wenn Sie aus einem SCL-Baustein heraus einen anderen Codebaustein aufrufen, müssen Sie die Formalparameter des aufgerufenen Bausteins mit Aktualparametern versorgen.

Die Angabe der Parameter hat die Form einer Wertzuweisung. Durch diese Wertzuweisung weisen Sie den Parametern, die Sie im aufgerufenen Baustein definiert haben, einen Wert (Aktualparameter) zu.

Die Formalparameter des aufgerufenen Codebausteins werden in einer Klammer direkt nach dem Aufruf aufgelistet. Eingangs- und Durchgangparameter haben die Zuweisungskennung ":", Ausgangsparameter haben die Zuweisungskennung "=". Ein grau hinterlegter Platzhalter hinter dem Parameter zeigt den erforderlichen Datentyp und die Art des Parameters an.

## Regeln zur Parameterversorgung

Zur Parameterversorgung von Funktionen gelten folgende Regeln:

- Alle Parameter der Funktion müssen versorgt werden.
- Die Reihenfolge der Zuweisungen ist beliebig.
- Als Aktualparameter können Konstanten, Variablen und Ausdrücke verwendet werden.
- Datentyp von Formal- und Aktualparameter müssen übereinstimmen.
- Die einzelnen Zuweisungen sind durch Komma getrennt.
- Wenn der aufgerufene Baustein nur einen Parameter besitzt, genügt die Angabe des Aktualparameters in der Klammer. Der Formalparameter muss nicht mit angegeben werden.
- Beim Aufruf von Funktionen in SCL können Sie den Freigabemechanismus über EN nicht nutzen. Verwenden Sie stattdessen eine IF-Anweisung, um Funktionen bedingt aufzurufen.

## Siehe auch

- Bausteinaufrufe manuell einfügen (Seite 1393)
- Bausteinaufrufe per Drag & Drop einfügen (Seite 1394)
- Beispiele zum Aufruf von Funktionen in SCL (Seite 1392)

## Beispiele zum Aufruf eines Funktionsbausteins in SCL

### Aufruf als Einzelinstanz

Das folgende Beispiel zeigt den Aufruf eines FB als Einzelinstanz:

```
SCL  
// Aufruf als Einzelinstanz  
"MyDB" (MyInput:=10, MyInout:= "Tag1");
```

### Ergebnis

Nach der Ausführung des Aufrufs steht der für den Durchgangparameter "MyInout" ermittelte Wert in "Tag1" im Datenbaustein "MyDB" zur Verfügung.

### Aufruf als Multiinstanz

Das folgende Beispiel zeigt den Aufruf eines FB als Multiinstanz:

```
SCL  
// Aufruf als Multiinstanz  
"MyFB" (MyInput:= 10, MyInout:= "Tag1");
```

### Ergebnis

Nach der Ausführung des Bausteins "MyFB" steht der für den Durchgangparameter "MyInout" ermittelte Wert in "Tag1" im Datenbaustein des aufrufenden Codebausteins zur Verfügung.

## Siehe auch

- Aufruf von Funktionsbausteinen (Seite 1388)
- Bausteinaufrufe manuell einfügen (Seite 1393)
- Bausteinaufrufe per Drag & Drop einfügen (Seite 1394)

## Beispiele zum Aufruf von Funktionen in SCL

### Standardaufruf

Das folgende Beispiel zeigt einen Standard-Funktionsaufruf:

```
SCL  
// Standard-Funktionsaufruf  
"MyFC" (MyInput := 10, MyInOut := "Tag1");
```

### Ergebnis

Nach der Ausführung des Bausteins "MyFC" steht der für den Durchgangparameter "MyInOut" ermittelte Wert in "Tag1" im aufrufenden Baustein zur Verfügung und muss dort weiter verarbeitet werden.

### Aufruf in einer Wertzuweisung

Das folgende Beispiel zeigt einen Funktionsaufruf in einer Wertzuweisung:

```
SCL  
(*Aufruf in einer Wertzuweisung, für "MyFC" wurde ein Rückgabewert  
definiert *)  
#MyOperand := "MyFC" (MyInput1 := 3, MyInput2 := 2, MyInput3 := 8.9,  
MyInOut := "Tag1");
```

### Ergebnis

Der Rückgabewert von "MyFC" wird an "#MyOperand" übertragen.

### Aufruf in einem arithmetischen Ausdruck

Das folgende Beispiel zeigt einen Funktionsaufruf in einem arithmetischen Ausdruck:

```
SCL  
(*Aufruf in einem arithmetischen Ausdruck, für "MyFC" wurde ein  
Rückgabewert definiert *)  
#MyOperand := "Tag2" + "MyFC" (MyInput1 := 3, MyInput2 := 2, MyInput3 :=  
8.9);
```

### Ergebnis

Der Rückgabewert von "MyFC" wird zu "Tag2" addiert und das Ergebnis an "MyOperand" übertragen.

## Siehe auch

- Aufruf von Funktionen (Seite 1389)
- Bausteinaufrufe manuell einfügen (Seite 1393)
- Bausteinaufrufe per Drag & Drop einfügen (Seite 1394)

## Bausteinaufrufe manuell einfügen

Sie können Aufrufe für Funktionen (FCs) und Funktionsbausteine (FBs) einfügen.

### Aufruf für eine Funktion (FC) einfügen

Um einen Funktionsaufruf einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie den Funktionsnamen ein.
2. Bestätigen Sie Ihre Eingabe mit der Eingabetaste.  
Die Syntax für den Funktionsaufruf einschließlich der Parameterliste wird in das SCL-Programm eingefügt. Die Platzhalter für die Aktualparameter sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
3. Ersetzen Sie den Platzhalter durch einen Aktualparameter. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablen-tabelle per Drag & Drop auf den Platzhalter ziehen.
4. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Aktualparameter.

### Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie den Namen des Funktionsbausteins ein.
2. Bestätigen Sie Ihre Eingabe mit der Eingabetaste.  
Der Dialog "Aufrufoptionen" wird geöffnet.
3. Geben Sie im Dialog an, ob Sie den Baustein als Einzel- oder Multiinstanz aufrufen wollen.
  - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
  - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
4. Bestätigen Sie Ihre Eingaben mit "OK".  
Die Syntax für den Aufruf des Funktionsbausteins einschließlich der Parameterliste wird in das SCL-Programm eingefügt. Die Platzhalter für die Aktualparameter sind gelb hinterlegt. Der erste Platzhalter ist selektiert.

5. Ersetzen Sie den Platzhalter durch einen Aktualparameter. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablen-tabelle per Drag & Drop auf den Platzhalter ziehen.
6. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Aktualparameter.

## Ergebnis

Der Bausteinaufruf wird eingefügt.

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt.

## Siehe auch

- Bausteinaufrufe aktualisieren (Seite 1395)
- Parameterliste erweitern und reduzieren (Seite 1396)
- Grundlagen zum Bausteinaufruf in SCL (Seite 1388)

## Bausteinaufrufe per Drag & Drop einfügen

Sie können Aufrufe für bestehende Funktionen (FCs) und Funktionsbausteine (FBs) per Drag & Drop aus der Projektnavigation einfügen.

## Voraussetzung

Die aufzurufende Funktion (FC) oder der aufzurufende Funktionsbaustein (FB) ist vorhanden.

## Aufruf für eine Funktion (FC) einfügen

Um einen Funktionsaufruf per Drag & Drop einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie die Funktion aus der Projektnavigation in das Programm.  
Die Syntax für den Funktionsaufruf einschließlich der Parameterliste wird in das SCL-Programm eingefügt. Die Platzhalter für die Aktualparameter sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
2. Ersetzen Sie den Platzhalter durch einen Aktualparameter. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablen-tabelle per Drag & Drop auf den Platzhalter ziehen.
3. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Aktualparameter.



## Aufruf für einen Funktionsbaustein (FB) einfügen

Um den Aufruf für einen Funktionsbaustein (FB) per Drag & Drop einzufügen, gehen Sie folgendermaßen vor:

1. Ziehen Sie den Funktionsbaustein aus der Projektnavigation in das Programm.  
Der Dialog "Aufrufoptionen" wird geöffnet.
2. Geben Sie im Dialog an, ob Sie den Baustein als Einzel- oder Multiinstanz aufrufen wollen.
  - Wenn Sie auf die Schaltfläche "Einzel-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name" einen Namen für den Datenbaustein ein, der dem Aufruf zugeordnet werden soll.
  - Wenn Sie auf die Schaltfläche "Multi-Instanz" klicken, geben Sie anschließend im Eingabefeld "Name in der Schnittstelle" den Namen der Variablen ein, mit dem der aufgerufene Funktionsbaustein als statische Variable in die Schnittstelle des aufrufenden Bausteins eingetragen wird.
3. Bestätigen Sie Ihre Eingaben mit "OK".  
Die Syntax für den Aufruf des Funktionsbausteins einschließlich der Parameterliste wird in das SCL-Programm eingefügt. Die Platzhalter für die Aktualparameter sind gelb hinterlegt. Der erste Platzhalter ist selektiert.
4. Ersetzen Sie den Platzhalter durch einen Aktualparameter. Sie können auch eine Variable aus der Schnittstelle oder der PLC-Variablen-tabelle per Drag & Drop auf den Platzhalter ziehen.
5. Navigieren Sie mit der Taste <TAB> nacheinander zu allen weiteren Platzhaltern und ersetzen Sie sie durch Aktualparameter.

## Ergebnis

Der Bausteinaufruf wird eingefügt.

Wenn Sie beim Aufruf eines Funktionsbausteins einen Instanz-Datenbaustein angeben, der noch nicht vorhanden ist, wird dieser erzeugt.

## Siehe auch

Bausteinaufrufe aktualisieren (Seite 1395)

Parameterliste erweitern und reduzieren (Seite 1396)

Grundlagen zum Bausteinaufruf in SCL (Seite 1388)

## Bausteinaufrufe aktualisieren

Wenn sich Schnittstellenparameter eines aufgerufenen Bausteins ändern, kann der Bausteinaufruf nicht korrekt ausgeführt werden. Solche inkonsistenten Bausteinaufrufe vermeiden Sie, indem Sie die Bausteinaufrufe aktualisieren.

Sie haben zwei Möglichkeiten, Bausteinaufrufe zu aktualisieren:

- Explizites Aktualisieren im Programmiereditor.  
Die Inkonsistenzen innerhalb des geöffneten Bausteins werden angezeigt und können aktualisiert werden.
- Implizites Aktualisieren während des Übersetzens.  
Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen werden aktualisiert.

### Bausteinaufrufe im Programmiereditor aktualisieren

Um in einem Baustein einen Bausteinaufruf zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie in der Funktionsleiste auf "Inkonsistente Bausteinaufrufe aktualisieren".  
Inkonsistente Aufrufe werden angezeigt.
3. Korrigieren Sie die Inkonsistenzen.

### Bausteinaufrufe während des Übersetzens aktualisieren

Um alle Bausteinaufrufe und Verwendungen von PLC-Datentypen während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

### Siehe auch

Bausteinaufrufe manuell einfügen (Seite 1393)

Bausteinaufrufe per Drag & Drop einfügen (Seite 1394)

### Parameterliste erweitern und reduzieren

Wenn Sie in SCL Bausteine aufrufen oder Anweisungen einfügen, die systemintern Funktionsbausteine sind, wird die Syntax einschließlich der Parameterliste mit den Platzhaltern für die Aktualparameter in das SCL-Programm eingefügt. Um die Übersichtlichkeit des Programmcodes zu erhöhen, wird die Parameterliste um die unbeschalteten optionalen Parameter reduziert, wenn Sie andere Anweisungen bearbeiten. Sie können Sie jederzeit wieder erweitern. Genauso können Sie die Parameterliste aber auch explizit reduzieren, wenn Sie mit der Beschaltung der Parameter fertig sind.

## Parameterliste erweitern

Um die Parameterliste zu erweitern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Bausteinaufruf oder die Anweisung.
2. Wählen Sie im Kontextmenü den Befehl "Parameterliste erweitern" oder verwenden Sie die Tastenkombination <Strg+Shift+Leertaste>. Die Parameterliste wird wieder vollständig angezeigt.

## Parameterliste reduzieren

Um die Parameterliste zu reduzieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in den Bausteinaufruf oder die Anweisung.
2. Wählen Sie im Kontextmenü den Befehl "Parameterliste reduzieren" oder verwenden Sie die Tastenkombination <Strg+Shift+Leertaste>. Alle unbeschalteten optionalen Parameter werden ausgeblendet.

## Siehe auch

SCL-Anweisungen manuell eingeben (Seite 1381)

SCL-Anweisungen über Task Card "Anweisungen" einfügen (Seite 1382)

Bausteinaufrufe manuell einfügen (Seite 1393)

Bausteinaufrufe per Drag & Drop einfügen (Seite 1394)

## Kommentare einfügen

### Programmcode kommentieren

Sie haben verschiedene Möglichkeiten, um SCL-Programme zu kommentieren:

- Zeilenkommentar  
Ein Zeilenkommentar wird mit "//" eingeleitet und erstreckt sich bis zum Ende der Zeile.
- Kommentarabschnitt  
Ein Kommentarabschnitt wird mit "(\*" eingeleitet und durch "\*)" abgeschlossen. Er kann sich über mehrere Zeilen erstrecken.

### Zeilenkommentar einfügen

Um einen Zeilenkommentar einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie "//" an der Stelle ein, an der Sie den Kommentar platzieren wollen. Dies muss nicht der Zeilenbeginn sein.
2. Geben Sie den Kommentartext ein.

### Kommentarabschnitt einfügen

Um einen Kommentarabschnitt einzufügen, gehen Sie folgendermaßen vor:

1. Geben Sie "(\*" an der Stelle ein, an der Sie den Kommentar platzieren wollen. Dies muss nicht der Zeilenbeginn sein.
2. Geben Sie den Kommentartext ein.
3. Schließen Sie den Kommentar mit "\*)" ab.

### Eine oder mehrere Zeilen durch Kommentieren deaktivieren

Um Programmcode durch Kommentieren zu deaktivieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Codezeilen, die Sie deaktivieren wollen.
2. Klicken Sie im Editor auf die Schaltfläche "Code deaktivieren".  
In die markierten Zeilen wird am Zeilenbeginn "/" eingefügt. Der nachfolgende Code wird als Kommentar interpretiert. Werden Zeilen deaktiviert, die bereits einen Zeilenkommentar enthalten, wird ebenfalls "/" eingefügt. Werden die Zeilen anschließend wieder aktiviert, bleiben ursprüngliche Kommentare erhalten.

### Kommentarzeilen aktivieren

Um auskommentierte Zeilen wieder als Code zu aktivieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Codezeilen, die Sie aktivieren wollen.
2. Klicken Sie im Editor auf die Schaltfläche "Code aktivieren".  
Die Kennzeichnung "/" für Zeilenkommentare am Beginn der Zeile wird entfernt.

### Beispiel

Der folgende Code enthält Kommentarabschnitte und Zeilenkommentare

```
(*****  
  Hier kann eine Beschreibung der folgenden Anweisungen stehen  
  *****)  
IF "MyVal1" > 0 THEN //Keine Division durch 0  
  "MyReal" := "MyVal2" (* Eingabewert *) / "MyVal1" (* Messwert *);  
END_IF;  
//Datentypkonvertierung  
"MyInt" := REAL_TO_INT("MyReal");
```

### SCL-Anweisungen bearbeiten

#### Anweisungen selektieren

Sie können einzelne Anweisungen oder alle Anweisungen eines Bausteins selektieren.

## Voraussetzung

Ein SCL-Baustein ist geöffnet.

## Einzelne Anweisungen selektieren

Um einzelne Anweisungen zu selektieren, gehen Sie folgendermaßen vor:

1. Setzen Sie die Einfügemarke vor das erste Zeichen, das Sie markieren möchten.
2. Drücken und halten Sie die linke Maustaste.
3. Bewegen Sie den Mauszeiger hinter das letzte Zeichen, das Sie markieren möchten.
4. Lassen Sie die linke Maustaste los.

## Alle Anweisungen eines Programms selektieren

Um alle Anweisungen zu selektieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Bearbeiten" den Befehl "Alles auswählen" oder verwenden Sie die Tastenkombination <Strg+A>.

---

### Hinweis

Wenn Sie das öffnende Schlüsselwort einer Anweisung markieren, wird automatisch auch das schließende Schlüsselwort markiert.

---

## Anweisungen kopieren, ausschneiden und einfügen

### Anweisung kopieren

Um eine Anweisung zu kopieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anweisung, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

### Anweisung ausschneiden

Um eine Anweisung zu ausschneiden, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anweisung, die Sie ausschneiden möchten.
2. Wählen Sie im Kontextmenü den Befehl "Ausschneiden".

### Anweisung aus der Zwischenablage einfügen

Um eine Anweisung aus der Zwischenablage einzufügen, gehen Sie folgendermaßen vor:

1. Kopieren Sie eine Anweisung oder schneiden Sie eine Anweisung aus.
2. Klicken Sie auf die Stelle, an der Sie die Anweisung einfügen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Einfügen".

### Anweisungen löschen

#### Voraussetzung

Ein SCL-Baustein ist geöffnet

#### Vorgehen

Um eine Anweisung zu löschen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Anweisung, die Sie löschen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Syntaxfehler im Programm beheben

#### Grundlagen zu Syntaxfehlern

#### Syntaxfehler

Als Syntaxfehler werden z. B. folgende Fehler bezeichnet:

- Fehlende Trennzeichen oder Verwendung von zu vielen Trennzeichen
- Falsche Schreibweise von Schlüsselwörtern
- Falsche Schreibweise von Sprungmarken
- Nicht zur eingestellten Mnemonik passende Schreibweise (z. B. "I2.3" statt "E2.3")
- Verwendung von Schlüsselwörtern als Operanden

#### Kennzeichnung von Syntaxfehler

Syntaxfehler werden mit roter Unterstreichung oder mit rotem Text hervorgehoben.

Durch die Kennzeichnung können Sie fehlerhafte Eingaben auf einen Blick erkennen und von Fehler zu Fehler springen, um sie zu beheben. Zusätzlich werden Syntaxfehler im Inspektorfenster im Register "Info" mit einer Fehlermeldung aufgelistet.

## Siehe auch

Syntaxfehler im Programm finden (Seite 1401)

## Syntaxfehler im Programm finden

### Vorgehen

Um die Syntaxfehler im Programm zu finden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Position im Programm, ab der Sie nach Fehlern suchen möchten.
2. Klicken Sie in der Funktionsleiste auf "Gehe zu nächstem Fehler".  
Die erste fehlerhafte Stelle nach der selektieren Position wird markiert.

Sie können über die Funktionsleiste mit "Gehe zu nächstem Fehler" und "Gehe zu vorherigem Fehler" alle fehlerhaften Stellen im Baustein auffinden und korrigieren.

Oder:

1. Öffnen Sie die Fehlerliste im Inspektorfenster unter "Info > Syntax".  
Alle Syntaxfehler werden in der Tabelle mit einer kurzen Fehlerbeschreibung aufgelistet.
2. Wenn vorhanden, klicken Sie auf das blaue Fragezeichen neben dem Fehlertext, um zusätzliche Hinweise zur Fehlerbehebung zu erhalten.
3. Doppelklicken Sie auf den Fehler, den Sie korrigieren möchten.  
Der entsprechende Fehler wird markiert.

## Siehe auch

Grundlagen zu Syntaxfehlern (Seite 1400)

## Programmiersprache umschalten

### Regeln zum Umschalten der Programmiersprache

#### Regeln

Beachten Sie folgende Regeln, wenn Sie die Programmiersprache für einen Baustein umstellen möchten:

- Alle CPU-Familien:
  - Sie können nur ganze Bausteine umschalten. Einzelne Netzwerke können nicht umgeschaltet werden.
  - Bausteine, die in den Programmiersprachen SCL oder GRAPH programmiert werden, können Sie nicht umschalten. Bei GRAPH-Bausteinen können Sie aber bei der Sprache der Netzwerke zwischen KOP und FUP wechseln.
- S7-300/400:
  - Sie können zwischen den Programmiersprachen KOP, FUP und AWL umschalten.
  - Sie können Netzwerke in einem Baustein mit einer anderen Programmiersprache erstellen und anschließend in den gewünschten Baustein kopieren.
  - Einzelne Netzwerke des Bausteins, die nicht umgeschaltet werden können, werden in der Originalsprache angezeigt.
- S7-1200/1500:
  - Sie können zwischen den Programmiersprachen KOP und FUP umschalten.
- S7-1500:
  - Sie können innerhalb von KOP- und FUP-Bausteinen AWL-Netzwerke erstellen. Das Kopieren zwischen AWL und KOP/FUP ist jedoch nicht möglich.

## Programmiersprache umschalten

#### Vorgehen

Um die Programmiersprache zu ändern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation den Baustein mit der rechten Maustaste an.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Ein Dialog mit den Eigenschaften des Bausteins wird geöffnet.
3. Wählen Sie in der Bereichsnavigation den Eintrag "Allgemein".
4. Wählen Sie in der Klappliste "Sprache" die neue Programmiersprache aus.
5. Bestätigen Sie Ihre Auswahl mit "OK".

#### Siehe auch

Regeln zum Umschalten der Programmiersprache (Seite 1402)



## Programmablauffehler behandeln

### Grundlagen zur Fehlerbehandlung

#### Einführung

Unter Programmablauffehler werden Programmier- und Peripheriezugriffsfehler verstanden. Um auf Programmablauffehler zu reagieren, haben Sie je nach eingesetzter CPU unterschiedliche Möglichkeiten.

#### Behandlung von Programmablauffehlern für S7-300/400

Für die S7-300/400-CPU's können Sie den Programmablauffehler-OB (OB 85) programmieren. Falls ein Programmablauffehler auftritt und Sie den Programmablauffehler-OB nicht verwenden, wechselt die CPU in den Betriebszustand "STOP".

Weitere Informationen über den Programmablauffehler-OB finden Sie in der Beschreibung der Arbeitsweise von S7-300/400-CPU's.

## Behandlung von Programmablauffehlern für S7-1200/1500

Für CPUs der Baureihen S7-1200 und S7-1500 können Sie die Art der Fehlerbehandlung auswählen. Sie haben die folgenden beiden Möglichkeiten:

- Sie verwenden die globale Fehlerbehandlung der CPU:
  - S7-1200: Die CPU erzeugt einen Diagnosepuffereintrag und verbleibt im Betriebszustand "RUN".
  - S7-1500: Für die S7-1500-CPU's können Sie den Programmierfehler-OB (OB 121) und den Peripheriezugriffsfehler-OB (OB 122) programmieren. Falls der Programmierfehler-OB in der CPU nicht vorhanden ist, wechselt die CPU bei einem Programmierfehler in den Betriebszustand "STOP". Bei einem Peripheriezugriffsfehler bleibt die CPU immer im Betriebszustand "RUN", unabhängig davon, ob der Peripheriezugriffsfehler-OB vorhanden ist.  
Beachten Sie jedoch, dass ein vorhandener Peripheriezugriffs- oder Programmierfehler-OB nicht synchron zum Fehler aufgerufen wird. Daher werden Peripheriezugriffs- oder Programmierfehler-OB möglicherweise nicht sofort beim Auftreten des Fehlers, sondern abhängig von der eingestellten Priorität erst verzögert bearbeitet. Wenn weitere Fehler auftreten, bevor der Peripheriezugriffs- oder Programmierfehler-OB fertig bearbeitet worden ist, dann wird kein weiterer Peripheriezugriffs- oder Programmierfehler-OB aufgerufen. Möchten Sie verhindern, dass Peripheriezugriffs- oder Programmierfehler-OBs verworfen werden, stellen Sie die Priorität entsprechend hoch ein.  
Für die Anweisungen "Feld lesen" (FieldRead), "Feld schreiben" (FieldWrite), "Speicheradresse lesen" (PEEK) und "Speicheradresse schreiben" (POKE) können Sie den Freigabeausgang ENO verwenden, um Peripheriezugriffs- und Programmierfehler zu erkennen.  
Weitere Informationen über diese Fehler-OBs finden Sie in der Beschreibung der Arbeitsweise von S7-1500-CPU's.
- Sie verwenden eine eigene lokale Fehlerbehandlung. Dabei handelt es sich um eine Fehlerbehandlung innerhalb eines Bausteins. Die lokale Fehlerbehandlung hat folgende Vorteile:
  - Im Systemspeicher wird eine Fehlerinformation gespeichert, die Sie abfragen und auswerten können.
  - Mithilfe der Fehlerinformation können Sie im Baustein eine Reaktion auf den aufgetretenen Fehler programmieren.
  - Programmierte Fehlerauswertungen und -reaktionen unterbrechen den Programmzyklus nicht.
  - Die Systemperformance wird durch die lokale Fehlerbehandlung nicht unnötig belastet. Wenn keine Fehler auftreten, werden programmierte Fehlerauswertungen und Reaktionen nicht ausgeführt.  
Die lokale Fehlerbehandlung gilt nur für Bausteine, für die sie explizit eingestellt wurde. Wenn für einen Baustein eine lokale Fehlerbehandlung eingestellt ist, wird für Fehler in diesem Baustein keine globale Fehlerbehandlung durchgeführt.

**Hinweis**

Beachten Sie folgende Hinweise:

- Alle Speicherzugriffsfehler und Peripheriezugriffsfehler müssen entweder durch eine globale oder eine lokale Fehlerbehandlung abgefangen werden.
  - Wenn durch die Parameter einer Anweisung keine Speicherzugriffsfehler aufgetreten sind, können Sie das zugehörige ENO abfragen.
- 

**Siehe auch**

GET\_ERROR: Fehler lokal abfragen (Seite 2099)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

GET\_ERROR: Fehler lokal abfragen (Seite 1828)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

**Lokale Fehlerbehandlung****Grundlagen der lokalen Fehlerbehandlung****Einführung**

Die lokale Fehlerbehandlung bietet die Möglichkeit, das Auftreten von Fehlern innerhalb eines Bausteins abzufragen und die dazugehörigen Fehlerinformationen auszuwerten. Die lokale Fehlerbehandlung können Sie für Organisationsbausteine (OB), Funktionsbausteine (FB) und Funktionen (FC) einstellen. Wenn die lokale Fehlerbehandlung aktiviert ist, wird die Systemreaktion ignoriert.

Die lokale Fehlerbehandlung gilt nur für Bausteine, für die sie explizit eingestellt ist. Die Einstellung der lokalen Fehlerbehandlung wird von einem aufrufenden Baustein nicht übernommen und auch nicht auf aufgerufene Bausteine übertragen. Für übergeordnete und untergeordnete Bausteine gelten weiterhin die Systemeinstellungen, sofern für diese keine eigene Fehlerbehandlung programmiert ist.

**Allgemeiner Ablauf der lokalen Fehlerbehandlung**

Wenn Fehler während der Bearbeitung eines Bausteins mit lokaler Fehlerbehandlung auftreten, wird eine voreingestellte Reaktion abhängig von den folgenden Fehlerarten ausgelöst:

- Schreibfehler: Die Fehler werden ignoriert und die Bearbeitung des Programms einfach fortgesetzt.
- Lesefehler: Die Bearbeitung des Programms wird mit dem Ersatzwert "0" fortgesetzt.
- Ausführungsfehler: Die Bearbeitung der Anweisung wird unterbrochen. Die Bearbeitung des Programms wird mit der nächsten Anweisung fortgesetzt.

Zum ersten aufgetretenen Fehler wird eine Information im Systemspeicher gespeichert. Diese Information kann mithilfe einer Anweisung (GET\_ERROR oder GET\_ERR\_ID) abgefragt und ausgegeben werden. Dabei wird die Fehlerinformation in einer Form ausgegeben, die weiter bearbeitet werden kann. Mithilfe weiterer Anweisungen können Sie die Fehlerinformation auswerten und eine Reaktion auf den aufgetretenen Fehler programmieren.

Wenn die Information zum ersten aufgetretenen Fehler abgefragt wird, wird der Fehler-Speicherplatz im Systemspeicher freigegeben. Wenn dann weitere Fehler auftreten, wird die Information zum nächsten aufgetretenen Fehler ausgegeben.

### Anweisungen zur lokalen Fehlerbehandlung

Zur lokalen Fehlerbehandlung können Sie eine der folgenden Anweisungen einsetzen:

- GET\_ERROR: Fehler lokal abfragen
- GET\_ERR\_ID: Fehler-ID lokal abfragen

Die Anweisungen unterscheiden sich in der Tiefe der ausgegebenen Fehlerinformation.

Weitere Informationen zu den Anweisungen finden Sie unter "Siehe auch".

### Siehe auch

GET\_ERROR: Fehler lokal abfragen (Seite 2099)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

GET\_ERROR: Fehler lokal abfragen (Seite 1828)

### Prioritäten bei der Fehlerausgabe

#### Übersicht über die Prioritäten

Bei der lokalen Fehlerbehandlung wird die Information zum ersten aufgetretenen Fehler ausgegeben. Wenn während der Bearbeitung einer Anweisung mehrere Fehler gleichzeitig auftreten, werden die Fehler nach ihrer Priorität ausgegeben. Die folgende Tabelle zeigt, welche Priorität die verschiedenen Fehlerarten haben:

| Priorität | Fehlerart  |
|-----------|--|
| 1         | Fehler im Programmcode                           |
| 2         | Fehlende Referenz                                |
| 3         | Ungültiger Bereich                               |
| 4         | DB existiert nicht                               |
| 5         | Operanden nicht kompatibel                       |
| 6         | Breite des angegebenen Bereichs reicht nicht aus |
| 7         | Zeiten oder Zähler existieren nicht              |
| 8         | Kein Schreibzugriff auf einen DB                 |
| 9         | Peripheriefehler                                 |

| Priorität | Fehlerart                    |
|-----------|------------------------------|
| 10        | Anweisung existiert nicht    |
| 11        | Baustein existiert nicht     |
| 12        | Ungültige Schachtelungstiefe |

Die höchste Priorität ist 1 und die niedrigste Priorität ist 12.

### Siehe auch

GET\_ERROR: Fehler lokal abfragen (Seite 2099)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

GET\_ERROR: Fehler lokal abfragen (Seite 1828)

### Lokale Fehlerbehandlung für einen Baustein aktivieren

#### Einführung

Die lokale Fehlerbehandlung wird für einen Baustein aktiviert, wenn Sie eine der folgenden Anweisungen in ein Netzwerk einfügen:

- GET\_ERROR: Fehler lokal abfragen
- GET\_ERR\_ID: Fehler-ID lokal abfragen

Weitere Informationen zu den Anweisungen finden Sie unter "Siehe auch".

Wenn die lokale Fehlerbehandlung für einen Baustein aktiviert ist, werden die Systemreaktionen für diesen Baustein ignoriert.

#### Voraussetzung

- Der Baustein ist geöffnet.
- Die Task Card "Anweisungen" ist geöffnet.

#### Vorgehen

Um die lokale Fehlerbehandlung für einen Baustein zu aktivieren, gehen Sie folgendermaßen vor:

1. Navigieren Sie in der Task Card "Anweisungen" zur Palette "Einfache Anweisungen".
2. Öffnen Sie den Ordner "Programmsteuerung".
3. Ziehen Sie die Anweisung "Fehler lokal abfragen" (GET\_ERROR) oder "Fehler-ID lokal abfragen" (GET\_ERR\_ID) in das gewünschte Netzwerk.

## Ergebnis

Die lokale Fehlerbehandlung wird für den geöffneten Baustein aktiviert. Im Inspektorfenster wird unter "Eigenschaften > Attribute" das Optionskästchen "Lokale Fehlerbehandlung im Baustein" aktiviert. Die Einstellung kann im Inspektorfenster nicht editiert werden. Die lokale Fehlerbehandlung kann durch das Löschen der eingefügten Anweisungen zur lokalen Fehlerbehandlung deaktiviert werden.

## Siehe auch

GET\_ERROR: Fehler lokal abfragen (Seite 2099)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

GET\_ERROR: Fehler lokal abfragen (Seite 1828)

### 9.1.4.3 Datenbausteine programmieren

#### Grundlagen zur Programmierung von Datenbausteinen

Ein Datenbaustein (DB) dient dazu, Werte zu speichern, die während des Programmablaufs geschrieben werden.

Im Gegensatz zum Codebaustein enthält der Datenbaustein nur Variablendeklarationen. Er enthält keine Netzwerke oder Anweisungen. Die Variablendeklarationen definieren die Struktur des Datenbausteins.

#### Arten von Datenbausteinen

Es gibt zwei Arten von Datenbausteinen:

- **Globale Datenbausteine**  
Der globale Datenbaustein ist keinem Codebaustein zugeordnet. Sie können von jedem Codebaustein aus auf die Werte eines globalen Datenbausteins zugreifen. Ein globaler Datenbaustein enthält ausschließlich statische Variablen.  
Die Struktur des globalen Datenbausteins ist frei definierbar. Sie deklarieren die Datenelemente, die der globale Datenbaustein enthalten soll, in der Deklarationstabelle für Datenbausteine.
- **Instanz-Datenbausteine**  
Der Instanz-Datenbaustein ist einem Funktionsbaustein (FB) unmittelbar zugeordnet. Die Struktur eines Instanz-Datenbausteins ist nicht frei definierbar, sondern wird durch die Schnittstellendeklaration des Funktionsbausteins bestimmt. Der Instanz-Datenbaustein enthält genau die Bausteinparameter und Variablen, die dort deklariert wurden.  
Sie können jedoch im Instanz-Datenbaustein instanzspezifische Werte definieren, z. B. Startwerte für die deklarierten Variablen.

#### ARRAY-Datenbausteine (S7-1500)

ARRAY-Datenbausteine sind globale Datenbausteine, die aus einem ARRAY bestehen. Dieses ARRAY kann auf einem beliebigen Datentypen basieren. Es kann z. B. ein ARRAY

eines PLC-Datentyps (UDT) sein. Neben dem ARRAY enthält der DB keine weiteren Elemente. ARRAY-Datenbausteine erleichtern durch ihre flache Struktur den Zugriff auf die ARRAY-Elemente sowie deren Übergabe an aufgerufene Bausteine.

Möglichkeiten zur Adressierung von ARRAY-DBs bietet die Task Card "Anweisungen" im Abschnitt "Verschieben".

### PLC-Datentypen als Vorlage für globale Datenbausteine

PLC-Datentypen können als Vorlage für die Erstellung von globalen Datenbausteinen mit gleicher Datenstruktur dienen. Sie erstellen nur einmal die Struktur als PLC-Datentyp und erzeugen anschließend die benötigten Datenbausteine durch Zuweisung des PLC-Datentyps.

### Systemdatentypen als Vorlage für globale Datenbausteine

Systemdatentypen können ebenfalls als Vorlage für die Erstellung von globalen Datenbausteinen mit gleicher Datenstruktur dienen. Systemdatentypen haben bereits eine fest definierte Struktur. Sie fügen den Systemdatentyp nur einmal in das Programm ein und erzeugen anschließend weitere Datenbausteine mit derselben Struktur durch Zuweisung des Systemdatentyps.

### Zugriffsarten

Es gibt zwei unterschiedliche Arten, auf Datenwerte in Datenbausteinen zuzugreifen:

- Datenbausteine mit optimiertem Zugriff (nur S7-1200)  
Datenbausteine mit optimiertem Zugriff haben keine fest definierte Struktur. Die Datenelemente erhalten in der Deklaration nur einen symbolischen Namen, keine feste Adresse innerhalb des Bausteins. Auf Datenwerte in diesen Bausteinen greifen Sie über den symbolischen Namen zu.  
Für ARRAY-Datenbausteine ist das Attribut "Optimierter Bausteinzugriff" immer aktiviert.
- Datenbausteine mit Standardzugriff (alle CPU-Familien)  
Datenbausteine mit Standardzugriff haben eine feste Struktur. Die Datenelemente enthalten in der Deklaration sowohl einen symbolischen Namen als auch eine feste Adresse innerhalb des Bausteins. Auf Datenwerte in diesen Bausteinen können Sie über den symbolischen Namen oder die Adresse zugreifen.  
ARRAY-Datenbausteine mit Standardzugriff sind nicht möglich.

### Remanenz der Datenwerte

Um Datenverluste bei Spannungsausfall zu vermeiden, können Sie die Datenwerte in einem remanenten Speicherbereich ablegen.

### Siehe auch

Datenbausteine anlegen (Seite 1214)

Globale Datenbausteine (DB) (Seite 1030)

Instanz-Datenbausteine (Seite 1032)

### Aufbau der Deklarationstabelle für Datenbausteine

### Aufbau der Deklarationstabelle für Datenbausteine

Das folgende Bild zeigt den Aufbau der Deklarationstabelle für Datenbausteine. Die Darstellung variiert je nach Baustein- und Zugriffsart.

| Name      | Datentyp | Startwert | Remanenz                            | Sichtbar in HMI                     | Kommentar |
|-----------|----------|-----------|-------------------------------------|-------------------------------------|-----------|
| ▼ Input   |          |           |                                     |                                     |           |
| MyInput1  | Bool     | false     | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |           |
| ▼ Output  |          |           |                                     |                                     |           |
| MyOutput1 | Byte     | 0         | <input type="checkbox"/>            | <input checked="" type="checkbox"/> |           |
| ▼ InOut   |          |           |                                     |                                     |           |
| ▼ Static  |          |           |                                     |                                     |           |

### Darstellung instanzspezifischer Werte

In Instanz-Datenbausteinen können Sie bereits definierte Werte aus der Schnittstelle des zugeordneten Funktionsbausteins übernehmen oder instanzspezifische Startwerte definieren. Werte, die aus dem Funktionsbaustein übernommen sind, werden grau dargestellt. Die grauen Werte können Sie durch instanzspezifische Werte ersetzen. Werte, die bereits instanzspezifisch geändert wurden, sind nicht grau.

### Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Spalten können Sie bei Bedarf ein- oder ausblenden. Die Anzahl der angezeigten Spalten variiert abhängig von der CPU-Familie.

| Spalte      | Erläuterung  |
|-------------|--|
|             | Symbol, das Sie anklicken können, um die Variable zu verschieben oder zu kopieren. Sie können sie z. B. per Drag & Drop in ein Programm ziehen und dort als Operand verwenden.   |
| Name        | Name der Variablen.  |
| Datentyp    | Datentyp der Variablen.  |
| Offset      | Relative Adresse der Variablen.<br>Diese Spalte ist nur in Datenbausteinen mit Standardzugriff sichtbar.   |
| Defaultwert | Wert, mit dem die Variable in der Schnittstelle eines übergeordneten Codebausteins oder einem PLC-Datentyp vorbelegt wurde.<br>Die in der Spalte "Defaultwerte" enthaltenen Werte sind nur im übergeordneten Codebaustein oder PLC-Datentyp änderbar. Im Datenbaustein werden die Werte nur angezeigt. |



| Spalte             | Erläuterung  |
|--------------------|--|
| Startwert          | Wert, den die Variable beim Anlauf einnehmen soll.<br>Beim Anlegen des Datenbausteins werden die in einem Codebaustein definierten Defaultwerte als Startwerte verwendet. Diese übernommenen Werte können Sie hier mit instanzspezifischen Startwerten ersetzen.<br>Die Angabe eines Startwerts ist optional. Wenn Sie keinen Wert festlegen, nimmt die Variable beim Anlauf den Defaultwert ein. Ist auch kein Defaultwert definiert, wird der für den Datentyp gültige Standardwert verwendet. Für BOOL ist z. B. der Standardwert "FALSE" festgelegt. |
| Beobachtungswert   | Aktueller Datenwert in der CPU.<br>Die Spalte wird eingeblendet, wenn eine Online-Verbindung besteht und Sie die Schaltfläche "Beobachten" anwählen.   |
| Momentaufnahme     | Zeigt Werte an, die aus dem Gerät geladen wurden.  |
| Remanenz           | Kennzeichnet die Variable als remanent. Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten.   |
| Sichtbar in HMI    | Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.   |
| Erreichbar aus HMI | Zeigt an, ob HMI zur Laufzeit auf diese Variable zugreifen kann.   |
| Einstellwert       | Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen. Nach der Inbetriebnahme können die Werte dieser Variablen als Startwerte in das Offline-Programm übertragen und dort gespeichert werden.  |
| Kommentar          | Kommentar zur Dokumentation der Variablen.   |

### Siehe auch

Datenbausteine anlegen (Seite 1214)

Grundlagen zu Startwerten (Seite 1419)

### Datenbausteine anlegen

#### Voraussetzung

Der Ordner "Programmbausteine" in der Projektnavigation ist geöffnet.

#### Vorgehen

Um einen Datenbaustein anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neuen Baustein hinzufügen".  
Der Dialog "Neuen Baustein hinzufügen" wird geöffnet.
2. Klicken Sie auf die Schaltfläche "Datenbaustein (DB)".

3. Wählen Sie den Typ des Datenbausteins aus. Folgende Möglichkeiten stehen zur Verfügung:
  - Um einen globalen Datenbaustein anzulegen, wählen Sie den Listeneintrag "Global-DB".
  - Um einen ARRAY-Datenbaustein anzulegen, wählen Sie den Listeneintrag "ARRAY-DB".
  - Um einen Instanz-Datenbaustein anzulegen, wählen Sie aus der Liste den Funktionsbaustein aus, dem Sie den Instanz-Datenbaustein zuordnen möchten. In der Liste werden nur Funktionsbausteine angeboten, die zuvor für die CPU erstellt wurden.
  - Um einen Datenbaustein basierend auf einem PLC-Datentyp anzulegen, wählen Sie aus der Liste den PLC-Datentyp aus. In der Liste werden nur PLC-Datentypen angeboten, die zuvor für die CPU erstellt wurden.
  - Um einen Datenbaustein basierend auf einem Systemdatentyp anzulegen, wählen Sie aus der Liste den Systemdatentyp aus. In der Liste werden nur Systemdatentypen angeboten, die zuvor in Programmbausteinen der CPU eingefügt wurden.
4. Geben Sie einen Namen für den Datenbaustein ein.
5. Geben Sie die Eigenschaften des neuen Datenbausteins ein.
6. Wenn Sie als "Typ" einen ARRAY-DB ausgewählt haben, geben Sie den ARRAY-Datentyp und die obere ARRAY-Grenze ein.  
Die obere ARRAY-Grenze können Sie jederzeit im Eigenschaftsfenster des erzeugten Bausteins ändern. Der ARRAY-Datentyp ist nachträglich nicht mehr änderbar.
7. Um weitere Eigenschaften des neuen Datenbausteins einzugeben, klicken Sie auf "Weitere Informationen".  
Ein Bereich mit weiteren Eingabefeldern wird angezeigt.
8. Geben Sie alle gewünschten Eigenschaften ein.
9. Aktivieren Sie das Optionskästchen "Neu hinzufügen und öffnen", wenn der Baustein nach dem Erstellen gleich geöffnet werden soll.
10. Bestätigen Sie Ihre Eingabe mit "OK".

## Ergebnis

Der neue Datenbaustein wird erzeugt. Sie finden den Datenbaustein in der Projektnavigation im Ordner "Programmbausteine".

## Siehe auch

Instanz-Datenbausteine (Seite 1032)

Globale Datenbausteine (DB) (Seite 1030)

Übersicht über die Bausteineigenschaften (Seite 1224)

## Datenbausteine aktualisieren

### Einführung

Änderungen in der Schnittstelle eines Funktionsbausteins oder eines PLC-Datentyps können dazu führen, dass zugehörige Datenbausteine inkonsistent werden. Diese Inkonsistenzen werden in der Deklarationstabelle und an der Aufrufstelle des Bausteins rot markiert. Um die Inkonsistenzen zu beheben, müssen Sie die Datenbausteine aktualisieren.

Sie haben drei Möglichkeiten, Datenbausteine zu aktualisieren:

- Explizites Aktualisieren in der Deklarationstabelle für Datenbausteine.  
Der Datenbaustein wird aktualisiert. Änderungen aus der Schnittstelle des zugeordneten Funktionsbausteins und Änderungen an verwendeten PLC-Datentypen werden übernommen.
- Explizites Aktualisieren im Programmiereditor.  
Die Bausteinaufrufe innerhalb des geöffneten Bausteins werden aktualisiert. Dabei werden auch die zugehörigen Instanz-Datenbausteine angepasst.
- Implizites Aktualisieren während des Übersetzens.  
Alle Bausteinaufrufe im Programm sowie die verwendeten PLC-Datentypen und die zugehörigen Instanz-Datenbausteine werden aktualisiert.

### Explizites Aktualisieren in der Deklarationstabelle für Datenbausteine

Um einen einzelnen Datenbaustein explizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Wählen Sie im Kontextmenü "Schnittstelle aktualisieren".

### Explizites Aktualisieren im Programmiereditor

Um in einem Baustein alle Bausteinaufrufe oder einen bestimmten Aufruf zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein im Programmiereditor.
2. Klicken Sie mit der rechten Maustaste auf die Anweisung mit dem Bausteinaufruf.
3. Wählen Sie im Kontextmenü den Befehl "Aktualisieren".
4. Der Dialog "Schnittstellenabgleich" wird geöffnet. In diesem Dialog werden die Unterschiede zwischen der verwendeten und der geänderten Bausteinschnittstelle des aufgerufenen Bausteins angezeigt.
5. Wenn Sie den Bausteinaufruf aktualisieren möchten, klicken Sie auf "OK". Um die Aktualisierung abzubrechen, klicken Sie auf "Abbrechen".

## Implizites Aktualisieren während des Übersetzens

Um alle Bausteinaufrufe und Verwendungen von PLC-Datentypen sowie die Instanz-Datenbausteine während des Übersetzens implizit zu aktualisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Projektnavigation.
2. Selektieren Sie den Ordner "Programmbausteine".
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

## Siehe auch

Eigenschaften von Variablen in Instanz-Datenbausteinen ändern (Seite 1426)

## Datenbausteine erweitern

### Beschreibung

Um PLC-Programme, die bereits in Betrieb genommen wurden und fehlerfrei auf einer Anlage laufen, nachträglich zu ändern, bieten CPUs der Baureihe S7-1500 und die meisten CPUs der Baureihe S7-1200 V4 die Möglichkeit, globale Datenbausteine im laufenden Betrieb zu erweitern.

Die geänderten Bausteine können geladen werden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen.

Programmänderungen lassen sich so auf einfache Weise einbringen. Der gesteuerte Prozess wird durch diesen Ladevorgang (Laden ohne Reinitialisierung) nicht gestört.

### Funktionsweise

Grundsätzlich enthält jeder Datenbaustein per Voreinstellung eine Speicherreserve. Die Speicherreserve wird zunächst nicht genutzt. Wenn Sie den Baustein übersetzt und geladen haben und anschließend feststellen, dass sie Schnittstellenänderungen nachladen wollen, aktivieren Sie die Speicherreserve. Alle Variablen, die Sie daraufhin deklarieren, werden in die Speicherreserve gelegt. Beim anschließenden Laden werden die Werte bereits geladener Variablen nicht beeinflusst.

Um Ihr Programm zu einem späteren Zeitpunkt zu überarbeiten, während die Anlage gerade nicht in Betrieb ist, haben Sie zusätzlich die Möglichkeit, das Speicherlayout einzelner oder mehrerer Bausteine in einem Schritt zu überarbeiten. Dabei werden alle Variablen aus dem Reservebereich in den regulären Bereich verschoben. Die Speicherreserve wird wieder frei für weitere Schnittstellenerweiterungen.

## Voraussetzungen

Die Funktion "Laden ohne Reinitialisierung" steht zur Verfügung, wenn folgende Voraussetzungen erfüllt sind:

- Das Projekt liegt im Format "TIA-Portal V12" vor.
- Sie arbeiten mit einer CPU , die das "Laden ohne Reinitialisierung" unterstützt.
- Die Bausteine wurden in KOP, FUP, AWL oder SCL erstellt.
- Die Bausteine wurden vom Anwender erstellt, d.h., es sind keine mitgelieferten Bausteine.
- Es handelt sich um Bausteine mit optimiertem Zugriff.

## Grundsätzliche Schritte

Um einen Datenbaustein zu erweitern und den Baustein anschließend ohne Reinitialisierung zu laden, führen Sie folgende Schritte aus.

1. Alle Bausteine haben per Voreinstellung eine Speicherreserve von 100 Byte. Bei Bedarf können Sie die Größe der Speicherreserve anpassen.
2. Aktivieren Sie die Speicherreserve.
3. Erweitern Sie die Bausteinschnittstelle.
4. Übersetzen Sie den Baustein.
5. Laden Sie den Baustein wie gewohnt in die CPU.

## Verweis

Detaillierte Beschreibungen der einzelnen Schritte finden Sie im Kapitel "Bausteine laden (S7-1200/1500) ".

## Datenstruktur von globalen Datenbausteinen erstellen

### Variablen von elementarem Datentyp deklarieren

#### Voraussetzung

Ein globaler Datenbaustein ist geöffnet.

---

#### Hinweis

Die Struktur von Instanz-Datenbausteinen und von Datenbausteinen, die auf einem PLC-Datentyp basieren, können Sie nicht direkt ändern, da die Strukturen dieser Bausteine durch den Funktionsbaustein oder den PLC-Datentyp definiert werden.

Der Typ des Datenbausteins ist in den Bausteineigenschaften eingetragen.

---

## Vorgehen

Um eine Variable elementaren Datentyps zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Klicken Sie in der Spalte "Datentyp" auf die Schaltfläche für die Datentypauswahl. Eine Liste der zulässigen Datentypen wird geöffnet.
3. Wählen Sie den gewünschten Datentyp aus.
4. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten angezeigt werden.
5. Wiederholen Sie die Schritte 1 bis 4 für alle zu deklarierenden Variablen.

## Siehe auch

Bausteineigenschaften anzeigen und bearbeiten (Seite 1229)

Variablen vom Datentyp ARRAY deklarieren (Seite 1416)

Variablen vom Datentyp STRUCT deklarieren (Seite 1417)

Bearbeiten von Tabellen (Seite 216)

## Variablen vom Datentyp ARRAY deklarieren

### Voraussetzung

Ein globaler Datenbaustein ist geöffnet.

### Vorgehen

Um eine Variable vom Datentyp ARRAY zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den Datentyp "Array" ein. Sie werden dabei von der Autovervollständigung unterstützt. Der Dialog "Array" wird geöffnet.
3. Geben Sie im Eingabefeld "Datentyp" den Datentyp der Array-Elemente an.
4. Geben Sie anschließend im Eingabefeld "Array-Grenzen" für jede Dimension die Ober- und Untergrenze an.  
Beispiel für ein eindimensionales ARRAY:  
[0..3]  
Beispiel für ein dreidimensionales ARRAY:  
[0..3, 0..15, 0..33]
5. Bestätigen Sie die Eingabe.
6. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten angezeigt werden.

## Startwerte von ARRAY-Elementen eingeben

Um die einzelnen Elemente eines ARRAYS mit Startwerten vorzubelegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp ARRAY.  
Das ARRAY wird aufgeklappt und die einzelnen ARRAY-Elemente werden in eigenen Zeilen dargestellt.
2. Tragen Sie die gewünschten Werte in die Spalte "Startwert" ein.

## Variablen vom Datentyp STRUCT deklarieren

### Voraussetzung

Ein globaler Datenbaustein ist geöffnet.

### Vorgehen

Um eine Variable vom Datentyp STRUCT zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" "Struct" ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.  
Nach der neuen Variablen wird eine leere, eingerückte Zeile eingefügt.
3. Fügen Sie in der leeren Zeile das erste Strukturelement ein.  
Nach dem Element wird erneut eine leere Zeile eingefügt.
4. Wählen Sie einen Datentyp für das Strukturelement aus.
5. Optional: Ändern Sie die Eigenschaften des Strukturelements, die in den weiteren Spalten der Bausteinschnittstelle angezeigt werden.
6. Wiederholen Sie Schritt 4 bis 7 für alle weiteren Strukturelemente.  
Ein explizites Beenden der Struktur ist nicht notwendig. Die Struktur endet mit dem letzten eingetragenen Element.
7. Um eine neue Variable nach der Struktur einzufügen, lassen Sie eine Zeile am Ende der Struktur leer und tragen Sie die neue Variable in die zweite leere Zeile ein.

### Ergebnis

Die Variable vom Datentyp STRUCT wird angelegt.

## Startwerte von Strukturelementen eingeben

Um die einzelnen Elemente einer Struktur mit Startwerten vorzubelegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Dreieck vor der Variablen vom Datentyp STRUCT.  
Die Struktur wird aufgeklappt und die einzelnen Strukturelemente werden in eigenen Zeilen dargestellt.
2. Tragen Sie die gewünschten Werte in die Spalte "Startwert" ein.

## Siehe auch

STRUCT (Seite 1119)

## Variablen basierend auf einem PLC-Datentyp deklarieren

### Voraussetzung

- Ein globaler Datenbaustein ist geöffnet.
- In der aktuellen CPU ist ein PLC-Datentyp deklariert.

### Vorgehen

Um eine Variable basierend auf einem PLC-Datentyp zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den PLC-Datentyp ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.
3. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten der Tabelle angezeigt werden.

## Ergebnis

Die Variable wird angelegt.

## Siehe auch

Aufbau der Bausteinschnittstelle (Seite 1257)



## Startwerte definieren

### Grundlagen zu Startwerten

#### Definition "Startwert"

Der Startwert einer Variablen ist ein von Ihnen definierter Wert, den die Variable nach einem Anlauf der CPU einnehmen soll.

Eine Sonderstellung nehmen die remanenten Variablen ein. Ihre Werte nehmen nur nach einem "Kaltstart" den definierten Startwert ein. Nach einem "Neustart (Warmstart)" bleiben ihre Werte erhalten und werden nicht auf den Startwert zurückgesetzt.

#### Definition "Defaultwert"

Die Struktur von Datenbausteinen kann von übergeordneten Elementen abgeleitet sein:

- Ein Instanz-Datenbaustein basiert z. B. auf der Schnittstelle eines übergeordneten Codebausteins.
- Ein globaler Datenbaustein kann auf einem vordefinierten PLC-Datentyp basieren.

In diesem Fall können Sie im übergeordneten Element einen Defaultwert für jede Variable definieren. Beim Anlegen des Datenbausteins werden diese Defaultwerte als Startwerte verwendet. Diese übernommenen Werte können Sie anschließend im Datenbaustein mit instanzspezifischen Startwerten ersetzen.

Die Angabe eines Startwerts ist optional. Wenn Sie keinen Wert festlegen, nimmt die Variable beim Anlauf den Defaultwert ein. Ist auch kein Defaultwert definiert, wird der für den Datentyp gültige Standardwert verwendet. Für BOOL ist z. B. der Standardwert "FALSE" festgelegt.

### Siehe auch

Startwerte definieren (Seite 1420)

Aufbau der Deklarationstabelle für Datenbausteine (Seite 1410)

Lokale Variablen in der Bausteinschnittstelle deklarieren (Seite 1262)

Werte aus dem Online-Programm als Startwerte übernehmen (Seite 1435)

## Startwerte definieren

### Startwerte definieren

Um Startwerte für die Variablen eines Datenbausteins festzulegen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.  
Die Spalte "Defaultwert" zeigt die Defaultwerte, die für die Variablen in der Schnittstelle eines übergeordneten Codebausteins oder in einem PLC-Datentyp definiert wurden.
2. Um alle Elemente strukturierter Datentypen einzublenden, klicken Sie auf die Schaltfläche "Erweiterter Modus".
3. Tragen Sie in die Spalte "Startwert" die gewünschten Startwerte ein. Der Wert muss dem Datentyp der Variablen entsprechen und darf den Wertebereich des Datentyps nicht überschreiten.  
Die Startwerte sind definiert. Die Variable nimmt beim Anlauf den definierten Wert ein, sofern sie nicht als remanent deklariert wurde.

### Eine Variable auf den Defaultwert zurücksetzen

Um eine Variable, für die Sie einen Startwert definiert haben, wieder auf den Defaultwert zurückzusetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle einen geänderten Wert.
2. Löschen Sie den Wert.  
Der Defaultwert wird eingetragen. Er wird grau dargestellt.

### Alle Variablen auf den Defaultwert zurücksetzen

Um alle Variablen, für die Sie einen Startwert definiert haben, wieder auf den Defaultwert zurückzusetzen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Funktionsleiste das Symbol "Startwerte zurücksetzen".  
Die Defaultwerte werden in die Spalte "Startwert" übertragen. Schreibgeschützte Startwerte werden nicht überschrieben.

### Siehe auch

Grundlagen zu Startwerten (Seite 1419)

Werte aus dem Online-Programm als Startwerte übernehmen (Seite 1435)

## Geänderte Werte laden

### Einleitung

Um geänderte Startwerte aus dem Offline-Programm in das Online-Programm zu übernehmen, müssen Sie die Änderungen laden. Dabei sind folgende Fälle zu unterscheiden:

- Geänderte Startwerte von nicht-remanenten Variablen laden
- Geänderte Startwerte von remanenten Variablen laden
- Geänderte Startwerte von Einstellwerten laden

### Voraussetzung

Die Startwerte im Offline-Programm wurden geändert.

### Vorgehen

Um geänderte Startwerte von nicht-remanenten Variablen zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu ladenden Bausteine in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät > Software (nur Änderungen)". Die Bausteine werden übersetzt und geladen. Die Startwerte der neu definierten Variablen werden in den Ladespeicher der CPU gelegt. Beim nächsten Wechsel von STOP nach RUN läuft das Programm mit den neuen Startwerten an.

Um geänderte Startwerte von remanenten Variablen zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu ladenden Bausteine in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".
3. Wählen Sie anschließend im Menü "Online" den Befehl "PLC-Programm ins Gerät laden und zurücksetzen". Die Online-Bausteine werden gelöscht und mit den neuen Bausteinen überschrieben. Dadurch werden alle Variablen, auch die remanenten Variablen, neu initialisiert.

Informationen zum Laden geänderter Einstellwerte sowie allgemeine Informationen zum Laden finden Sie unter "Siehe auch".

### Siehe auch

Einstellwerte im Online-Programm initialisieren (Seite 1434)

## Remanenz einstellen

### Remanenz von Variablen in Datenbausteinen

#### Remanenzverhalten

Um Datenverlust bei Spannungsausfall zu vermeiden, können Sie Daten als remanent kennzeichnen. Diese werden in einem remanenten Speicherbereich abgelegt. Die Möglichkeiten zur Einstellung der Remanenz sind abhängig von der Art des Datenbausteins und der eingestellten Art des Bausteinzugriffs.

#### Siehe auch

Remanenz in einem Instanz-Datenbaustein einstellen (Seite 1422)

Remanenz in einem globalen Datenbaustein einstellen (Seite 1423)

### Remanenz in einem Instanz-Datenbaustein einstellen

#### Einführung

In einem Instanz-Datenbaustein hängt die Editierbarkeit des Remanenzverhaltens von der Zugriffsart des übergeordneten Funktionsbausteins ab:

- Funktionsbaustein mit Standardzugriff  
Sie können die Instanzdaten gemeinsam als remanent oder nicht-remanent definieren. Individuelle Remanenzeinstellungen für einzelne Variablen sind nicht möglich.
- Funktionsbaustein mit optimiertem Zugriff  
Sie können im Instanz-Datenbaustein die Remanenzeinstellungen der Variablen definieren, die in der Bausteinschnittstelle mit "Im IDB setzen" markiert sind. Auch bei diesen Variablen können Sie das Remanenzverhalten nicht für jede Variable individuell einstellen. Die Remanenzeinstellung wirkt sich auf alle Variablen aus, die in der Bausteinschnittstelle mit "Im IDB setzen" markiert sind.

### Remanenz bei Standardzugriff einstellen

Um in Datenbausteinen mit Standardzugriff die Remanenz aller Variablen zentral einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Instanz-Datenbaustein.
2. Aktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.  
Alle Variablen werden als remanent definiert.
3. Um die Remanenzeinstellung für alle Variablen zurückzusetzen, deaktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.  
Alle Variablen werden als nicht remanent definiert.

## Remanenz bei optimiertem Zugriff einstellen

Um in Datenbausteinen mit optimiertem Zugriff das Remanenzverhalten der Variablen, die mit "Im IDB setzen" markiert sind, einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Instanz-Datenbaustein.
2. Aktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.  
Alle Variablen, die in der Bausteinschnittstelle mit "Im IDB setzen" markiert sind, werden als remanent definiert.
3. Um die Remanenzeinstellung für die Variablen zurückzusetzen, deaktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.  
Alle Variablen, die in der Bausteinschnittstelle mit "Im IDB setzen" markiert sind, werden als nicht remanent definiert.

## Siehe auch

Grundlagen zum Bausteinzugriff (Seite 1034)

Remanenz von Variablen in Datenbausteinen (Seite 1422)

## Remanenz in einem globalen Datenbaustein einstellen

### Einführung

In einem globalen Datenbaustein hängt die Editierbarkeit des Remanenzverhaltens von der Zugriffsart ab:

- Globaler Datenbaustein mit Standardzugriff  
Sie können die Daten gemeinsam als remanent oder nicht-remanent definieren. Individuelle Remanenzeinstellungen für einzelne Variablen sind nicht möglich.
- Globaler Datenbaustein mit optimiertem Zugriff  
Sie können die Remanenzeinstellungen der Variablen individuell definieren. Bei Variablen mit strukturierten Datentypen werden die Remanenzeinstellungen auf alle Variablenelemente übertragen.

## Remanenz bei Standardzugriff einstellen

Um in Datenbausteinen mit Standardzugriff die Remanenz aller Variablen zentral einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den globalen Datenbaustein.
2. Aktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.  
Alle Variablen werden als remanent definiert.
3. Um die Remanenzeinstellung für alle Variablen zurückzusetzen, deaktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.  
Alle Variablen werden als nicht remanent definiert.

### Remanenz bei optimiertem Zugriff einstellen

Um in Datenbausteinen mit optimiertem Zugriff die Remanenz der Variablen individuell einzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den globalen Datenbaustein.
2. Aktivieren Sie das Optionskästchen in der Spalte "Remanenz" für die Variable, für die Sie ein Remanenzverhalten einstellen wollen.  
Die selektierte Variable wird als remanent definiert.
3. Um die Remanenzeinstellung für die Variablen zurückzusetzen, deaktivieren Sie das Optionskästchen in der Spalte "Remanenz" einer Variablen.  
Die selektierten Variablen werden als nicht remanent definiert.

### Siehe auch

Grundlagen zum Bausteinzugriff (Seite 1034)

Remanenz von Variablen in Datenbausteinen (Seite 1422)

### Eigenschaften von Variablen in Datenbausteinen bearbeiten

### Eigenschaften von Variablen in Datenbausteinen

### Eigenschaften

Die folgende Tabelle zeigt eine Übersicht über die Eigenschaften von Variablen in Datenbausteinen:

| Gruppe    | Eigenschaft | Beschreibung   |
|-----------|-------------|--|
| Allgemein | Name        | Name der Variablen.  |
|           | Datentyp    | Datentyp der Variablen.  |
|           | Defaultwert | Wert, mit dem die Variable in der Schnittstelle eines übergeordneten Codebausteins oder einem PLC-Datentyp vorbelegt wurde.<br>Die in der Spalte "Defaultwerte" enthaltenen Werte sind nur im übergeordneten Codebaustein oder PLC-Datentyp änderbar. Im Datenbaustein werden die Werte nur angezeigt. |

| Gruppe    | Eigenschaft                             | Beschreibung   |
|-----------|---|--|
|           | Startwert                               | Wert, den die Variable beim Anlauf der CPU einnehmen soll.<br>Beim Anlegen des Datenbausteins werden die in einem Codebaustein definierten Defaultwerte als Startwerte verwendet. Diese übernommenen Werte können Sie hier mit instanzspezifischen Startwerten ersetzen.<br>Die Angabe eines Startwerts ist optional. Wenn Sie keinen Wert festlegen, nimmt die Variable beim Anlauf den Defaultwert ein. Ist auch kein Defaultwert definiert, wird der für den Datentyp gültige Standardwert verwendet. Für BOOL ist z. B. der Standardwert "FALSE" festgelegt. |
|           | Kommentar                               | Kommentar zur Variablen.   |
| Attribute | Remanenz                                | Kennzeichnet die Variable als remanent.<br>Die Werte remanenter Variablen bleiben auch nach Ausschalten der Versorgungsspannung erhalten.<br>Dieses Attribut ist nur in der Schnittstelle von Funktionsbausteinen mit optimiertem Zugriff verfügbar.   |
|           | Sichtbar                                | Kennzeichnet, ob ein Parameter im CFC sichtbar ist.  |
|           | Parametrierbar                          | Kennzeichnet, ob ein Parameter in CFC parametrierbar ist.  |
|           | Für Test                                | Kennzeichnet, ob ein Parameter für den Testmodus von CFC angemeldet ist.   |
|           | Verschaltbar                            | Kennzeichnet, ob ein Parameter in CFC verschaltbar ist.  |
|           | Rücklesbarkeit für Variablen aktivieren | Kennzeichnet, ob ein Parameter für die Funktion "Plan rücklesen" in CFC relevant ist.  |
|           | Enumerationstexte                       | Ordnet einen Parameter einer Enumeration in CFC zu.  |
|           | Engineering Einheit                     | Ordnet einen Parameter einer Einheit in CFC zu.  |
|           | Untergrenze                             | Definiert den unteren Grenzwert für den Parameter in CFC.  |
|           | Obergrenze                              | Definiert den oberen Grenzwert für den Parameter in CFC.   |

### Siehe auch

Eigenschaften von Variablen in Instanz-Datenbausteinen ändern (Seite 1426)

Eigenschaften von Variablen in globalen Datenbausteinen ändern (Seite 1427)

## Eigenschaften von Variablen in Instanz-Datenbausteinen ändern

### Instanzspezifische Variableneigenschaften

Für die Definition der Variableneigenschaften gibt zwei Möglichkeiten:

- Die Variableneigenschaften werden aus der Schnittstelle des zugeordneten Funktionsbausteins übernommen.  
Eigenschaften, die aus dem Funktionsbaustein übernommen sind, werden in den Spalten der Deklarationstabelle gegraut dargestellt. Die Eigenschaften "Name" und "Datentyp" werden immer übernommen.
- Sie definieren instanzspezifische Eigenschaften.  
Einige Eigenschaften können Sie instanzspezifisch ändern. Änderbare Werte sind z. B. der "Kommentar" oder "Sichtbar in HMI". Eigenschaften die instanzspezifisch geändert wurden, sind in den Spalten der Deklarationstabelle nicht gegraut. Die instanzspezifischen Änderungen bleiben erhalten, auch wenn die Schnittstelle des übergeordneten Funktionsbausteins geändert und der Instanz-Datenbaustein anschließend aktualisiert wird.

### Eigenschaften in der Deklarationstabelle bearbeiten

Um die Eigenschaften einer oder mehrerer Variablen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Instanz-Datenbaustein.
2. Ändern Sie die Einträge in den Spalten.

### Eigenschaften im Eigenschaftsfenster bearbeiten

Um die Eigenschaften einer einzelnen Variable zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle eine Variable.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Das Eigenschaftsfenster wird geöffnet. Es zeigt die Eigenschaften der Variablen in den Bereichen "Allgemein" und "Attribute".
3. Wählen Sie in der Bereichsnavigation den gewünschten Bereich.
4. Ändern Sie die Einträge in den Eingabefeldern.

### Einzelne Eigenschaften auf den Vorbelegungswert zurücksetzen

Um einzelne Variableneigenschaften auf den Wert zurückzusetzen, der im Funktionsbaustein als Vorbelegungswert definiert wurde, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle einen instanzspezifisch geänderten Wert.
2. Löschen Sie den Wert.  
Der instanzspezifische Wert wird gelöscht und der Vorbelegungswert aus der Schnittstelle des Funktionsbausteins wird eingetragen. Der Vorbelegungswert wird gegraut dargestellt.



## Siehe auch

Datenbausteine aktualisieren (Seite 1413)

Eigenschaften von Variablen in Datenbausteinen (Seite 1424)

## Eigenschaften von Variablen in globalen Datenbausteinen ändern

### Einführung

Für die Definition der Variableneigenschaften gibt zwei Möglichkeiten:

- Die Variableneigenschaften werden aus dem PLC-Datentyp übernommen. Eigenschaften, die aus dem PLC-Datentyp übernommen sind, werden in den Spalten der Deklarationstabelle gegraut dargestellt. Die Eigenschaften "Name" und "Datentyp" werden immer übernommen.
- Sie definieren spezifische Eigenschaften. Einige Eigenschaften können Sie im globalen Datenbaustein ändern. Änderbare Werte sind z. B. der "Kommentar" oder "Sichtbar in HMI". Eigenschaften die geändert wurden, sind in den Spalten der Deklarationstabelle nicht gegraut. Die Änderungen bleiben erhalten, auch wenn der PLC-Datentyp geändert und der globale Datenbaustein anschließend aktualisiert wird.

### Eigenschaften in der Deklarationstabelle bearbeiten

Um die Eigenschaften einer oder mehrerer Variablen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den globalen Datenbaustein.
2. Ändern Sie die Einträge in den Spalten.

### Eigenschaften im Eigenschaftsfenster bearbeiten

Um die Eigenschaften einer einzelnen Variable zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle eine Variable.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften". Das Eigenschaftsfenster wird geöffnet. Es zeigt die Eigenschaften der Variablen in den Bereichen "Allgemein" und "Attribute".
3. Wählen Sie in der Bereichsnavigation den gewünschten Bereich.
4. Ändern Sie die Einträge in den Eingabefeldern.

### Einzelne Eigenschaften auf den Vorbelegungswert zurücksetzen

Um einzelne Variableneigenschaften auf den Wert zurückzusetzen, der im PLC-Datentyp als Vorbelegungswert definiert wurde, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle einen geänderten Wert.
2. Löschen Sie den Wert.  
Der Vorbelegungswert aus dem PLC-Datentyp wird eingetragen. Der Vorbelegungswert wird grau dargestellt.

### Siehe auch

Eigenschaften von Variablen in Datenbausteinen (Seite 1424)

### Deklarationstabelle für Datenbausteine bearbeiten

#### Tabellenzeilen einfügen

#### Vorgehen

Um eine Zeile vor der selektierten Zeile einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile einfügen".

#### Ergebnis

Vor der markierten Zeile wird eine neue Zeile eingefügt.

### Siehe auch

Bearbeiten von Tabellen (Seite 216)

#### Tabellenzeilen hinzufügen

#### Vorgehen

Um eine neue Zeile nach der selektierten Zeile hinzuzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, nach der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile hinzufügen".

#### Ergebnis

Nach der selektierten Zeile wird eine neue, leere Zeile hinzugefügt.

**Siehe auch**

Bearbeiten von Tabellen (Seite 216)

**Variablen löschen****Voraussetzung**

Ein globaler Datenbaustein ist geöffnet.

**Vorgehen**

Um eine Variable zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Variablen. Sie können auch mehrere Zeilen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

---

**Hinweis**

Die Struktur von Instanz-Datenbausteinen oder von globalen Datenbausteinen, die auf einem PLC-Datentyp basieren, können Sie nicht direkt ändern, da die Strukturen dieser Bausteine im übergeordneten Objekt definiert werden.

Der Typ des Datenbausteins ist in den Bausteineigenschaften eingetragen.

Siehe auch: Bausteineigenschaften anzeigen und bearbeiten (Seite 1229)

---

**Siehe auch**

Bearbeiten von Tabellen (Seite 216)

**Aufeinander folgende Zellen automatisch ausfüllen**

Sie können den Inhalt einer oder mehrerer Tabellenzellen in die darunter liegenden Zellen übertragen und so aufeinander folgende Zellen automatisch ausfüllen.

Wenn Sie Zellen in der Spalte "Name" automatisch ausfüllen, wird jeder Name um eine fortlaufende Nummer erweitert. Aus dem Namen "Motor" wird z. B. der Name "Motor\_1".

Als Quellbereich können Sie eine einzelne oder mehrere Zellen und komplette Zeilen definieren.

Sollten in der geöffneten Tabelle weniger Zeilen vorhanden sein, als Sie füllen möchten, müssen Sie zunächst weitere Leerzeilen einfügen.

### Voraussetzung

- Die Tabelle ist geöffnet.
- Es stehen genügend Deklarationszeilen zur Verfügung.

### Vorgehen

Um aufeinander folgende Zellen automatisch auszufüllen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu übertragenden Zellen.
2. Klicken Sie auf das Ausfüllzeichen in der rechten unteren Ecke der Zelle.  
Der Mauszeiger wird zu einem Fadenkreuz.
3. Halten Sie die Maustaste gedrückt und ziehen Sie den Mauszeiger nach unten über die Zellen, die Sie automatisch ausfüllen wollen.
4. Lassen Sie die Maustaste los.  
Die Zellen werden automatisch ausgefüllt.
5. Wenn in den automatisch auszufüllenden Zellen bereits Einträge vorhanden sind, erscheint ein Dialog. In diesem Dialog können Sie angeben, ob Sie die bestehenden Einträge überschreiben wollen, oder ob Sie für die neuen Variablen neue Zeilen einfügen möchten.

### Tabellenspalten ein- und ausblenden

Sie können die Spalten in Tabellen bei Bedarf ein- und ausblenden.

### Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".  
Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.

### Variablen mit externen Editoren bearbeiten

Um einzelne Variablen in externen Editoren außerhalb des TIA-Portals zu bearbeiten, können Sie sie per Copy & Paste exportieren oder importieren. Strukturierte Variablen können Sie allerdings nicht in einen Editor kopieren.

### Voraussetzung

Der Datenbaustein und ein externer Editor sind geöffnet.

## Vorgehen

Um einzelne Variablen per Drag & Drop zu exportieren und wieder zu importieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere Variablen.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wechseln Sie in den externen Editor und fügen Sie die kopierten Variablen dort ein.
4. Bearbeiten Sie die Variablen nach Bedarf.
5. Kopieren Sie die Variablen im externen Editor.
6. Wechseln Sie wieder in die Deklarationstabelle.
7. Wählen Sie im Kontextmenü den Befehl "Einfügen".

## Datenwerte online beobachten

### Datenwerte in Datenbausteinen online beobachten

Sie können die Datenwerte, die die Variablen aktuell in der CPU einnehmen, direkt in der Deklarationstabelle beobachten.

## Voraussetzung

- Eine Online-Verbindung besteht.
- Der Datenbaustein ist in die CPU geladen.
- Die Programmbearbeitung ist aktiv (CPU im "RUN")
- Der Datenbaustein ist geöffnet

## Vorgehen

Um die Datenwerte zu beobachten, gehen Sie folgendermaßen vor:

1. Starten Sie die Beobachtung, indem Sie auf die Schaltfläche "Alle Variablen beobachten" klicken.  
Die zusätzliche Spalte "Beobachtungswert" wird in der Tabelle eingeblendet. Sie zeigt die aktuellen Datenwerte an.  
Siehe auch: Aufbau der Deklarationstabelle für Datenbausteine (Seite 1410)
2. Beenden Sie die Beobachtung, indem Sie erneut auf die Schaltfläche "Alle Variablen beobachten" klicken.

## Vom Gerät geladene Datenwerte anzeigen

Beim Laden eines Datenbausteins von einem Gerät werden auch die aktuellen Variablenwerte geladen. Diese Werte können Sie anzeigen.

### Voraussetzung

Ein Datenbaustein wurde vom Gerät geladen.

### Vorgehen

Um die aktuellen Werte anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Klicken Sie in einen Spaltenkopf.
3. Wählen Sie im Kontextmenü den Befehl "Spalten ein-/ausblenden". Die Auswahl der verfügbaren Spalten wird angezeigt.
4. Aktivieren Sie das Optionskästchen der Spalte "Momentaufnahme".

### Ergebnis

Die aktuellen Werte werden in der Spalte "Momentaufnahme" angezeigt.

---

#### Hinweis

Wenn Sie anschließend die Struktur des Datenbausteins ändern, geht die Anzeige der aktuellen Werte verloren. Die Spalte "Momentaufnahme" ist dann leer.

---

## Datenwerte während der Inbetriebnahme einstellen

### Grundlagen zum Anpassen von Datenwerten während der Inbetriebnahme

#### Einführung

Bei der Inbetriebnahme einer Anlage müssen Datenwerte häufig justiert werden, um das Programm optimal an die Rahmenbedingungen vor Ort anzupassen. Zu diesem Zweck bietet die Deklarationstabelle für Datenbausteine folgende Funktionen:

- Datenwerte im Betriebszustand "RUN" initialisieren  
Diese Funktion ermöglicht es Ihnen, Datenwerte online zu ändern, um so schnell die optimalen Variablenwerte zu ermitteln.
- Variablenwerte aus dem Online-Programm als Startwerte in das Offline-Programm übernehmen  
Wenn Sie die optimalen Variablenwerte ermittelt haben, können Sie diese als Startwerte in das Offline-Programm übernehmen. So sorgen Sie dafür, dass das Programm nach dem nächsten Laden mit den optimierten Werten anläuft.

Um die Funktionen einzusetzen, definieren Sie zunächst bestimmte Variablen im Programm als "Einstellwerte". Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen.

## Daten als einstellbare Werte kennzeichnen

Sie können bestimmte Variablen im Programm als "Einstellwerte" kennzeichnen. Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen.

## Regeln

Sie können Variablen in den folgenden Bausteinarten als "Einstellwert" kennzeichnen:

- In Funktionsbausteinen (FB), jedoch nur im Abschnitt "Static"
- In globalen Datenbausteinen (DB)
- In PLC-Datentypen (UDT)  
Bei PLC-Datentypen wird die Einstellung jedoch nur wirksam, wenn der UDT im Abschnitt "Static" eines Funktions- oder Datenbausteins verwendet wird.

In folgenden Bausteinarten ist die Definition von Einstellwerten nicht möglich:

- In Datenbausteinen, die auf einem PLC-Datentyp basieren, und Instanzdatenbausteinen. Diese erben die Einstellung vom übergeordneten FB oder UDT.
- Variablen in ARRAY-Datenbausteinen können Sie nicht als "Einstellwert" kennzeichnen.
- An der Aufrufstelle einer Multiinstanz können Sie Variablen ebenfalls nicht als "Einstellwert" kennzeichnen. Sie müssen die Einstellung in der Schnittstelle des Funktionsbausteins vornehmen, der als Multiinstanz aufgerufen wird.
- In know-how-geschützten Bausteinen können Sie die Kennzeichnung als "Einstellwert" nicht ändern. Dazu müssen Sie zunächst den Know-how-Schutz entfernen.

## Voraussetzung

Ein Funktionsbaustein, ein globaler Datenbaustein oder ein PLC-Datentyp (UDT) ist geöffnet.

## Vorgehen

Um eine Variable als "Einstellwert" zu kennzeichnen, gehen Sie folgendermaßen vor:

1. Wählen Sie eine Variable aus dem Abschnitt "Static" aus.
2. Aktivieren Sie das Optionskästchen in der Spalte "Einstellwert".
  - Das übergeordnete Element einer Struktur oder eines PLC-Datentyps können Sie nicht als "Einstellwert" definieren. Sie müssen die Einstellung für die unterlagerten Elemente einzeln vornehmen.
  - Bei ARRAYS können Sie nur das übergeordnete Element als "Einstellwert" kennzeichnen. Die unterlagerten Elemente erben die Einstellung.
  - Bei ARRAYS of STRUCT können Sie nur die Elemente unterhalb der ersten Struktur als Einstellwerte kennzeichnen. Die Elemente weiterer Strukturen erben die Einstellung.

## Ergebnis


Die Variablen sind als Einstellwerte markiert. Während der Inbetriebnahme können diese Variablen im Betriebszustand "RUN" initialisiert werden. Außerdem können die aktuellen

Variablenwerte als Startwerte in das Offline-Programm übertragen und dort gespeichert werden.

## Einstellwerte im Online-Programm initialisieren

### Grundlagen zum Initialisieren von Einstellwerten

Sie können alle Variablen, die als "Einstellwert" markiert sind, im Online-Programm mit neuen Werten initialisieren. Dabei werden die Startwerte aus dem Offline-Programm in das Online-Programm geladen. Die CPU bleibt im Betriebszustand "RUN". Alle Variablen, die als Einstellwert markiert sind, werden am nächsten Zykluskontrollpunkt einmalig initialisiert. Dies gilt sowohl für remanente als auch für nicht-remanente Variablen. Die Programmbearbeitung wird anschließend mit den neuen Variablenwerten fortgesetzt.

|   |
|---|
|  <b>GEFAHR</b>   |
| <b>Gefahr beim Verändern von Variablenwerten</b>  |
| Ein Verändern der Variablenwerte bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen! |
| Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Einstellwerte neu initialisieren.                                     |

### Voraussetzung

- Eine Online-Verbindung zur CPU ist vorhanden.
- Die Struktur des Datenbausteins ist Offline und Online identisch.
- Eine oder mehrere Variablen sind als "Einstellwert" markiert.

### Vorgehen

Um alle Einstellwerte des Datenbausteins zu initialisieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie einen globalen Datenbaustein oder einen Instanz-Datenbaustein.
2. Tragen Sie die gewünschten Werte in die Spalte "Startwert" ein. Die Startwerte müssen dem angegebenen Datentypen entsprechen.
3. Klicken Sie auf die Schaltfläche "Einstellwerte initialisieren".

### Ergebnis

Die Einstellwerte im Online-Programm werden am nächsten Zykluskontrollpunkt mit den Startwerten aus dem Offline-Programm initialisiert.

Die maximale Anzahl initialisierbarer Variablen ist CPU-abhängig. Wenn die Anzahl der markierten Einstellwerte zu hoch ist, werden Sie mit einer Meldung darüber informiert. In diesem Fall können Sie die Variablen in eine Beobachtungstabelle einfügen und über die



Funktion "Steuern" in der Beobachtungstabelle initialisieren. Alternativ können Sie auch den gesamten Datenbaustein laden. Weitere Informationen hierzu finden Sie unter "Siehe auch".

## Siehe auch

Geänderte Werte laden (Seite 1421)

## Werte aus dem Online-Programm als Startwerte übernehmen

Um Variablenwerte aus dem Online-Programm als Startwerte in das Offline-Programm zu übernehmen, erzeugen Sie zunächst eine Momentaufnahme der Variablenwerte aus dem Online-Programm. Diese können Sie anschließend in das Offline-Programm übernehmen. Beachten Sie, dass immer die Werte aus der Momentaufnahme kopiert werden. Dabei wird nicht geprüft, ob alle Werte aus demselben Zyklus stammen. Schreibgeschützte Startwerte werden nicht überschrieben.

Grundsätzlich haben Sie zur Übernahme der Werte folgende Möglichkeiten:

- Übernehmen der Werte eines geöffneten Datenbausteins  
In einem geöffneten Datenbaustein können Sie alle Werte oder nur die Werte der als "Einstellwert" markierten Variablen als Startwerte übernehmen.
- Übernehmen der Werte mehrerer Bausteine in der Projektnavigation  
In der Projektnavigation können Sie wahlweise alle Einstellwerte oder alle remaneten Werte als Startwerte übernehmen.

## Voraussetzung

- Eine Online-Verbindung zur CPU besteht.
- Mindestens ein Datenbaustein ist in die CPU geladen.

## Vorgehen

Um in einem Datenbaustein alle Werte oder nur die Werte der als "Einstellwert" markierten Variablen zu übernehmen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Datenbaustein.
2. Starten Sie die Beobachtung, indem Sie auf die Schaltfläche "Alle Variablen beobachten" klicken.  
Die Spalte "Beobachtungswert" wird in der Tabelle eingeblendet. Sie zeigt die aktuellen Datenwerte an.
3. Klicken Sie in der Funktionsleiste auf "Momentaufnahme der Beobachtungswerte".  
Die aktuellen Beobachtungswerte werden in die Spalte "Momentaufnahme" übernommen.
4. Klicken Sie in der Funktionsleiste auf eine der folgenden Schaltflächen:
  - "Einstellwerte aus der Momentaufnahme als Startwerte übernehmen"
  - "Alle Werte aus der Momentaufnahme als Startwerte übernehmen"

Die Werte werden aus der Spalte "Momentaufnahme" in die Spalte "Startwert" übernommen.

Um die Beobachtungswerte mehrerer Datenbausteine in der Projektnavigation zu übernehmen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Bausteine in der Projektnavigation.
2. Wählen Sie im Kontextmenü den Befehl "Momentaufnahme der Beobachtungswerte". Die aktuellen Beobachtungswerte aller markierten Bausteine werden in die Spalte "Momentaufnahme" übernommen. Wenn der Vorgang beendet ist, wird eine Meldung im Inspektorfenster ausgegeben.
3. Wählen Sie anschließend im Kontextmenü einen der folgenden Befehle:
  - "Beobachtungswerte als Startwerte übernehmen > Nur Einstellwerte"
  - "Beobachtungswerte als Startwerte übernehmen > Nur Remanenzwerte"

Die Werte werden aus der Spalte "Momentaufnahme" in die Spalte "Startwert" übernommen.

## Ergebnis

Die neuen Startwerte werden im Offline-Programm gespeichert.

---

### Hinweis

#### Werte einzelner Variablen übernehmen

Sie können auch die Werte einzelner Variablen, die nicht zuvor als Einstellwert markiert wurden, aus der Spalte "Momentaufnahme" in die Spalte "Startwerte" übernehmen. Kopieren Sie dazu die Werte mit Hilfe der Befehle "Kopieren" und "Einfügen" aus dem Kontextmenü und fügen Sie sie in die Spalte "Startwert" ein.

---

## Siehe auch

Grundlagen zu Startwerten (Seite 1419)

Startwerte definieren (Seite 1420)

### 9.1.4.4 PLC-Datentypen programmieren

## Grundlagen zu PLC-Datentypen

### Beschreibung

PLC-Datentypen sind von Ihnen definierte Datenstrukturen, die Sie im Programm mehrmals verwenden können. Die Struktur eines PLC-Datentyps setzt sich aus mehreren Komponenten zusammen, die verschiedene Datentypen aufweisen können. Die Art der Komponenten legen Sie bei der Deklaration des PLC-Datentyps fest.

Es gibt folgende Verwendungsmöglichkeiten für PLC-Datentypen:

- PLC-Datentypen können als Datentypen für Variablen in der Variablendeklaration von Codebausteinen oder in Datenbausteinen verwendet werden.
- PLC-Datentypen können als Vorlage für die Erstellung von globalen Datenbausteinen mit gleicher Datenstruktur dienen.








**Siehe auch**

PLC-Datentypen anlegen (Seite 1438)

**Aufbau der Deklarationstabelle für PLC-Datentypen**


**Aufbau der Deklarationstabelle für PLC-Datentypen**

Das folgende Bild zeigt den Aufbau der Deklarationstabelle für PLC-Datentypen.

| Name   | Datentyp   | Defaultwert | Sichtbar in HMI                     | Kommentar            |
|--|--|-------------|-------------------------------------|----------------------|
|  Motor        | Bool   | false       | <input checked="" type="checkbox"/> |                      |
|  ▼ MyTag1     | Struct   |             | <input checked="" type="checkbox"/> |                      |
|  ■ Element1  | Bool   | false       | <input checked="" type="checkbox"/> |                      |
|  ■ Element2 | Bool   | false       | <input checked="" type="checkbox"/> |                      |
|  MyTag2     | Bool   | false       | <input checked="" type="checkbox"/> | <input type="text"/> |

**Bedeutung der Spalten**

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten. Die Spalten können Sie bei Bedarf ein- oder ausblenden. Die Anzahl der angezeigten Spalten variiert abhängig von der CPU-Familie.

| Spalte  | Erläuterung   |
|---|---|
|  | Symbol, das Sie anklicken können, um die Variable zu verschieben oder zu kopieren.  |
| Name  | Name der Variablen.   |
| Datentyp  | Datentyp der Variablen.   |
| Defaultwert   | Wert, mit dem Sie die Variable bei der Deklaration des PLC-Datentyps vorbelegen. Die Angabe des Defaultwerts ist optional. Wenn Sie keinen Wert festlegen, wird der Wert verwendet, der für den angegebenen Datentyp vordefiniert ist. Für BOOL ist z. B. der Wert "false" vordefiniert.. |
| Sichtbar in HMI   | Zeigt an, ob die Variable in der Auswahlliste von HMI per Voreinstellung sichtbar ist.  |
| Erreichbar aus HMI  | Zeigt an, ob HMI zur Laufzeit auf diese Variable zugreifen kann.  |
| Einstellwert  | Einstellwerte sind die Werte, die bei der Inbetriebnahme voraussichtlich feinjustiert werden müssen. Nach der Inbetriebnahme können die Werte dieser Variablen als Startwerte in das Offline-Programm übertragen und dort gespeichert werden.   |
| Kommentar   | Kommentar zur Dokumentation der Variablen.  |

## Siehe auch

PLC-Datentypen anlegen (Seite 1438)

Tabellenspalten ein- und ausblenden (Seite 1445)

## PLC-Datentypen anlegen

### Voraussetzung

Der Ordner "PLC-Datentypen" ist in der Projektnavigation geöffnet.

### Vorgehen

Um einen PLC-Datentyp anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Ordner "PLC-Datentypen" auf den Befehl "Neuen Datentyp hinzufügen".  
Eine neue Deklarationstabelle zur Erstellung eines PLC-Datentyps wird angelegt und geöffnet.
2. Selektieren Sie den PLC-Datentyp und wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie einen Namen für den PLC-Datentyp ein.

### Ergebnis

Der neue PLC-Datentyp wird erzeugt. Sie finden den PLC-Datentyp in der Projektnavigation im Ordner "PLC-Datentypen".

## Siehe auch

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 1437)

Grundlagen zu PLC-Datentypen (Seite 1436)

## PLC-Datentypen löschen

### Voraussetzung

Der PLC-Datentyp, den Sie löschen möchten, ist nicht geöffnet.

## Vorgehen

Um einen PLC-Datentyp zu löschen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "PLC-Datentypen".
2. Selektieren Sie den zu löschenden PLC-Datentyp. Sie können auch mehrere PLC-Datentypen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste den ersten und den letzte Datentyp anklicken.
3. Wählen Sie im Kontextmenü den Befehl "Löschen".

---

### Hinweis

Wenn Sie einen PLC-Datentyp löschen, werden die Bausteine, die den Datentyp verwenden, inkonsistent. Diese Inkonsistenzen werden im verwendenden Baustein rot markiert. Um die Inkonsistenzen zu beheben, müssen Sie die Bausteine aktualisieren.

Siehe auch:

Bausteinschnittstelle aktualisieren (Seite 1269)

Datenbausteine aktualisieren (Seite 1413)

---

## Struktur von PLC-Datentypen programmieren

### Variablen von elementarem Datentyp deklarieren

#### Voraussetzung

Ein PLC-Datentyp ist geöffnet.

#### Vorgehen

Um eine Variable zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den gewünschten Datentyp ein. Sie werden bei der Eingabe durch die Autovervollständigung unterstützt.
3. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten angezeigt werden.
4. Wiederholen Sie die Schritte 1 bis 3 für alle zu deklarierenden Variablen.

#### Siehe auch

Bearbeiten von Tabellen (Seite 216)

## Variablen vom Datentyp ARRAY deklarieren

### Voraussetzung

Ein PLC-Datentyp ist geöffnet.

### Vorgehen

Um eine Variable vom Datentyp ARRAY zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den Datentyp "Array" ein. Sie werden dabei von der Autovervollständigung unterstützt.  
Der Dialog "Array" wird geöffnet.
3. Geben Sie im Eingabefeld "Datentyp" den Datentyp der Array-Elemente an.
4. Geben Sie anschließend im Eingabefeld "Array-Grenzen" für jede Dimension die Ober- und Untergrenze an.  
Beispiel für ein eindimensionales ARRAY:  
[0..3]  
Beispiel für ein dreidimensionales ARRAY:  
[0..3, 0..15, 0..33]
5. Bestätigen Sie die Eingabe.
6. Optional: Ändern Sie die Eigenschaften der Variablen, die in den weiteren Spalten angezeigt werden.

---

### Hinweis

Für ARRAY-Elemente können Sie keine spezifischen Defaultwerte definieren. Sie können ihnen jedoch an der Verwendungsstelle im Datenbaustein Startwerte zuweisen.

---

### Siehe auch

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 1437)

## Variablen vom Datentyp STRUCT deklarieren

### Voraussetzung

Ein PLC-Datentyp ist geöffnet.

## Vorgehen

Um eine Variable vom Datentyp STRUCT zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" "Struct" ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.  
Nach der neuen Variablen wird eine leere, eingerückte Zeile eingefügt.
3. Fügen Sie in der leeren Zeile das erste Strukturelement ein.  
Nach dem Element wird erneut eine leere Zeile eingefügt.
4. Wählen Sie einen Datentyp für das Strukturelement aus.
5. Optional: Ändern Sie die Eigenschaften des Strukturelements, die in den weiteren Spalten der angezeigt werden.
6. Wiederholen Sie Schritt 3 bis 5 für alle weiteren Strukturelemente.  
Ein explizites Beenden der Struktur ist nicht notwendig. Die Struktur endet mit dem letzten eingetragenen Element.
7. Um eine neue Variable nach der Struktur einzufügen, lassen Sie eine Zeile am Ende der Struktur leer und tragen Sie die neue Variable in die zweite leere Zeile ein.

## Ergebnis

Die Variable vom Datentyp STRUCT wird angelegt.

## Siehe auch

STRUCT (Seite 1119)

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 1437)

## Variablen basierend auf einem anderen PLC-Datentyp deklarieren

### Voraussetzung

- Ein globaler Datenbaustein ist geöffnet.
- In der aktuellen CPU ist ein PLC-Datentyp deklariert.

## Vorgehen

Um eine Variable basierend auf einem anderen PLC-Datentyp zu deklarieren, gehen Sie folgendermaßen vor:

1. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
2. Tragen Sie in der Spalte "Datentyp" den PLC-Datentyp ein. Bei der Eingabe werden Sie von der Autovervollständigung unterstützt.

**Ergebnis**

Die Variable wird angelegt.

**Hinweis**

Die Defaultwerte der Variablen innerhalb eines PLC-Datentyps definieren Sie beim Anlegen des PLC-Datentyps. An der Verwendungsstelle des PLC-Datentyps können Sie diese Werte nicht ändern.

**Siehe auch**

Grundlagen zu PLC-Datentypen (Seite 1436)

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 1437)

**Eigenschaften von Variablen in PLC-Datentypen bearbeiten**

**Eigenschaften von Variablen in PLC-Datentypen**

**Eigenschaften**

Die folgende Tabelle zeigt eine Übersicht über die Eigenschaften von Variablen in PLC-Datentypen:

| Gruppe    | Eigenschaft    | Beschreibung   |
|-----------|----------------|--|
| Allgemein | Name           | Name der Variablen.  |
|           | Datentyp       | Datentyp der Variablen.  |
|           | Defaultwert    | Wert, mit dem die Variable in der Schnittstelle eines übergeordneten Codebausteins oder einem PLC-Datentyp vorbelegt wurde.<br>Die in der Spalte "Defaultwerte" enthaltenen Werte sind nur im übergeordneten Codebaustein oder PLC-Datentyp änderbar. Im Datenbaustein werden die Werte nur angezeigt. |
|           | Startwert      | In PLC-Datentypen nicht relevant   |
|           | Kommentar      | Kommentar zur Variablen.   |
| Attribute | Remanenz       | In PLC-Datentypen nicht relevant   |
|           | Sichtbar       | Kennzeichnet, ob ein Parameter im CFC sichtbar ist.  |
|           | Parametrierbar | Kennzeichnet, ob ein Parameter in CFC parametrierbar ist.  |
|           | Für Test       | Kennzeichnet, ob ein Parameter für den Testmodus von CFC angemeldet ist.   |
|           | Verschaltbar   | Kennzeichnet, ob ein Parameter in CFC verschaltbar ist.  |



**Siehe auch**

Eigenschaften von Variablen in PLC-Datentypen ändern (Seite 1443)

Grundlagen zu PLC-Datentypen (Seite 1436)

Aufbau der Deklarationstabelle für PLC-Datentypen (Seite 1437)

**Eigenschaften von Variablen in PLC-Datentypen ändern****Allgemeine Eigenschaften in der Deklarationstabelle bearbeiten**

Um allgemeine Eigenschaften einer oder mehrerer Variablen zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den PLC-Datentyp.
2. Ändern Sie die Einträge in den Spalten.

**Detaillierte Eigenschaften im Eigenschaftsfenster bearbeiten**

Um detaillierte Eigenschaften einer einzelnen Variable zu bearbeiten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Tabelle eine Variable.
2. Wählen Sie im Kontextmenü den Befehl "Eigenschaften".  
Das Inspektorfenster zeigt die Eigenschaften der Variablen in den Bereichen "Allgemein" und "Attribute".
3. Wählen Sie in der Bereichsnavigation den gewünschten Bereich.
4. Ändern Sie die Einträge in den Eingabefeldern.

**Siehe auch**

Bausteinschnittstelle aktualisieren (Seite 1269)

Datenbausteine aktualisieren (Seite 1413)

**Deklarationstabelle für PLC-Datentypen bearbeiten****Tabellenzeilen einfügen****Vorgehen**

Um eine Zeile vor der selektierten Zeile einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile einfügen".

## Ergebnis

Vor der markierten Zeile wird eine neue Zeile eingefügt.

## Tabellenzeilen hinzufügen

### Vorgehen

Um eine neue Zeile nach der selektierten Zeile hinzuzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, nach der Sie die neue Zeile einfügen wollen.
2. Klicken Sie in der Funktionsleiste der Tabelle auf die Schaltfläche "Zeile hinzufügen".

## Ergebnis

Nach der selektierten Zeile wird eine neue, leere Zeile hinzugefügt.

## Variablen löschen

### Vorgehen

Um eine Variable zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zeile mit der zu löschenden Variablen. Sie können auch mehrere Zeilen markieren, indem Sie sie bei gedrückter <Strg>-Taste nacheinander anklicken oder bei gedrückter <Shift>-Taste die erste und die letzte Zeile anklicken.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

## Siehe auch

Bausteinschnittstelle aktualisieren (Seite 1269)

Datenbausteine aktualisieren (Seite 1413)

## Aufeinander folgende Zellen automatisch ausfüllen

Sie können den Inhalt einer oder mehrerer Tabellenzellen in die darunter liegenden Zellen übertragen und so aufeinander folgende Zellen automatisch ausfüllen.

Wenn Sie Zellen in der Spalte "Name" automatisch ausfüllen, wird jeder Name um eine fortlaufende Nummer erweitert. Aus dem Namen "Motor" wird z. B. der Name "Motor\_1".

Als Quellbereich können Sie eine einzelne oder mehrere Zellen und komplette Zeilen definieren.

Sollten in der geöffneten Tabelle weniger Zeilen vorhanden sein, als Sie füllen möchten, müssen Sie zunächst weitere Leerzeilen einfügen.

## Voraussetzung

- Die Tabelle ist geöffnet.
- Es stehen genügend Deklarationszeilen zur Verfügung.

## Vorgehen

Um aufeinander folgende Zellen automatisch auszufüllen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die zu übertragenden Zellen.
2. Klicken Sie auf das Ausfüllzeichen in der rechten unteren Ecke der Zelle.  
Der Mauszeiger wird zu einem Fadenkreuz.
3. Halten Sie die Maustaste gedrückt und ziehen Sie den Mauszeiger nach unten über die Zellen, die Sie automatisch ausfüllen wollen.
4. Lassen Sie die Maustaste los.  
Die Zellen werden automatisch ausgefüllt.
5. Wenn in den automatisch auszufüllenden Zellen bereits Einträge vorhanden sind, erscheint ein Dialog. In diesem Dialog können Sie angeben, ob Sie die bestehenden Einträge überschreiben wollen, oder ob Sie für die neuen Variablen neue Zeilen einfügen möchten.

## Tabellenspalten ein- und ausblenden

Sie können die Spalten in Tabellen bei Bedarf ein- und ausblenden.

## Vorgehen

Um Tabellenspalten ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Anzeigen/Verbergen".  
Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalte auszublenden, deaktivieren Sie das Optionskästchen der Spalte.

#### 9.1.4.5 Externe Quelldateien verwenden

### Grundlagen zur Verwendung von externen Quelldateien

#### Funktion

Für die textuellen Programmiersprachen AWL und SCL haben Sie die Möglichkeit, den Programmcode in einem beliebigen ASCII-Editor einzugeben und als externe Quelldatei zu speichern. Sie können dabei z. B. folgende Arbeiten durchführen:

- Variablen deklarieren
- Bausteineigenschaften festlegen
- Bausteine programmieren

Diese externen Quelldateien können Sie in Ihr Projekt importieren und daraus Bausteine generieren. Auf diese Weise können Sie mehrere Bausteine aus einer Quelldatei erzeugen. Beachten Sie beim Generieren von Bausteinen aus einer Quelldatei folgende Besonderheiten:

- Wenn ein Baustein mit dem gleichen symbolischen Namen bereits im Projekt vorhanden ist, wird der Baustein im Projekt überschrieben. Bei einem Organisationsbaustein (OB) bleibt jedoch der Bausteintyp erhalten.
- Wenn ein Baustein in der Quelldatei mit seiner absoluten Bausteinnummer statt mit einem symbolischen Namen programmiert wurde und diese Nummer durch einen Baustein im Projekt bereits vergeben ist, erhält der neu generierte Baustein den nächst freien symbolischen Namen.
- Wenn Sie in der externen Quelldatei die Zugriffsart für einen Baustein nicht explizit definiert haben, wird die Zugriffsart für den Baustein abhängig von der CPU-Familie gesetzt:
  - Bausteine, die für eine CPU der Baureihe S7-1200/1500 generiert werden, erhalten als Voreinstellung die Zugriffsart "optimiert".
  - Bausteine, die für eine CPU der Baureihe S7-300/400 generiert werden, erhalten als Voreinstellung die Zugriffsart "standard".

Eine Ausnahme bilden Organisationsbausteine, die in einem solchen Fall unabhängig von der CPU-Familie als Voreinstellung immer die Zugriffsart "standard" erhalten. Sie haben die Möglichkeit den Bausteinzugriff bei Bedarf manuell zu ändern.

- Eventuell werden nicht alle Kommentare aus der Quelldatei in den Baustein übernommen.
- Wenn Sie in der externen Quelldatei absolute Adressierung verwenden, wird bei der Generierung des Bausteins für jede absolute Adresse eine symbolische Variable erzeugt. Die Namen dieser Variablen setzen sich zusammen aus "Tag\_" und einem Zeitstempel. Dadurch ergeben sich eventuell relativ lange Variablennamen, die Sie jedoch bei Bedarf manuell ändern können.
- Wenn Sie in der externen Quelldatei Anweisungen in einer anderen Version als im Zielgerät verwenden, kann dies zu Übersetzungsfehler führen. Korrigieren Sie in diesem Fall die entsprechenden Anweisungen und starten Sie den Übersetzungsvorgang erneut. Alternativ können Sie für das Zielgerät eine andere Version auswählen.

Zusätzlich haben Sie auch die Möglichkeit, bestehende Bausteine als externe Quelldateien zu speichern.

## Siehe auch

Regeln zur Programmierung von externen Quelldateien (Seite 1447)

Baustein als externe Quelldatei speichern (Seite 1448)

Externe Quelldateien einbinden (Seite 1449)

Externe Quelldateien öffnen und bearbeiten (Seite 1450)

Bausteine aus externen Quelldateien generieren (Seite 1451)

## Regeln zur Programmierung von externen Quelldateien

Eine externe Quelldatei besteht prinzipiell aus fortlaufendem Text. Damit sie sich in Bausteine übersetzen lässt, müssen jedoch bestimmte Strukturen und Syntaxvorschriften eingehalten werden.

## Syntaxregeln

Die Syntax der Anweisungen in den externen Quelldateien ist der beim Erstellen des Anwenderprogramms im Programmiereditor mit AWL oder SCL sehr ähnlich. Beachten Sie jedoch folgende zusätzlichen Syntaxregeln:

- **Bausteinaufruf**  
Übergeben Sie im ASCII-Editor bei einem Bausteinaufruf die Parameter in definierter Reihenfolge. Ansonsten stimmen möglicherweise die Kommentazuordnungen dieser Zeilen nicht überein.  
Geben Sie die Parameter in Klammern an. Die einzelnen Parameter werden dabei durch ein Komma voneinander getrennt.
- **Groß- oder Kleinschreibung**  
Der Programmiereditor berücksichtigt allgemein keine Groß- oder Kleinschreibung. Ausgenommen davon sind Sprungmarken. Bei Eingabe von Zeichenketten (Datentyp "STRING") muss ebenfalls auf Groß- und Kleinschreibung geachtet werden. Schlüsselwörter werden in Großbuchstaben dargestellt. Beim Übersetzen wird die Groß- und Kleinschreibung jedoch nicht berücksichtigt, sodass Sie Schlüsselwörter in Groß- oder Kleinbuchstaben oder gemischt angeben können.
- **Semikolon**  
Kennzeichnen Sie das Ende jeder Anweisung und jeder Variablendeklaration mit einem Semikolon. Sie können mehrere Anweisungen pro Zeile eingeben.
- **Schrägstriche**  
Beginnen Sie jeden Kommentar mit zwei Schrägstrichen (//) und beenden Sie jede Kommentareingabe mit der Taste <Return>.

## Siehe auch

Grundlagen zur Verwendung von externen Quelldateien (Seite 1446)

Baustein als externe Quelldatei speichern (Seite 1448)

Externe Quelldateien einbinden (Seite 1449)

Externe Quelldateien öffnen und bearbeiten (Seite 1450)

Bausteine aus externen Quelldateien generieren (Seite 1451)

## Baustein als externe Quelldatei speichern

Abhängig von der verwendeten Programmiersprache für den Baustein haben Sie folgende Möglichkeiten Bausteine als externe Quelldatei zu speichern:

- AWL und SCL: Einen Baustein als Text kopieren
- SCL: Externe Quelldatei aus einem oder mehreren Bausteinen generieren

## Baustein als Text kopieren

Um einen Baustein als Text zu kopieren und in eine externe Quelldatei zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation mit der rechten Maustaste auf den Baustein, den Sie in eine externe Quelldatei exportieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Als Text kopieren".
3. Öffnen Sie einen externen Texteditor.
4. Fügen Sie den kopierten Text aus der Zwischenablage in den Texteditor ein.
5. Speichern Sie die Datei mit einer der folgenden Dateinamenerweiterungen:
  - ".scl", wenn Sie eine Quelldatei für SCL erzeugen möchten
  - ".awl", wenn Sie eine Quelldatei für AWL erzeugen möchten

## Externe Quelldatei aus SCL-Bausteinen generieren

Um aus SCL-Bausteinen eine externe Quelldatei zu generieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation oder im Übersichtsfenster die SCL-Bausteine, aus denen Sie eine externe Quelldatei generieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Quelle aus Bausteinen generieren". Der Dialog "Quelle aus Bausteinen generieren" wird geöffnet.
3. Geben Sie einen Pfad und einen Namen für die externe Quelle an.
4. Klicken Sie auf "OK".

Sie können auch aus einem geöffneten SCL-Baustein eine externe Quelldatei generieren. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie im Programmiereditor auf die Schaltfläche "Quelle aus Baustein generieren". Der Dialog "Quelle aus Bausteinen generieren" wird geöffnet.
2. Geben Sie einen Pfad und einen Namen für die externe Quelle an.
3. Klicken Sie auf "OK".

## Ergebnis

Der Baustein wurde als externe Quelldatei gespeichert. Diese Quelldatei können Sie im TIA-Portal in ein Projekt einbinden und daraus erneut Bausteine generieren. Beachten Sie aber, dass Sie AWL-Quelldateien nur in S7-300/400/1500-CPUs verwenden können.

## Siehe auch

- Grundlagen zur Verwendung von externen Quelldateien (Seite 1446)
- Regeln zur Programmierung von externen Quelldateien (Seite 1447)
- Externe Quelldateien einbinden (Seite 1449)
- Externe Quelldateien öffnen und bearbeiten (Seite 1450)
- Bausteine aus externen Quelldateien generieren (Seite 1451)

## Externe Quelldateien einbinden

### Voraussetzung

- Eine externe Quelldatei ist vorhanden und entspricht den Syntax- und Strukturregeln.
- Der Ordner "Externe Quellen" in der Projektnavigation ist geöffnet.

### Vorgehen

Um eine externe Quelldatei einzubinden, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie auf den Befehl "Neue externe Datei hinzufügen". Das Dialogfeld "Öffnen" wird geöffnet.
2. Navigieren Sie zur vorhandenen externen Quelldatei und selektieren Sie diese.
3. Bestätigen Sie Ihre Auswahl mit "Öffnen".

## Ergebnis

Die neue Quelldatei wird in den Ordner "Externe Quellen" eingefügt.

## Siehe auch

- Grundlagen zur Verwendung von externen Quelldateien (Seite 1446)
- Regeln zur Programmierung von externen Quelldateien (Seite 1447)
- Baustein als externe Quelldatei speichern (Seite 1448)
- Externe Quelldateien öffnen und bearbeiten (Seite 1450)
- Bausteine aus externen Quelldateien generieren (Seite 1451)

## Externe Quelldateien öffnen und bearbeiten

Wenn Sie Dateien mit den Dateinamenerweiterungen ".awl" und ".scl" mit einem Editor verknüpfen, haben Sie die Möglichkeit, externe Quelldateien mit diesen Formaten direkt zu öffnen und zu bearbeiten. Als Editor sollten Sie Notepad verwenden, da Sie mit anderen Texteditoren möglicherweise nicht gleichzeitig mehrere Quellen öffnen können.

Dadurch müssen Sie die externen Quelldateien nach einer Bearbeitung nicht erneut einbinden.

## Dateien mit den Dateinamenerweiterungen ".awl" und ".scl" mit einem Editor verknüpfen

Um die Dateien mit den Dateinamenerweiterungen ".awl" und ".scl" mit einem Editor zu verknüpfen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Windows Explorer.
2. Klicken Sie mit der rechten Maustaste auf eine AWL-Datei.
3. Wählen Sie im Kontextmenü "Eigenschaften".  
Der Dialog "Eigenschaften" wird geöffnet.
4. Klicken Sie im Register "Allgemein" im Bereich "Dateityp" auf "Ändern".  
Der Dialog "Öffnen mit" wird geöffnet.
5. Wählen Sie den Texteditor, den Sie mit dem Dateityp ".awl" verknüpfen möchten.
6. Bestätigen Sie Ihre Auswahl mit "OK".
7. Schließen Sie den Dialog "Eigenschaften" mit "OK".
8. Wiederholen Sie die Schritte 2 bis 7 mit einer SCL-Datei.

## Externe Quelldatei öffnen und bearbeiten

Um eine externe Quelldatei zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Externe Quellen".
2. Klicken Sie doppelt auf die externe Quelldatei, die Sie öffnen möchten.  
Die externe Quelldatei wird im verknüpften Editor geöffnet und kann bearbeitet werden.



## Siehe auch

- Grundlagen zur Verwendung von externen Quelldateien (Seite 1446)
- Regeln zur Programmierung von externen Quelldateien (Seite 1447)
- Baustein als externe Quelldatei speichern (Seite 1448)
- Externe Quelldateien einbinden (Seite 1449)
- Bausteine aus externen Quelldateien generieren (Seite 1451)

## Bausteine aus externen Quelldateien generieren

### Voraussetzung

- Der Ordner "Externe Quellen" in der Projektnavigation ist geöffnet.
- Eine externe Quelldatei ist vorhanden.

### Vorgehen

Um aus einer externen Quelldatei die Bausteine zu generieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die externe Quelldatei, aus der Sie die Bausteine generieren möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Bausteine aus Quelle generieren".
3. Eine Sicherheitsabfrage informiert Sie darüber, dass eventuell vorhandene Bausteine überschrieben werden.
4. Bestätigen Sie die Sicherheitsabfrage mit "Ja".

### Ergebnis

Die Bausteine der externen Quelldatei werden generiert und in der Projektnavigation in den Ordner "Programmbausteine" eingefügt. Im Fehlerfall finden Sie Informationen zu den aufgetretenen Fehlern im Inspektorfenster. Diese Informationen beziehen sich jedoch auf die externe Quelldatei und nicht auf den generierten Baustein.

## Siehe auch

- Grundlagen zur Verwendung von externen Quelldateien (Seite 1446)
- Regeln zur Programmierung von externen Quelldateien (Seite 1447)
- Baustein als externe Quelldatei speichern (Seite 1448)
- Externe Quelldateien einbinden (Seite 1449)
- Externe Quelldateien öffnen und bearbeiten (Seite 1450)

## 9.1.5 PLC-Programm vergleichen

### 9.1.5.1 Grundlagen zum Vergleichen von PLC-Programmen

#### Einführung zum Vergleichen von PLC-Programmen

##### Funktion

Sie haben die Möglichkeit, folgende Objekte eines PLC-Programms miteinander zu vergleichen, um mögliche Unterschiede zu ermitteln:

- Codebausteine mit anderen Codebausteinen
- Datenbausteine mit anderen Datenbausteinen
- PLC-Variablen einer PLC-Variablen-Tabelle mit den PLC-Variablen einer anderen PLC-Variablen-Tabelle
- PLC-Datentypen mit anderen PLC-Datentypen

##### Vergleichsarten und Vergleichsstufen

Prinzipiell können Sie zwei unterschiedliche Vergleichsarten einsetzen:

- **Online/Offline-Vergleich:**  
Die Objekte im Projekt werden mit den Objekten des entsprechenden Geräts verglichen. Dazu ist eine Online-Verbindung zu dem Gerät notwendig.
- **Offline/Offline-Vergleich:**  
Es werden entweder die Objekte zweier Geräte innerhalb eines Projekts oder aus verschiedenen Projekten bzw. Bibliotheken verglichen. Für diesen Vergleich ist keine Online-Verbindung notwendig.

Beachten Sie, dass Sie nicht beliebig viele Vergleiche zur gleichen Zeit durchführen können, sondern nur einen Vergleich pro Vergleichsart (Online/Offline, Offline/Offline) und Ausgangspunkt (Gerät oder Ordner "Programmbausteine").

Abhängig davon, in welcher Tiefe Sie die Objekte vergleichen möchten, können Sie zwischen den folgenden Vergleichsstufen wählen:

- Vergleichseditor
- Detailvergleich

Wenn Sie einen Vergleich starten, erhalten Sie zunächst eine Übersicht innerhalb des Vergleichseditors. Für einige Vergleichsobjekte können Sie dann noch einen Detailvergleich starten, bei dem die verglichenen Objekte nebeneinander in je einer eigenen Instanz des Programmiereditors geöffnet werden. Falls Unterschiede bestehen, werden diese hervorgehoben.

Die folgende Tabelle zeigt eine Übersicht, welche Vergleichsart und -stufe Sie für die einzelnen Objekte einsetzen können:

| Objekt                      | Online/Offline  |                  | Offline/Offline |   |
|-----------------------------|-----------------|------------------|-----------------|---|
|                             | Detailvergleich | Vergleichseditor | Detailvergleich |   |
| KOP-Baustein                | X               | X                | X               | X |
| FUP-Baustein                | X               | X                | X               | X |
| AWL-Baustein <sup>1</sup>   | X               | X                | X               | X |
| SCL-Baustein                | X               | X <sup>3</sup>   | X               | X |
| GRAPH-Baustein <sup>2</sup> | X               | -                | X               | - |
| Globaler Datenbaustein      | X               | X                | X               | X |
| Instanz-Datenbaustein       | X               | X                | X               | X |
| PLC-Variablen               | -               | -                | X               | X |
| PLC-Datentyp                | X <sup>4</sup>  | X <sup>4</sup>   | X               | X |

Legende:  
X: verfügbar  
-: nicht verfügbar  
<sup>1</sup>: AWL ist für S7-1200 nicht verfügbar  
<sup>2</sup>: GRAPH ist für S7-1200/1500 nicht verfügbar  
<sup>3</sup>: nicht für S7-1200 kleiner Version 2.0  
<sup>4</sup>: nicht für S7-300/400

**Hinweis**

**Beachten Sie folgende Hinweise:**

- Für know-how-geschützte Bausteine können Sie keinen Detailvergleich durchführen.
- Wenn im Detailvergleich die Unterschiede nur in den Datentypen von lokalen Variablen bestehen, wobei offline ein Alarmdatentyp (C\_ALARM C\_ALARM\_S C\_ALARM\_8 C\_ALARM\_8P C\_ALARM\_T C\_AR\_SEND C\_NOTIFY C\_NOTIFY\_8P) und online ein DWORD ist, wird dieser Unterschied nicht als Unterschied gekennzeichnet.
- Für Typen und Kopiervorlagen us Bibliotheken können Sie keinen Detailvergleich durchführen.

## Siehe auch

- Grundlagen zum Projektdatenvergleich (Seite 261)
- Vergleich von Codebausteinen (Seite 1454)
- Vergleich von Datenbausteinen (Seite 1455)
- Vergleich von PLC-Variablen und PLC-Datentypen (Seite 1455)
- Online/Offline-Vergleich durchführen (Seite 262)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Vergleichseditor verwenden (Seite 264)

## Vergleich von Codebausteinen

### Einführung

Beim Vergleich von Codebausteinen werden die zu vergleichenden Bausteine anhand folgender Kriterien einander zugeordnet:

- Online/Offline-Vergleich: Adressen, z. B. FB100
- Offline/Offline-Vergleich: Symbolische Namen der Bausteine

Beim Vergleich werden die Zeitstempel der Bausteine ausgewertet und die Ergebnisse im Vergleichseditor als Übersicht angezeigt. Anschließend können Sie über Aktionen festlegen, wie mit den Unterschieden umgegangen werden soll. Zusätzlich können Sie für die einzelnen Bausteine Detailvergleiche starten. Dabei werden die verglichenen Versionen eines Bausteins nebeneinander geöffnet und die Unterschiede hervorgehoben.

Für den Vergleich von Codebausteinen werden sowohl die Bausteinschnittstellen als auch die einzelnen Netzwerke hinzugezogen. Zusätzlich werden auch unterschiedliche Variablenamen ermittelt. Bei einem Online/Offline-Vergleich sind alle Kommentare sowie weitere Bausteinattribute vom Vergleich ausgeschlossen.

Falls sich die Bausteinschnittstelle ändert, wird der Zeitstempel der Bausteinschnittstelle des Codebausteins geändert. Diese Änderung sorgt auch für eine Änderung des Zeitstempels des Programmcodes. Für den Vergleich der Bausteinschnittstellen werden daher zunächst die Zeitstempel des Programmcodes verglichen. Wenn diese Zeitstempel gleich sind, wird von einer Gleichheit der Schnittstellen ausgegangen. Wenn die Zeitstempel der Schnittstellen unterschiedlich sind, werden als Nächstes abschnittsweise die Datentypen der Schnittstellen verglichen. Multiinstanzen und PLC-Datentypen werden in den Vergleich mit einbezogen. Bei Gleichheit der Datentypen in den Abschnitten werden die Startwerte der Variablen verglichen. Es werden alle Unterschiede angezeigt.

Beim Vergleich der Netzwerke werden zunächst eingefügte bzw. gelöschte Netzwerke ermittelt. Danach werden die übrigen Netzwerke verglichen. Gleich sind Anweisungen dann, wenn Operator und Operand gleich sind. Es wird jeweils der erste Unterschied pro Anweisung angezeigt. Es können jedoch mehrere Unterschiede pro Netzwerk angezeigt werden.

## Siehe auch

- Einführung zum Vergleichen von PLC-Programmen (Seite 1452)
- Vergleich von Datenbausteinen (Seite 1455)
- Vergleich von PLC-Variablen und PLC-Datentypen (Seite 1455)
- Online/Offline-Vergleich durchführen (Seite 262)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Vergleichseditor verwenden (Seite 264)

## Vergleich von Datenbausteinen

### Einführung

Beim Vergleich von Datenbausteinen werden die zu vergleichenden Bausteine anhand folgender Kriterien einander zugeordnet:

- Online/Offline-Vergleich: Adressen, z. B. DB100
- Offline/Offline-Vergleich: Symbolische Namen der Bausteine

Für den Vergleich von Datenbausteinen werden die Zeitstempel des Datenbausteins verglichen. Wenn diese Zeitstempel gleich sind, wird von einer Gleichheit der Datenstrukturen ausgegangen. Wenn die Zeitstempel unterschiedlich sind, werden die Strukturen bis zum ersten Unterschied in der Datenstruktur verglichen. Bei Gleichheit der Datenstruktur in den Abschnitten werden danach die Anfangs- und Aktualwerte verglichen. Es werden alle Unterschiede angezeigt. Zusätzlich werden auch unterschiedliche Variablennamen ermittelt. Vom Vergleich ausgenommen sind Kommentare sowie die Strukturen von PLC-Datentypen, die im Datenbaustein verwendet wurden.

## Siehe auch

- Einführung zum Vergleichen von PLC-Programmen (Seite 1452)
- Vergleich von Codebausteinen (Seite 1454)
- Vergleich von PLC-Variablen und PLC-Datentypen (Seite 1455)
- Online/Offline-Vergleich durchführen (Seite 262)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Vergleichseditor verwenden (Seite 264)

## Vergleich von PLC-Variablen und PLC-Datentypen

### Einführung

Wenn Sie einen Offline/Offline-Vergleich durchführen, werden auch die PLC-Variablentabellen und die PLC-Datentypen der Geräte im Vergleichseditor angezeigt. Dabei werden die PLC-

Variablentabellen und die PLC-Datentypen anhand ihres Namens einander zugeordnet und Sie erhalten folgende Informationen:

- Status: Es wird über ein Symbol angezeigt, ob die PLC-Variablen/PLC-Datentypen identisch sind oder Unterschiede aufweisen.
- Fehlende PLC-Variablentabellen/PLC-Datentypen: Sie können auf einen Blick erkennen, ob die PLC-Variablentabellen/PLC-Datentypen in beiden Geräten vorhanden sind.

Bei einem Online/Offline-Vergleich erhalten Sie für CPUs der Baureihe S7-1200/1500 folgende Informationen:

- PLC-Variablen: Es wird über ein Symbol angezeigt, ob die PLC-Variablen identisch sind oder Unterschiede aufweisen. Da PLC-Variablentabellen bei einem Ladevorgang nicht in das Gerät geladen werden, können sie bei einem Online/Offline-Vergleich nicht angezeigt werden.
- PLC-Datentypen: Sie erhalten für jeden PLC-Datentyp das Statussymbol. So können Sie auf einen Blick erkennen, ob die PLC-Datentypen in beiden Geräten vorhanden sind.

## Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 1452)

Vergleich von Codebausteinen (Seite 1454)

Vergleich von Datenbausteinen (Seite 1455)

Online/Offline-Vergleich durchführen (Seite 262)

Offline/Offline-Vergleich durchführen (Seite 262)

Vergleichseditor verwenden (Seite 264)

### 9.1.5.2 Bausteine vergleichen

#### Bausteine im Vergleichseditor vergleichen

Sie haben folgende Möglichkeiten Bausteine im Vergleichseditor zu vergleichen:

- Online/Offline-Vergleich  
Die Bausteine im Projekt werden mit den Bausteinen des gewählten Geräts verglichen.
- Automatischer Offline/Offline-Vergleich  
Alle Bausteine der gewählten Geräte werden offline verglichen.
- Manueller Offline/Offline-Vergleich  
Die ausgewählten Bausteine der Geräte werden offline verglichen.

#### Online/Offline-Vergleich von Bausteinen durchführen

Um einen Online/Offline-Vergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Online/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Online".

3. Falls Sie bisher noch keine Online-Verbindung zu diesem Gerät hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Verbinden".  
Die Online-Verbindung wird hergestellt und der Vergleichseditor geöffnet.
4. Öffnen Sie den Ordner "Programmbausteine".  
Den Status können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen. Beachten Sie dabei jedoch, dass Sie in einer Synchronisationsaktion nur Aktionen in eine Richtung ausführen können.

### Automatischer Offline/Offline-Vergleich von Bausteinen durchführen

Um einen automatischen Offline/Offline-Vergleich von Bausteinen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Öffnen Sie den Ordner "Programmbausteine".  
Den Status der Objekte können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen. Wenn Sie ein Objekt selektieren, werden die Eigenschaften des Objekts und des entsprechenden Objekts des zugeordneten Geräts übersichtlich im Eigenschaftsvergleich angezeigt.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

### Manueller Offline/Offline-Vergleich von Bausteinen durchführen

Um einen manuellen Offline/Offline-Vergleich von Bausteinen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.

4. Klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich.
5. Selektieren Sie die Objekte, die Sie miteinander vergleichen möchten.  
Der Eigenschaftsvergleich wird angezeigt. Den Status der Objekte können Sie an den Symbolen ablesen. Abhängig vom Status der Objekte können Sie bestimmte Aktionen festlegen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

### Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 1452)

Vergleichseditor verwenden (Seite 264)

PLC-Variablen vergleichen (Seite 1465)

PLC-Datentypen vergleichen (Seite 1466)

## Detailvergleich von Bausteinen durchführen

### Detailvergleich starten

Sie können für Bausteine einen Detailvergleich starten. Dabei werden die verglichenen Versionen eines Bausteins nebeneinander geöffnet und die Unterschiede werden hervorgehoben.

---

#### Hinweis

Beachten Sie folgende Hinweise:

- Für Bausteine, die in der Programmiersprache GRAPH erstellt sind, steht Ihnen der Detailvergleich nicht zur Verfügung.
  - Für Bausteine, die in der Programmiersprache SCL erstellt sind, steht Ihnen der Detailvergleich für CPUs der S7-1200-Baureihe mit einer Version kleiner als 2.0 nicht zur Verfügung.
- 

### Detailvergleich über den Vergleichseditor starten

Um einen Detailvergleich für einen Baustein über den Vergleichseditor zu starten, gehen Sie folgendermaßen vor:

1. Führen Sie einen Online/Offline- oder einen Offline/Offline-Vergleich durch.  
Der Vergleichseditor wird geöffnet.
2. Selektieren Sie im Vergleichseditor den Baustein, für den Sie einen Detailvergleich durchführen möchten.
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten".



### Detailvergleich innerhalb des Programmiereditors starten

Für die Vergleichsart Online/Offline können Sie den Detailvergleich direkt im Programmiereditor starten. Gehen Sie dazu folgendermaßen vor:

1. Öffnen Sie den Baustein, für den Sie einen Detailvergleich durchführen möchten.
2. Stellen Sie eine Online-Verbindung her.  
Siehe auch: Online verbinden und Online-Verbindung trennen

---

#### Hinweis

Beachten Sie, dass der Baustein online vorhanden sein muss, damit Sie den Detailvergleich für den Baustein innerhalb des Programmiereditors starten können.

---

3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich".
4. Bestätigen Sie den Dialog zum Schließen des Bausteins mit "Ja".

### Ergebnis

Für die miteinander verglichenen Versionen des Bausteins wird je eine Instanz des Programmiereditors geöffnet und die beiden Instanzen werden nebeneinander angeordnet. Falls Unterschiede bestehen, werden diese in den beiden Versionen farblich hervorgehoben.

### Siehe auch

- Offline/Offline-Vergleich durchführen (Seite 262)
- Online/Offline-Vergleich durchführen (Seite 262)
- Vergleichseditor verwenden (Seite 264)
- Darstellung des Detailvergleichs (Seite 1460)
- Im Detailvergleich navigieren (Seite 1462)
- Ändern von Bausteinen während des Detailvergleichs (Seite 1463)
- Vergleichsergebnisse aktualisieren (Seite 1464)

## Darstellung des Detailvergleichs

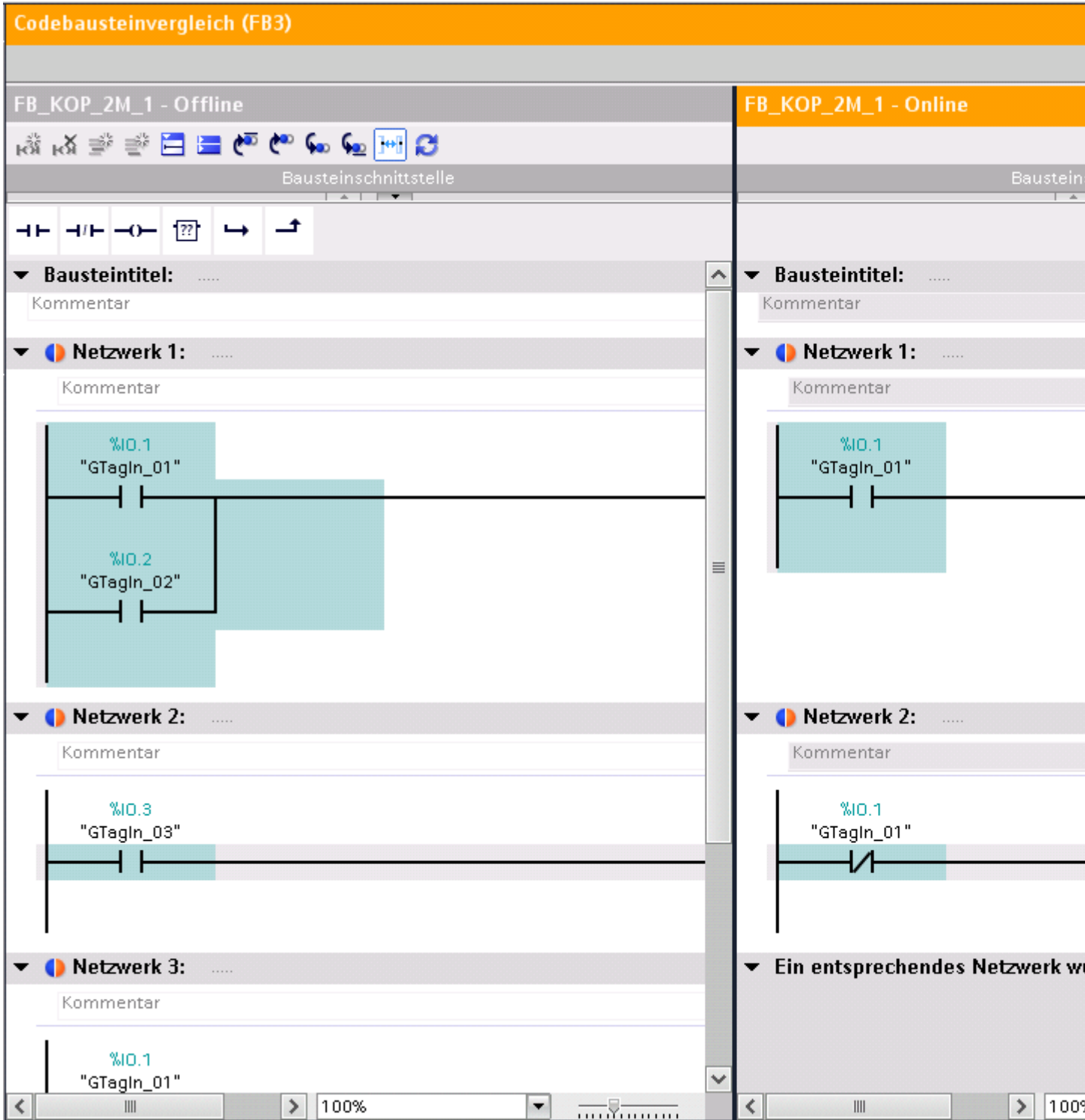
### Kennzeichnung der Unterschiede

Über den Detailvergleich können Sie genau die Stellen identifizieren, die sich in den verglichenen Versionen eines Bausteins unterscheiden. Damit Sie diese Stellen möglichst schnell finden, gelten folgende Kennzeichnungen:

- Die Zeilen, in denen es Unterschiede gibt, werden grau hinterlegt.
- Die unterschiedlichen Operanden und Anweisungen werden grün hinterlegt.
- Bei einer unterschiedlichen Anzahl von Netzwerken werden Pseudonetzwerke eingefügt, damit eine synchronisierte Darstellung identischer Netzwerke möglich ist. Diese Pseudonetzwerke sind grau hinterlegt und enthalten in der Titelleiste des Netzwerks den Text "Ein entsprechendes Netzwerk wurde nicht gefunden.". Pseudonetzwerke können nicht bearbeitet werden.

**Beispiel**

Das folgende Bild zeigt ein Beispiel für den Detailvergleich für die Programmiersprache KOP:



---

### Hinweis

Die Anzeige der symbolischen Namen der Online-Version des Bausteins ist nur für S7-1200 und S7-1500 möglich.

---

## Reduzierung der angezeigten Unterschiede

Um die Übersichtlichkeit zu bewahren, werden nicht alle Unterschiede hervorgehoben, sondern jeweils der erste Unterschied einer Operation. Wenn z. B. bei einer Box mit mehreren Eingängen alle Eingänge in der Offline- und Online-Version des Bausteins unterschiedlich sind, wird nur der erste Eingang als Unterschied hervorgehoben. Sie können diesen Unterschied beheben und die Vergleichsliste aktualisieren. Anschließend wird der nächste Eingang als Unterschied hervorgehoben.

Die Anzahl der hervorgehobenen Unterschiede innerhalb eines Netzwerks ist daher abhängig von der Anzahl der vorhandenen Anweisungen.

## Siehe auch

Online/Offline-Vergleich durchführen (Seite 262)

Offline/Offline-Vergleich durchführen (Seite 262)

Vergleichseditor verwenden (Seite 264)

Detailvergleich starten (Seite 1458)

Im Detailvergleich navigieren (Seite 1462)

Ändern von Bausteinen während des Detailvergleichs (Seite 1463)

Vergleichsergebnisse aktualisieren (Seite 1464)

## Im Detailvergleich navigieren

### Voraussetzung

Ein Detailvergleich wurde durchgeführt.

### Zu den Unterschieden navigieren

Um zu einem Unterschied zwischen den beiden Bausteinen zu navigieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Ergebnisliste des Detailvergleichs im Inspektorfenster unter "Info > Vergleichsergebnis".
2. Doppelklicken Sie auf einen Unterschied.  
Der entsprechende Unterschied wird in beiden Editoren selektiert.

Oder:

1. Klicken Sie in der Funktionsleiste auf eine der folgenden Navigationsschaltflächen:
  - Auf erstem Unterschied positionieren  
Navigiert zum ersten Unterschied im Baustein und zeigt den Unterschied in beiden Editoren an.
  - Auf vorherigem Unterschied positionieren  
Navigiert ausgehend von der aktuellen Position zum vorherigen Unterschied und zeigt den Unterschied in beiden Editoren an.
  - Auf nächstem Unterschied positionieren  
Navigiert ausgehend von der aktuellen Position zum nächsten Unterschied und zeigt den Unterschied in beiden Editoren an.
  - Auf letztem Unterschied positionieren  
Navigiert zum letzten Unterschied im Baustein und zeigt den Unterschied in beiden Editoren an.

### **Synchronisation des vertikalen Bildlaufs zwischen den Editoren aus- und einschalten**

Damit bei einem vertikalen Bildlauf die sich entsprechenden Netzwerke parallel sichtbar bleiben, erfolgt ein synchroner Bildlauf für beide Editoren. Sie können dieses Verhalten beliebig aus- und einschalten. Gehen Sie dazu folgendermaßen vor:

1. Um den synchronen Bildlauf auszuschalten, klicken Sie in der Funktionsleiste auf die Schaltfläche "Bildlauf der Editoren synchronisieren".
2. Um den synchronen Bildlauf wieder einzuschalten, klicken Sie erneut in der Funktionsleiste auf die Schaltfläche "Bildlauf der Editoren synchronisieren".

### **Siehe auch**

- Online/Offline-Vergleich durchführen (Seite 262)
- Offline/Offline-Vergleich durchführen (Seite 262)
- Vergleichseditor verwenden (Seite 264)
- Detailvergleich starten (Seite 1458)
- Darstellung des Detailvergleichs (Seite 1460)
- Ändern von Bausteinen während des Detailvergleichs (Seite 1463)
- Vergleichsergebnisse aktualisieren (Seite 1464)

### **Ändern von Bausteinen während des Detailvergleichs**

#### **Ändern von Offline-Bausteinen**

Sie können Offline-Bausteine jederzeit ändern.

## Ändern von Online-Bausteinen

Sie können Online-Bausteine nicht ändern.

### Siehe auch

Online/Offline-Vergleich durchführen (Seite 262)

Offline/Offline-Vergleich durchführen (Seite 262)

Vergleichseditor verwenden (Seite 264)

Detailvergleich starten (Seite 1458)

Darstellung des Detailvergleichs (Seite 1460)

Im Detailvergleich navigieren (Seite 1462)

Vergleichsergebnisse aktualisieren (Seite 1464)

## Vergleichsergebnisse aktualisieren

Sobald Sie ein Objekt ändern, sind die Vergleichsergebnisse nicht mehr gültig und müssen aktualisiert werden.

### Voraussetzung

Ein Detailvergleich wurde durchgeführt.

### Vorgehen

Um die Vergleichsergebnisse zu aktualisieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf "Vergleichsergebnisse aktualisieren".

### Siehe auch

Online/Offline-Vergleich durchführen (Seite 262)

Offline/Offline-Vergleich durchführen (Seite 262)

Vergleichseditor verwenden (Seite 264)

Detailvergleich starten (Seite 1458)

Darstellung des Detailvergleichs (Seite 1460)

Im Detailvergleich navigieren (Seite 1462)

Ändern von Bausteinen während des Detailvergleichs (Seite 1463)

### 9.1.5.3 PLC-Variablen vergleichen

Sie haben folgende Möglichkeiten, PLC-Variablen zu vergleichen:

- Automatischer Offline/Offline-Vergleich im Vergleichseditor  
Die PLC-Variablen tabellen der gewählten Geräte werden offline verglichen.
- Manueller Offline/Offline-Vergleich im Vergleichseditor  
Die ausgewählten PLC-Variablen tabellen der Geräte werden offline verglichen.
- Detailvergleich  
Über den Detailvergleich können Sie die Unterschiede innerhalb der PLC-Variablen tabellen ermitteln.

#### Automatischen Offline/Offline-Vergleich im Vergleichseditor durchführen

Um einen automatischen Offline/Offline-Vergleich von PLC-Variablen tabellen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Öffnen Sie den Ordner "PLC-Variablen".  
Den Status der PLC-Variablen tabellen können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status können Sie bestimmte Aktionen festlegen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

#### Manuellen Offline/Offline-Vergleich im Vergleichseditor durchführen

Um einen manuellen Offline/Offline-Vergleich von PLC-Variablen tabellen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich.
5. Selektieren Sie die PLC-Variablen tabellen, die Sie miteinander vergleichen möchten.  
Der Eigenschaftsvergleich wird angezeigt. Den Status können Sie an den Symbolen ablesen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

### Detailvergleich durchführen

Um einen Detailvergleich für eine PLC-Variablen-Tabelle über zu starten, gehen Sie folgendermaßen vor:

1. Führen Sie einen automatischen oder manuellen Offline/Offline-Vergleich durch.
2. Selektieren Sie bei einem automatischen Offline/Offline-Vergleich im Vergleichseditor die PLC-Variablen-Tabelle, für die Sie einen Detailvergleich durchführen möchten. Beachten Sie bei einem manuellen Offline/Offline-Vergleich, dass zwei PLC-Variablen-Tabellen für einen Vergleich ausgewählt sein müssen.
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten". Es wird ein eigener Vergleichseditor geöffnet. Alle vorhandenen PLC-Variablen der ausgewählten PLC-Variablen-Tabellen werden abhängig von den Einstellungen des Vergleichseditors angezeigt. Anwender- und Systemkonstanten werden dagegen nicht angezeigt. Den Status der PLC-Variablen können Sie an den Symbolen ablesen. Abhängig vom Status der PLC-Variablen können Sie bestimmte Aktionen festlegen.

### Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 1452)

Vergleichseditor verwenden (Seite 264)

PLC-Datentypen vergleichen (Seite 1466)

#### 9.1.5.4 PLC-Datentypen vergleichen

Sie haben folgende Möglichkeiten, PLC-Datentypen zu vergleichen:

- Online/Offline-Vergleich (nur S7-1200/1500)  
Die PLC-Datentypen im Projekt werden mit den PLC-Datentypen des gewählten Geräts verglichen.
- Automatischer Offline/Offline-Vergleich im Vergleichseditor  
Die PLC-Datentypen der gewählten Geräte werden offline verglichen.
- Manueller Offline/Offline-Vergleich im Vergleichseditor  
Die ausgewählten PLC-Datentypen der Geräte werden offline verglichen.
- Detailvergleich  
Über den Detailvergleich können Sie die Unterschiede zwischen PLC-Datentypen ermitteln.



## Online/Offline-Vergleich von PLC-Datentypen durchführen

Um einen Online/Offline-Vergleich durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Online/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Online".  
Falls Sie bisher noch keine Online-Verbindung zu diesem Gerät hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Verbinden".  
Die Online-Verbindung wird hergestellt und der Vergleichseditor geöffnet.
3. Öffnen Sie den Ordner "PLC-Datentypen".  
Den Status können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Wenn Sie ein Objekt selektieren, werden die Eigenschaften des PLC-Datentyps und des entsprechenden PLC-Datentyps des zugeordneten Geräts im Eigenschaftsvergleich angezeigt.

## Automatischen Offline/Offline-Vergleich im Vergleichseditor durchführen

Um einen automatischen Offline/Offline-Vergleich von PLC-Variablentabellen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.
4. Öffnen Sie den Ordner "PLC-Datentypen".  
Den Status der PLC-Variablentabellen können Sie an den Symbolen im Status- und Aktionsbereich ablesen. Abhängig vom Status können Sie bestimmte Aktionen festlegen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

## Manuellen Offline/Offline-Vergleich im Vergleichseditor durchführen

Um einen manuellen Offline/Offline-Vergleich von PLC-Datentypen durchzuführen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein Gerät, das ein Offline/Offline-Vergleich erlaubt.
2. Wählen Sie im Kontextmenü den Befehl "Vergleichen > Offline/Offline".  
Der Vergleichseditor wird geöffnet und das ausgewählte Gerät wird im linken Bereich angezeigt.
3. Ziehen Sie per Drag & Drop ein weiteres Gerät in die Drop-Fläche im rechten Bereich. Das zu vergleichende Gerät kann dabei aus dem gleichen Projekt, aus einem Referenzprojekt oder aus der Bibliothek stammen.

4. Klicken Sie im Status- und Aktionsbereich auf die Schaltfläche zum Umschalten zwischen automatischem und manuellem Vergleich.
5. Selektieren Sie die PLC-Datentypen, die Sie miteinander vergleichen möchten. Der Eigenschaftsvergleich wird angezeigt. Den Status können Sie an den Symbolen ablesen.

Sie können jederzeit andere Geräte in die Drop-Flächen ziehen, um weitere Vergleiche durchzuführen.

### Detailvergleich durchführen

Um einen Detailvergleich für einen PLC-Datentyp über zu starten, gehen Sie folgendermaßen vor:

1. Führen Sie einen Offline/Offline- oder Online/Offline-Vergleich (nur S7-1200/1500) durch.
2. Selektieren Sie bei einem automatischen Offline/Offline-Vergleich im Vergleichseditor den PLC-Datentyp, für den Sie einen Detailvergleich durchführen möchten. Beachten Sie bei einem manuellen Offline/Offline-Vergleich, dass zwei PLC-Datentypen für einen Vergleich ausgewählt sein müssen.
3. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Detailvergleich starten". Die beiden PLC-Datentypen werden nebeneinander geöffnet, sodass Sie die Unterschiede schnell erkennen können.

### Siehe auch

Einführung zum Vergleichen von PLC-Programmen (Seite 1452)

Vergleichseditor verwenden (Seite 264)

PLC-Variablen vergleichen (Seite 1465)

## 9.1.6 Bausteine übersetzen und laden

### 9.1.6.1 Bausteine übersetzen

#### Grundlagen zum Übersetzen von Bausteinen

##### Einführung

Damit die CPU das Anwenderprogramm bearbeiten kann, muss das Programm zunächst übersetzt werden. Nach jeder Änderung müssen Sie Ihr Programm erneut übersetzen.

Bei der Übersetzung finden folgende Vorgänge statt:

- Das Anwenderprogramm wird nach Syntaxfehlern durchsucht.
- Nicht benötigte Anweisungen werden aus dem Anwenderprogramm entfernt.

- Alle Bausteinaufrufe innerhalb der übersetzten Bausteine werden überprüft. Bei Änderungen an den Schnittstellen von aufgerufenen Bausteinen werden im Register "Übersetzen" des Info-Fensters Fehler angezeigt. Diese Fehler müssen Sie zunächst beheben.
- Anschließend wird das Anwenderprogramm in einen von der CPU lesbaren Code übersetzt.

## Übersetzungswege

Die Übersetzung können Sie in folgenden Fenstern oder Editoren starten:

- Übersetzen von Bausteinen in der Projektnavigation  
Dient dem Übersetzen einzelner Bausteine oder dem gleichzeitigen Übersetzen mehrerer oder aller Bausteine im Ordner "Programmbausteine".
- Übersetzen von Bausteinen im Programmiereditor  
Dient dem Übersetzen eines einzelnen geöffneten Bausteins.
- Übersetzen von Bausteinen in der Aufruf- oder Abhängigkeitsstruktur  
Dient dem Übersetzen einzelner Bausteine.  
Siehe auch: Aufrufstruktur (Seite 1502), Abhängigkeitsstruktur (Seite 1509)

## Übersetzungsoptionen

Wenn Sie Bausteine über die Projektnavigation übersetzen, erhalten Sie weitere Optionen:

- Software (nur Änderungen)  
Es werden alle Programmänderungen der selektierten Bausteine übersetzt. Wenn Sie einen Bausteinordner selektiert haben, werden alle Programmänderungen an den im Ordner enthaltenen Bausteinen übersetzt.
- Software (Bausteine komplett übersetzen)  
Alle Bausteine werden komplett übersetzt. Dies empfiehlt sich bei der ersten Übersetzung und bei größeren Änderungen.
- Software (Speicherreserve zurücksetzen)  
Alle Variablen, die im Reservebereich der Schnittstellen der selektierten Bausteine deklariert sind, werden in den Standardbereich der Schnittstelle verschoben. Die Speicherreserve wird frei für weitere Schnittstellenerweiterungen.

---

### Hinweis

Diese Option steht Ihnen nur bei CPUs der Baureihe S7-1500 und S7-1200 V4 zur Verfügung.

---

## Konsistenzprüfung

Wenn Sie die Schnittstellen aufgerufener Bausteine oder verwendeter PLC-Datentypen ändern, kann es zu Inkonsistenzen zwischen aufrufenden und aufgerufenen Bausteinen oder zwischen den PLC-Datentypen und den globalen Datenbausteinen, die diese PLC-Datentypen verwenden, kommen.

Um solche Inkonsistenzen im Anwenderprogramm zu vermeiden, wird vom System vor jedem Übersetzungsvorgang eine automatische Konsistenzprüfung durchgeführt. Dabei werden die

Zeitstempel verglichen und abhängig von diesem Vergleichsergebnis wird der Übersetzungsvorgang durchgeführt oder abgebrochen:

- Das erfolgreiche Übersetzen des aufrufenden Bausteins ist nur möglich, wenn die Zeitstempel der Schnittstellen der aufgerufenen Bausteine älter sind als der Zeitstempel des aufrufenden Bausteins.
- Das erfolgreiche Übersetzen eines globalen Datenbausteins, der auf einem PLC-Datentyp basiert, ist nur möglich, wenn der Zeitstempel des globalen Datenbausteins neuer als der Zeitstempel des verwendeten PLC-Datentyps ist.
- Das erfolgreiche Übersetzen eines Instanz-Datenbausteins ist nur möglich, wenn die Zeitstempel für die Schnittstellen des Instanz-Datenbausteins und des zugeordneten Funktionsbausteins identisch sind.

Wenn der Übersetzungsvorgang abgebrochen wird, erhalten Sie eine Meldung im Inspektorfenster. Aktualisieren Sie dann zunächst die Bausteinaufrufe in den jeweiligen Bausteinen und die PLC-Datentypen in den globalen Datenbausteinen und starten Sie anschließend den Übersetzungsvorgang erneut. Die Konsistenzprüfung ermittelt auch know-how-geschützte Bausteine, für die der Übersetzungsvorgang nicht durchgeführt werden kann. Auch hierzu finden Sie die entsprechenden Meldungen im Inspektorfenster.

Wenn Sie gleich einen Ladevorgang statt eines Übersetzungsvorgangs starten, werden die ausgewählten Bausteine automatisch übersetzt und die Bausteinaufrufe und globalen Datenbausteine werden implizit aktualisiert. Dabei gibt es folgende Unterschiede bei den CPU-Familien:

- S7-1200/1500: Bei einem Ladevorgang werden alle betroffenen Bausteine mitgeladen, sodass keine Inkonsistenzen entstehen können.
- S7-300/400: Nur der ausgewählte Baustein wird geladen.

## Siehe auch

Bausteine in der Projektnavigation übersetzen (Seite 1470)

Bausteine im Programmiereditor übersetzen (Seite 1472)

Übersetzungsfehler beheben (Seite 1472)

Zeitstempel von Bausteinen (Seite 1227)

Bausteinaufrufe in KOP aktualisieren (Seite 1300)

Bausteinaufrufe in FUP aktualisieren (Seite 1342)

Projektdaten übersetzen (Seite 254)

## Bausteine in der Projektnavigation übersetzen

In der Projektnavigation können Sie einen, mehrere oder alle Bausteine übersetzen.

Für CPUs der Baureihen S7-1500 und S7-1200 V4 können Sie mit einem Übersetzungsvorgang zusätzlich das Speicherlayout von Bausteinen mit Speicherreserve rücksetzen. Die Beschreibung der Speicherreserve finden Sie im Kapitel "Bausteine laden (S7-1200/1500) > Bausteinänderungen ohne Reinitialisierung laden".

## Voraussetzung

Die Projektnavigation ist eingeblendet.

## Einen oder mehrere Bausteine in der Projektnavigation übersetzen

Um einen oder mehrere Bausteine in der Projektnavigation zu übersetzen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Selektieren Sie die Bausteine, die Sie übersetzen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (nur Änderungen)".

## Alle Bausteine in der Projektnavigation übersetzen

Um alle Bausteine des Ordners "Programmbausteine" in der Projektnavigation zu übersetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Sie können für die Übersetzung zwei verschiedene Optionen wählen:
  - Wenn Sie nur die Änderungen seit der letzten Übersetzung übersetzen möchten, wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (nur Änderungen)".
  - Wenn Sie die Bausteine vollständig übersetzen möchten, wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Bausteine komplett übersetzen)".

## Speicherlayout rücksetzen (S7-1500/S7-1200 V4)

Um das Speicherlayout von Bausteinen rückzusetzen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den gesamten Ordner "Programmbausteine" oder einzelne Bausteine innerhalb des Ordners.
2. Wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (Speicherreserve zurücksetzen)".

## Ergebnis

Wenn die Konsistenzprüfung erfolgreich war, wird der Code für die Bausteine generiert. Vom System erzeugte und nicht mehr benötigte Instanz-Datenbausteine werden gelöscht.

Ob die Übersetzung erfolgreich war, können Sie an der Meldung im Inspektorfenster unter "Info > Übersetzen" erkennen.

## Siehe auch

Grundlagen zum Übersetzen von Bausteinen (Seite 1468)

Bausteine im Programmiereditor übersetzen (Seite 1472)

Übersetzungsfehler beheben (Seite 1472)

Syntaxfehler im Programm finden (Seite 1401)

## Bausteine im Programmiereditor übersetzen

---

### Hinweis

Beachten Sie, dass der Baustein auch dann neu übersetzt wird, wenn Sie keine Änderungen vorgenommen haben und dass dadurch die Zeitstempel des Bausteins verändert werden.

---

### Voraussetzung

Der zu übersetzende Baustein ist geöffnet.

### Vorgehen

Um einen Baustein im Programmiereditor zu übersetzen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in das Anweisungsfenster des Programmiereditors.
2. Wählen Sie im Kontextmenü den Befehl "Übersetzen".

### Ergebnis

Der Code für den Baustein wird generiert. Vom System erzeugte und nicht mehr benötigte Instanz-Datenbausteine werden gelöscht.

Ob die Übersetzung erfolgreich war, können Sie an der Meldung im Inspektorfenster unter "Info > Übersetzen" erkennen.

### Siehe auch

Grundlagen zum Übersetzen von Bausteinen (Seite 1468)

Bausteine in der Projektnavigation übersetzen (Seite 1470)

Übersetzungsfehler beheben (Seite 1472)

## Übersetzungsfehler beheben

Im Inspektorfenster unter "Info > Übersetzen" erkennen Sie, ob eine Übersetzung erfolgreich war oder ob Fehler im Programm erkannt worden sind. Wenn Fehler auftreten, müssen Sie diese beheben und anschließend die Übersetzung erneut starten.

### Vorgehen

Um Fehler nach einer Übersetzung zu beheben, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Fehlerliste im Inspektorfenster unter "Info > Übersetzen".
2. Wenn vorhanden, klicken Sie auf das blaue Fragezeichen neben dem Fehlertext, um Hinweise zur Fehlerbehebung zu erhalten.
3. Doppelklicken Sie auf den Fehler, den Sie korrigieren möchten.  
Der entsprechende Fehler wird markiert.

4. Korrigieren Sie den Fehler.
5. Starten Sie die Übersetzung erneut.

### Siehe auch

Grundlagen zum Übersetzen von Bausteinen (Seite 1468)

Bausteine im Programmiereditor übersetzen (Seite 1472)

Bausteine in der Projektnavigation übersetzen (Seite 1470)

## 9.1.6.2 Bausteine laden

### Einführung zum Laden von Bausteinen

#### Bausteine in Gerät laden

Damit die CPU das Anwenderprogramm bearbeiten kann, muss das Programm zunächst übersetzt und anschließend in das Gerät geladen werden. Zum Laden stehen Ihnen folgende Möglichkeiten zur Verfügung:

- Laden von Bausteinen im Programmiereditor  
Im Programmiereditor können Sie einen einzelnen geöffneten Baustein laden.
- Laden von Bausteinen in der Projektnavigation  
Über die Projektnavigation können Sie mehrere oder alle Bausteine des Bausteinordners laden.

Beim Ladevorgang werden alle Informationen, die zur Rekonstruktion des Programms notwendig sind, auch symbolische Informationen, wie z. B. Namen und Kommentare für Code- und Datenbausteine, in der aktuellen Projektsprache mitgeladen. Wenn Sie die Projektsprache ändern, ist daher ein neuer Ladevorgang notwendig.

Die symbolischen Informationen werden nicht in den Arbeitsspeicher, sondern in den Ladespeicher geladen.

Nach dem Laden der Daten aus einem Gerät stehen Ihnen auch die symbolischen Informationen wieder in Ihrem Programm zur Verfügung, dadurch erhöht sich die Lesbarkeit Ihres Programmcodes. Beachten Sie jedoch, dass Sie durch das Laden in und von Gerät nicht die Ablage der Daten in einem Offline-Projekt ersetzen können, da Sie durch dieses Vorgehen z. B. Beobachtungstabellen oder die Mehrsprachigkeit von Projekten nicht abbilden können.

Die Daten aus know-how-geschützten Bausteinen können Sie nach dem Laden aus einem Gerät nur mit korrektem Passwort vollständig anzeigen lassen.

---

#### Hinweis

Um Inkonsistenzen zwischen aufrufenden und aufgerufenen Bausteinen zu vermeiden, werden bei globalen Änderungen, z. B. Änderungen in der Bausteinschnittstelle eines Bausteins, immer alle betroffenen Bausteine übersetzt und geladen.

---

---

**Hinweis**

**S7-1200 Version 1.0**

Wenn Sie ein Element Ihres Projekts in die CPU laden, z. B. einen Programmbaustein, einen Datenbaustein oder die Hardware-Konfiguration, führt die CPU beim nächsten Wechsel in den Betriebszustand RUN einen Kaltstart durch. Neben dem Löschen der Eingänge, Initialisieren der Ausgänge und Löschen des nicht remanenten Speichers werden beim Kaltstart auch die remanenten Speicherbereiche gelöscht. Bei jedem weiteren Wechsel von STOP in RUN wird ein Warmstart durchgeführt, dabei wird der remanente Speicher nicht gelöscht.

---

---

**Hinweis**

**S7-1500**

Bei CPUs der Baureihe S7-1500 befindet sich der Ladespeicher auf der SIMATIC Memory Card. Daher ist für den Betrieb der CPU eine gesteckte SIMATIC Memory Card zwingend erforderlich.

---

## Bausteine von Gerät laden

Sie können die Bausteine eines Gerätes in Ihr Projekt laden. Dies ist z. B. notwendig, um Bausteine zu bearbeiten, die nur in diesem Gerät vorhanden sind. Sie haben die Möglichkeit entweder alle vorhandenen Bausteine (Organisationsbausteine, Funktionsbausteine, Funktionen, Datenbausteine) und globalen PLC-Variablen oder einzelne Bausteine in das Projekt geladen.

## Bausteine auf oder von einer Memory Card laden

Memory Cards sind steckbare Speicherkarten, die z. B. bei einer CPU der Baureihe S7-1200 dazu dienen, den Ladespeicher eines Geräts zu ersetzen. Bei CPUs der Baureihe S7-1500 enthalten sie den Ladespeicher. Für Geräte der S7-1200- und S7-1500-Baureihe können Sie dafür ausschließlich SD-Karten von Siemens verwenden.



Damit Sie eine Memory Card als Ladespeicher nutzen können, müssen Sie das Anwenderprogramm oder einzelne Bausteine auf eine Memory Card laden. Umgekehrt können Sie auch Bausteine von einer Memory Card wieder ins Projekt laden.

---

**Hinweis****S7-1200**

Beachten Sie beim Laden auf oder von einer Memory Card folgende Hinweise:

- Wenn die CPU noch kein Programm enthält und Sie eine leere Memory Card in die CPU stecken, wird das Programm aus dem PG/PC auf die Memory Card und nicht in die CPU geladen.
  - Wenn Sie vor dem Anlauf der CPU eine leere Memory Card in die CPU stecken, wird das Programm, das auf der CPU vorhanden ist, automatisch auf die Memory Card übertragen. Das Programm auf der CPU wird anschließend gelöscht.
  - Wenn Sie vor dem Anlauf der CPU eine Memory Card mit einem Programm in die CPU stecken, die ebenfalls ein Programm enthält, wird das Programm auf der Memory Card und nicht das Programm der CPU ausgeführt. Das Programm auf der CPU wird gelöscht.
- 

**GRAPH-Funktionsbausteine laden**

Wenn Sie einen GRAPH-Funktionsbaustein zusammen mit seinem Instanz-Datenbaustein laden, beginnt die Bearbeitung der Ablaufkette wieder mit dem Initialschritt. Dadurch können Probleme bei der Synchronisation der Ablaufkette mit dem Prozess auftreten. Sie können diese Probleme vermeiden, indem Sie die Ablaufkette vor dem Laden ausschalten.

**Bausteinänderungen ohne Reinitialisierung laden**

Häufig ergibt sich die Notwendigkeit, ein PLC-Programm, das bereits in Betrieb genommen wurde und fehlerfrei auf einer Anlage läuft, nachträglich zu ändern oder zu erweitern. Dabei soll der laufende Betrieb möglichst nicht beeinträchtigt werden.

S7-1500 bietet deshalb die Möglichkeit, die Schnittstellen von Funktions- oder Datenbausteinen im laufenden Betrieb zu erweitern und die geänderten Bausteine zu laden, ohne die CPU in STOP zu setzen und ohne die Werte von bereits geladenen Variablen zu beeinflussen. Programmänderungen lassen sich so auf einfache Weise einbringen. Der gesteuerte Prozess wird durch diesen Ladevorgang (Laden ohne Reinitialisierung) nicht gestört.

### Auswirkungen eines Ladevorgangs auf die Variablenwerte eines Datenbausteins

Wenn Datenbausteine im Betriebszustand STOP in ein Gerät geladen werden, wirkt sich der nächste Übergang des Geräts nach RUN auf die Aktualwerte der Variablen folgendermaßen aus:

- Variablen, die nicht als remanent gekennzeichnet sind, erhalten ihre definierten Startwerte.
- Remanente Variablen der S7-1200 behalten ihre Werte nur, wenn folgende Bedingungen erfüllt sind:
  - Sie haben den Datenbaustein über "Laden in Gerät > Software (nur Änderungen)" geladen.
  - Sie haben die Struktur des Datenbausteins nicht geändert.

Ansonsten erhalten auch die remanenten Variablen ihre definierten Startwerte.

- Remanente Variablen der S7-1500 behalten ihre Werte nur, wenn folgende Bedingungen erfüllt sind:
  - Sie haben den Datenbaustein über "Laden in Gerät > Software (nur Änderungen)" geladen.
  - Sie haben die Struktur des Datenbausteins nicht oder innerhalb der Speicherreserve geändert.

Ansonsten erhalten auch die remanenten Variablen ihre definierten Startwerte.

### Bausteine im Betriebszustand "RUN" in Gerät laden

#### Grundlagen zum Laden von Bausteinen im Betriebszustand "RUN"

Wenn Sie geänderte Bausteine in ein Gerät laden, dann ist es nicht immer notwendig, dass das Gerät in den Betriebszustand "STOP" wechselt. Bei einem Ladevorgang prüft das Engineering System daher, ob das Gerät vor dem Laden gestoppt werden muss. Das Ergebnis dieser Prüfung wird im Dialog "Vorschau Laden" angezeigt. Wenn ein Wechsel in den Betriebszustand "STOP" notwendig ist, können Sie den Ladevorgang erst fortsetzen, wenn Sie die entsprechende Option eingestellt haben.

---

#### Hinweis

Durch einen Ladevorgang im Betriebszustand "RUN" werden keine Aktualparameter überschrieben. Änderungen an den Aktualparameter erfolgen erst beim nächsten Betriebszustandwechsel von "STOP" nach "RUN".

---

Die folgende Tabelle zeigt, nach welchen Aktionen der Ladevorgang im Betriebszustand "RUN" durchgeführt werden kann:

| Aktion   | Laden in Betriebszustand "RUN" möglich |                                   |
|--|--|-----------------------------------|
|  | S7-1200                                | S7-1500                           |
| Laden einzelner Bausteine  | Ja                                     | Ja                                |
| Laden aller Bausteine  | Nein                                   | Nein                              |
| Hinzufügen oder löschen von OBs  | Nein                                   | Ja                                |
| Hinzufügen oder löschen von DBs, FCs oder FBs  | Ja                                     | Ja                                |
| Ändern von Bausteinschnittstellen für FBs  | Nein                                   | Nur innerhalb der Speicherreserve |
| Ändern von Bausteinschnittstellen für FCs  | Ja                                     | Ja                                |
| Ändern der Struktur eines DB oder eines Instanz-DB   | Nein                                   | Nur innerhalb der Speicherreserve |
| Ändern der Hardware-Konfiguration  | Nein                                   | Nein                              |
| Ändern der Remanenzeinstellungen von Merker und DBs  | Nein                                   | Nein                              |
| Ändern des Programmcodes von FC, FB oder OB  | Ja                                     | Ja                                |
| Ändern der Attribute von OBs   | Ja                                     | Ja                                |
| Hinzufügen von Kommentaren   | Ja                                     | Ja                                |
| Hinzufügen von Eingangs-, Ausgangs- oder Merkerbereiche  | Ja                                     | Ja                                |
| Ändern von Variablennamen  | Ja                                     | Ja                                |
| Maximale Anzahl der Bausteine nicht überschritten, die für das verwendete Gerät in "RUN" geladen werden kann | Ja                                     | Ja                                |

### Siehe auch

- Bausteine vom Programmiereditor in Gerät laden (Seite 1477)
- Bausteine aus der Projektnavigation in Gerät laden (Seite 1479)
- Projektdaten in ein Gerät laden (Seite 257)

### Bausteine vom Programmiereditor in Gerät laden

#### Voraussetzung

Der zu ladende Baustein ist geöffnet.

## Vorgehen

Um einen Baustein vom Programmiereditor aus in das Gerät zu laden, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste in das Anweisungsfenster des Programmiereditors.
2. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät".
  - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden". Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.  
Siehe auch: Online verbinden und Online-Verbindung trennen
  - Wurde die Online-Verbindung definiert, werden die Projektdaten übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
3. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".

---

### Hinweis

#### Aktionen

Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

---

Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

4. Klicken Sie auf "Laden".  
Der Baustein wird geladen und der Dialog "Ergebnisse laden" wird geöffnet. Dieser Dialog zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
5. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Alle starten".
6. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

## Ergebnis

Der Code für den Baustein wird in das Gerät geladen. Wenn die Änderungen weitere Bausteine betreffen, werden diese übersetzt und ebenfalls in das Gerät geladen. Bausteine, die nur online im Gerät existieren, werden gelöscht. Vorhandene CPU-Datenbausteine bleiben dabei jedoch erhalten. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine im Gerät werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

## Siehe auch

Bausteine aus der Projektnavigation in Gerät laden (Seite 1479)

Projektdateien in ein Gerät laden (Seite 257)

Bausteine im Betriebszustand "RUN" in Gerät laden (Seite 1476)

## Bausteine aus der Projektnavigation in Gerät laden

In der Projektnavigation können Sie einen, mehrere oder alle Bausteine in ein Gerät laden.

## Einen oder mehrere Bausteine aus der Projektnavigation in Gerät laden

Um einen oder mehrere Bausteine aus der Projektnavigation heraus in ein Gerät zu laden, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Selektieren Sie die Bausteine, die Sie laden möchten.
3. Wählen Sie im Kontextmenü den Befehl "Laden in Gerät > Software (nur Änderungen)".
  - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden". Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.  
Siehe auch: Online verbinden und Online-Verbindung trennen
  - Wurde die Online-Verbindung definiert, werden die Projektdateien übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".

---

### Hinweis

Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

---

Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

5. Klicken Sie auf "Laden".  
Die Bausteine werden geladen und der Dialog "Ergebnisse laden" wird geöffnet. Dieser Dialog zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
6. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Alle starten".
7. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

### Alle Bausteine aus der Projektnavigation in Gerät laden

Um alle Bausteine des Ordners "Programmbausteine" aus der Projektnavigation heraus in ein Gerät zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" in der Projektnavigation.
2. Wählen Sie im Kontextmenü das Untermenü "Laden in Gerät".
3. Wenn Sie nur die Änderungen seit dem letzten Ladevorgang laden möchten, wählen Sie die Option "Software (nur Änderungen)". Wenn alle Bausteine komplett geladen und alle Werte auf ihre Startwerte zurückgesetzt werden sollen, wählen Sie die Option "PLC-Programm ins Gerät laden und zurücksetzen".
  - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden". Sie können den Dialog "Erweitertes Laden" auch explizit über das Menü "Online" aufrufen.  
Siehe auch: Online verbinden und Online-Verbindung trennen
  - Wurde die Online-Verbindung definiert, werden die Projektdaten übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".

---

#### Hinweis

Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Ausführung der Aktionen aktivieren!

---

Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.

5. Klicken Sie auf "Laden".  
Die Bausteine werden geladen und der Dialog "Ergebnisse laden" wird geöffnet. Dieser Dialog zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
6. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Alle starten".
7. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

### Ergebnis

Der Code für die Bausteine wird in das Gerät geladen. Wenn die Änderungen weitere Bausteine betreffen, werden diese übersetzt und ebenfalls in das Gerät geladen. Bausteine, die nur online im Gerät existieren, werden gelöscht. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine im Gerät werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

## Siehe auch

Bausteine vom Programmiereditor in Gerät laden (Seite 1477)

Projektdateien in ein Gerät laden (Seite 257)

Bausteine im Betriebszustand "RUN" in Gerät laden (Seite 1476)

## Bausteine von Gerät laden

Sie können entweder alle oder nur einzelne Bausteine von einem Gerät in Ihr Projekt laden.

---

### Hinweis

Beachten Sie folgende Hinweise:

- Beachten Sie, dass beim Laden von einzelnen Bausteinen keine Variablen und keine anderen notwendigen Bausteine mitgeladen werden, auf die Sie evtl. verweisen. Während des Ladevorgangs werden Referenzen auf Variablen und Bausteine anhand der Namen möglichst wieder zugeordnet. Prüfen Sie nach dem Ladevorgang, ob diese Zuordnungen korrekt sind.
  - S7-1500: Beim Laden von GRAPH-Funktionsbausteinen von einem Gerät in Ihr Projekt werden die schritt spezifischen Meldungstexte für die Interlock- und Supervisionsmeldungen nicht mitgeladen.
- 

## Voraussetzung

Die Online- und Offline-Version der zu ladenden Bausteine ist unterschiedlich oder die Bausteine existieren nur online.

## Alle Bausteine von Gerät laden

Um alle Bausteine von einem Gerät zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, von dem Sie die Bausteine laden möchten.  
Siehe auch: Online verbinden und Online-Verbindung trennen
2. Selektieren Sie in der Projektnavigation den Ordner des Geräts, von dem Sie Bausteine laden möchten.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".  
Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".  
Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".  
Der Ladevorgang wird durchgeführt.

## Einzelne Bausteine von Gerät laden

Um einzelne Bausteine von einem Gerät zu laden, gehen Sie folgendermaßen vor:

1. Stellen Sie eine Online-Verbindung zu dem Gerät her, von dem Sie die Bausteine laden möchten.  
Siehe auch: Online verbinden und Online-Verbindung trennen
2. Selektieren Sie in der Projektnavigation die Bausteine, die Sie vom Gerät laden möchten.
3. Wählen Sie im Menü "Online" den Befehl "Laden von Gerät".  
Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".  
Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
5. Klicken Sie auf die Schaltfläche "Laden von Gerät".  
Der Ladevorgang wird durchgeführt.

## Ergebnis

Die Bausteine werden von dem Gerät in das Projekt geladen. Sie können sie normal bearbeiten und erneut übersetzen und zurück in das Gerät laden.

## Bausteine auf eine Memory Card laden

### Voraussetzung

- Die Memory Card ist als Programmkarte gekennzeichnet.
- Der Ordner "Programmbausteine" der Memory Card ist geöffnet.

### Vorgehen

Um Bausteine auf eine Memory Card zu laden, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine" des Geräts.
2. Selektieren Sie die Bausteine, die Sie auf die Memory Card laden möchten.
3. Ziehen Sie die Bausteine in der Projektnavigation auf den Ordner "Programmbausteine" der Memory Card.  
Der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
4. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
5. Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.



6. Klicken Sie auf die Schaltfläche "Laden".  
Der Ladevorgang wird durchgeführt. Anschließend wird der Dialog "Ergebnisse laden" geöffnet. In diesem Dialog können Sie prüfen, ob der Ladevorgang erfolgreich war, und eventuell weitere Aktionen auswählen.
7. Klicken Sie auf die Schaltfläche "Fertig stellen".

## Ergebnis

Der Baustein wird auf die Memory Card geladen. Wenn die Änderungen weitere Bausteine betreffen, werden diese ebenfalls auf die Memory Card geladen. Bausteine, die nur auf der Memory Card existieren, werden gelöscht. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine auf der Memory Card werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

## Siehe auch

- Bausteine von einer Memory Card laden (Seite 1483)
- Auf Memory Cards zugreifen (Seite 312)

## Bausteine von einer Memory Card laden

Sie können immer nur alle Bausteine von einer Memory Card in Ihr Projekt laden.

## Voraussetzung

- Die Memory Card wird angezeigt.
- Siehe auch: Auf Memory Cards zugreifen (Seite 312)

## Vorgehen

Um Bausteine von einer Memory Card in das Projekt zu laden, gehen Sie folgendermaßen vor:

1. Ziehen Sie in der Projektnavigation den Ordner der Memory Card auf den Ordner des Geräts im Projekt.  
Der Dialog "Vorschau für das Laden von Gerät" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
2. Kontrollieren Sie die Meldungen und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".  
Sobald das Laden möglich ist, wird die Schaltfläche "Laden von Gerät" aktiv.
3. Klicken Sie auf die Schaltfläche "Laden von Gerät".

## Siehe auch

- Bausteine auf eine Memory Card laden (Seite 1482)

### Ablaufkette vor dem Laden eines GRAPH-DB ausschalten

Sie können das Ausschalten der Ablaufkette vor dem Laden eines Instanz-Datenbausteins entweder global festlegen oder während des Ladevorgangs.

### Ablaufkette global ausschalten

Um die Ablaufkette global für jeden Ladevorgang eines Instanz-Datenbausteins auszuschalten, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Extras" den Befehl "Einstellungen".  
Das Fenster "Einstellungen" wird im Arbeitsbereich angezeigt.
2. Wählen Sie in der Bereichsnavigation die Gruppe "PLC-Programmierung > GRAPH".
3. Aktivieren Sie das Optionskästchen "Kette vor Laden des DB ausschalten".  
Bei zukünftigen Ladevorgängen wird die Ablaufkette vor dem Laden des Instanz-Datenbausteins ausgeschaltet.

### Ablaufkette während des Ladevorgangs ausschalten

Um die Ablaufkette während des Ladevorgangs auszuschalten, gehen Sie folgendermaßen vor:

1. Laden Sie den GRAPH-Funktionsbaustein in das Gerät.  
Während des Ladevorgangs erscheint der Dialog "Vorschau laden". Hier werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen. Wenn der Instanz-Datenbaustein zusammen mit dem GRAPH-Funktionsbaustein geladen werden muss, wird Ihnen im Dialog "Vorschau laden" die Aktion "Kette vor Laden des DB ausschalten" angeboten.
2. Aktivieren Sie das Optionskästchen "Kette vor Laden des DB ausschalten".

## 9.1.7 Bausteine schützen

### 9.1.7.1 Schützen von Bausteinen

#### Einführung

Sie können einen oder mehrere Bausteine des Typs OB, FB, FC und globale Datenbausteine mit einem Passwort vor unbefugtem Zugriff schützen. Beachten Sie dabei folgende Besonderheiten:

- Instanz-Datenbausteine können Sie nicht manuell schützen, sondern sie sind vom Know-how-Schutz des zugeordneten FB abhängig. Das heißt, wenn Sie für einen know-how-geschützten FB einen Instanz-Datenbaustein erzeugen, erhält auch der Instanz-Datenbaustein den Know-how-Schutz. Dies ist unabhängig davon, ob Sie den Instanz-Datenbaustein explizit anlegen oder ob er durch einen Bausteinaufruf erzeugt wird.
- Globale Array-Datenbausteine können Sie nicht mit einem Know-how-Schutz versehen.

Bei einem know-how-geschützten Baustein sind lediglich die folgenden Daten ohne korrektes Passwort lesbar:

- Übergabeparameter Input, Output, InOut, Return, Static
- Bausteintitel
- Bausteinkommentar
- Bausteineigenschaften
- Globale Variablen ohne Angabe der Verwendungsstelle

Außerdem sind die folgenden Aktionen mit einem know-how-geschützten Baustein durchführbar:

- Kopieren und Löschen
- Aufrufen in einem Programm
- Online/Offline-Vergleich
- Laden

Der Code des Bausteins ist dagegen vor unbefugtem Lesen und Verändern geschützt. Für S7-1200/1500-CPU's können Sie zusätzlich einen Kopierschutz einrichten, der die Ausführung des Bausteins an die CPU oder die SD-Karte mit der festgelegten Seriennummer bindet.

---

### Hinweis

Beachten Sie folgende Hinweise:

- S7-1200 Version 1.0: Wenn Sie einen know-how-geschützten Baustein in ein Gerät laden, werden keine Wiederherstellungsinformationen mitgeladen. Das bedeutet, dass Sie einen know-how-geschützten Baustein auch mit dem korrekten Passwort nicht wieder öffnen können, wenn Sie ihn vom Gerät laden.
  - Bei einem Vergleich zwischen der Offline- und der Online-Version von know-how-geschützten Bausteinen werden nur die nicht geschützten Daten verglichen.
  - Ohne Passwort ist kein weiterer Zugriff auf den Baustein möglich.
  - Wenn Sie einen know-how-geschützten Baustein in eine Bibliothek einfügen, erhält auch die entstehende Kopiervorlage den Know-how-Schutz.
  - Für die S7-1500 erhalten Sie über das Optionskästchen "Baustein als know-how-geschütztes Bibliothekselement verwendbar" in den Bausteineigenschaften einen Hinweis darauf, ob der Baustein als geschütztes Bibliothekselement verwendet werden kann. Wenn der Baustein als know-how-geschütztes Bibliothekselement verwendet werden kann, werden bei der Verwendung des Bausteins aus der Bibliothek auch eventuell auftretende Nummernkonflikte mit Bausteinen, die sich bereits im Anwenderprogramm befinden, aufgelöst.
  - Querverweise zu verwendeten Variablen, Merkern, Ein- und Ausgängen in know-how-geschützten Bausteinen werden auch nach Eingabe des korrekten Passworts nicht angezeigt.
  - Wenn Sie den Namen oder die Nummer eines Bausteins ändern, ist die ladbare Binärkomponente des Bausteins nicht mehr aktuell. Das bedeutet, dass der Baustein vor dem Laden in ein Gerät neu übersetzt werden muss, was bei einem know-how-geschützten Baustein nur mit korrektem Passwort möglich ist. Beachten Sie dies vor allem, wenn Sie einen know-how-geschützten Baustein in ein anderes Gerät kopieren möchten, in dem es bereits einen Baustein mit diesem Namen oder dieser Nummer gibt.
  - Geben Sie ein Projekt, das know-how-geschützte Bausteine enthält, immer als Projekt- oder Bibliotheksarchiv weiter. Dadurch stellen Sie sicher, dass der Know-how-Schutz nicht umgangen werden kann.
-

**Siehe auch**

- Kopierschutz für Bausteine einrichten und entfernen (Seite 1487)
- Know-how-Schutz für Bausteine einrichten (Seite 1488)
- Know-how-geschützte Bausteine öffnen (Seite 1489)
- Know-how-geschützte Bausteine drucken (Seite 1490)
- Know-how-Schutz für Bausteine entfernen (Seite 1492)
- Passwort ändern (Seite 1491)
- Projekte archivieren (Seite 249)
- Globale Bibliotheken archivieren (Seite 331)

**9.1.7.2 Kopierschutz für Bausteine einrichten und entfernen**

Sie können für S7-1200/1500-CPUs einen Kopierschutz einrichten, mit dem das Ausführen des Bausteins an eine bestimmte CPU oder an eine bestimmte SD-Karte gebunden wird. Nur wenn der Baustein sich auf dem Gerät mit der festgelegten Seriennummer befindet, kann er ausgeführt werden.

Wenn Sie einen solchen Kopierschutz für einen Baustein einrichten, ist es wichtig, diesen Baustein auch mit einem Know-how-Schutz zu versehen. Ansonsten kann jeder den Kopierschutz zurücksetzen. Allerdings müssen Sie den Kopierschutz zuerst einrichten, da die Einstellungen für den Kopierschutz schreibgeschützt sind, wenn der Baustein einen Know-how-Schutz besitzt.

---

**Hinweis**

S7-1500 und S7-1200 V2.2 und höher: Wenn Sie einen Baustein mit Kopierschutz in ein Gerät laden, das mit der festgelegten Seriennummer nicht übereinstimmt, wird der gesamte Ladevorgang zurückgewiesen. Das bedeutet, dass auch Bausteine ohne Kopierschutz nicht geladen werden.

---

**Voraussetzung**

Der Baustein ist nicht know-how-geschützt.

**Kopierschutz einrichten**

Um den Kopierschutz für einen Baustein einzurichten, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Baustein, für den Sie den Kopierschutz einrichten möchten.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
3. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Schutz".
4. Wählen Sie im Bereich "Kopierschutz" aus der Klappliste entweder den Eintrag "An Seriennummer der CPU binden" oder den Eintrag "An Seriennummer der Memory Card binden".

5. Geben Sie bei einer S7-1500-CPU die Seriennummer der CPU oder der Memory Card ein. Bei einer S7-1200-CPU können Sie entweder die Seriennummer ebenfalls direkt eingeben oder Sie aktivieren die Option "Seriennummer eingefügt beim Laden in ein Gerät oder eine Memory Card", wenn die Seriennummer beim Ladevorgang automatisch eingefügt werden soll.
6. Im Bereich "Know-how-Schutz" können Sie nun den Know-how-Schutz für den Baustein einrichten.

### Kopierschutz entfernen

Um den Kopierschutz wieder zu entfernen, gehen Sie folgendermaßen vor:

1. Entfernen Sie den Know-how-Schutz des Bausteins, für den Sie den Kopierschutz entfernen möchten.
2. Öffnen Sie den Baustein.
3. Öffnen Sie im Inspektorfenster das Register "Eigenschaften".
4. Wählen Sie im Inspektorfenster in der Bereichsnavigation den Eintrag "Schutz".
5. Wählen Sie im Bereich "Kopierschutz" aus der Klappliste den Eintrag "Keine Bindung".

### Siehe auch

Schützen von Bausteinen (Seite 1484)

Know-how-Schutz für Bausteine einrichten (Seite 1488)

Know-how-geschützte Bausteine öffnen (Seite 1489)

Know-how-geschützte Bausteine drucken (Seite 1490)

Know-how-Schutz für Bausteine entfernen (Seite 1492)

Passwort ändern (Seite 1491)

#### 9.1.7.3 Know-how-Schutz für Bausteine einrichten

Sie können in Ihrem Projekt für die Bausteine in den Geräten einen Know-how-Schutz einrichten.

### Vorgehen

Um einen Know-how-Schutz für Bausteine einzurichten, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Bausteine ohne Know-how-Schutz, die Sie schützen möchten.
2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz".  
Der Dialog "Know-how-Schutz" wird geöffnet.
3. Klicken Sie auf "Definieren".  
Der Dialog "Passwort definieren" wird geöffnet.
4. Tragen Sie im Feld "Neu" ein Passwort ein.

5. Wiederholen Sie das Passwort im Feld "Bestätigen".
6. Bestätigen Sie die Eingabe mit "OK".
7. Schließen Sie den Dialog "Know-how-Schutz" mit "OK".

## Ergebnis

Die ausgewählten Bausteine werden mit einem Know-how-Schutz versehen. In der Projektnavigation werden know-how-geschützte Bausteine mit einem Schloss markiert. Das eingegebene Passwort ist für alle ausgewählten Bausteine gültig.

## Siehe auch

- Schützen von Bausteinen (Seite 1484)
- Kopierschutz für Bausteine einrichten und entfernen (Seite 1487)
- Know-how-geschützte Bausteine öffnen (Seite 1489)
- Know-how-geschützte Bausteine drucken (Seite 1490)
- Know-how-Schutz für Bausteine entfernen (Seite 1492)
- Passwort ändern (Seite 1491)

### 9.1.7.4 Know-how-geschützte Bausteine öffnen

Sie können mehrere know-how-geschützte Bausteine nur dann gemeinsam öffnen, wenn sie mit dem gleichen Passwort geschützt werden.

## Vorgehen

Um einen know-how-geschützten Baustein zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie doppelt auf den Baustein, den Sie öffnen möchten.  
Der Dialog "Zugriffsschutz" wird geöffnet.
2. Geben Sie das Passwort für den know-how-geschützten Baustein ein.
3. Bestätigen Sie Ihre Eingabe mit "OK".

## Ergebnis

Wenn Sie das korrekte Passwort eingegeben haben, wird der know-how-geschützte Baustein geöffnet. Der Know-how-Schutz des Bausteins wird jedoch nicht aufgehoben. Wenn Sie den Baustein z. B. kopieren oder in eine Bibliothek einfügen, sind auch die Kopien know-how-geschützt.

Nach dem Öffnen des Bausteins können Sie den Programmcode und die Bausteinschnittstelle des Bausteins so lange bearbeiten, bis Sie den Baustein oder das TIA-Portal schließen. Beim nächsten Öffnen des Bausteins muss das Passwort wieder eingegeben werden. Wenn Sie den Dialog "Zugriffsschutz" mit "Abbrechen" schließen, wird der Baustein zwar geöffnet, aber der Code des Bausteins wird nicht angezeigt und Sie können den Baustein nicht bearbeiten.

## Siehe auch

- Schützen von Bausteinen (Seite 1484)
- Kopierschutz für Bausteine einrichten und entfernen (Seite 1487)
- Know-how-Schutz für Bausteine einrichten (Seite 1488)
- Know-how-geschützte Bausteine drucken (Seite 1490)
- Know-how-Schutz für Bausteine entfernen (Seite 1492)
- Passwort ändern (Seite 1491)

### 9.1.7.5 Know-how-geschützte Bausteine drucken

Sie können know-how-geschützte Bausteine nur dann vollständig ausdrucken, wenn Sie den Baustein mit dem korrekten Passwort geöffnet haben. Drucken Sie einen geschlossenen Baustein oder wurde der Baustein nicht mit dem korrekten Passwort geöffnet, werden nur die nicht geschützten Daten des Bausteins gedruckt.

## Vorgehen

Um einen know-how-geschützten Baustein vollständig zu drucken, gehen Sie folgendermaßen vor:

1. Öffnen Sie den know-how-geschützten Baustein, den Sie drucken möchten.  
Siehe auch: Know-how-geschützte Bausteine öffnen (Seite 1489)
2. Wählen Sie im Menü "Projekt" den Befehl "Drucken".  
Der Dialog "Drucken" wird geöffnet.
3. Wählen Sie im Feld "Name" den Drucker aus.
4. Klicken Sie auf "Erweitert", um die Windows-Einstellungen des Druckers zu bearbeiten.
5. Wählen Sie in der Klappliste "Dokument-Information" das Dokumentations-Informationen-Set aus, das für das Rahmen-Layout verwendet werden soll.
6. Wählen Sie unter "Objekte/Bereich drucken aus", ob Sie alle Objekt bzw. den gesamten Bereich oder nur eine Auswahl drucken möchten.
7. Wählen Sie unter "Eigenschaften" den Druckumfang aus.
  - Wählen Sie "Alle", um den kompletten Baustein zu drucken.
  - Wählen Sie "Sichtbar", um alle auf dem Bildschirm sichtbaren Informationen innerhalb des Bausteins zu drucken.
  - Wählen Sie "Kompakt", um den Baustein in einer gekürzten Form zu drucken.
8. Klicken Sie auf "Vorschau", um vorab eine Druckvorschau zu erzeugen.  
Im Arbeitsbereich wird eine Druckvorschau erstellt.
9. Klicken Sie auf "Drucken", um den Ausdruck zu starten.



**Siehe auch**

- Projekthinhalte drucken (Seite 276)
- Schützen von Bausteinen (Seite 1484)
- Kopierschutz für Bausteine einrichten und entfernen (Seite 1487)
- Know-how-Schutz für Bausteine einrichten (Seite 1488)
- Know-how-Schutz für Bausteine entfernen (Seite 1492)
- Passwort ändern (Seite 1491)

**9.1.7.6 Passwort ändern****Vorgehen**

Um das Passwort zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie die know-how-geschützten Bausteine, für die Sie das Passwort ändern möchten.

---

**Hinweis**

Wenn Sie für mehrere Bausteine das Passwort gleichzeitig ändern möchten, müssen die selektierten Bausteine das gleiche Passwort haben.

---

2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz". Der Dialog "Know-how-Schutz" wird geöffnet.
3. Klicken Sie auf die Schaltfläche "Ändern".
4. Tragen Sie im Feld "Alt" das alte Passwort ein.
5. Tragen Sie im Feld "Neu" das neue Passwort ein.
6. Wiederholen Sie das neue Passwort im Feld "Bestätigen".
7. Bestätigen Sie die Eingabe mit "OK".
8. Schließen Sie den Dialog "Know-how-Schutz" mit "OK".

**Siehe auch**

- Schützen von Bausteinen (Seite 1484)
- Kopierschutz für Bausteine einrichten und entfernen (Seite 1487)
- Know-how-Schutz für Bausteine einrichten (Seite 1488)
- Know-how-geschützte Bausteine öffnen (Seite 1489)
- Know-how-geschützte Bausteine drucken (Seite 1490)
- Know-how-Schutz für Bausteine entfernen (Seite 1492)

### 9.1.7.7 Know-how-Schutz für Bausteine entfernen

#### Vorgehen

Um den Know-how-Schutz für Bausteine zu entfernen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Bausteine, bei denen Sie den Know-how-Schutz entfernen möchten.

---

#### Hinweis

Wenn Sie für mehrere Bausteine den Know-how-Schutz gleichzeitig entfernen möchten, müssen die selektierten Bausteine das gleiche Passwort haben.

---

2. Wählen Sie im Menü "Bearbeiten" den Befehl "Know-how-Schutz".  
Der Dialog "Know-how-Schutz" wird geöffnet.
3. Deaktivieren Sie das Optionskästchen "Code nicht anzeigen (Know-how-Schutz)".
4. Geben Sie das Passwort ein.
5. Bestätigen Sie die Eingabe mit "OK".

#### Ergebnis

Der Know-how-Schutz wird für die ausgewählten Bausteine aufgehoben.

#### Siehe auch

Schützen von Bausteinen (Seite 1484)

Kopierschutz für Bausteine einrichten und entfernen (Seite 1487)

Know-how-Schutz für Bausteine einrichten (Seite 1488)

Know-how-geschützte Bausteine öffnen (Seite 1489)

Know-how-geschützte Bausteine drucken (Seite 1490)

Passwort ändern (Seite 1491)

## 9.2 Programminformationen anzeigen

### 9.2.1 Übersicht der möglichen Programminformationen

#### Programminformationen

Die Programminformationen eines Anwenderprogramms enthalten die in der nachfolgenden Tabelle angegebenen Ansichten.

| Ansicht                            | Anwendung   |
|------------------------------------|---|
| Belegungsplan (Seite 1494)         | Gibt einen Überblick, welche Bits der Operanden der Speicherbereiche E, A und M innerhalb des Anwenderprogramms bereits belegt sind.<br><br>Zeigt zudem an, ob eine Adresse durch einen Zugriff aus einem S7-Programm heraus belegt ist oder ob die Adresse einer SIMATIC-S7-Baugruppe zugeordnet ist |
| Aufrufstruktur (Seite 1502)        | Zeigt die Aufrufhierarchie der Bausteine innerhalb eines Anwenderprogramms und gibt einen Überblick über die verwendeten Bausteine und deren Abhängigkeiten.  |
| Abhängigkeitsstruktur (Seite 1509) | Zeigt die Liste der im Anwenderprogramm verwendeten Bausteine. In der ersten Ebene steht der jeweilige Baustein und darunter eingerückt die Bausteine, die diesen Baustein aufrufen oder verwenden. Instanz- Datenbausteine werden im Gegensatz zur Aufrufstruktur separat aufgelistet.               |
| Speicherauslastung (Seite 1515)    | Zeigt die Hardware-Speicherauslastung der CPU für Objekte (OB, FC, FB, DB, anwenderdefinierte Datentypen und PLC-Variablen), für Speicherbereiche der CPU und für die vorhandenen Ein-/Ausgabebaugruppen an.  |

#### Gleichzeitige Anzeige mehrerer Ansichten

Sie können mehrere Ansichten zu einem oder mehreren Anwenderprogrammen erzeugen und anzeigen lassen, um sich das Testen und Ändern Ihres Anwenderprogramms zu erleichtern.

Das Anzeigen von mehreren Ansichten ermöglicht Ihnen z. B.:

- Alle Programminformationen zu einem Anwenderprogramm nebeneinander anzuzeigen
- Eine Gegenüberstellung unterschiedlicher Anwenderprogramme vorzunehmen

## 9.2.2 Belegungsplan anzeigen

### 9.2.2.1 Einführung zum Belegungsplan

#### Programminformationen im Belegungsplan

Der Belegungsplan zeigt an, ob eine Adresse durch einen Zugriff aus einem S7-Programm heraus belegt ist oder ob die Adresse einer SIMATIC-S7-Baugruppe zugeordnet ist. Er ist somit eine wichtige Grundlage für die Fehlersuche oder für Änderungen im Anwenderprogramm.

Sie erhalten im Belegungsplan eine CPU-spezifische Übersicht, welches Bit in welchem Byte der nachfolgend aufgelisteten Speicherbereiche verwendet wird:

- Eingang (E)
- Ausgang (A)
- Merker (M)
- Zeiten (T)
- Zähler (Z)
- Peripherie (P)

#### Anzeige des Belegungsplans

Der Belegungsplan wird in mehreren Arbeitsfenstern angezeigt, jeweils getrennt für Ein- und Ausgänge, sowie für Merker.

#### Filtern

Die Anzeige innerhalb des Belegungsplans können Sie filtern. Sie können dazu vordefinierte Filter verwenden oder eigene Filter erstellen.

#### Anzeige der Querverweisinformationen

Im Belegungsplan haben Sie die Möglichkeit die Querverweisinformationen zu ausgewählten Adressen anzuzeigen.

Die Querverweise zu einer selektierten Adresse zeigen Sie im Inspektorfenster mit dem Kontextmenübefehl "Querverweis-Informationen" an. Über den Befehl "Werkzeuge > Querverweise" können Sie zusätzlich auch die Querverweisliste zum selektierten Objekt öffnen.

#### Anzeige der PLC-Variablentabelle

Aus dem Belegungsplan heraus können Sie die PLC-Variablentabelle öffnen und dort die Eigenschaften der verwendeten Variablen bearbeiten.

Dazu selektieren Sie eine Adresse des Belegungsplans und wählen Sie im Kontextmenü den Befehl "Editor öffnen"

### Anzeige der Remanenz aktivieren

Im Belegungsplan haben Sie die Möglichkeit die Anzeige der Remanenz für Merker über das Symbol "Remanenz ein/ausblenden" der Funktionsleiste zu aktivieren und deaktivieren.

### Siehe auch

Symbole im Belegungsplan (Seite 1496)

Aufbau des Belegungsplans (Seite 1495)

## 9.2.2.2 Aufbau des Belegungsplans

### Aufbau des Belegungsplans

Der Belegungsplan wird CPU-abhängig in mehreren Arbeitsfenstern mit folgenden Operanden angezeigt.

Für S7-300/400er CPUs:

- Eingänge
- Ausgänge
- Merker
- Zeiten
- Zähler

Für S7-1200er CPUs:

- Eingänge
- Ausgänge
- Merker

### Anzeige von Eingängen, Ausgängen , Merkern, Zeiten und Zählern

Es werden alle verwendeten Operanden sowie deren Belegung im S7-Programm angezeigt.

Jede Zeile des Belegungsplans beinhaltet für alle angezeigten Operanden jeweils ein Byte des Speicherbereichs, in dem die zugehörigen acht Bits von 7 bis 0 je nach Zugriff gekennzeichnet werden. Im Anschluss wird durch einen "Balken" angezeigt, ob der Zugriff durch einen Byte- (B), Wort- (W) oder Doppelwortzugriff (D) erfolgt.

Eine Erklärung zu den Symbolen im Belegungsplan finden Sie hier: (Seite 1496)












### Siehe auch

Einführung zum Belegungsplan (Seite 1494)

### 9.2.2.3 Symbole im Belegungsplan

#### Bedeutung der Symbole im Belegungsplan

Die nachfolgende Tabelle zeigt die Bedeutung der Symbole im Belegungsplan:

| Symbol  | Bedeutung  |
|---|--|
|    | Zeigt die Adressbelegung im selektierten Zustand an.   |
|    | Zeigt die Adressbelegung im nicht selektierten Zustand an.   |
|    | Zeigt an, dass eine Pointerstartadresse und eine Variablenadresse auf denselben Adressbereich zugreifen und dass diese selektiert sind.  |
|    | Zeigt an, dass eine Pointerstartadresse und eine Variablenadresse auf denselben Adressbereich zugreifen und dass diese nicht selektiert sind.  |
|    | Zeigt die Pointerbelegung im selektierten Zustand an.  |
|    | Zeigt die Pointerbelegung im nicht selektierten Zustand an.  |
|   | Zeigt an, dass das Byte durch einen Bytezugriff belegt ist und die entsprechende Variable selektiert ist. Über das Kontextmenü können die Querverweisinformationen zu der selektierten Variablen sowie die PLC-Variablen-tabelle angezeigt werden.       |
|  | Zeigt an, dass das Byte durch einen Bytezugriff belegt ist und die entsprechende Variable nicht selektiert ist.  |
|  | Zeigt an, dass das Byte durch einen Wortzugriff belegt ist und die entsprechende Variable selektiert ist. Über das Kontextmenü können die Querverweisinformationen zu der selektierten Variablen sowie die PLC-Variablen-tabelle angezeigt werden.       |
|  | Zeigt an, dass das Byte durch einen Wortzugriff belegt ist und die entsprechende Variable nicht selektiert ist.  |
|  | Zeigt an, dass das Byte durch einen Doppelwortzugriff belegt ist und die entsprechende Variable selektiert ist. Über das Kontextmenü können die Querverweisinformationen zu der selektierten Variablen sowie die PLC-Variablen-tabelle angezeigt werden. |
|  | Zeigt an, dass das Byte durch einen Doppelwortzugriff belegt ist und die entsprechende Variable nicht selektiert ist.  |
| Hintergrundfarbe: grau  | Zeigt an, dass ein Byte-, Wort- oder Doppelwortzugriff vorhanden ist und dass die Adresse auch von der Hardware belegt ist. Die graue Hintergrundfarbe ist ein Hinweis auf einen überlappenden Speicherzugriff.  |
| Hintergrundfarbe: gelb  | Zeigt an, dass die Adresse von der Hardware nicht belegt ist.  |
|  | Zeigt an, dass der Speicherbereich als Systemmerker definiert ist.   |
|  | Zeigt an, dass der Speicherbereich als Taktmerker definiert ist.   |

## Siehe auch

Aufbau des Belegungsplans (Seite 1495)

Einführung zum Belegungsplan (Seite 1494)

### 9.2.2.4 Belegungsplan anzeigen

#### Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

#### Vorgehen

Um den Belegungsplan anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Belegungsplan".

#### Ergebnis

Der Belegungsplan zum selektierten Programm wird angezeigt.

#### Ansichtsoptionen im Belegungsplan

Beachten Sie die jeweils eingestellten Ansichtsoptionen, um die gewünschten Informationen im Belegungsplan angezeigt zu bekommen.

## Siehe auch

Ansichtsoptionen für Belegungsplan einstellen (Seite 1498)

Aufbau des Belegungsplans (Seite 1495)

### 9.2.2.5 Ansichtsoptionen für Belegungsplan einstellen

#### Einführung

Für den Belegungsplan stehen folgende Ansichtsoptionen zur Verfügung:


- **Verwendete Adressen:**  
Wenn dieses Optionskästchen aktiviert ist, werden die im Programm verwendeten Adressen, Peripherien und Pointer angezeigt.
- **Freie Hardware-Adressen:**  
Wenn dieses Optionskästchen aktiviert ist, werden nur die freien Hardware-Adressen angezeigt.

#### Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Der Belegungsplan ist geöffnet.

#### Vorgehen

Um die Ansichtsoptionen für den Belegungsplan einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf den Pfeil des Symbols  ("Ansichtsoptionen"). Die Ansichtsoptionen zum Belegungsplan werden geöffnet. Vor den aktivierten Ansichtsoptionen sind Häkchen gesetzt.
2. Wenn Sie eine Ansichtsoption aktivieren bzw. deaktivieren wollen, klicken Sie in das entsprechende Optionskästchen und setzen bzw. entfernen Sie das Häkchen.

#### Ergebnis

Die Ansichtsoptionen werden eingestellt und die gewünschten Informationen werden im Belegungsplan angezeigt.

### 9.2.2.6 Filtermöglichkeiten im Belegungsplan

#### Filtereinstellungen

Sie haben die Möglichkeit, selbst definierte Filtereinstellungen für den Belegungsplan festzulegen. Folgende Möglichkeiten für das Definieren von Filtern stehen Ihnen zur Verfügung:

- Anzeige aller verwendeten Adressen aus den nachfolgend genannten Adressbereichen.
- Anzeige einzelner, definierter Adressen, aus dem selektierten Adressbereich, z. B. "0" und "200".
- Anzeige kompletter Bereiche aus dem selektierten Adressbereich, z. B. "0 - 256".



Die folgende Tabelle zeigt alle Auswahlmöglichkeiten im Überblick:

| Selektion im  | Auswahl  | Symbol  | Bedeutung  |
|---------------|--|---|--|
| Adressbereich | Alle CPU-abhängig angezeigten Adressen (E, A, M, T, Z) wie in der Voreinstellung oder nur einzelne Adressbereiche können aktiviert werden. | Optionskästchen ist aktiviert   | Nur die aktivierten Adressbereiche (E, A, M, T, Z) werden im Belegungsplan angezeigt.            |
| Filterbereich | Belegung für alle Adressen anzeigen  | *   | Belegung aller verwendeten Adressen für die aktivierten Adressbereiche (E, A, M) wird angezeigt. |
|               | Belegung für ausgewählte Adressen anzeigen, z. B. für die Eingänge "EB 0" und "EB 256"   | 0;256<br>Einzelne Adressen und Bereiche sind durch Semikolon zu trennen.        | Belegung der genannten Adressen für die aktivierten Adressbereiche (E) wird angezeigt.           |
|               | Belegung für ausgewählte Bereiche anzeigen, z. B. für die Eingänge von "EB 0 bis EB 100" und von "EB 200 bis EB 256".                      | 0-100;200-256<br>Zusammenhängende Bereiche sind durch Bindestrich zu verbinden. | Belegung der genannten Bereiche für die aktivierten Adressbereiche (E) wird angezeigt.           |



### 9.2.2.7 Filter für Belegungsplan definieren

#### Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Der Belegungsplan ist geöffnet.

#### Filter definieren



Um einen Filter für den Belegungsplan zu definieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol  ("Filter").  
Der Dialog "Assignmentlist Filter" wird geöffnet.
2. Klicken Sie auf das Symbol  ("Neuen Filter erstellen").  
Ein neuer Filter wird unter dem Namen "Filter\_1" erstellt. Als Voreinstellung für den Filter werden CPU-abhängig die Optionskästchen aller Adressen (z. B. Eingänge, Ausgänge, Merker, Zeiten und Zähler) aktiviert.
3. Wenn Sie den Namen des Filters ändern wollen, klicken Sie in der Funktionsleiste auf die Klappliste und geben einen neuen Filternamen ein.
4. Deaktivieren Sie die Optionskästchen der Adressen, die das Filtern nicht beeinflussen sollen.

5. Tragen Sie im Filterbereich der aktivierten Adressen eine der folgenden Optionen ein:
  - Anzeige aller verwendeten Adressen = ""
  - Anzeige einzelner, definierter Adressen, z. B. EB 0 und EB 25 = "0,25". Einzelne Adressen und Adressbereiche werden durch Kommata oder Semikolon getrennt.
  - Anzeige komplette Adressbereiche, z. B. von EB 0 bis EB 256 = "0-256". Komplette Adressbereiche sind durch Bindestrich zu verbinden.
6. Bestätigen Sie die Eingaben mit "OK".  
Der neu definierte Filter wird unter dem vorgegebenen Namen in der Funktionsleiste des Belegungsplans angezeigt.

### Filter löschen

Um einen Filter zu löschen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol  ("Filter").  
Der Filter-Dialog für den Belegungsplan wird geöffnet.
2. Wählen Sie aus Klappliste in der Funktionsleiste den Filter aus, den Sie löschen wollen.
3. Klicken Sie auf das Symbol  ("Ausgewählten Filter löschen").  
Der ausgewählte Filter wird gelöscht.

### Siehe auch

Filtermöglichkeiten im Belegungsplan (Seite 1498)

Belegungsplan anzeigen (Seite 1497)

Einführung zum Belegungsplan (Seite 1494)

### 9.2.2.8 Belegungsplan filtern

#### Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Der Belegungsplan ist geöffnet.

#### Vorgehen

1. Klicken Sie in der Funktionsleiste auf den Pfeil der Klappliste.  
Die vorhandenen Filter werden angezeigt.
2. Wählen Sie den gewünschten Filter aus.

## Ergebnis

Der Belegungsplan wird nach den Einstellungen des ausgewählten Filters gefiltert.

---

### Hinweis

Die Filtereinstellungen werden beim Schließen des Projekts gespeichert.

---

## 9.2.2.9 Remanente Speicherbereiche für Merker definieren

### Einführung

Im Belegungsplan können Sie die Breite des remanenten Speicherbereichs für Merker definieren. Der Inhalt der Variablen, die im remanenten Speicherbereich adressiert sind, bleibt nach Ausschalten der Versorgungsspannung und nach Einschalten bei einem Übergang von STOP nach RUN erhalten.

Die Anzeige der remanenten Merker kann im Belegungsplan aktiviert und deaktiviert werden. Remanente Merker erkennen Sie bei aktivierter Anzeige am Icon in der Spalte "Adresse".

### Voraussetzung

Der Belegungsplan ist geöffnet.

### Vorgehen

Um die Breite des remanenten Speicherbereichs für Merker zu definieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Remanenz".  
Der Dialog "Remanenter Speicher" wird geöffnet.
2. Legen Sie die Breite des remanenten Speicherbereichs fest, indem Sie von 0 aus gezählt das letzte Byte des Bereichs in das Eingabefeld eingeben. Achten Sie dabei auf die Adressen vorhandener Variablen, die als remanent definiert sind.
3. Laden Sie die geänderten Bausteine anschließend ins Zielsystem. Selektieren Sie dazu den Ordner "Programmbausteine" in der Projektnavigation und wählen Sie im Kontextmenü das Untermenü "Laden in Gerät".

## Ergebnis

Die Breite des remanenten Speicherbereichs wird definiert. Falls die Anzeige der Remanenz im Belegungsplan aktiviert ist, wird in der Spalte "Adresse" für alle remanenten Variablen das entsprechende Icon angezeigt.

### 9.2.2.10 Anzeige der Remanenz für Merker aktivieren

#### Einführung

Im Belegungsplan können Sie die Anzeige der remanenten Merker aktivieren und deaktivieren. Wenn Sie die Anzeige der Remanenz aktivieren, werden die als remanent definierte Merker mit einem Icon in der Spalte "Adresse" gekennzeichnet.

#### Voraussetzung

Der Belegungsplan ist geöffnet.

#### Vorgehen

Um die Anzeige der remanenten Merker zu aktivieren und deaktivieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Remanenz ein/ausblenden".

#### Ergebnis

Wenn die Anzeige der Remanenz aktiviert ist, werden die als remanent definierten Variablen mit einem Icon in der Spalte "Adresse" des Merker-Bereichs gekennzeichnet. Wird die Anzeige der Remanenz deaktiviert, werden die Icons in der Spalte "Adresse" ausgeblendet.

## 9.2.3 Aufrufstruktur anzeigen

### 9.2.3.1 Einführung zur Aufrufstruktur

#### Aufrufstruktur

Die Aufrufstruktur beschreibt die Aufrufhierarchie der Bausteine innerhalb eines S7-Programms.

Sie gibt einen Überblick über:

- Verwendete Bausteine
- Sprünge zu den Verwendungsstellen der Bausteine
- Abhängigkeiten zwischen den Bausteinen
- Lokaldatenbedarf der Bausteine
- Status der Bausteine

## Informationen in der Aufrufstruktur

Mit der Anzeige der Aufrufstruktur erhalten Sie eine Liste der im Anwenderprogramm verwendeten Bausteine. Die erste Ebene der Aufrufstruktur ist farblich hervorgehoben und zeigt die Bausteine an, die von keinem anderen Baustein im Programm aufgerufen werden. Organisationsbausteine werden immer in der ersten Ebene der Aufrufstruktur angezeigt. Funktionen, Funktionsbausteine und Datenbausteine werden nur dann in der ersten Ebene angezeigt, wenn sie von keinem Organisationsbaustein aufgerufen werden. Wenn ein Baustein andere Bausteine oder Funktionen aufruft, werden diese unter dem Baustein eingerückt aufgelistet. Anweisungen und Bausteine werden in der Aufrufstruktur nur dann dargestellt, wenn sie von einem Baustein aufgerufen werden.

## Ansichtsoptionen

Für die Aufrufstruktur stehen folgende Ansichtsoptionen zur Verfügung:

- Nur Konflikte anzeigen:  
Wenn dieses Optionskästchen aktiviert ist, werden nur Konflikte innerhalb der Aufrufstruktur angezeigt.
- Mehrere Aufrufe zusammenfassen:  
Wenn dieses Optionskästchen aktiviert ist, werden mehrere Bausteinaufrufe zusammengefasst. Die Anzahl der Bausteinanrufe wird in der Spalte "Anzahl der Aufrufe" angezeigt. Die Links zu den verschiedenen Aufrufstellen werden in der Spalte "Details" in einer Klappliste angeboten.

## Anzeige der Bausteinaufrufe

Die in einem Baustein enthaltenen Bausteinaufrufe können Sie anzeigen, indem Sie auf den Pfeil vor dem Bausteintitel klicken. Um die Aufrufinformationen aller Bausteine einzublenden, klicken Sie in der Funktionsleiste auf das Symbol "Liste erweitern".

Das Zuklappen der Gesamtübersicht erfolgt über das Symbol "Liste reduzieren".

## Anzeige der Querverweisinformationen

Die Querverweisinformationen zu einem Baustein können Sie im Inspektorfenster einblenden, indem Sie auf den entsprechenden Baustein mit der rechten Maustaste klicken und im Kontextmenü den Befehl "Querverweis-Informationen" wählen.

Um die Ansicht "Querverweise" zu öffnen, klicken Sie im Kontextmenü auf den Befehl "Querverweise".

## Anzeige der Bausteine im Programmeditor

Aus der Aufrufstruktur heraus können Sie den Programmeditor öffnen und dort die Bausteine bearbeiten.

Dazu selektieren Sie den gewünschten Baustein in der Aufrufstruktur und wählen Sie im Kontextmenü den Befehl "Editor öffnen".

### Anzeige gelöschter Bausteine

Die Zeilen zu gelöschten Bausteinen werden mit einem Symbol gekennzeichnet.

---

#### Hinweis

Bitte beachten Sie, dass ggf. vorhandene Lokaldaten erst nach dem Übersetzen eines Bausteins angezeigt bzw. aktualisiert werden können.

---














### Siehe auch



Symbole in der Aufrufstruktur (Seite 1504)

### 9.2.3.2 Symbole in der Aufrufstruktur

#### Bedeutung der Symbole in der Aufrufstruktur

Die nachfolgende Tabelle zeigt die Bedeutung der Symbole in der Aufrufstruktur:

| Symbol  | Bedeutung  |
|---|--|
|  | Zeigt einen Organisationsbaustein (OB) an.   |
|  | Zeigt einen Funktionsbaustein (FB) an.   |
|  | Zeigt eine Funktion (FC) an.   |
|  | Zeigt einen Datenbaustein (DB) an.   |
|  | Zeigt an, dass der Baustein als Multiinstanz deklariert ist.                                     |
|  | Zeigt an, dass das Objekt eine Schnittstellen-Abhängigkeit zum links verbundenen Objekt besitzt. |
|  | Zeigt an, dass der Baustein neu übersetzt werden muss.   |
|  | Zeigt an, dass der Datenbaustein neu übersetzt werden muss.                                      |
|  | Zeigt an, dass das Objekt nicht vorhanden ist.   |
|  | Zeigt an, dass die Schnittstelle einen Zeitstempelkonflikt verursacht.                           |
|  | Zeigt an, dass eine Variable einen Zeitstempelkonflikt verursacht.                               |
|  | Zeigt an, dass der Baustein von einem OB nicht direkt oder indirekt aufgerufen wird.             |
|  | Zeigt an, dass ein Objekt mit Know-how-Schutz versehen ist.                                      |

| Symbol  | Bedeutung   |
|---|---|
|  | Zeigt an, dass der Baustein regulär rekursiv aufgerufen wird.   |
|  | <p>Zeigt an, dass eine Variablendeklaration in der Schnittstelle eine rekursive Abhängigkeit besitzt:</p> <ul style="list-style-type: none"> <li>• Fall 1: FB1 ruft FB2 auf und dieser wieder den FB1. Die Instanzdatenbausteine dieser FBs haben in der Schnittstelle eine Rekursion.</li> <li>• Fall 2: Ein Multiinstanz-FB verwendet den Instanz-DB seines Vater-FBs als Globalen DB.</li> </ul> |

### 9.2.3.3 Aufbau der Aufrufstruktur

#### Aufbau der Aufrufstruktur

Die Ansicht der Aufrufstruktur besteht aus folgenden Spalten:

| Spalte                     | Inhalt/Bedeutung   |
|----------------------------|--|
| Aufrufstruktur             | Zeigt die Übersicht der aufgerufenen Bausteine an. Bei aktivierter Ansichtsoption "Mehrere Aufrufe zusammenfassen" werden mehrere Bausteinaufrufe zusammengefasst und die Spalte "Anzahl der Aufrufe" wird eingeblendet.   |
| Aufrufart (!)              | Zeigt die Art des Aufrufs an, z. B. rekursiver Bausteinaufruf.   |
| Adresse                    | Zeigt die absolute Adresse des Bausteins an. Bei einem Funktionsbaustein wird die absolute Adresse des zugehörigen Instanzdatenbausteins mit angezeigt.  |
| Anzahl der Aufrufe         | Zeigt die Anzahl der Mehrfachaufrufe von Bausteinen an.  |
| Details                    | Zeigt das Netzwerk bzw. die Schnittstelle des aufrufenden Bausteins an. In dieser Spalte sind alle Informationen als Link angeboten. Über diesen Link erreichen Sie im Programmierer die Stelle, die den Bausteinaufruf enthält. Bei aktivierter Ansichtsoption "Mehrere Aufrufe zusammenfassen" werden die Aufrufe zusammengefasst und die Links in einer Klappliste angeboten. |
| Lokaldaten (im Pfad)       | <p>Zeigt den Lokaldatenbedarf des gesamten Pfades an. Bausteine mit optimiertem Zugriff haben einen höheren Lokaldatenbedarf, da die Information für die symbolische Adressierung mit hinterlegt wird.</p> <p>Bitte beachten Sie, dass ggf. vorhandene Lokaldaten erst nach dem Übersetzen eines Bausteins angezeigt bzw. aktualisiert werden können.</p>                        |
| Lokaldaten (für Bausteine) | <p>Zeigt den Lokaldatenbedarf des Bausteins an. Bausteine mit optimiertem Zugriff haben einen höheren Lokaldatenbedarf, da die Information für die symbolische Adressierung mit hinterlegt wird.</p> <p>Bitte beachten Sie, dass ggf. vorhandene Lokaldaten erst nach dem Übersetzen eines Bausteins angezeigt bzw. aktualisiert werden können.</p>                              |

### Siehe auch

Symbole in der Aufrufstruktur (Seite 1504)

Einführung zur Konsistenzprüfung in der Aufrufstruktur (Seite 1508)

### 9.2.3.4 Aufrufstruktur anzeigen

#### Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

#### Vorgehen

Um die Aufrufstruktur anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Ordner "Programmbausteine" bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Aufrufstruktur".

#### Ergebnis

Die Aufrufstruktur zum selektierten Programm wird angezeigt.

---

#### Hinweis

Bitte beachten Sie, dass ggf. vorhandene Lokaldaten erst nach dem Übersetzen eines Bausteins angezeigt bzw. aktualisiert werden können.

---

### Siehe auch

Ansichtsoptionen für die Aufrufstruktur einstellen (Seite 1507)



### 9.2.3.5 Ansichtsoptionen für die Aufrufstruktur einstellen

#### Einführung

Für die Aufrufstruktur stehen folgende Ansichtsoptionen zur Verfügung:


- Nur Konflikte anzeigen:  
Wenn dieses Optionskästchen aktiviert ist, werden nur konfliktbehaftete Bausteine innerhalb der Aufrufstruktur angezeigt.  
Folgende Bausteine verursachen einen Konflikt:
  - Bausteine, die bezüglich eines ihrer Aufrufe einen älteren oder jüngeren Code-Zeitstempel haben.
  - Bausteine, die einen Baustein aufrufen, dessen Schnittstelle sich geändert hat.
  - Bausteine, die eine Variable verwenden, deren Adresse und/oder Datentyp sich geändert hat.
  - Bausteine, die weder direkt noch indirekt von einem OB aufgerufen werden.
  - Bausteine, die einen Baustein aufrufen, der nicht mehr existiert.
- Mehrere Aufrufe zusammenfassen:  
Wenn diese Ansichtsoption aktiviert ist, werden mehrere Bausteinaufrufe und Datenbausteinzugriffe zusammengefasst. Die Anzahl der Bausteinanrufe wird in der Spalte "Anzahl der Aufrufe" angezeigt. Die Links zu den verschiedenen Aufrufstellen werden in der Spalte "Details" in einer Klappliste angeboten.

#### Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Die Aufrufstruktur ist geöffnet.

#### Vorgehen

Um die Ansichtsoptionen für die Aufrufstruktur einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf den Pfeil des Symbols  ("Ansichtsoptionen"). Die Ansichtsoptionen zur Aufrufstruktur werden geöffnet. Vor den aktivierten Ansichtsoptionen sind Häkchen gesetzt.
2. Wenn Sie eine Ansichtsoption aktivieren bzw. deaktivieren wollen, klicken Sie in das entsprechende Optionskästchen und setzen bzw. entfernen Sie das Häkchen.

#### Ergebnis

Die Ansichtsoptionen werden eingestellt und die gewünschten Informationen werden in der Aufrufstruktur angezeigt.

### 9.2.3.6 Einführung zur Konsistenzprüfung in der Aufrufstruktur

#### Konsistenzprüfung

Änderungen des Zeitstempels eines Bausteins während oder nach der Programmerstellung können zu Zeitstempel-Konflikten führen, die wiederum Inkonsistenzen zwischen den aufrufenden und aufgerufenen Bausteinen verursachen.

#### Verwendung der Konsistenzprüfung

Die Funktion "Konsistenzprüfung" dient zur Visualisierung von Inkonsistenzen bei Zeitstempel-Konflikten. Wenn die Konsistenzprüfung durchgeführt wird, werden die inkonsistenten Bausteine in der Aufrufstruktur angezeigt und mit den entsprechenden Symbolen gekennzeichnet.

- Ein großer Teil der Zeitstempel- und Schnittstellenkonflikte können entfernt werden, indem die Bausteine neu übersetzt werden.
- Wenn sich Inkonsistenzen durch die Übersetzung nicht bereinigen lassen, können Sie über den Link in der Spalte "Details" zu der genauen Problemstelle im Programmeditor wechseln und die Inkonsistenzen manuell bereinigen.
- Die rot gekennzeichneten Bausteine müssen neu übersetzt werden.

#### Siehe auch

Symbole in der Aufrufstruktur (Seite 1504)


### 9.2.3.7 Bausteinkonsistenz in der Aufrufstruktur prüfen

#### Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Die Aufrufstruktur ist geöffnet.

#### Vorgehen

Um die Bausteinkonsistenz zu prüfen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf das Symbol  ("Konsistenzprüfung"). Die Bausteinkonsistenz wird geprüft. Die Bausteine, die Inkonsistenzen aufweisen, werden durch das entsprechende Symbol gekennzeichnet.
2. Falls ein Baustein Inkonsistenzen aufweist, klicken Sie in der Aufrufstruktur auf den Pfeil vor dem entsprechenden Bausteintitel. Die inkonsistenten Bausteine werden angezeigt. In der Spalte "Details" werden die genauen Problemstellen als Links aufgelistet.

3. Klicken Sie in der Spalte "Details" auf den entsprechenden Link, um an die zu bereinigende Stelle im jeweiligen Baustein zu springen.
4. Überprüfen und bereinigen Sie die Inkonsistenzen in den Bausteinen.
5. Übersetzen Sie die Bausteine neu indem Sie die gewünschten Bausteine selektieren und im Kontextmenü auf den Befehl "Übersetzen" klicken.
6. Laden Sie die korrigierten Bausteine ins Zielsystem, indem Sie im Kontextmenü auf den Befehl "Laden in Gerät" klicken.

## Ergebnis

Die Bausteinkonsistenz wird geprüft. Die Inkonsistenzen der Bausteine werden bereinigt. Die korrigierten Bausteine sind ins Zielsystem geladen.

## Siehe auch

Symbole in der Aufrufstruktur (Seite 1504)

## 9.2.4 Abhängigkeitsstruktur anzeigen

### 9.2.4.1 Einführung zur Abhängigkeitsstruktur

#### Einführung

Die Abhängigkeitsstruktur zeigt für jeden Baustein im Programm die Abhängigkeit zu anderen Bausteinen an.

#### Informationen in der Abhängigkeitsstruktur

Mit der Anzeige der Abhängigkeitsstruktur erhalten Sie eine Liste der im Anwenderprogramm verwendeten Bausteine. Ganz links steht der jeweilige Baustein und darunter eingerückt die Bausteine, die diesen Baustein aufrufen oder verwenden.

Die Abhängigkeitsstruktur zeigt zusätzlich den Status der einzelnen Bausteine durch Symbole an.

Objekte, die einen Zeitstempelkonflikt verursachen und zu einer Inkonsistenz im Programm führen können, werden mit verschiedenen Symbolen gekennzeichnet.

Die Abhängigkeitsstruktur stellt eine Erweiterung der Querverweisliste für Objekte dar.

### Ansichtsoptionen

Für die Abhängigkeitsstruktur stehen folgende Ansichtsoptionen zur Verfügung:

- Nur Konflikte anzeigen:  
Wenn dieses Optionskästchen aktiviert ist, werden nur Konflikte innerhalb der Abhängigkeitsstruktur angezeigt.
- Mehrere Aufrufe zusammenfassen:  
Wenn dieses Optionskästchen aktiviert ist, werden mehrere Bausteinaufrufe zusammengefasst. Die Anzahl der Bausteinanrufe wird in der Spalte "Abhängigkeitsstruktur" numerisch angezeigt. Die Links zu den verschiedenen Aufrufstellen werden in der Spalte "Details" in einer Klappliste angeboten.

### Anzeige der Abhängigkeiten

Wenn Sie auf den Pfeil vor dem Bausteintitel klicken, zeigen sie die Bausteine an, in denen dieser Baustein aufgerufen bzw. verwendet wird. Um die Abhängigkeiten aller Bausteine einzublenden,

klicken Sie in der Funktionsleiste auf das Symbol "Liste erweitern".

Das Zuklappen der Gesamtübersicht erfolgt über das Symbol "Liste reduzieren".

### Anzeige der Querverweisinformationen

Die Querverweisinformationen zu einem Baustein können Sie im Inspektorfenster einblenden, indem Sie auf den entsprechenden Baustein mit der rechten Maustaste klicken und im Kontextmenü den Befehl "Verwendung anzeigen" wählen.

### Anzeige der Bausteine im Programmeditor

Aus der Abhängigkeitsstruktur heraus können Sie den Programmeditor öffnen und dort die Bausteine bearbeiten. Dazu selektieren Sie den gewünschten Baustein in der Abhängigkeitsstruktur und wählen Sie im Kontextmenü den Befehl "Öffnen".

#### 9.2.4.2 Aufbau der Abhängigkeitsstruktur

### Aufbau der Abhängigkeitsstruktur

Die Ansicht der Abhängigkeitsstruktur besteht aus folgenden Spalten:

| Spalte        | Inhalt/Bedeutung   |
|---------------|--|
| Abhängigkeit  | Zeigt in einer Übersicht für jeden Baustein die Abhängigkeiten zu anderen Bausteinen an. |
| Aufrufart (!) | Zeigt die Art des Aufrufs an, z. B. rekursiver Bausteinaufruf.                           |
| Adresse       | Zeigt die absolute Adresse des Bausteins an.   |

| Spalte             | Inhalt/Bedeutung  |
|--------------------|---|
| Anzahl der Aufrufe | Zeigt die Anzahl der Mehrfachaufrufe von Bausteinen an.   |
| Details            | Zeigt das Netzwerk bzw. die Schnittstelle des aufgerufenen Bausteins an. In dieser Spalte sind alle Informationen als Link angeboten. Über diesen Link erreichen Sie im Programmierer die Stelle, die den Bausteinaufruf enthält. Bei aktivierter Ansichtsoption "Mehrere Aufrufe zusammenfassen" werden die Aufrufe zusammengefasst und die Links in einer Klappliste angeboten. |











**Siehe auch**

Symbole in der Abhängigkeitsstruktur (Seite 1511)

**9.2.4.3 Symbole in der Abhängigkeitsstruktur**

**Bedeutung der Symbole in der Abhängigkeitsstruktur**

Die nachfolgende Tabelle zeigt die Bedeutung der Symbole in der Abhängigkeitsstruktur:

| Symbol  | Bedeutung  |
|---|--|
|  | Zeigt einen Organisationsbaustein (OB) an.   |
|  | Zeigt einen Funktionsbaustein (FB) an.   |
|  | Zeigt eine Funktion (FC) an.   |
|  | Zeigt einen Datenbaustein (DB) an.   |
|  | Zeigt an, dass das Objekt eine Schnittstellen-Abhängigkeit zum links verbundenen Objekt besitzt.   |
|  | Zeigt an, dass der Baustein neu übersetzt werden muss.   |
|  | Zeigt an, dass der Datenbaustein neu übersetzt werden muss.  |
|  | Zeigt an, dass für dieses Objekt eine Inkonsistenz vorliegt.   |
|  | Zeigt an, dass ein Objekt mit Know-how-Schutz versehen ist.  |
|  | Zeigt an, dass eine Variablendeklaration in der Schnittstelle eine rekursive Abhängigkeit besitzt: <ul style="list-style-type: none"> <li>Fall 1: FB1 ruft FB2 auf und dieser wieder den FB1. Die Instanzdatenbausteine dieser FBs haben in der Schnittstelle eine Rekursion.</li> <li>Fall 2: Ein Multiinstanz-FB verwendet den Instanz-DB seines Vater-FBs als Globalen DB.</li> </ul> |

#### 9.2.4.4 Abhängigkeitsstruktur anzeigen

##### Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

##### Vorgehen

Um die Abhängigkeitsstruktur anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Bausteinordner bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Abhängigkeitsstruktur".

##### Ergebnis

Die Abhängigkeitsstruktur zum selektierten Programm wird angezeigt.

##### Siehe auch

Ansichtsoptionen für die Abhängigkeitsstruktur einstellen (Seite 1512)

#### 9.2.4.5 Ansichtsoptionen für die Abhängigkeitsstruktur einstellen

##### Einführung

Für die Abhängigkeitsstruktur stehen folgende Ansichtsoptionen zur Verfügung:


- Nur Konflikte anzeigen:  
Wenn dieses Optionskästchen aktiviert ist, werden nur Konflikte innerhalb der Abhängigkeitsstruktur angezeigt.  
Folgende Bausteine verursachen einen Konflikt:
  - Bausteine, die bezüglich eines ihrer Aufrufe einen älteren oder jüngeren Code-Zeitstempel haben.
  - Bausteine, die von einem Baustein aufgerufen werden, dessen Schnittstelle sich geändert hat.
  - Bausteine, die eine Variable verwenden, deren Adresse und/oder Datentyp sich geändert hat.
  - Bausteine, die weder direkt noch indirekt von einem OB aufgerufen werden.
- Mehrere Aufrufe zusammenfassen:  
Wenn dieses Optionskästchen aktiviert ist, werden mehrere Bausteinaufrufe zusammengefasst. Die Anzahl der Bausteinaufrufe wird in der entsprechenden Spalte angezeigt. Die Links zu den verschiedenen Aufrufstellen werden in der Spalte "Details" in einer Klappliste angeboten.

## Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Die Abhängigkeitsstruktur ist geöffnet.

## Vorgehen

Um die Ansichtsoptionen für die Abhängigkeitsstruktur einzustellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf den Pfeil des Symbols  ("Ansichtsoptionen"). Die Ansichtsoptionen zur Abhängigkeitsstruktur werden geöffnet. Vor den aktivierten Ansichtsoptionen sind Häkchen gesetzt.
2. Wenn Sie eine Ansichtsoption aktivieren bzw. deaktivieren wollen, klicken Sie in das entsprechende Optionskästchen und setzen bzw. entfernen Sie das Häkchen.

## Ergebnis

Die Ansichtsoptionen werden eingestellt und die gewünschten Informationen werden in der Abhängigkeitsstruktur angezeigt.

### 9.2.4.6 Einführung zur Konsistenzprüfung in der Abhängigkeitsstruktur

## Konsistenzprüfung

Änderungen des Zeitstempels eines Bausteins während oder nach der Programmerstellung können zu Zeitstempel-Konflikten führen, die wiederum Inkonsistenzen zwischen den aufrufenden und aufgerufenen Bausteinen verursachen.

## Verwendung der Konsistenzprüfung

Die Funktion "Konsistenzprüfung" dient zur Visualisierung von Inkonsistenzen. Wenn die Konsistenzprüfung durchgeführt wird, werden die inkonsistenten Bausteine in der Abhängigkeitsstruktur angezeigt und mit den entsprechenden Symbolen gekennzeichnet.

- Ein großer Teil der Zeitstempel- und Schnittstellenkonflikte können entfernt werden, indem die Bausteine neu übersetzt werden.
- Wenn sich Inkonsistenzen durch die Übersetzung nicht bereinigen lassen, können Sie über den Link in der Spalte "Details" zu der genauen Problemstelle im Programmmeditor wechseln und die Inkonsistenzen manuell bereinigen.
- Die rot gekennzeichneten Bausteine müssen neu übersetzt werden.

### Siehe auch

Aufbau der Abhängigkeitsstruktur (Seite 1510)

Symbole in der Abhängigkeitsstruktur (Seite 1511)


### 9.2.4.7 Bausteinkonsistenz in der Abhängigkeitsstruktur prüfen

#### Voraussetzung

- Ein Projekt mit programmierten Bausteinen ist angelegt.
- Die Abhängigkeitsstruktur ist geöffnet.

#### Vorgehen

Um die Bausteinkonsistenz zu prüfen, gehen Sie folgendermaßen vor:

1. Klicken in der Funktionsleiste auf das Symbol  ("Konsistenzprüfung"). Die Bausteinkonsistenz wird geprüft. Die Bausteine, die Inkonsistenzen aufweisen, werden durch das entsprechende Symbol gekennzeichnet.
2. Falls ein Baustein Inkonsistenzen aufweist, klicken Sie in der Abhängigkeitsstruktur auf den Pfeil vor dem entsprechenden Bausteintitel. Die inkonsistenten Bausteine werden angezeigt. In der Spalte "Details" werden die genauen Problemstellen als Links aufgelistet.
3. Überprüfen und bereinigen Sie die Inkonsistenzen in den Bausteinen.
4. Übersetzen Sie die Bausteine neu indem Sie die gewünschten Bausteine selektieren und im Kontextmenü auf den Befehl "Übersetzen" klicken.
5. Laden Sie die korrigierten Bausteine ins Zielsystem, indem Sie im Kontextmenü auf den Befehl "Laden in Gerät" klicken.

#### Ergebnis

Die Bausteinkonsistenz wird geprüft. Die Inkonsistenzen der Bausteine werden bereinigt. Die korrigierten Bausteine sind ins Zielsystem geladen.

### Siehe auch

Symbole in der Abhängigkeitsstruktur (Seite 1511)



## 9.2.5 Speicherauslastung der CPU anzeigen

### 9.2.5.1 Einführung zur Speicherauslastung

#### Einführung

Das Register "Speicherauslastung" zeigt die jeweilige Hardware-Speicherauslastung der projektierten CPU für:

- die verwendeten Programmierobjekte,
- die Belegung der unterschiedlichen Speicherbereiche innerhalb der CPU und
- die belegten Ein- und Ausgänge auf den vorhandenen Ein-/Ausgabebaugruppen an.

#### Informationen im Register "Speicherauslastung"

Mit der Anzeige der Speicherauslastung erhalten Sie einen Überblick über die Hardware-Speicherauslastung. Die Anzeige in diesem Register ist abhängig von der von Ihnen verwendeten CPU. Folgende Informationen werden angezeigt:

- die in der CPU vorhandenen Programmier-Objekte (z. B. OB, FC, FB, DB, Datentypen und PLC-Variablen),
- die in der CPU vorhandenen Speicherbereiche (Ladespeicher, Arbeitsspeicher, je nach CPU unterteilt in Code-Arbeitsspeicher und Daten-Arbeitsspeicher und Remanenzspeicher), deren maximale Größe und deren Belegung durch die o. g. verwendeten Programmier-Objekte,
- die für die CPU konfigurierbaren Ein- und Ausgänge auf den zur Verfügung stehenden Baugruppen (Ein-/Ausgabebaugruppen, Digitale Eingabebaugruppen, Digitale Ausgabebaugruppen, Analoge Eingabebaugruppen und Analoge Ausgabebaugruppen) und die davon bereits belegten Ein- und Ausgänge.

#### Anzeige des maximal verfügbaren Ladespeichers

Die Größe des maximal verfügbaren Ladespeichers können Sie in der Spalte "Ladespeicher" in der Zeile "Gesamt" über eine Auswahlliste auswählen.

#### Anzeige des maximal verfügbaren Arbeitsspeichers

Die Größe des maximal verfügbaren Arbeitsspeichers wird in der Spalte "Arbeitsspeicher" bzw. in den Spalten "Code-Arbeitsspeicher" und "Daten-Arbeitsspeicher" in der Zeile "Gesamt" angezeigt.

### Anzeige des maximal verfügbaren Remanenzspeichers

Die Größe des maximal verfügbaren Remanenzspeichers wird in der Spalte "Remanenzspeicher" in der Zeile "Gesamt" angezeigt.

---

#### Hinweis

##### Remanenzdaten

Als remanent definierte Merker und Datenbausteine werden in die Berechnung der Remanenzdaten mit einbezogen.

---

### Aktualisieren der Anzeige im Register "Speicherauslastung"

Um die Anzeige der gezeigten Objekte zu aktualisieren, klicken Sie in der Funktionsleiste auf das Symbol "Ansicht aktualisieren".

### Nutzen der Anzeige im Register "Speicherauslastung"

Im Register "Speicherauslastung" der Programminformationen erhalten Sie eine detaillierte Auflistung für jedes verwendete Objekt und den dazugehörigen belegten Speicherbereich.

Eine Überbelegung der CPU wird ebenfalls angezeigt und kann somit vermieden werden.

Nicht kompilierte Bausteine sind identifizierbar, da deren Größe durch Fragezeichen angezeigt wird.

### Siehe auch

Aufbau des Registers "Speicherauslastung" (Seite 1517)

Speicherauslastung anzeigen (Seite 1518)

Maximal verfügbaren Ladespeicher auswählen (Seite 1519)

### 9.2.5.2 Aufbau des Registers "Speicherauslastung"

#### Aufbau des Registers "Speicherauslastung" in den Programminformationen

Die Ansicht des Registers "Speicherauslastung" besteht aus folgenden Spalten:

| Spalte   | Inhalt/Bedeutung   |
|--|--|
| Objekte  | Zeigt unter "Details" eine Übersicht der in der CPU vorhandenen Programmier-Objekte und deren jeweilige Speicherbelegung an.   |
| Ladespeicher   | <p>Zeigt die maximale Speicherauslastung für den Ladespeicher der CPU in Prozentwerten und absolut an.</p> <p>Die unter "Gesamt" angezeigten Werte geben Auskunft über den im Ladespeicher maximal zur Verfügung stehenden Speicherplatz.</p> <p>Die unter "Belegt" angezeigten Werte geben Auskunft über den im Ladespeicher tatsächlich belegten Speicherplatz.</p> <p>Wird ein Wert in rot dargestellt, so ist die vorhandene Speicherkapazität überschritten.</p>  |
| Arbeitsspeicher<br>bzw.<br>Code- und Daten-Arbeitsspeicher | <p>Zeigt die maximale Speicherauslastung für den Arbeitsspeicher der CPU in Prozentwerten und absolut an.</p> <p>Der Arbeitsspeicher ist CPU-abhängig und ist z. B. für eine CPU aus der Familie S7-400 und S7-1500 unterteilt in "Code-Arbeitsspeicher" und "Daten-Arbeitsspeicher".</p> <p>Die unter "Gesamt" angezeigten Werte geben Auskunft über den im Arbeitsspeicher maximal zur Verfügung stehenden Speicherplatz.</p> <p>Die unter "Belegt" angezeigten Werte geben Auskunft über den im Arbeitsspeicher tatsächlich belegten Speicherplatz.</p> <p>Wird ein Wert in rot dargestellt, so ist die vorhandene Speicherkapazität überschritten.</p> |
| Remanenzspeicher   | <p>Zeigt die maximale Speicherauslastung für den remanenten Speicherbereich der CPU in Prozentwerten und absolut an.</p> <p>Die unter "Gesamt" angezeigten Werte geben Auskunft über den im Remanenzspeicher maximal zur Verfügung stehenden Speicherplatz.</p> <p>Die unter "Belegt" angezeigten Werte geben Auskunft über den im Remanenzspeicher tatsächlich belegten Speicherplatz.</p> <p>Wird ein Wert in rot dargestellt, so ist die vorhandene Speicherkapazität überschritten.</p>  |

| Spalte            | Inhalt/Bedeutung   |
|-------------------|--|
| E/A               | <p>Zeigt die auf der CPU zur Verfügung stehenden Ein- und Ausgänge an, die in den nachfolgenden Spalten typenspezifisch angezeigt werden.</p> <p>Die unter "Konfiguriert" angezeigten Werte geben Auskunft über die maximal zur Verfügung stehenden Ein- und Ausgänge.</p> <p>Die unter "Belegt" angezeigten Werte geben Auskunft über die tatsächlich belegten Ein- und Ausgänge.</p>   |
| DE / DA / AE / AA | <p>Zeigt die Anzahl der projektierten und genutzten Ein-/Ausgänge an:</p> <p>DE = Digitaleingänge<br/>                     DA = Digitalausgänge<br/>                     AE = Analogeingänge<br/>                     AA = Analogausgänge</p> <p>Die unter "Konfiguriert" angezeigten Werte geben Auskunft über die maximal zur Verfügung stehenden Ein- und Ausgänge.</p> <p>Die unter "Belegt" angezeigten Werte geben Auskunft über die tatsächlich belegten Ein- und Ausgänge.</p> |

**Siehe auch**

- Speicherauslastung anzeigen (Seite 1518)
- Maximal verfügbaren Ladespeicher auswählen (Seite 1519)
- Einführung zur Speicherauslastung (Seite 1515)

**9.2.5.3 Speicherauslastung anzeigen**

**Voraussetzung**

Ein Projekt mit programmierten Bausteinen ist angelegt.

**Vorgehen**

Um die Speicherauslastung für die jeweiligen Speicherbereiche der CPU anzuzeigen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Bausteinordner unterhalb der gewünschten CPU bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Speicherauslastung".

**Ergebnis**

Die Speicherauslastung der zugeordneten CPU wird angezeigt.

#### 9.2.5.4 Maximal verfügbaren Ladespeicher auswählen

##### Voraussetzung

Ein Projekt mit programmierten Bausteinen ist angelegt.

##### Vorgehen

Um in der Speicherauslastung den maximal verfügbaren Ladespeicher auszuwählen, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Bausteinordner unterhalb der gewünschten CPU, bzw. einen oder mehrere der darin enthaltenen Bausteine.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Speicherauslastung".
3. Öffnen Sie im angezeigten Dialog in der Spalte "Ladespeicher" im Feld "Gesamt" die Klappliste, indem Sie auf das Symbol klicken.
4. Wählen Sie aus der Klappliste den entsprechenden Wert für die verwendete CPU aus, indem Sie darauf klicken.

##### Ergebnis

Im Feld "Gesamt" wird die ausgewählte maximale Speicherauslastung angezeigt.

---

##### Hinweis

##### Anzeige der maximalen Speicherauslastung

Wird bei der Speicherauslastung ein Wert in rot dargestellt, so ist die verfügbare Speicherkapazität überschritten.

Passen Sie in diesem Fall die Speicherkapazität wie oben beschrieben an.

---

## 9.3 Querverweise anzeigen

### 9.3.1 Allgemeine Informationen zu Querverweisen

#### Einführung

Die Querverweisliste bietet einen Überblick über die Verwendung von Operanden und Variablen innerhalb des Anwenderprogramms.

### Nutzen der Querverweise

Die Querverweisliste bietet Ihnen folgende Vorteile:

- Bei der Programmerstellung und bei Änderungen behalten Sie die Übersicht über verwendete Operanden, Variablen und Bausteinaufrufe.
- Von den Querverweisen aus können Sie direkt an die jeweilige Verwendungsstelle von Operanden und Variablen springen.
- Bei einem Programmtest oder bei der Fehlersuche erfahren Sie:
  - welcher Operand in welchem Baustein mit welchem Befehl bearbeitet wird,
  - welche Variable in welchem Bild verwendet wird,
  - welcher Baustein von welchem anderen Baustein aufgerufen wird.
- Als Teil der Projektdokumentation liefern die Querverweise einen umfassenden Überblick über alle verwendeten Operanden, Speicherbereiche, Bausteine, Variablen und Bilder.

### Siehe auch

Aufbau der Querverweisliste (Seite 1520)

Querverweisliste anzeigen (Seite 1522)

Querverweise im Inspektorfenster anzeigen (Seite 1523)

## 9.3.2 Aufbau der Querverweisliste

### Ansichten der Querverweisliste

Es gibt zwei Ansichten der Querverweisliste, die sich darin unterscheiden, welche Objekte jeweils in der ersten Spalte angezeigt werden:

- **Verwendet von:**  
Anzeige der referenzierten Objekte. Hier wird angezeigt, an welchen Stellen das Objekt verwendet wird.
- **Verwendet:**  
Anzeige der referenzierenden Objekte. Hier werden die Verwender des Objektes angezeigt.

Die zugeordneten Tooltips geben weitere Informationen zu den jeweiligen Objekten.

### Aufbau der Querverweisliste

Die Querverweisliste hat folgenden Aufbau:

| Spalte            | Inhalt/Bedeutung  |
|-------------------|---|
| Objekt            | Name des Objektes, das die unterlagerten Objekte verwendet, bzw. von diesen verwendet wird. |
| Anzahl            | Anzahl der Verwendungen   |
| Verwendungsstelle | Jeweilige Verwendungsstelle, z. B. Netzwerk   |

| Spalte      | Inhalt/Bedeutung   |
|-------------|--|
| Eigenschaft | Spezielle Eigenschaften der referenzierten Objekte, z. B. den Namen von Variablen bei Multiinstanz-Deklarationen.          |
| als         | Zeigt zusätzliche Informationen zum Objekt an, z. B. dass ein Instanz-DB als Vorlage oder als Multiinstanz verwendet wird. |
| Zugriff     | Zugriffsart, ob es sich um einen lesenden (R) und/oder schreibenden (W) Zugriff auf den Operanden handelt.                 |
| Adresse     | Adresse des Operanden  |
| Typ         | Informationen über den Typ und die Erstsprache des Objektes  |
| Pfad        | Pfad des Objektes im Projektnavigationsbaum  |

Abhängig von den installierten Produkten werden zusätzliche oder andere Spalten für die Querverweise angezeigt.

### Einstellungen in der Querverweisliste

Folgende Einstellungen können über die Symbole in der Funktionsleiste für die Querverweisliste vorgenommen werden:

- Querverweisliste aktualisieren  
Aktualisiert die aktuelle Querverweisliste.
- Einstellungen für die Querverweisliste festlegen  
Hier legen Sie durch Anklicken der entsprechenden Optionskästchen fest, ob alle verwendeten, alle nicht verwendeten, alle definierten bzw. alle nicht definierten Objekte angezeigt werden sollen. Wenn die Option "nicht definierte Objekte anzeigen" aktiviert ist, werden auch Referenzen zu bereits gelöschten Objekten angezeigt.
- Einträge reduzieren  
Reduziert die Einträge in der aktuellen Querverweisliste, indem die unterlagerten Objekte geschlossen werden.
- Einträge erweitern  
Erweitert die Einträge in der aktuellen Querverweisliste, indem die unterlagerten Objekte geöffnet werden.

### Sortieren in der Querverweisliste

Die Einträge in der Spalte "Objekt" und, je nach installierten Produkten, andere Spalten können auf- bzw. absteigend sortiert werden. Klicken Sie dazu auf die jeweilige Spaltenüberschrift.

### Siehe auch

Allgemeine Informationen zu Querverweisen (Seite 1519)

Querverweisliste anzeigen (Seite 1522)

### 9.3.3 Querverweisliste anzeigen

#### Voraussetzung

Ein Projekt ist angelegt.

#### Einführung

Sie haben verschiedene Möglichkeiten, Querverweise anzuzeigen, abhängig davon, ob Sie sich in der Portalansicht oder der Projektansicht befinden und welches Objekt Sie ggf. in der Projektnavigation selektiert haben.

In der Portalansicht können Sie Querverweise immer nur für die gesamte CPU anzeigen, in der Projektansicht können z. B. für folgende Objekte Querverweise angezeigt werden:

- Ordner "PLC-Variablen"
- Ordner "PLC-Datentypen"
- Ordner "Programmbausteine"
- Ordner "Variablen und Verbindungen"
- Einzelne Variablen
- Einzelne PLC-Datentypen
- Einzelne Bausteine
- Technologische Objekte

#### Querverweise anzeigen

Um Querverweise anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Portalansicht die gewünschte Aktion aus, z.B. "PLC programmieren" und den Befehl "Querverweise anzeigen" oder selektieren Sie in der Projektansicht eines der o. g. Objekte und wählen Sie im Menü "Werkzeuge" den Befehl "Querverweise". Die Querverweisliste wird angezeigt.
2. Klicken Sie auf die Schaltfläche "Verwendet von", um anzuzeigen, an welchen Stellen die in der Querverweisliste angezeigten Objekte verwendet werden.
3. Klicken Sie auf die Schaltfläche "Verwendet", um die Verwender der in der Querverweisliste angezeigten Objekte zu sehen.
4. Über die Symbole in der Funktionsleiste können Sie folgende Aktionen durchführen:
  - Querverweisliste aktualisieren
  - Einstellungen für die Querverweisliste festlegen
  - Einträge reduzieren
  - Einträge erweitern



5. Um die Einträge in der Spalte "Objekt" und in der Spalte "Adresse" auf- bzw. absteigend zu sortieren, klicken Sie auf die jeweilige Spaltenüberschrift.
6. Um an die Verwendungsstelle des jeweiligen Objektes zu gelangen, klicken Sie ggf. auf den angezeigten Link.

## Siehe auch

Allgemeine Informationen zu Querverweisen (Seite 1519)

Aufbau der Querverweisliste (Seite 1520)

## 9.3.4 Querverweise im Inspektorfenster anzeigen

### Einführung

Im Inspektorfenster werden die Querverweisinformationen zu einem selektierten Objekt im Register "Info > Querverweise" angezeigt. In diesem Register wird für jedes selektierte Objekt gezeigt, an welchen Stellen und von welchen anderen Objekten es verwendet wird.

Im Inspektorfenster werden auch solche Bausteine bei den Querverweisen angezeigt, die nur online vorhanden sind.

Über den Kontextmenübefehl "Überlappende Zugriffe anzeigen..." können Sie für selektierte Objekte auch überlappende Zugriffe über Bausteingrenzen hinweg anzeigen lassen.

### Aufbau

Die Querverweisinformationen werden im Inspektorfenster tabellarisch dargestellt. Jede Spalte enthält bestimmte Detailinformationen zu dem selektierten Objekt und dessen Verwendung. Die folgende Tabelle zeigt, welche Detailinformationen im Register "Info > Querverweis" aufgelistet werden:

| Spalte            | Bedeutung  |
|-------------------|--|
| Objekt            | Name des Objektes, das die unterlagerten Objekte verwendet, bzw. von diesen verwendet wird.                                |
| Anzahl            | Anzahl der Verwendungen  |
| Verwendungsstelle | Jeweilige Verwendungsstelle, z.B. Netzwerk   |
| Eigenschaft       | Spezielle Eigenschaften der referenzierten Objekte, z.B. den Namen der Variablen bei Multiinstanz-Deklarationen            |
| als               | Zeigt zusätzliche Informationen zum Objekt an, z. B. dass ein Instanz-DB als Vorlage oder als Multiinstanz verwendet wird. |
| Zugriff           | Zugriffsart<br>Zeigt an, ob es sich um einen lesenden (R) und/oder schreibenden Zugriff auf den Operanden handelt.         |

| Spalte           | Bedeutung   |
|------------------|---|
| Adresse          | Adresse des Operanden   |
| Beobachtungswert | Diese Spalte wird nur angezeigt, wenn der Programmiereditor geöffnet ist. |
| Typ              | Informationen über den Typ und die Erstsprache des Objekts                |
| Pfad             | Pfad des Objekts im Projektnavigationsbaum                                |

Abhängig von den installierten Produkten werden zusätzliche oder andere Spalten für die Querverweise angezeigt.

## 9.4 Anwenderprogramm testen

### 9.4.1 Grundlagen zum Testen des Anwenderprogramms

#### Funktion

Sie haben die Möglichkeit, den Ablauf Ihres Anwenderprogramms auf dem Gerät zu testen. Sie können Signalzustände und Werte von Variablen beobachten und Variablen mit Werten vorbelegen, um damit bestimmte Situationen für den Programmablauf zu simulieren.

#### Voraussetzung

Ein ablauffähiges Programm muss auf das Gerät geladen sein.

#### Testmöglichkeiten

Folgende Testmöglichkeiten stehen Ihnen zur Verfügung:

- Testen mit dem Programmstatus  
Der Programmstatus ermöglicht es Ihnen, den Programmablauf zu beobachten. Dabei können Sie sich die Werte der Operanden und die Verknüpfungsergebnisse (VKE) anzeigen lassen und dadurch logische Fehler in Ihrem Programm finden und beheben.
- Testen im Einzelschrittmodus (nur S7-300/400)  
Bausteine, die Sie in AWL oder SCL erstellt haben, können Sie im Einzelschrittmodus testen. Dazu setzen Sie Haltepunkte in den Programmcode, an denen die Programmausführung gestoppt wird. Anschließend können Sie das Programm schrittweise weiter ausführen. Innerhalb einer CPU können Sie entweder mit Programmstatus oder im Einzelschrittmodus testen. Sie können jedoch nicht gleichzeitig beide Testmöglichkeiten innerhalb einer CPU verwenden.

- **Testen mit der Beobachtungstabelle**  
Die Beobachtungstabelle ermöglicht es Ihnen, die aktuellen Werte einzelner Variablen des Anwenderprogramms bzw. einer CPU zu beobachten und zu steuern. Sie können einzelne Variablen zum Testen mit Werten vorbelegen und somit verschiedene Ablaufsituationen durchspielen. Außerdem haben Sie die Möglichkeit, den Peripherieausgängen einer CPU im Betriebszustand STOP feste Werte zuzuweisen, um damit z. B. die Verdrahtung zu überprüfen.
- **Testen mit der Forcetabelle**  
Die Forcetabelle ermöglicht es Ihnen, die aktuellen Werte einzelner Variablen des Anwenderprogramms bzw. einer CPU zu beobachten und zu forcen. Beim Forcen überschreiben Sie einzelne Variablen mit vorgegebenen Werten. Damit können Sie Ihr Anwenderprogramm testen und verschiedene Ablaufsituationen durchspielen. Beachten Sie vor dem Forcen auf jeden Fall die notwendigen Sicherheitsvorkehrungen für das Forcen (Seite 1578)!

### Siehe auch

Einführung zum Testen mit Programmstatus (Seite 1525)

Einführung zum Testen mit der Beobachtungstabelle (Seite 1533)

Einführung zum Testen mit der Forcetabelle (Seite 1560)

## 9.4.2 Testen mit Programmstatus

### 9.4.2.1 Einführung zum Testen mit Programmstatus

#### Funktion Programmstatus

Durch die Anzeige des Programmstatus können Sie den Programmablauf beobachten. Dadurch erhalten Sie eine Übersicht über die Werte der einzelnen Operanden und der Verknüpfungsergebnisse und können feststellen, ob die Komponenten des Automatisierungssystems korrekt gesteuert werden.

Die Anzeige des Programmablaufs im Programmstatus kann sich geringfügig unterscheiden, abhängig von der eingesetzten CPU-Familie.

#### Testen mit Programmstatus für S7-300/400

Beim Testen mit Programmstatus kann sich die CPU-Zykluszeit verlängern, da aufgrund der CPU-Kapazität die Erfassung aller Testdaten von der Dauer der programmierten Anweisungen abweichen kann und somit nicht in Echtzeit abläuft.

Beim Ausführen folgender Testfunktionen wird für jede Online-Sitzung einmalig eine Meldung angezeigt, die Sie darauf hinweist, dass die Gefahr einer Zykluszeitüberschreitung besteht:

- Beim Testen mit Aufrufbedingungen
- Beim Testen mit Haltepunkten

Sie können diese Testfunktionen erst nach Bestätigen der Meldung ausführen.

---

**Hinweis**

Bei älteren CPUs aus den CPU-Familien S7-300/400 müssen Sie das Betriebsverhalten über die Hardware-Konfiguration ändern und anschließend die Hardware-Konfiguration in das Gerät laden. Sie haben die Möglichkeit, für das Betriebsverhalten "Prozessbetrieb" oder "Testbetrieb" einzustellen.

---

### Einschränkungen bei der Funktion "Programmstatus"

Das Beobachten von Schleifen kann die Zykluszeit deutlich erhöhen, jeweils abhängig von der Anzahl der zu beobachtenden Variablen und abhängig von der tatsächlichen Schleifendurchlaufzahl.

Um eine möglichst geringe Beeinflussung der Zykluszeit zu gewährleisten, wird die Funktion "Programmstatus" wie folgt eingeschränkt:

- Die Statusanzeige einer programmierten Schleife wird an der Rücksprungstelle abgebrochen.



**WARNUNG**

**Testen mit Programmstatus**

Ein Test mit der Funktion "Programmstatus" kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen.

Stellen Sie sicher, dass keine gefährlichen Zustände eintreten können, bevor Sie einen Test mit der Funktion "Programmstatus" durchführen!

#### 9.4.2.2 Test mit Programmstatus ein-/ausschalten

Sie können den Programmstatus für alle Programmiersprachen einschalten. Für die grafischen Programmiersprachen KOP und FUP können Sie den Programmstatus zusätzlich ab einer bestimmten Position oder für eine bestimmte Selektion einschalten.

#### Voraussetzung

- Der Baustein ist identisch im Gerät vorhanden.
- Der Baustein ist geöffnet.

### Programmstatus ein- oder ausschalten

Um den Programmstatus für einen Baustein ein- oder auszuschalten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus". Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Mithilfe dieses Dialogs können Sie eine Online-Verbindung herstellen. Siehe auch: Online verbinden und Online-Verbindung trennen

### Programmstatus ab einer bestimmten Stelle in einem Netzwerk einschalten

Um den Programmstatus für KOP und FUP ab einer bestimmten Stelle einzuschalten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
2. Klicken Sie mit der rechten Maustaste auf die Variable, ab der Sie den Programmstatus einschalten möchten.
3. Wählen Sie im Kontextmenü "Steuern > Beobachten ab hier".

### Programmstatus für selektierte Variablen einschalten

Um den Programmstatus für KOP und FUP für selektierte Variablen einzuschalten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus".
2. Selektieren Sie die Variablen, für die Sie den Programmstatus einschalten möchten.
3. Wählen Sie im Kontextmenü "Steuern > Selektion beobachten".

---

#### Hinweis

Die Ressourcen für das Testen mit dem Programmstatus sind begrenzt. Falls die Ressourcen für den aktuellen Test nicht ausreichen, werden frühere Tests beendet.

---

### Ergebnis

Wenn Sie die Anzeige des Programmstatus einschalten, wird eine Online-Verbindung hergestellt und der Programmstatus angezeigt. Wenn Sie die Anzeige des Programmstatus ausschalten, können Sie gleichzeitig die Online-Verbindung trennen.

Wenn sich die CPU im Betriebszustand "HALT" oder "STOP" befindet, wird in der Task Card "Testen" in der Palette "Aufrufhierarchie" die Aufrufhierarchie des Bausteins angezeigt. Für S7-1200-CPU's wird die Aufrufhierarchie auch während des Tests mit Programmstatus angezeigt. Über diese Aufrufhierarchie können Sie zu einem der aufrufenden Bausteine wechseln.

### 9.4.2.3 Bausteine während des Programmtests bearbeiten

Wenn Sie Bausteine bearbeiten, während der Test mit Programmstatus noch läuft, wird das Online-Beobachten unterbrochen und Sie können den Baustein offline bearbeiten. Falls der Baustein offline im Projekt nicht vorhanden ist, müssen Sie ihn zunächst vom Gerät ins Projekt laden. Nach der Bearbeitung ist ein erneutes Übersetzen und Laden des Bausteins nötig.

#### Vorgehen

Um Bausteine zu bearbeiten, während der Test mit Programmstatus noch läuft, gehen Sie folgendermaßen vor:

1. Bearbeiten Sie den Baustein gemäß den Erfordernissen.  
Der Test mit Programmstatus wird unterbrochen und der Baustein offline geschaltet, falls er offline bereits vorhanden ist.
2. Falls der Baustein offline nicht vorhanden ist, laden Sie ihn vom Gerät ins Projekt.
3. Übersetzen Sie den Baustein.  
Siehe auch: Auto-Hotspot
4. Laden Sie den Baustein in das Gerät.  
Siehe auch: Auto-Hotspot

#### Ergebnis

Der Baustein enthält nun online und offline Ihre Änderungen. Die Online-Verbindung kann wieder hergestellt und das Testen mit Programmstatus fortgesetzt werden.

### 9.4.2.4 Variablen im Programmstatus steuern

Sie haben die Möglichkeit, Variablen während des Testens mit dem Programmstatus sofort und einmalig auf folgende Werte zu steuern:

- Steuern auf 1  
Steuert Variablen vom Datentyp "Bool" auf den Wert "True".
- Steuern auf 0  
Steuert Variablen vom Datentyp "Bool" auf den Wert "False"
- Operand steuern  
Für Variablen, die nicht vom Datentyp "Bool" sind, können Sie einen Steuerwert eingeben.

Beachten Sie jedoch, dass Sie keine Peripherieeingänge, z. B. über TagName:P, steuern können.

## Vorgehen

Um Variablen während des Testens mit dem Programmstatus zu steuern, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Variable, die Sie steuern möchten.
2. Wählen Sie im Kontextmenü einen der folgenden Befehle:
  - "Steuern > Steuern auf 1"
  - "Steuern > Steuern auf 0"
  - "Steuern > Operand steuern"
3. Wenn Sie "Operand steuern" wählen, wird der Dialog "Operand steuern" geöffnet. Geben Sie in das Feld "Steuerwert" den gewünschten Wert ein und bestätigen Sie Ihre Eingabe mit "OK".

### 9.4.2.5 Anzeigeformate im Programmstatus umstellen

## Einführung

Die Anzeigeformate für Variablen werden üblicherweise in "Integer" dargestellt. Im Programmstatus haben Sie die Möglichkeit, das aktuelle Anzeigeformat über das Kontextmenü umzustellen. Die für eine Variable möglichen Anzeigeformate werden in einer Liste angeboten. Dies ist hilfreich, wenn Sie z. B. eine hexadezimale Darstellung benötigen, um nach einem hexadezimalen Fehlercode zu suchen.

## Vorgehen

Um das Anzeigeformat umzustellen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den gewünschten Baustein im Programmiereditor.
2. Schalten Sie den Programmstatus ein, indem Sie in der Funktionsleiste auf die Schaltfläche "Beobachten ein/aus" klicken.  
Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Online verbinden" geöffnet. Mithilfe dieses Dialogs können Sie eine Online-Verbindung herstellen.
3. Selektieren Sie die Variablen, für die Sie den Programmstatus einschalten möchten.
4. Wählen Sie im Kontextmenü "Steuern > Selektion beobachten", um das Beobachten dieser Variablen zu starten.
5. Selektieren Sie die gewünschte Variable am entsprechenden Bausteinausgang und wählen Sie das gewünschte Anzeigeformat im Kontextmenü aus, z. B. "Steuern > Anzeigeformat > Hexadezimal".

## Ergebnis

Das Anzeigeformat für die selektierte Variable wird in Hexadezimal angezeigt.

### Hinweis

#### Umstellen des Anzeigeformats im Programmstatus

Beachten Sie bitte, dass für nicht beschaltete Ausgänge eine Umstellung des Anzeigeformats nicht möglich ist, da hier kein Beobachtungswert ausgegeben wird.

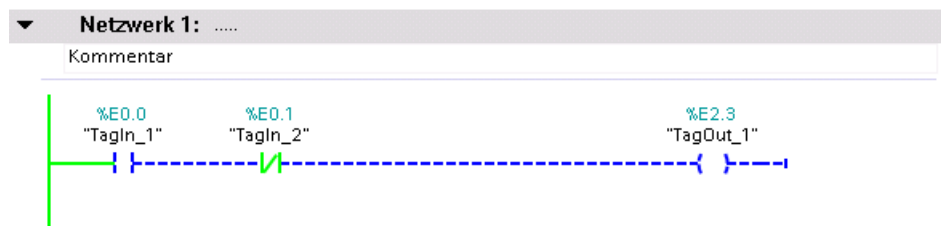
## 9.4.2.6 Beispiele für die Programmstatusanzeige

### Programmstatusanzeige für KOP-Programme

#### Anzeigen im Programmstatus

Die Anzeige des Programmstatus wird zyklisch aktualisiert.

Das folgende Bild zeigt ein Beispiel für die Programmstatusanzeige unter KOP:



#### Darstellung des Programmstatus

Den Status der einzelnen Anweisungen und Linien eines Netzwerks können Sie schnell an Farbe und Art der Linien und Symbole erkennen. Die folgende Tabelle zeigt die Zuordnung von Darstellung und Status:

| Darstellung   | Status   |
|---|--|
| Grün durchgezogen                                       | Erfüllt  |
| Blau gestrichelt  | Nicht erfüllt  |
| Grau durchgezogen                                       | Unbekannt oder nicht durchlaufen   |
| Schwarz   | Nicht beschaltet   |
| Parameter in einem Rahmen mit einer Sättigung von 100 % | Wert ist aktuell   |
| Parameter in einem Rahmen mit einer Sättigung von 50 %  | Wert stammt aus einem vorhergehenden Zyklus. Die Programmstelle wurde im aktuellen Zyklus nicht durchlaufen. |

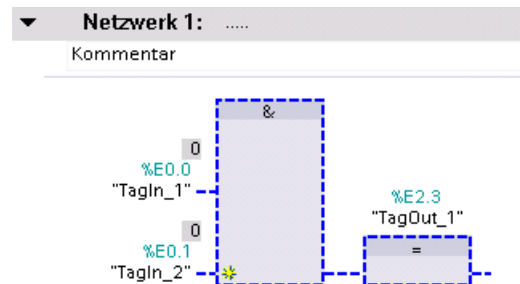


## Programmstatusanzeige für FUP-Programme

### Anzeigen im Programmstatus

Die Anzeige des Programmstatus wird zyklisch aktualisiert.

Das folgende Bild zeigt ein Beispiel für die Programmstatusanzeige unter FUP:



### Darstellung des Programmstatus

Den Status der einzelnen Anweisungen und Linien eines Netzwerks können Sie schnell an Farbe und Art der Linien und Symbole erkennen. Die folgende Tabelle zeigt die Zuordnung von Darstellung und Status:

| Darstellung   | Status   |
|---|--|
| Grün durchgezogen                                       | Erfüllt  |
| Blau gestrichelt  | Nicht erfüllt  |
| Grau durchgezogen                                       | Unbekannt oder nicht durchlaufen   |
| Schwarz   | Nicht beschaltet   |
| Parameter in einem Rahmen mit einer Sättigung von 100 % | Wert ist aktuell   |
| Parameter in einem Rahmen mit einer Sättigung von 50 %  | Wert stammt aus einem vorhergehenden Zyklus. Die Programmstelle wurde im aktuellen Zyklus nicht durchlaufen. |

Die Werte der Operanden werden über dem jeweiligen Operandennamen in einem grauen Kästchen angezeigt.

#### Hinweis

##### Programmstatusanzeige bei nicht beschalteten Ausgängen

Beachten Sie bitte, dass für nicht beschaltete Ausgänge auch kein Beobachtungswert ausgegeben werden kann.

## Programmstatusanzeige für SCL-Programme

### Anzeigen im Programmstatus

Die Anzeige des Programmstatus wird zyklisch aktualisiert und in einer Tabelle dargestellt. Die Tabelle wird direkt neben dem SCL-Programm eingeblendet und Sie können für jede Programmzeile den Programmstatus ablesen. Die Tabelle enthält folgende Informationen:

- Variablennamen
- Wert

Sie können die Tabelle jederzeit nach rechts oder links verschieben.

Das folgende Bild zeigt ein Beispiel für die Programmstatusanzeige unter SCL:

|   |   |           |       |
|---|---|-----------|-------|
| 1 | <input type="checkbox"/> IF "TagIn_1"         | "TagIn_1" | FALSE |
| 2 | THEN "TagIn_2" :=1;                           | "TagIn_2" | TRUE  |
| 3 | END_IF;                                       |           |       |
| 4 | <input type="checkbox"/> IF "TagIn_2" = false | "TagIn_2" | FALSE |
| 5 | THEN "TagIn_3" :=1;                           | "TagIn_3" | TRUE  |
| 6 | END_IF;                                       |           |       |
| 7 |   |           |       |

In der ersten Spalte finden Sie den Variablennamen, für den der aktuelle Wert angezeigt wird. Falls die Zeile die Anweisung "IF", "WHILE" oder "REPEAT" enthält, wird in der Zeile das Ergebnis der Anweisung mit "True" oder "False" angezeigt. Wenn die Zeile mehrere Variablen enthält, wird der Wert der ersten Variablen angezeigt. In beiden Fällen werden alle Variablen dieser Zeilen in einer eigenen Liste mit ihren Werten angezeigt, sobald Sie die entsprechende Zeile selektieren. Wenn Sie im Programmcode den Cursor in eine Variable setzen, wird diese in der Liste fett markiert. Sie können die weiteren Variablen einer Zeile auch explizit anzeigen, wenn Sie auf den Pfeil-nach-rechts klicken, der sich vor den Variablennamen der Zeilen befindet, in der es mehrere Variable gibt.

Wird der Code der Zeile nicht ausgeführt, wird der Variablennamen in der Wertetabelle in grauer Schriftfarbe angezeigt.

In der letzten Spalte werden die aktuellen Werte der Variablen angezeigt. Wenn für eine Variable keine Werte angezeigt werden können, erhält die entsprechende Zelle einen gelben Hintergrund und es werden drei Fragezeichen angezeigt. Aktivieren Sie in diesem Fall das Optionskästchen "Erweiterte Statusinformationen erstellen" in den Eigenschaften des Bausteins und laden Sie den Baustein erneut in das Gerät. Anschließend werden alle Werte angezeigt.

## 9.4.3 Testen mit der Beobachtungstabelle

### 9.4.3.1 Einführung zum Testen mit der Beobachtungstabelle

#### Übersicht

Innerhalb der Beobachtungstabelle stehen Ihnen folgende Funktionen zur Verfügung:

- **Beobachten von Variablen**  
Damit lassen sich die aktuellen Werte einzelner Variablen eines Anwenderprogramms bzw. einer CPU am PG/PC anzeigen.
- **Steuern von Variablen**  
Mit dieser Funktion können Sie einzelnen Variablen eines Anwenderprogramms bzw. einer CPU feste Werte zuweisen. Das Steuern ist auch beim Testen mit Programmstatus möglich.
- **"PA freischalten" und "Steuern sofort"**  
Diese beiden Funktionen geben Ihnen die Möglichkeit, einzelnen Peripherieausgängen einer CPU im Betriebszustand STOP feste Werte zuzuweisen. Sie können damit auch Ihre Verdrahtung überprüfen.

#### Beobachten und Steuern von Variablen

Folgende Variablen können beobachtet bzw. gesteuert werden:

- Eingänge, Ausgänge und Merker
- Inhalte von Datenbausteinen
- Peripherie

#### Einsatzmöglichkeit

Die Beobachtungstabelle bietet den Vorteil, unterschiedliche Testumgebungen abspeichern zu können. So können Tests während einer Inbetriebsetzung oder zu Service- und Wartungszwecken reproduziert werden.

#### Siehe auch

Beobachtungstabellen erstellen und bearbeiten (Seite 1537)

Aufbau der Beobachtungstabelle (Seite 1534)

Basismodus und Erweiterter Modus in der Beobachtungstabelle (Seite 1535)

Symbole in der Beobachtungstabelle (Seite 1536)

### 9.4.3.2 Aufbau der Beobachtungstabelle

#### Einführung

Eine Beobachtungstabelle enthält die von Ihnen definierten und ausgewählten CPU-weit gültigen Variablen. Für jede im Projekt angelegte CPU wird automatisch ein Ordner "Beobachtungs- und Forcetabellen" angelegt. Über den Befehl "Neue Beobachtungstabelle hinzufügen" legen Sie eine neue Beobachtungstabelle in diesem Ordner an.

#### Aufbau der Beobachtungstabelle

Die in der Beobachtungstabelle angezeigten Spalten sind abhängig davon, ob Sie im Basismodus oder im Erweiterten Modus arbeiten.


Im Erweiterten Modus werden folgende Spalten zusätzlich angezeigt:

- Beobachten mit Trigger
- Steuern mit Trigger

Außerdem können die Namen der Spalten je nach Aktion dynamisch wechseln.

#### Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten im Basismodus und im Erweiterten Modus:

| Mode   | Spalte  | Erläuterung   |
|--|---|---|
| Basismodus   | <b>i</b>  | Kennzeichnungsspalte  |
|  | Name  | Name der eingefügten Variablen  |
|  | Adresse   | Adresse der eingefügten Variablen   |
|  | Anzeigeformat   | Ausgewähltes Anzeigeformat  |
|  | Beobachtungswert  | Werte der Variablen, abhängig vom ausgewählten Anzeigeformat.                             |
|  | Steuerwert  | Wert, mit dem die Variable gesteuert wird.  |
|  |  | Auswahl der zu steuernden Variablen durch Aktivieren des entsprechenden Optionskästchens. |
|  | Kommentar   | Kommentar zur Dokumentation der Variablen   |
| Im Erweiterten Modus werden zusätzlich folgende Spalten angezeigt: | Beobachten mit Trigger  | Anzeige des ausgewählten Beobachtungsmodus  |
|  | Steuern mit Trigger   | Anzeige des ausgewählten Steuermodus  |

#### Siehe auch

Symbole in der Beobachtungstabelle (Seite 1536)

### 9.4.3.3 Basismodus und Erweiterter Modus in der Beobachtungstabelle

#### Unterschied zwischen Basismodus und Erweitertem Modus in der Beobachtungstabelle

Abhängig vom eingestellten Modus, werden in der Beobachtungstabelle unterschiedliche Spalten und Spaltenüberschriften angezeigt, mit denen unterschiedliche Aktionen durchgeführt werden können.

Eine detaillierte Auflistung der jeweiligen Spalten finden Sie unter Aufbau der Beobachtungstabelle (Seite 1534).

#### Umschalten zwischen Basismodus und Erweitertem Modus

Um zwischen Basismodus und Erweitertem Modus umzuschalten, haben Sie folgende Möglichkeiten:

- Klicken Sie auf das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden". Ein erneutes Klicken auf dieses Symbol schaltet wieder in den Basismodus um.  
Oder:
- Aktivieren Sie im Menü "Online" das Kontrollkästchen vor dem Befehl "Erweiterter Modus". Ein Deaktivieren des Kontrollkästchens schaltet wieder in den Basismodus um.

#### Funktionalität im Erweiterten Modus

Folgende Funktionalität ist nur im Erweiterten Modus möglich:

- Beobachten mit Trigger
- Steuern mit Trigger
- Peripherieausgänge freischalten
- Peripherieeingänge beobachten
- Peripherieausgänge steuern

#### **ACHTUNG**

#### **Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen und Steuern von Peripherieausgängen**

Beachten Sie, dass das Beobachten von Peripherieeingängen und das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!

Die CPU geht dann in den Betriebszustand "STOP".

#### Siehe auch

Festlegen des Beobachtungs- und Steuermodus (Seite 1548)

### 9.4.3.4 Symbole in der Beobachtungstabelle

#### Bedeutung der Symbole

Die folgende Tabelle zeigt die Bedeutung der Symbole in der Beobachtungstabelle:

| Symbol  | Bedeutung  |
|---|--|
|    | Kennzeichnet eine Tabelle innerhalb der Projektnavigation als Beobachtungstabelle.   |
|    | Zeigt Informationen in der Kennzeichnungsspalte an.  |
|    | Steuert die Adressen aller ausgewählten Variablen sofort und einmalig. Dieser Befehl wird einmalig und schnellstmöglich ausgeführt, ohne Bezug auf einem bestimmten Triggerpunkt im Anwenderprogramm.  |
|    | Steuert die Adressen aller ausgewählten Variablen mit Bezug auf einen bestimmten Triggerpunkt im Anwenderprogramm.   |
|    | Schaltet die Befehlsausgabesperre der Peripherieausgänge (BASP) ab. Dies ermöglicht das Steuern der Peripherieausgänge im STOP-Zustand der CPU.  |
|    | Blendet alle Spalten des Erweiterten Modus ein. Wenn Sie erneut auf dieses Symbol klicken, werden die Spalten des Erweiterten Modus ausgeblendet.  |
|  | Blendet alle Steuerspalten ein. Wenn Sie erneut auf dieses Symbol klicken, werden die Steuerspalten ausgeblendet.  |
|  | Startet das Beobachten der sichtbaren Variablen in der aktiven Beobachtungstabelle. Im Basismodus ist für den Beobachtungsmodus "permanent" voreingestellt. Im Erweiterten Modus können Sie für das Beobachten der Variablen bestimmte Triggerpunkte einstellen.   |
|  | Startet das Beobachten der sichtbaren Variablen in der aktiven Beobachtungstabelle. Dieser Befehl wird sofort ausgeführt und die Variablen werden einmalig beobachtet.   |
|  | Zeigt das Optionskästchen für die Auswahl der zu steuernden Variablen an.  |
|  | Zeigt an, dass die selektierte Variable auf den Wert "1" gesteuert wurde.  |
|  | Zeigt an, dass die selektierte Variable auf den Wert "0" gesteuert wurde.  |
|  | Zeigt an, dass die Adresse mehrfach verwendet wird.  |
|  | Zeigt an, dass ein Ersatzwert verwendet wird. Ersatzwerte sind Werte, die bei fehlerhaften Signalausgabebaugruppen an den Prozess ausgegeben werden, bzw. bei fehlerhaften Signaleingabebaugruppen im Anwenderprogramm anstelle eines Prozesswertes verwendet werden. Die Ersatzwerte sind vom Anwender vorgebar (z. B. alten Wert beibehalten). |
|  | Zeigt an, dass die Adresse gesperrt ist, da bereits gesteuert wird.  |
|  | Zeigt an, dass die Adresse nicht gesteuert werden kann.  |
|  | Zeigt an, dass die Adresse nicht beobachtet werden kann.   |
|  | Zeigt an, dass eine Adresse geforct wird.  |

| Symbol | Bedeutung  |
|--------|--|
|        | Zeigt an, dass eine Adresse teilweise geforct wird.  |
|        | Zeigt an, dass eine zugehörige Peripherie-Adresse vollständig oder teilweise geforct wird.   |
|        | Zeigt an, dass eine Adresse evtl. nicht vollständig geforct werden kann.<br>Beispiel: Es ist zwar möglich, die Adresse AW0:P zu forcen, es ist jedoch nicht möglich, die Adresse AD0:P zu forcen, da dieser Adressbereich auf der CPU evtl. nicht vorhanden ist. |
|        | Zeigt an, dass ein Syntaxfehler aufgetreten ist.   |
|        | Zeigt an, dass die Adresse zwar ausgewählt ist, im Moment aber z. B. noch nicht gesteuert wird.  |

## Siehe auch

Aufbau der Beobachtungstabelle (Seite 1534)

### 9.4.3.5 Beobachtungstabellen erstellen und bearbeiten

#### Beobachtungstabelle erstellen

#### Einführung

Mit Hilfe der Beobachtungstabelle können Sie Variablen im Anwenderprogramm beobachten und steuern. Eine einmal erstellte Beobachtungstabelle können Sie speichern, duplizieren, ausdrucken und immer wieder zum Beobachten und Steuern von Variablen verwenden.

#### Voraussetzung

Ein Projekt ist geöffnet.

#### Vorgehen

Um eine Beobachtungstabelle zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Statusleiste auf "Projektansicht".  
Die Projektansicht wird angezeigt.
2. Doppelklicken Sie in der Projektnavigation auf die entsprechende CPU, für die Sie eine Beobachtungstabelle erstellen wollen.
3. Doppelklicken Sie zunächst auf den Ordner "Beobachtungs- und Forcetabellen" und dann auf den Befehl "Neue Beobachtungstabelle hinzufügen".  
Eine neue Beobachtungstabelle wird hinzugefügt.
4. Tragen Sie in der Spalte "Name" oder in der Spalte "Adresse" den Namen oder die absolute Adresse für die Variablen ein, die Sie beobachten oder steuern wollen.

5. Falls Sie das voreingestellte Anzeigeformat ändern wollen, wählen Sie in der Spalte "Anzeigeformat" das gewünschte Anzeigeformat aus der Klappliste aus.
6. Entscheiden Sie nun, ob Sie die eingetragenen Variablen beobachten oder steuern wollen und tragen Sie ggf. die gewünschten Werte für das Steuern der Variablen ein.

## Beobachtungstabelle öffnen

### Voraussetzung

Eine Beobachtungstabelle ist angelegt.

### Vorgehen

Um eine Beobachtungstabelle zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Beobachtungs- und Forcetabellen" unterhalb der gewünschten CPU.
2. Doppelklicken Sie auf die gewünschte Beobachtungstabelle im Ordner.

### Ergebnis

Die ausgewählte Beobachtungstabelle wird geöffnet.

## Beobachtungstabelle kopieren und einfügen

### Voraussetzung

Eine Beobachtungstabelle ist angelegt.

### Vorgehen

Um eine Beobachtungstabelle zu kopieren, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf die Beobachtungstabelle, die Sie kopieren möchten.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Öffnen Sie in der Projektnavigation die Ordnerstruktur für die CPU, in die Sie die kopierte Beobachtungstabelle einfügen möchten.
4. Klicken Sie mit der rechten Maustaste auf den Ordner "Beobachtungs- und Forcetabellen".
5. Wählen Sie im Kontextmenü den Befehl "Einfügen".
6. Alternativ können Sie auch den gesamten Inhalt der Beobachtungstabelle markieren und per Drag & Drop in eine andere Beobachtungstabelle ziehen.



## Ergebnis

Eine Kopie der selektierten Beobachtungstabelle wird in den Ordner "Beobachtungs- und Forcetabellen" der entsprechenden CPU eingefügt.

## Beobachtungstabelle speichern

### Voraussetzung

Eine Beobachtungstabelle ist angelegt.

### Vorgehen

Um eine Beobachtungstabelle zu speichern, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation die Beobachtungstabelle, die Sie speichern wollen.
2. Falls Sie den voreingestellten Namen der Tabelle ändern wollen, wählen Sie im Kontextmenü den Befehl "Umbenennen" und geben Sie einen neuen Namen für die Tabelle ein.
3. Wählen Sie im Menü "Projekt" den Befehl "Speichern". Beachten Sie, dass mit diesem Speichervorgang das gesamte Projekt gespeichert wird.

## Ergebnis

Die Inhalte der Beobachtungstabelle und das Projekt werden gespeichert.

---

### Hinweis

Gespeicherte Beobachtungstabellen können Sie beim erneuten Testen Ihres Programms wiederholt zum Beobachten und Steuern von Variablen verwenden.

---

## 9.4.3.6 Variablen in die Beobachtungstabelle eingeben

### Grundlagen für die Eingabe von Variablen in die Beobachtungstabelle

#### Empfohlene Vorgehensweise

- Bestimmen Sie die Variablen, deren Werte Sie beobachten oder steuern wollen und tragen Sie diese in die Beobachtungstabelle ein.
- Beachten Sie bei der Eingabe von Variablen in die Beobachtungstabelle, dass es sinnvoll ist, diese Variablen zuvor in der PLC-Variablentabelle zu definieren.
- Bei der Eingabe von Variablen gehen Sie von "außen" nach "innen" vor. Dies bedeutet, dass Sie zuerst die Variablen für die Eingänge in die Beobachtungstabelle eintragen. Danach geben Sie die Variablen ein, die von den Eingängen beeinflusst werden, bzw. die die Ausgänge beeinflussen. Anschließend geben Sie die Variablen für die Ausgänge ein.

### Beispiel für das Ausfüllen einer Beobachtungstabelle

- Tragen Sie in der Spalte "Adresse" die absolute Adresse ein, die beobachtet oder gesteuert werden soll.
- Tragen Sie in der Spalte "Name" den symbolischen Namen für die Variable ein.
- Wählen Sie in der Spalte "Anzeigeformat" das gewünschte Anzeigeformat aus der Klappliste aus, falls Sie die Voreinstellung nicht verwenden wollen.
- Entscheiden Sie nun, ob Sie die eingetragenen Variablen beobachten oder steuern wollen. Tragen Sie für das Steuern die gewünschten Werte sowie einen Kommentar in die entsprechenden Spalten der Beobachtungstabelle ein.

### Syntaxprüfung

Wenn Sie Variablen in die Beobachtungstabelle eintragen, wird vor dem Verlassen der Zelle eine Syntaxprüfung durchgeführt. Fehlerhafte Einträge werden rot gekennzeichnet.

---

#### Hinweis

Wenn Sie den Cursor in eine rot markierte Zelle stellen, so erhalten Sie über eine kurze Information weitere Hinweise zum Fehler.

---

### Siehe auch

Erlaubte Operanden für die Beobachtungstabelle (Seite 1540)

Erlaubte Steuerwerte für die Beobachtungstabelle (Seite 1541)

### Erlaubte Operanden für die Beobachtungstabelle

### Zulässige Operanden für die Beobachtungstabelle

Die folgende Tabelle zeigt die Operanden, die für die Beobachtungstabelle zugelassen sind:

| Erlaubter Operand      | Beispiel für Datentyp | Beispiel (Mnemonic deutsch)         |
|------------------------|-----------------------|-------------------------------------|
| Eingang/Ausgang/Merker | BOOL                  | E1.0; A1.7; M10.1<br>E0.0:P; A0.0:P |
| Eingang/Ausgang/Merker | BYTE                  | EB1; AB10; MB100<br>EB1:P; AB1:P    |
| Eingang/Ausgang/Merker | WORD                  | EW1; AW10; MW100<br>EW2:P; AW3:P    |
| Eingang/Ausgang/Merker | DWORD                 | ED4; AD10; MD100<br>ED2:P; AD1:P    |
| Zeiten                 | TIMER                 | T1                                  |
| Zähler                 | COUNTER               | Z1                                  |
| Datenbaustein          | BOOL                  | DB1.DBX1.0                          |

| Erlaubter Operand | Beispiel für Datentyp | Beispiel (Mnemonic deutsch) |
|-------------------|-----------------------|-----------------------------|
| Datenbaustein     | BYTE                  | DB1.DBB1                    |
| Datenbaustein     | WORD                  | DB1.DBW1                    |
| Datenbaustein     | DWORD                 | DB1.DBD1                    |

**Hinweis**

**Beachten Sie bitte die folgenden Hinweise für das Arbeiten mit der Beobachtungstabelle**

- Die Eingabe von "DB0..." ist aufgrund interner Benutzung nicht erlaubt!
- Peripherieausgänge können zwar gesteuert, aber nicht beobachtet werden!
- Peripherieeingänge können zwar beobachtet, aber nicht gesteuert werden!

**ACHTUNG**

**Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen und Steuern von Peripherieausgängen**

Beachten Sie, dass das Beobachten von Peripherieeingängen und das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!

Die CPU geht dann in den Betriebszustand "STOP".

**Siehe auch**

Grundlagen für die Eingabe von Variablen in die Beobachtungstabelle (Seite 1539)

**Erlaubte Steuerwerte für die Beobachtungstabelle**

**Eingabe von Steuerwerten in die Beobachtungstabelle**

Die folgende Tabelle zeigt die Operanden, die für die Eingabe von Steuerwerten in die Beobachtungstabelle zugelassen sind:

Tabelle 9-14 Bitoperanden

| Mögliche Bitoperanden | Beispiel für erlaubte Steuerwerte |
|-----------------------|-----------------------------------|
| E1.0                  | true                              |
| M1.7                  | false                             |
| A1.0                  | 0                                 |
| A1.1:P                | 1                                 |

| Mögliche Bitoperanden | Beispiel für erlaubte Steuerwerte |
|-----------------------|-----------------------------------|
| DB1.DBX1.1            | 2#0                               |
| M1.6                  | 2#1                               |

Tabelle 9-15 Byteoperanden

| Mögliche Byteoperanden | Beispiel für erlaubte Steuerwerte |
|------------------------|-----------------------------------|
| EB1                    | 2#00110011                        |
| MB12                   | B#16#1F                           |
| AB10                   | 1F                                |
| AB11:P                 | 'a'                               |
| DB1.DBB1               | 10                                |

Tabelle 9-16 Wortoperanden

| Mögliche Wortoperanden | Beispiel für erlaubte Steuerwerte |
|------------------------|-----------------------------------|
| EW1                    | 2#0011001100110011                |
| MW12                   | W#16#ABCD                         |
| MW14                   | ABCD                              |
| AW10                   | B#(12, 34)                        |
| AW12:P                 | 12345                             |
| DB1.DBW1               | 'ab'                              |
| MW16                   | S5T#9s_340ms                      |
| MW18                   | C#123                             |
| MW9                    | D#2006-12-31                      |

Tabelle 9-17 Doppelwortoperanden

| Mögliche Doppelwortoperanden | Beispiel für erlaubte Steuerwerte |
|------------------------------|-----------------------------------|
| ED1                          | 2#0011001100110011001100110011    |
| AD10                         | Dw#16#abcdef10                    |
| AD12:P                       | ABCDEF10                          |
| DB1.DBD2                     | b#(12,34,56,78)                   |
| MD8                          | L#-12                             |
| MD12                         | L#12                              |
| MD16                         | 123456789                         |
| MD20                         | 123456789                         |
| MD24                         | T#12s345ms                        |
| MD28                         | Tod#1:2:34.567                    |
| MD32                         | P#e0.0                            |

Tabelle 9-18 Zeiten

| Mögliche Operanden vom Typ "Zeit" | Erlaubte Steuer- / Forcewerte | Erläuterung                    |
|-----------------------------------|-------------------------------|--------------------------------|
| T1                                | 0 ms                          | Zeitwert in Millisekunden (ms) |
| T12                               | 20 ms                         | Zeitwert in Millisekunden (ms) |
| T14                               | 12345 ms                      | Zeitwert in Millisekunden (ms) |
| T16                               | S5t#12s340ms                  | Zeitwert 12s 340 ms            |

Tabelle 9-19 Zähler

| Mögliche Operanden vom Typ "Zähler" | Erlaubte Steuer- / Forcewerte |
|-------------------------------------|-------------------------------|
| Z1                                  | 0                             |
| Z14                                 | 20                            |
| Z16                                 | C#123                         |

## Anmerkungen zu Zeiten und Zählern

- Zeiten

### Hinweis

Das Steuern einer Zeit beeinflusst nur den Wert, nicht den Zustand. Die Zeit T1 kann auf den Wert "0" gesteuert werden, das Verknüpfungsergebnis bei U T1 wird nicht verändert.

Die Zeitfolgen "s5t" und "s5time" können sowohl klein als auch groß geschrieben werden.

- Zähler

### Hinweis

Das Steuern eines Zählers beeinflusst nur den Wert, nicht den Zustand. Der Zähler Z1 kann auf den Wert "0" gesteuert werden, das Verknüpfungsergebnis bei U Z1 wird nicht verändert.

## Übersicht über die Anzeigeformate

### Anzeigeformate in der Beobachtungstabelle

Mit der Auswahl eines Anzeigeformats legen Sie fest, wie der Wert einer Variablen dargestellt wird.

Beim Eintragen der Adresse wird automatisch ein Anzeigeformat vorgelegt. Um dieses zu ändern, können Sie das gewünschte Anzeigeformat aus der Klappliste in der Spalte "Anzeigeformate" auswählen. In der Klappliste werden nur die Anzeigeformate angeboten, die für diesen Datentyp zulässig sind. Das zuerst genannte Anzeigeformat ist das voreingestellte Format.

**Beispiel**

Die folgende Tabelle zeigt die für alle CPU-Familien in der Beobachtungstabelle zulässigen 32-Bit-Datentypen und ihre möglichen Anzeigeformate:

| Datentyp      | Mögliche Anzeigeformate   |
|---------------|---|
| BOOL          | Bool, Hex, BCD, Oktal, Bin, Dez, Dez+/-   |
| BYTE          | Hex, BCD, Oktal, Bin, Dez, Dez+/-, Zeichen  |
| WORD          | Hex, BCD, Oktal, Bin, Dez, Dez+/-, Dez_Sequenz, Zeichen, Unicode_Zeichen, SIMATIC_Zeit, Datum, Zähler                   |
| DWORD         | Hex, BCD, Oktal, Bin, Dez, Dez+/-, Dez_Sequenz, Zeichen, Unicode_Zeichen, Gleitpunktzahl, Tageszeit, Zeit, Zeiger       |
| SINT          | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen  |
| INT           | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, SIMATIC_Zeit, Zähler, Datum                   |
| DINT          | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Zeiger       |
| USINT         | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen  |
| UINT          | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, SIMATIC_Zeit, Zähler, Datum                   |
| UDINT         | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Zeiger       |
| REAL          | Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez +/-, Dez_Sequenz, Tageszeit, Zeit, Zeiger      |
| DATE          | Datum, Dez, Hex, BCD, Bin   |
| TIME_OF_DAY   | Tageszeit, Dez, Hex, BCD, Bin   |
| TIME          | Zeit, Hex, BCD, Bin   |
| DATE_AND_TIME | Datum und Uhrzeit,  |
| TIMER         | SIMATIC_Zeit, Hex, BCD, Bin   |
| CHAR          | Zeichen, Hex, BCD, Oktal, Bin, Dez, Dez+/-,   |
| STRING        | Zeichenkette  |
| POINTER       | Zeiger, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Zahl |
| COUNTER       | Zähler, Hex, BCD, Bin   |
| S5TIME        | SIMATIC_Zeit, Hex, BCD, Bin   |

Für die CPU-Familie S7-1200 sind alle 32-Bit-Datentypen zulässig, siehe obige Tabelle, sowie zusätzlich der 64-Bit-Datentyp LREAL mit folgenden möglichen Anzeigeformaten:

| Datentyp | Mögliche Anzeigeformate  |
|----------|--|
| LREAL    | in einem mit dem TIA-Portal V12 erstellten Projekt:<br>Gleitpunktzahl<br>Hinweis: Die Anzeige von LREAL ist auf 13 Stellen zuzüglich Exponent begrenzt.  |
| LREAL    | in einem mit dem TIA-Portal V12 erstellten Projekt:<br>Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez +/-, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit<br>Hinweis: Die Anzeige von LREAL ist auf 13 Stellen zuzüglich Exponent begrenzt. |

Für die CPU-Familie S7-1500 sind zusätzlich zu 32-Bit-Datentypen auch die in der Tabelle aufgelisteten 64-Bit-Datentypen zulässig mit folgenden möglichen Anzeigeformaten:

| Datentyp | Mögliche Anzeigeformate   |
|----------|---|
| LWORD    | Hex, Oktal, BCD, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Datum und Uhrzeit  |
| LINT     | Dez+/-, Dez, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit                  |
| ULINT    | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit                  |
| LREAL    | Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez +/-, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit |
| LTIME    | Zeit, Dez+/-, Dez, Hex  |
| LTOD     | Tageszeit, Dez, Hex, BCD, Bin   |
| LDT      | Datum und Uhrzeit, Dez, Hex   |

Weitere Informationen hierzu finden Sie bei der Beschreibung der gültigen Datentypen.

### Hinweis

#### Rundungen bei Gleitpunktzahlen

In der Beobachtungstabelle werden Gleitpunktzahlen binär im IEEE-Format gespeichert. Da sich nicht jede an der Oberfläche darstellbare Gleitpunkt-Zahl (real, longreal) eindeutig auf das IEEE-Format abbilden lässt, kann es zu Rundungen bei Gleitpunktzahlen kommen. Wird eine solche gerundete Gleitpunktzahl in der Beobachtungstabelle kopiert und in ein anderes Eingabefeld wieder eingefügt, erhält man durch die Rundung unter Umständen eine geringe Abweichung.

---

**Hinweis**

**Nur symbolische Adressierung möglich**

In der Beobachtungstabelle können LongDataTypes wie z. B. LWORD oder LREAL nur symbolisch adressiert werden.

---

## Anzeigeformat für Variablen auswählen

### Vorgehen

Um das Anzeigeformat der Variablen auszuwählen, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschte Adresse in die Beobachtungstabelle ein.
2. Klicken Sie in der Spalte "Anzeigeformat" auf die gewünschte Zelle und öffnen Sie die Klappliste.  
Die zulässigen Anzeigeformate werden in der Klappliste angezeigt.
3. Wählen Sie aus der Klappliste das gewünschte Anzeigeformat aus.

---

**Hinweis**

Falls das ausgewählte Anzeigeformat nicht übernommen werden kann, wird automatisch das zuletzt ausgewählte Anzeigeformat angezeigt.

---

### 9.4.3.7 Variablen in der Beobachtungstabelle beobachten

#### Einführung zum Beobachten von Variablen in der Beobachtungstabelle

##### Einführung

In der Beobachtungstabelle können Sie die Variablen der projektierten Ein- und Ausgabebaugruppen in der CPU beobachten, abhängig vom jeweils ausgewählten Beobachtungs- und Steuermodus (Seite 1548). Um Variablen zu beobachten, muss eine Online-Verbindung zur CPU bestehen.



**ACHTUNG****Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen**

Beachten Sie, dass das Beobachten von Peripherieeingängen evtl. zu einer Zykluszeitüberschreitung führen kann!

Die CPU geht dann in den Betriebszustand "STOP".

**Optionen zum Beobachten von Variablen**

Folgende Optionen für das Beobachten von Variablen stehen zur Verfügung:

- **Sofort beobachten**  
Dieser Befehl startet das Beobachten der sichtbaren Variablen in der aktiven Beobachtungstabelle sofort und einmalig.
- **Alle beobachten**  
Dieser Befehl startet das Beobachten aller sichtbaren Variablen in der aktiven Beobachtungstabelle, abhängig vom ausgewählten Beobachtungsmodus:
  - Im Basismodus ist der Beobachtungsmodus "permanent" voreingestellt.
  - Im Erweiterten Modus können Sie bestimmte Triggerpunkte für das Beobachten der Variablen festlegen.

**Hinweis**

Wird der Beobachtungsmodus im Erweiterten Modus umgestellt und wieder in den Basismodus umgeschaltet, wird der zuvor eingestellte Beobachtungsmodus auch im Basismodus übernommen.

**CPU-spezifische Einschränkungen beim Beobachten von Variablen**

Es bestehen folgende CPU-spezifische Unterschiede:

- **CPU S7-300/400:**  
CPUs aus dieser Familie können lediglich die ersten 30 Zeichen eines Strings beobachten.
- **CPU S7-1200/1500:**  
CPUs aus dieser Familie können einen String bis zur Gesamtgröße von 254 Zeichen beobachten.

## Festlegen des Beobachtungs- und Steuermodus

### Einführung

Mit der Auswahl des Beobachtungs- und Steuermodus legen Sie den Triggerpunkt und die Dauer des Beobachtens von Variablen in der Beobachtungstabelle und der Forcetabelle fest.

### Möglicher Beobachtungs- und Steuermodus (Dauer des Beobachtens bzw. Steuerns)

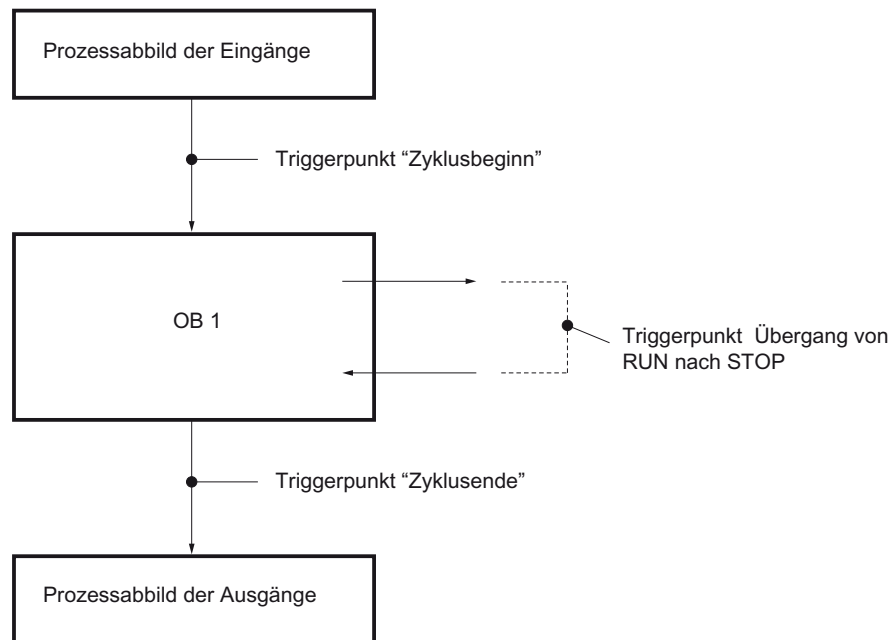
Folgende Beobachtungs- und Steuermodi stehen zur Verfügung:

- Permanent
  - In diesem Modus können die Eingänge am Anfang und die Ausgänge am Ende des Zyklus beobachtet werden.
- Zyklusbeginn einmalig
- Zyklusende einmalig
- Zyklusbeginn permanent
- Zyklusende permanent
- Übergang von RUN nach STOP einmalig
- Übergang von RUN nach STOP permanent

### Triggerpunkt festlegen

Die Triggerpunkte "Zyklusbeginn", "Zyklusende" und "Übergang von RUN nach STOP" legen fest, zu welchem Zeitpunkt die Variablen aus der CPU gelesen bzw. in der CPU aktualisiert werden sollen.

Das folgende Bild zeigt die Lage der genannten Triggerpunkte:



### Lage der Triggerpunkte

Aus der Lage der Triggerpunkte geht hervor:

- Das Steuern von Eingängen ist nur sinnvoll am Zyklusbeginn (entspricht dem Beginn des Anwenderprogramms OB 1), da sonst das Prozessabbild der Eingänge nach dem Steuern wieder aktualisiert und damit überschrieben wird.
- Das Steuern von Ausgängen ist nur sinnvoll am Zyklusende (entspricht dem Ende des Anwenderprogramms OB 1), da sonst das Prozessabbild der Ausgänge vom Anwenderprogramm überschrieben werden kann.
- Der gesteuerte Wert wird in der Spalte "Beobachtungswert" angezeigt, sofern aktiv beobachtet wird und der gesteuerte Wert nicht vom Anwenderprogramm überschrieben wird.

### Steuern von Variablen

Beim Steuern von Variablen gilt bezüglich der Triggerpunkte:

- Wenn Sie als Steuermodus "einmalig" eingestellt haben, so erhalten Sie eine Meldung, falls die ausgewählten Variablen nicht zu steuern sind.
- Beim Steuermodus "permanent" erhalten Sie keine Rückmeldung.

### Anmerkung zum Befehl "Sofort steuern"

Sie können die Werte ausgewählter Variablen über den Befehl "Online > Steuern > Sofort steuern" sofort steuern. Dieser Auftrag wird einmalig und schnellstmöglich durchgeführt, ohne Bezug zu einer bestimmten Stelle (Triggerpunkt) im Anwenderprogramm. Diese Funktion wird hauptsächlich im STOP-Zustand der CPU zum Steuern angewendet.

## Variablen "Alle beobachten"

### Einführung

Mit dem Befehl "Alle beobachten" können Sie das Beobachten der sichtbaren Variablen in der aktiven Beobachtungstabelle starten. Im Basismodus der Beobachtungstabelle ist für den Beobachtungsmodus "permanent" voreingestellt. Im Erweiterten Modus können Sie bestimmte Triggerpunkte für das Beobachten der Variablen festlegen. In diesem Fall werden die Variablen bezüglich der festgelegten Triggerpunkte beobachtet.

|  |
|--|
| <b>ACHTUNG</b>   |
| <b>Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen</b>                           |
| Beachten Sie, dass das Beobachten von Peripherieeingängen evtl. zu einer Zykluszeitüberschreitung führen kann! |
| Die CPU geht dann in den Betriebszustand "STOP".   |

### Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

### Vorgehensweise

Um den Befehl "Alle beobachten" auszuführen, gehen Sie folgendermaßen vor:

1. Geben Sie die zu beobachtenden Variablen und die entsprechenden Adressen in die Beobachtungstabelle ein.
2. Schalten Sie in den Erweiterten Modus um, indem Sie in der Funktionsleiste auf das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden" klicken.
3. Wenn Sie den voreingestellten Beobachtungsmodus einer Variablen ändern wollen, klicken Sie in der Spalte "Beobachten mit Trigger" auf die entsprechende Zelle und wählen Sie aus der Klappliste den gewünschten Beobachtungsmodus aus.
4. Klicken Sie in der Funktionsleiste auf das Symbol "Alle beobachten".

### Ergebnis

Die Variablen der aktiven Beobachtungstabelle werden mit dem jeweils eingestellten Beobachtungsmodus beobachtet.

## Siehe auch

Symbole in der Beobachtungstabelle (Seite 1536)

Variablen in die Beobachtungstabelle eingeben (Seite 1539)

Basismodus und Erweiterter Modus in der Beobachtungstabelle (Seite 1535)

## Variablen "Sofort beobachten"

### Einführung

Mit dem Befehl "Sofort beobachten" wird das Beobachten der Variablen sofort, ohne Bezug auf bestimmte Triggerpunkte gestartet. Die Variablenwerte werden einmalig ausgelesen und in der Beobachtungstabelle angezeigt.

| <b>ACHTUNG</b>   |
|--|
| <b>Gefahr einer Zykluszeitüberschreitung beim Beobachten von Peripherieeingängen</b>                           |
| Beachten Sie, dass das Beobachten von Peripherieeingängen evtl. zu einer Zykluszeitüberschreitung führen kann! |
| Die CPU geht dann in den Betriebszustand "STOP".   |

### Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

### Vorgehensweise

Um den Befehl "Sofort beobachten" auszuführen, gehen Sie folgendermaßen vor:

1. Geben Sie die zu beobachtenden Variablen und die entsprechenden Adressen in die Beobachtungstabelle ein.
2. Klicken Sie in der Funktionsleiste auf das Symbol "Sofort beobachten".

### Ergebnis

Die Variablen der aktiven Beobachtungstabelle werden sofort und einmalig beobachtet.

## Siehe auch

Symbole in der Beobachtungstabelle (Seite 1536)

Variablen in die Beobachtungstabelle eingeben (Seite 1539)

Basismodus und Erweiterter Modus in der Beobachtungstabelle (Seite 1535)

### 9.4.3.8 Variablen in der Beobachtungstabelle steuern

#### Einführung zum Steuern von Variablen

##### Einführung

Mithilfe der Beobachtungstabelle können Sie die Variablen der projizierten Ein- und Ausgabebaugruppen in der CPU steuern, abhängig vom jeweils ausgewählten Beobachtungs- und Steuermodus (Seite 1548). Um die Variablen zu beobachten, muss eine Online-Verbindung zur CPU bestehen.



#### **GEFAHR**

##### **Gefahr beim Steuern:**

Ein Verändern der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

#### **ACHTUNG**

##### **Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen**

Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!

Die CPU geht dann in den Betriebszustand "STOP".

#### Optionen zum Steuern von Variablen

Folgende Optionen für das Steuern von Variablen stehen zur Verfügung:

- Steuern auf "0"  
Dieser Befehl steuert die selektierte Adresse auf den Steuerwert "0".
- Steuern auf "1"  
Dieser Befehl steuert die selektierte Adresse auf den Steuerwert "1".
- Einmalig und sofort steuern  
Dieser Befehl steuert alle aktivierten Adressen in der aktiven Beobachtungstabelle "einmalig und sofort".

- **Steuern mit Trigger**  
Dieser Befehl steuert alle aktivierten Adressen in der aktiven Beobachtungstabelle mit dem jeweils ausgewählten Beobachtungs- und Steuermodus (Seite 1548).  
Die Funktion "Steuern mit Trigger" ist nur im Erweiterten Modus verfügbar. Sie erhalten keine Rückmeldung, ob die aktivierten Adressen tatsächlich mit dem vorgegebenen Wert gesteuert wurden oder nicht. Falls Sie eine solche Rückmeldung benötigen, sollten Sie die Funktion "Einmalig und sofort steuern" verwenden.
- **Peripherieausgänge freischalten**  
Dieser Befehl schaltet die Befehlsausgabesperre (BASP) ab.  
Diese Funktion ist nur durchführbar im Erweiterten Modus, wenn die CPU im STOP ist und wenn das Forcen (Seite 1576) von Variablen nicht aktiviert ist. Deaktivieren Sie diese Funktion, falls gewünscht, in der Forcetabelle.

---

**Hinweis****Beachten Sie beim Steuern:**

Steuern kann **nicht** rückgängig gemacht werden.

---

## Variablen steuern auf "0"

### Einführung

Sie können Variablen unabhängig vom Beobachtungs- und Steuerungsmodus einmalig Werte zuweisen und diese steuern. Beim Ausführen wird der Steuerauftrag wie ein "Trigger sofort" schnellst möglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.

 **GEFAHR****Gefahr beim Steuern:**

Ein Verändern der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.

**ACHTUNG****Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen**

Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!

Die CPU geht dann in den Betriebszustand "STOP".

### Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

### Vorgehen

Um Variablen auf "0" zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschte Adresse in die Beobachtungstabelle ein.
2. Wählen Sie den Befehl "Online > Steuern > Steuern auf 0", um die ausgewählte Adresse mit dem vorgegebenen Wert zu steuern.

### Ergebnis

Die ausgewählte Adresse wird auf "0" gesteuert.

---

#### Hinweis

#### Beachten Sie beim Steuern:

Steuern kann **nicht** rückgängig gemacht werden!

---

## Variablen steuern auf "1"

### Einführung

Sie können Variablen unabhängig vom Beobachtungs- und Steuerungsmodus einmalig Werte zuweisen und diese steuern. Beim Ausführen wird der Steuerauftrag wie ein "Trigger sofort" schnellst möglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.



#### GEFAHR

#### Gefahr beim Steuern:

Ein Verändern der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen.



**ACHTUNG****Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen**

Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!

Die CPU geht dann in den Betriebszustand "STOP".

**Voraussetzung**

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

**Vorgehen**

Um Variablen auf "1" zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschte Adresse in die Beobachtungstabelle ein.
2. Wählen Sie den Befehl "Online > Steuern > Steuern auf 1", um die ausgewählte Adresse mit dem vorgegebenen Wert zu steuern.

**Ergebnis**

Die ausgewählte Adresse wird auf "1" gesteuert.


**Hinweis****Beachten Sie beim Steuern:**

Steuern kann **nicht** rückgängig gemacht werden!

**Variablen "Sofort steuern"****Einführung**

Sie können Variablen unabhängig vom Beobachtungs- und Steuerungsmodus einmalig Werte zuweisen und diese sofort steuern. Beim Ausführen wird der Steuerauftrag wie ein "Trigger

sofort" schnellst möglich durchgeführt, ohne Bezug zu einer bestimmten Stelle im Anwenderprogramm.

|  |
|--|
|  <b>GEFAHR</b>  |
| <b>Gefahr beim Steuern:</b><br>Ein Verändern der der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!<br>Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen. |

|   |
|---|
| <b>ACHTUNG</b>  |
| <b>Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen</b><br>Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!<br>Die CPU geht dann in den Betriebszustand "STOP". |

### Voraussetzung

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

### Vorgehen

Um Variablen sofort zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Adressen und Steuerwerte in die Beobachtungstabelle ein.
2. Wählen Sie die zu steuernden Adressen aus, indem Sie das Optionskästchen für das Steuern in der Spalte nach dem "Steuerwert" aktivieren.  
Ein gelbes Dreieck erscheint hinter dem aktivierten Optionskästchen und zeigt an, dass die Adresse nun für das Steuern ausgewählt ist, aber noch nicht gesteuert wurde.
3. Wählen Sie den Befehl "Online > Steuern > Steuern sofort", um die ausgewählten Adressen einmalig und sofort mit den vorgegebenen Werten zu steuern.

## Ergebnis

Die ausgewählten Adressen werden einmalig und sofort gesteuert.

---

### Hinweis

#### Beachten Sie beim Steuern:


Steuern kann **nicht** rückgängig gemacht werden!

---

## Variablen "Steuern mit Trigger"

### Einführung

Sie können Adressen in Abhängigkeit vom definierten Beobachtungs- und Steuerungsmodus Werte zuweisen und diese steuern. Beim Ausführen wird der Steuerauftrag durchgeführt, wie im Beobachtungs- und Steuerungsmodus festgelegt, mit Bezug zu der als Trigger bestimmten Stelle im Anwenderprogramm.

|  |
|--|
|  <b>GEFAHR</b>  |
| <b>Gefahr beim Steuern:</b><br>Ein Verändern der Variablen bzw. der Adressen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!<br>Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Steuern" ausführen. |

|   |
|---|
| <b>ACHTUNG</b>  |
| <b>Gefahr einer Zykluszeitüberschreitung beim Steuern von Peripherieausgängen</b><br>Beachten Sie, dass das Steuern von Peripherieausgängen in der Beobachtungstabelle evtl. zu einer Zykluszeitüberschreitung führen kann!<br>Die CPU geht dann in den Betriebszustand "STOP". |

### Voraussetzungen

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die Beobachtungstabelle muss im Erweiterten Modus sein.

## Vorgehen

Um Variablen "mit Trigger" zu steuern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Adressen und Steuerwerte in die Beobachtungstabelle ein.
2. Wählen Sie die zu steuernden Adressen aus, indem Sie das Optionskästchen für das Steuern in der Spalte nach dem "Steuerwert" aktivieren.  
Ein gelbes Dreieck erscheint hinter dem aktivierten Optionskästchen und zeigt an, dass die Adresse nun für das Steuern ausgewählt ist, aber noch nicht gesteuert wurde.
3. Schalten Sie über das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden" in der Funktionsleiste oder über den Befehl "Online > Erweiterter Modus" in den erweiterten Modus um.  
Die Spalten "Beobachten mit Trigger" und "Steuern mit Trigger" werden angezeigt.
4. Wählen Sie in der Spalte "Steuern mit Trigger" den gewünschten Steuermodus aus der Auswahlliste aus. Folgende Optionen stehen zur Verfügung:
  - Permanent
  - Zyklusbeginn, permanent
  - Zyklusbeginn, einmalig
  - Zyklusende, permanent
  - Zyklusende, einmalig
  - Übergang nach STOP, permanent
  - Übergang nach STOP, einmalig
5. Starten Sie das Steuern über den Befehl "Online > Steuern > Steuern mit Trigger".
6. Bestätigen Sie die Anfrage mit "Ja", falls Sie das Steuern mit Trigger starten wollen.

## Ergebnis

Die aktivierten Variablen werden mit dem ausgewählten Beobachtungs- und Steuermodus gesteuert. Das gelbe Dreieck wird nicht mehr angezeigt.

---

### Hinweis

#### Beachten Sie beim Steuern:

Steuern kann **nicht** rückgängig gemacht werden!


---

## Peripherieausgänge freischalten

### Einführung

Die Funktion "Peripherieausgänge freischalten" schaltet die Befehlsausgabesperre der Peripherieausgänge (BASP) ab. Dies ermöglicht das Steuern der Peripherieausgänge, wenn

sich die CPU im Betriebszustand STOP befindet. Diese Funktion steht in der Beobachtungstabelle nur im "Erweiterten Modus" zur Verfügung.

|  |
|--|
|  <b>GEFAHR</b>  |
| <b>Gefahr beim Freischalten der Peripherieausgänge:</b><br>Beachten Sie, dass das Freischalten der Peripherieausgänge schwere Sach- und Personenschäden verursachen kann!<br>Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie die Funktion "Peripherieausgänge freischalten" ausführen. |

## Voraussetzungen

- Eine Beobachtungstabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die CPU muss für das Freischalten der Peripherieausgänge im Betriebszustand STOP sein.
- Die Beobachtungstabelle muss im Erweiterten Modus sein.
- Das Forcen (Seite 1576) von Variablen darf nicht aktiv sein.

---

### Hinweis

#### Funktion "Peripherieausgänge freischalten"

- Diese Funktion ist nur im Betriebszustand STOP möglich. Sie wird beendet durch einen Betriebszustandswechsel der CPU und beim Schließen der Online-Verbindung.
  - Solange die Funktion aktiv ist, kann nicht geforct werden.
- 

## Vorgehen

Um die Peripherieausgänge im Betriebszustand STOP freizuschalten, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Adressen und Steuerwerte in die Beobachtungstabelle ein.
2. Wählen Sie die zu steuernden Adressen aus, indem Sie das Optionskästchen für das Steuern in der Spalte nach dem "Steuerwert" aktivieren.  
Ein gelbes Dreieck erscheint hinter dem aktivierten Optionskästchen und zeigt an, dass die Adresse nun für das Steuern ausgewählt ist, aber noch nicht gesteuert wurde.
3. Schalten Sie über das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden" in der Funktionsleiste oder über den Befehl "Online > Erweiterter Modus" in den erweiterten Modus um.  
Die Spalten "Beobachten mit Trigger" und "Steuern mit Trigger" werden angezeigt.
4. Schalten Sie die zugeordnete CPU über das Bedienpanel in den Betriebszustand STOP.
5. Rufen Sie mit der rechten Maustaste das Kontextmenü auf und aktivieren Sie den Befehl "Peripherieausgänge freischalten".

6. Bestätigen Sie die Anfrage mit "Ja", falls Sie die Befehlsausgabesperre für die Peripherieausgänge freischalten wollen.
7. Steuern Sie die Peripherieausgänge über den Befehl "Online > Steuern > Steuern sofort".

## Ergebnis

Die Peripherieausgänge werden mit den ausgewählten Steuerwerten gesteuert. Das gelbe Dreieck wird nicht mehr angezeigt.

## Freischaltung der Peripherieausgänge

Die Funktion "Peripherieausgänge freischalten" bleibt so lange aktiviert bis:

- Der Befehl "Peripherieausgänge freischalten" über das Kontextmenü oder über den Befehl "Online > Steuern > Peripherieausgänge freischalten" wieder deaktiviert wird.
- Die CPU nicht mehr im Betriebszustand STOP ist.
- Die Online-Verbindung beendet wird.

---

### Hinweis

**Beachten Sie beim Steuern:**

Steuern kann **nicht** rückgängig gemacht werden!

---

## 9.4.4 Testen mit der Forcetabelle

### 9.4.4.1 Einführung zum Testen mit der Forcetabelle

#### Übersicht

Mit Hilfe der Forcetabelle können Sie einzelne Variablen des Anwenderprogramms mit festen Werten vorbelegen. Diesen Vorgang nennt man "Forcen".

Innerhalb der Forcetabelle stehen Ihnen folgende Funktionen zur Verfügung:

- **Beobachten von Variablen**  
Damit lassen sich die aktuellen Werte einzelner Variablen eines Anwenderprogramms bzw. einer CPU am PG/PC anzeigen. Das Beobachten kann mit oder ohne Triggerbedingung ausgeführt werden.
- **Forcen von Variablen**  
Damit können Sie einzelnen Peripherie-Variablen eines Anwenderprogramms einen festen Wert zuweisen.

## Beobachten und Forcen von Variablen

Das Beobachten und Forcen von Variablen ist immer abhängig vom Operandenumfang der eingesetzten CPU.

Folgende Variablen können beobachtet werden:

- Eingänge, Ausgänge und Merker
- Inhalte von Datenbausteinen
- Peripherieeingänge

Folgende Variablen können geforct werden:

- Peripherieeingänge
- Peripherieausgänge

## Beispiel

- Unabhängig von der jeweils eingesetzten CPU kann ausschließlich Peripherie geforct werden, also z. B.: "Tag\_1":P oder "AW0:P" oder "EW0:P". Beachten Sie dabei, dass "Tag\_1":P nicht die symbolische Bezeichnung für einen Merker sein darf.

## Einsatzmöglichkeit

Die Forcetabelle bietet den Vorteil, sowohl unterschiedliche Testumgebungen simulieren zu können als auch Variablen in der CPU mit einem festen Wert überschreiben zu können. Damit haben Sie die Möglichkeit, regulierend in den laufenden Prozess einzugreifen.

## Siehe auch

Aufbau der Forcetabelle (Seite 1562)

Basismodus und Erweiterter Modus in der Forcetabelle (Seite 1563)


Symbole in der Forcetabelle (Seite 1564)


Forcetabelle öffnen und bearbeiten (Seite 1565)

### 9.4.4.2 Sicherheitsvorkehrungen beim Forcen von Variablen

#### Sicherheitsvorkehrungen zum Forcen von Variablen

Da Sie mit der Funktionalität des Forcens permanent in den Prozess eingreifen, sind die folgenden Hinweise unbedingt zu beachten:

|  |
|--|
|  <b>GEFAHR</b>  |
| <b>Personen- und Sachschaden verhindern!</b><br>Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung: <ul style="list-style-type: none"><li>• Das Leben oder die Gesundheit von Personen gefährden kann.</li><li>• Schäden an der Maschine oder der gesamten Anlage verursachen kann.</li></ul> |

|  |
|--|
|  <b>VORSICHT</b>  |
| <b>Personen- und Sachschäden verhindern!</b> <ul style="list-style-type: none"><li>• Bevor Sie die Funktion "Forcen" starten, sollten Sie sicherstellen, dass niemand gleichzeitig auf derselben CPU diese Funktion ebenfalls ausführt.</li><li>• Das Forcen kann nur durch Klicken auf das Symbol "Forcen beenden" oder über den Befehl "Online &gt; Forcen &gt; Forcen beenden" beendet werden. Das Schließen der aktiven Forcetabelle beendet das Forcen <b>nicht!</b></li><li>• "Forcen" kann <b>nicht</b> rückgängig gemacht werden!</li><li>• Informieren Sie sich über die Unterschiede zwischen "Steuern von Variablen" (Seite 1552) und "Forcen von Variablen" (Seite 1576)".</li><li>• Falls eine CPU die Funktionalität des "Forcens" nicht unterstützt, so sind alle das Forcen betreffenden Symbole nicht anwählbar.</li><li>• Wenn die Funktion "Peripherieausgänge freischalten" auf einer CPU aktiv ist, ist das Forcen auf dieser CPU <b>nicht</b> möglich. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.</li></ul> |

### 9.4.4.3 Aufbau der Forcetabelle

#### Einführung

In die Forcetabelle tragen Sie die von Ihnen definierten und ausgewählten CPU-weit gültigen Variablen ein, die in der zugeordneten CPU geforct werden sollen. Es können nur Peripherieeingänge und Peripherieausgänge geforct werden.

Für jede im Projekt angelegte CPU wird automatisch eine Forcetabelle im Ordner "Beobachtungs- und Forcetabellen" angelegt. Einer CPU kann nur eine einzige Forcetabelle



zugeordnet werden. Diese Forcetabelle zeigt alle in der zugeordneten CPU geforcten Adressen an.

### Aufbau der Forcetabelle

Die in der Forcetabelle angezeigten Spalten sind abhängig davon, ob Sie im Basismodus oder im Erweiterten Modus arbeiten.

Im Erweiterten Modus wird die Spalte "Beobachten mit Trigger" zusätzlich angezeigt.

### Bedeutung der Spalten

Die folgende Tabelle zeigt die Bedeutung der einzelnen Spalten im Basismodus und im Erweiterten Modus:

| Mode  | Spalte                 | Erläuterung  |
|---|------------------------|--|
| Basismodus  | <b>i</b>               | Kennzeichnungsspalte   |
|   | Name                   | Name der eingefügten Variablen   |
|   | Adresse                | Adresse der eingefügten Variablen  |
|   | Anzeigeformat          | Ausgewähltes Anzeigeformat   |
|   | Beobachtungswert       | Werte der Variablen, abhängig vom ausgewählten Anzeigeformat.                            |
|   | Forcewert              | Wert, mit dem die Variable geforct wird.   |
|   | <b>F</b> ("Forcen")    | Auswahl der zu forcenden Variablen durch Aktivieren des entsprechenden Optionskästchens. |
| Im Erweiterten Modus wird zusätzlich folgende Spalte angezeigt: | Kommentar              | Kommentar zur Dokumentation der Variablen  |
|   | Beobachten mit Trigger | Anzeige des ausgewählten Beobachtungsmodus   |

### Siehe auch

Symbole in der Forcetabelle (Seite 1564)

Basismodus und Erweiterter Modus in der Forcetabelle (Seite 1563)

#### 9.4.4.4 Basismodus und Erweiterter Modus in der Forcetabelle

### Unterschied zwischen Basismodus und Erweitertem Modus in der Forcetabelle

Im Erweiterten Modus wird in der Forcetabelle zusätzlich die Spalte "Beobachten mit Trigger" angezeigt.

Eine detaillierte Auflistung der jeweiligen Spalten finden Sie unter Aufbau der Forcetabelle (Seite 1562).

### Umschalten zwischen Basismodus und Erweiterter Modus

Um zwischen Basismodus und Erweiterter Modus umzuschalten, haben Sie folgende Möglichkeiten:

- Klicken Sie auf das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden". Ein erneutes Klicken auf dieses Symbol schaltet wieder in den Basismodus um.  
Oder:
- Aktivieren Sie im Menü "Online" das Kontrollkästchen vor dem Befehl "Erweiterter Modus". Ein Deaktivieren des Kontrollkästchens schaltet wieder in den Basismodus um.













### Funktionalität im Erweiterten Modus





Die Auswahl der Funktionalität "Beobachten mit Trigger" ist nur im Erweiterten Modus möglich.

#### 9.4.4.5 Symbole in der Forcetabelle

### Bedeutung der Symbole

Die folgende Tabelle zeigt die Bedeutung der Symbole in der Forcetabelle:

| Symbol  | Bedeutung  |
|---|--|
|  | Kennzeichnet eine Tabelle innerhalb der Projektnavigation als Forcetabelle.  |
|  | Kennzeichnungsspalte   |
|  | Blendet alle Spalten des Erweiterten Modus ein. Wenn Sie erneut auf dieses Symbol klicken, werden die Spalten des Erweiterten Modus wieder ausgeblendet.   |
|  | Startet das Forcen für alle Adressen der ausgewählten Variablen. Falls das Forcen bereits läuft, wird der vorherige Vorgang unterbrechungsfrei ersetzt.  |
|  | Beendet das Forcen von Adressen in der Forcetabelle.   |
|  | Startet das Beobachten der sichtbaren Variablen in der Forcetabelle. Im Basismodus ist für das Beobachten "permanent" voreingestellt. Im Erweiterten Modus wird eine zusätzliche Spalte eingeblendet und Sie können für das Beobachten der Variablen bestimmte Triggerpunkte einstellen. |
|  | Startet das Beobachten der sichtbaren Variablen in der Forcetabelle. Dieser Befehl wird sofort ausgeführt und die Variablen werden einmalig beobachtet.  |
|  | Zeigt das Optionskästchen für die Auswahl der zu forcenden Variablen an.   |
|  | Zeigt an, dass eine Adresse nicht geforct werden kann.   |
|  | Zeigt an, dass eine Adresse evtl. nicht vollständig geforct werden kann. Beispiel: Es ist zwar möglich, die Adresse AW0:P zu forcen, es ist jedoch nicht möglich, die Adresse AD0:P zu forcen, da dieser Adressbereich auf der CPU evtl. nicht vorhanden ist.                            |
|  | Zeigt an, dass eine Adresse nicht beobachtet werden kann.  |
|  | Zeigt an, dass eine Adresse geforct wird.  |

| Symbol  | Bedeutung   |
|---|---|
|  | Zeigt an, dass eine Adresse teilweise geforct wird.   |
|  | Zeigt an, dass die zugehörige Peripherie-Adresse geforct wird.                                |
|  | Zeigt an, dass ein Syntaxfehler aufgetreten ist.  |
|  | Zeigt an, dass die Adresse zwar ausgewählt ist, im Moment aber z. B. noch nicht geforct wird. |

## Siehe auch

Aufbau der Forcetabelle (Seite 1562)

### 9.4.4.6 Forcetabelle öffnen und bearbeiten

#### Forcetabelle anzeigen

#### Einführung

Eine Forcetabelle kann nicht neu erstellt werden, sie ist für jede CPU bereits einmal vorhanden. Sie ist dieser CPU fest zugeordnet und kann nicht kopiert oder dupliziert werden.

#### Voraussetzung

Ein Projekt mit einer zugeordneten CPU muss geöffnet sein.

#### Anzeige einer Forcetabelle

Die Forcetabelle wird immer unterhalb einer CPU im Ordner "Beobachtungs- und Forcetabellen" angezeigt.

#### Forcetabelle öffnen

#### Voraussetzung

Ein Projekt mit einer zugeordneten CPU muss angelegt sein.

#### Vorgehen

Um eine Forcetabelle zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Beobachtungs- und Forcetabellen" unterhalb der gewünschten CPU.
2. Doppelklicken Sie auf die in diesem Ordner enthaltene "Forcetabelle".

### Ergebnis

Die ausgewählte Forcetabelle wird geöffnet.

### Forcetabelle speichern

### Voraussetzung

Ein Projekt mit einer zugeordneten CPU ist angelegt.

### Vorgehen

Um eine Forcetabelle zu speichern, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschten Änderungen in die Forcetabelle ein.
2. Wählen Sie im Menü "Projekt" den Befehl "Speichern" oder klicken auf das Symbol "Projekt speichern" in der Funktionsleiste. Beachten Sie, dass mit diesem Speichervorgang das gesamte Projekt gespeichert wird.

### Ergebnis

Die Inhalte der Forcetabelle und das zugehörige Projekt werden gespeichert.

---

#### Hinweis

Eine Forcetabelle kann nicht umbenannt werden!

---

## 9.4.4.7 Variablen in die Forcetabelle eingeben

### Grundlagen für die Eingabe von Variablen in die Forcetabelle

#### Empfohlene Vorgehensweise

Bestimmen Sie die Variablen, deren Werte Sie beobachten oder forcen wollen und tragen Sie diese in die Forcetabelle ein.

Beachten Sie bei der Eingabe von Variablen in die Forcetabelle, dass es sinnvoll ist, diese Variablen zuvor in der PLC-Variablentabelle zu definieren.

#### Beispiel für das Ausfüllen einer Forcetabelle

- Sie können in der Spalte "Adresse" die absolute Adresse eintragen, die geforct oder beobachtet werden soll, oder Sie tragen in der Spalte "Name" den symbolischen Namen für die Variable ein.
- Wählen Sie in der Spalte "Anzeigeformat" das gewünschte Anzeigeformat aus der Klappliste aus, falls Sie die Voreinstellung nicht verwenden wollen.

- Entscheiden Sie nun, ob Sie die eingetragenen Variablen beobachten oder forcen wollen. Tragen Sie dazu den gewünschten Forcewert sowie einen Kommentar in die entsprechenden Spalten der Forcetabelle ein.
- Beachten Sie bitte, dass nur Peripherieeingänge und Peripherieausgänge geforct werden können und informieren Sie sich über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 1578).

## Syntaxprüfung

Wenn Sie Variablen in die Forcetabelle eintragen, so wird vor dem Verlassen der Zelle eine Syntaxprüfung durchgeführt. Fehlerhafte Einträge werden rot gekennzeichnet.

### Hinweis

Wenn Sie den Cursor in eine rot markierte Zelle stellen, so erhalten Sie über eine kurze Information weitere Hinweise zum Fehler.

## Erlaubte Operanden für die Forcetabelle

### Zulässige Operanden für die Forcetabelle

Die folgende Tabelle zeigt die Operanden, die für das Forcen in der Forcetabelle zugelassen sind:

| Erlaubter Operand                   | Beispiel für Datentyp | Beispiel (Mnemonic deutsch) |
|-------------------------------------|-----------------------|-----------------------------|
| Peripherieeingang/Peripherieausgang | BOOL                  | E0.0:P; A0.0:P              |
| Peripherieeingang/Peripherieausgang | BYTE                  | EB1:P; AB1:P                |
| Peripherieeingang/Peripherieausgang | WORD                  | EW2:P; AW3:P                |
| Peripherieeingang/Peripherieausgang | DWORD                 | ED2:P; AD1:P                |

Die nachfolgende Tabelle zeigt die Operanden, die für das Beobachten in der Forcetabelle zugelassen sind:

| Erlaubter Operand      | Beispiel für Datentyp | Beispiel (Mnemonic deutsch) |
|------------------------|-----------------------|-----------------------------|
| Eingang/Ausgang/Merker | BOOL                  | E1.0; A1.7; M10.1<br>E0.0:P |
| Eingang/Ausgang/Merker | BYTE                  | EB1; AB10; MB100<br>EB1:P   |
| Eingang/Ausgang/Merker | WORD                  | EW1; AW10; MW100<br>EW2:P   |

| Erlaubter Operand      | Beispiel für Datentyp | Beispiel (Mnemonik deutsch) |
|------------------------|-----------------------|-----------------------------|
| Eingang/Ausgang/Merker | DWORD                 | ED4; AD10; MD100<br>ED2:P   |
| Datenbaustein          | BOOL                  | DB1.DBX1.0                  |
| Datenbaustein          | BYTE                  | DB1.DBB1                    |
| Datenbaustein          | WORD                  | DB1.DBW1                    |
| Datenbaustein          | DWORD                 | DB1.DBD1                    |

**Hinweis**

Die Eingabe von "DB0..." ist aufgrund interner Benutzung nicht erlaubt!

**Erlaubte Forcewerte für die Forcetabelle**

**Eingabe von Forcewerten in die Forcetabelle**

Die folgende Tabelle zeigt die Operanden, die für die Eingabe von Forcewerten in die Forcetabelle zugelassen sind:

Tabelle 9-20 Bitoperanden

| Mögliche Bitoperanden | Beispiel für erlaubte Forcewerte |
|-----------------------|----------------------------------|
| E1.0:P                | true                             |
| E1.1:P                | false                            |
| A1.0:P                | 0                                |
| A1.1:P                | 1                                |
| E2.0:P                | 2#0                              |
| E2.1:P                | 2#1                              |

Tabelle 9-21 Byteoperanden

| Mögliche Byteoperanden | Beispiel für erlaubte Forcewerte |
|------------------------|----------------------------------|
| EB1:P                  | 2#00110011                       |
| EB2:P                  | B#16#1F                          |
| AB14:P                 | 1F                               |
| AB10:P                 | 'a'                              |
| EB3:P                  | 10                               |

Tabelle 9-22 Wortoperanden

| Mögliche Wortoperanden | Beispiel für erlaubte Forcewerte |
|------------------------|----------------------------------|
| EW0:P                  | 2#0011001100110011               |
| EW2:P                  | W#16#ABCD                        |
| AW10:P                 | ABCD                             |
| AW12:P                 | B#(12, 34)                       |
| EW4:P                  | 'ab'                             |
| EW6:P                  | 12345                            |
| EW8:P                  | S5T#9S_340ms                     |
| EW10:P                 | C#123                            |
| EW12:P                 | D#2006-12-31                     |

Tabelle 9-23 Doppelwortoperanden

| Mögliche Doppelwortoperanden | Beispiel für erlaubte Forcewerte   |
|------------------------------|------------------------------------|
| ED0:P                        | 2#00110011001100110011001100110011 |
| ED4:P                        | 1.2                                |
| AD10:P                       | 1.234.e4                           |
| AD14:P                       | Dw#16#abcdef10                     |
| ED8:P                        | 16#ABCDEF10                        |
| ED12:P                       | b#(12,34,56,78)                    |
| ED16:P                       | L#-12                              |
| ED20:P                       | L#12                               |
| ED24:P                       | 123456789                          |
| ED28:P                       | 123456789                          |
| ED32:P                       | T#12s345ms                         |
| ED36:P                       | Tod#14:20:40.645                   |
| ED40:P                       | P#e0.0                             |

## Übersicht über die Anzeigeformate

### Anzeigeformate in der Forcetabelle

Mit der Auswahl eines Anzeigeformats legen Sie fest, wie der Wert einer Variablen dargestellt wird.

Beim Eintragen der Adresse wird automatisch ein Anzeigeformat vorbelegt. Um dieses zu ändern, können Sie das gewünschte Anzeigeformat aus der Klappliste in der Spalte "Anzeigeformate" auswählen. In der Klappliste werden nur die Anzeigeformate angeboten, die für diesen Datentyp zulässig sind. Das zuerst genannte Anzeigeformat ist das voreingestellte Format.

**Beispiel**

Die folgende Tabelle zeigt die für alle CPU-Familien in der Forcetabelle zulässigen 32-Bit-Datentypen und ihre möglichen Anzeigeformate:

| Datentyp      | Mögliche Anzeigeformate   |
|---------------|---|
| BOOL          | Bool, Hex, BCD, Oktal, Bin, Dez, Dez+/-   |
| BYTE          | Hex, BCD, Oktal, Bin, Dez, Dez+/-, Zeichen  |
| WORD          | Hex, BCD, Oktal, Bin, Dez, Dez+/-, Dez_Sequenz, Zeichen, SIMATIC_Zeit, Datum, Unicode_Zeichen, Zähler                   |
| DWORD         | Hex, BCD, Oktal, Bin, Dez, Dez+/-, Dez_Sequenz, Zeichen, Gleitpunktzahl, Tageszeit, Zeit, Zeiger, Unicode_Zeichen       |
| SINT          | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen  |
| INT           | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, SIMATIC_Zeit, Zähler, Datum                   |
| DINT          | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Zeiger       |
| USINT         | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen  |
| UINT          | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, SIMATIC_Zeit, Zähler, Datum                   |
| UDINT         | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Zeiger       |
| REAL          | Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez +/-, Dez_Sequenz, Tageszeit, Zeit, Zeiger      |
| DATE          | Datum, Dez, Hex, BCD, Bin   |
| TIME_OF_DAY   | Tageszeit, Dez, Hex, BCD, Bin   |
| TIME          | Zeit, Hex, BCD, Bin   |
| DATE_AND_TIME | Datum und Uhrzeit,  |
| TIMER         | SIMATIC_Zeit, Hex, BCD, Bin   |
| CHAR          | Zeichen, Hex, BCD, Oktal, Bin, Dez, Dez+/-,   |
| STRING        | Zeichenkette  |
| POINTER       | Zeiger, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Zahl |
| COUNTER       | Zähler, Hex, BCD, Bin   |
| S5TIME        | SIMATIC_Zeit, Hex, BCD, Bin   |

Für die CPU-Familie S7-1200 sind alle 32-Bit-Datentypen zulässig, siehe obige Tabelle, sowie zusätzlich der 64-Bit-Datentyp LREAL mit folgenden möglichen Anzeigeformaten:



| Datentyp | Mögliche Anzeigeformate  |
|----------|--|
| LREAL    | in einem mit dem TIA-Portal V12 erstellten Projekt:<br>Gleitpunktzahl<br>Hinweis: Die Anzeige von LREAL ist auf 13 Stellen zuzüglich Exponent begrenzt.  |
| LREAL    | in einem mit dem TIA-Portal V12 erstellten Projekt:<br>Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez +/-, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit<br>Hinweis: Die Anzeige von LREAL ist auf 13 Stellen zuzüglich Exponent begrenzt. |

Für die CPU-Familie S7-1500 sind zusätzlich zu 32-Bit-Datentypen auch die in der Tabelle aufgelisteten 64-Bit-Datentypen zulässig mit folgenden möglichen Anzeigeformaten:

| Datentyp | Mögliche Anzeigeformate   |
|----------|---|
| LWORD    | Hex, Oktal, BCD, Bin, Zeichen, Unicode_Zeichen, Dez, Dez+/-, Dez_Sequenz, Gleitpunktzahl, Tageszeit, Zeit, Datum und Uhrzeit  |
| LINT     | Dez+/-, Dez, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit                  |
| ULINT    | Dez, Dez+/-, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit                  |
| LREAL    | Gleitpunktzahl, Hex, BCD, Oktal, Bin, Zeichen, Unicode_Zeichen, Dez, Dez +/-, Dez_Sequenz, Tageszeit, Zeit, Datum und Uhrzeit |
| LTIME    | Zeit, Dez+/-, Dez, Hex  |
| LTOD     | Dez, Hex, BCD, Bin, Tageszeit   |
| LDT      | Dez, Hex, Datum und Uhrzeit   |

Weitere Informationen hierzu finden Sie bei der Beschreibung der gültigen Datentypen .

### Hinweis

#### Rundungen bei Gleitpunktzahlen

In der Forcetabelle werden Gleitpunktzahlen binär im IEEE-Format gespeichert. Da sich nicht jede an der Oberfläche darstellbare Gleitpunkt-Zahl (real, longreal) eindeutig auf das IEEE-Format abbilden lässt, kann es zu Rundungen bei Gleitpunktzahlen kommen. Wird eine solche gerundete Gleitpunktzahl in der Forcetabelle kopiert und in ein anderes Eingabefeld wieder eingefügt, erhält man durch die Rundung unter Umständen eine geringe Abweichung.

---

**Hinweis**

**Nur symbolische Adressierung möglich**

In der Forcetabelle können LongDataTypes wie z. B. LWORD oder LREAL nur symbolisch adressiert werden.

---

## Anzeigeformat für Variablen auswählen

### Vorgehen

Um das Anzeigeformat der Variablen auszuwählen, gehen Sie folgendermaßen vor:

1. Tragen Sie die gewünschte Adresse in die Forcetabelle ein.
2. Klicken Sie in der Spalte "Anzeigeformat" auf die gewünschte Zelle und öffnen Sie die Klappliste.  
Die zulässigen Anzeigeformate werden in der Klappliste angezeigt.
3. Wählen Sie aus der Klappliste das gewünschte Anzeigeformat aus.

---

**Hinweis**

Falls das ausgewählte Anzeigeformat nicht übernommen werden kann, wird automatisch das zuletzt ausgewählte Anzeigeformat angezeigt.

---

### 9.4.4.8 Variablen in der Forcetabelle beobachten

#### Einführung zum Beobachten von Variablen in der Forcetabelle

##### Einführung

In der Forcetabelle können Sie die Variablen der projektierten Ein- und Ausgabebaugruppen in der CPU beobachten, abhängig vom jeweils ausgewählten Beobachtungsmodus (Seite 1573). Um Variablen zu beobachten zu können, muss eine Online-Verbindung zur CPU bestehen.

## Optionen zum Beobachten von Variablen

Folgende Optionen für das Beobachten von Variablen stehen zur Verfügung:

- **Alle beobachten**  
Dieser Befehl startet das Beobachten aller sichtbaren Variablen in der aktiven Forcetabelle, abhängig vom ausgewählten Beobachtungsmodus:
  - Im Basismodus ist der Beobachtungsmodus "permanent" voreingestellt.
  - Im Erweiterten Modus können Sie bestimmte Triggerpunkte für das Beobachten der Variablen festlegen.

---

### Hinweis

Wird der Beobachtungsmodus im Erweiterten Modus umgestellt und wieder in den Basismodus umgeschaltet, wird der zuvor eingestellte Beobachtungsmodus auch im Basismodus übernommen.

---

- **Sofort beobachten**  
Dieser Befehl startet das Beobachten der sichtbaren Variablen in der aktiven Forcetabelle sofort und einmalig.

## CPU-spezifische Einschränkungen beim Beobachten von Variablen

Es bestehen folgende CPU-spezifische Unterschiede:

- **CPU S7-300/400:**  
CPUs aus dieser Familie können lediglich die ersten 30 Zeichen eines Strings beobachten.
- **CPU S7-1200:**  
CPUs aus dieser Familie können einen String bis zur Gesamtgröße von 254 Zeichen beobachten.

## Festlegen des Beobachtungsmodus in der Forcetabelle

### Einführung

Mit der Auswahl des Beobachtungsmodus legen Sie den Triggerpunkt und die Dauer des Beobachtens von Variablen in der Forcetabelle fest.

### Möglicher Beobachtungsmodus (Dauer des Beobachtens)

Folgende Auswahlmöglichkeiten stehen zur Verfügung:

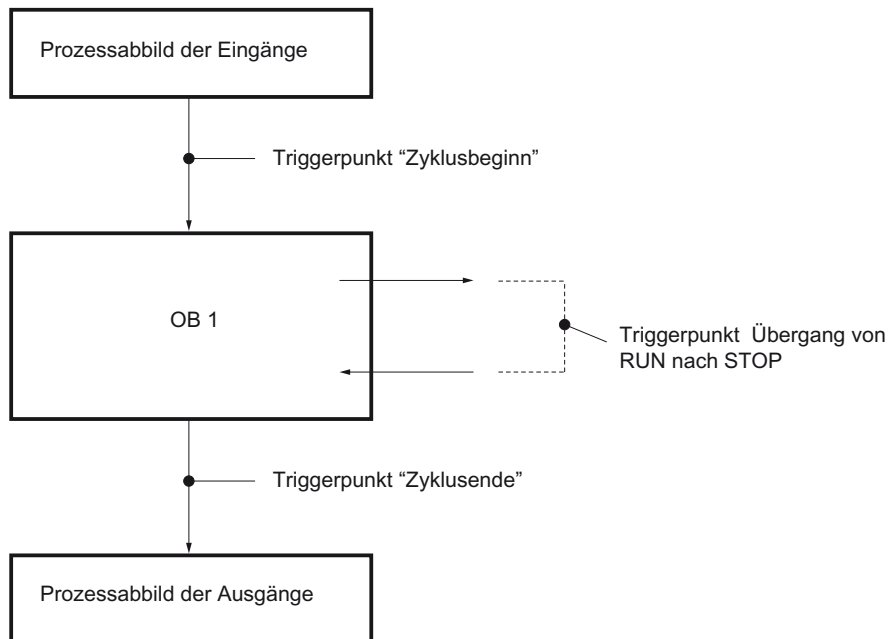
- **Permanent:** In diesem Modus können die Eingänge am Anfang und die Ausgänge am Ende des Zyklus beobachtet werden.
- **Zyklusbeginn einmalig**
- **Zyklusende einmalig**
- **Zyklusbeginn permanent**
- **Zyklusende permanent**

- Übergang von RUN nach STOP einmalig
- Übergang von RUN nach STOP permanent

### Triggerpunkt festlegen

Die Triggerpunkte "Zyklusbeginn", "Zyklusende" und "Übergang von RUN nach STOP" legen fest, zu welchem Zeitpunkt die Variablen aus der CPU gelesen bzw. in der CPU aktualisiert werden sollen.

Das folgende Bild zeigt die Lage der genannten Triggerpunkte:



### Variablen "Alle beobachten"

#### Einführung

Mit dem Befehl "Alle beobachten" können Sie das Beobachten der sichtbaren Variablen in der aktiven Forcetabelle starten. Im Basismodus der Forcetabelle ist für den Beobachtungsmodus "permanent" voreingestellt. Im Erweiterten Modus können Sie bestimmte Triggerpunkte für das Beobachten der Variablen festlegen. In diesem Fall werden die Variablen bezüglich der festgelegten Triggerpunkte beobachtet.

#### Voraussetzung

- Eine Forcetabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

## Vorgehensweise

Um den Befehl "Alle beobachten" auszuführen, gehen Sie folgendermaßen vor:

1. Geben Sie die zu beobachtenden Variablen und die entsprechenden Adressen in die Forcetabelle ein.
2. Schalten Sie in den Erweiterten Modus um, indem Sie in der Funktionsleiste auf das Symbol "Alle Spalten des Erweiterten Modus ein- oder ausblenden" klicken.
3. Wenn Sie den voreingestellten Beobachtungsmodus einer Variablen ändern wollen, klicken Sie in der Spalte "Beobachten mit Trigger" auf die entsprechende Zelle und wählen Sie aus der Klappliste den gewünschten Beobachtungsmodus aus.
4. Klicken Sie in der Funktionsleiste auf das Symbol "Alle beobachten".

## Ergebnis

Die Variablen der aktiven Forcetabelle werden mit dem jeweils eingestellten Beobachtungsmodus beobachtet.

## Variablen "Sofort beobachten"

### Einführung

Mit dem Befehl "Sofort beobachten" wird das Beobachten der Variablen sofort, ohne Bezug auf bestimmte Triggerpunkte gestartet. Die Variablenwerte werden einmalig ausgelesen und in der Forcetabelle angezeigt.

### Voraussetzung

- Eine Forcetabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.

## Vorgehensweise

Um den Befehl "Sofort beobachten" auszuführen, gehen Sie folgendermaßen vor:

1. Geben Sie die zu beobachtenden Variablen und die entsprechenden Adressen in die Forcetabelle ein.
2. Klicken Sie in der Funktionsleiste auf das Symbol "Sofort beobachten".

## Ergebnis

Die Variablen der aktiven Forcetabelle werden sofort und einmalig beobachtet.

### 9.4.4.9 Variablen in der Forcetabelle forcen

#### Einführung zum Forcen von Variablen

##### Einführung

Mit Hilfe der Forcetabelle können Sie einzelne Variablen des Anwenderprogramms mit festen Werten vorbelegen. Diesen Vorgang nennt man Forcen. Es können nur Peripherieeingänge und Peripherieausgänge geforct werden.

Voraussetzung für das Forcen ist, dass eine Online-Verbindung zur CPU besteht und dass die verwendete CPU diese Funktionalität unterstützt.


Wenn eine Forcetabelle im Ordner "Beobachtungs- und Forcetabellen" unterhalb einer CPU öffnen, werden automatisch alle in der zugeordneten CPU geforcten Werte in dieser Forcetabelle angezeigt, sofern eine Online-Verbindung zur CPU besteht.

##### Einsatzmöglichkeiten

Durch das feste Vorbelegen von Variablen mit definierten Werten, können Sie für Ihr Anwenderprogramm bestimmte Vorgaben setzen und damit die programmierten Funktionen testen. Das Forcen ist im Basismodus und im Erweiterten Modus (Seite 1563) möglich.

##### Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen (Seite 1578) bei diesem Vorgang informieren.

|   |
|---|
|  <b>GEFAHR</b>   |
| <b>Personen- und Sachschaden verhindern!</b>  |
| Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:  |
| <ul style="list-style-type: none"><li>• Das Leben oder die Gesundheit von Personen gefährden kann.</li><li>• Schäden an der Maschine oder der gesamten Anlage verursachen kann.</li></ul> |

##### Optionen zum Forcen von Variablen

Folgende Optionen für das Forcen von Variablen stehen zur Verfügung:

- Forcen auf "0"  
Dieser Befehl forct die selektierte Adresse in der CPU auf den Forcewert "0"
- Forcen auf "1"  
Dieser Befehl forct die selektierte Adresse in der CPU auf den Forcewert "1".

- **Alle forcen**  
Dieser Befehl startet das Forcen der aktivierten Adressen in der aktiven Forcetabelle oder ersetzt unterbrechungsfrei einen bestehenden Forceauftrag.
- **Forcen beenden**  
Dieser Befehl beendet das Forcen aller Adressen in der aktiven Forcetabelle.

### Einschränkungen beim Forcen von Variablen

Beachten Sie bitte die nachfolgend genannten Einschränkungen beim Forcen:

- Das Forcen ist immer abhängig vom Operandenumfang der eingesetzten CPU.
- Es können grundsätzlich nur Peripherieeingänge und Peripherieausgänge geforct werden.
- Wenn die Funktion "Peripherieausgänge freischalten" auf Ihrer CPU aktiv ist, ist ein Forcen nicht möglich! Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

### Besonderheiten beim Forcen von Variablen

Beachten Sie, dass das Forcen von Variablen Werte in der CPU überschreibt und auch nach Beenden der Online-Verbindung zur CPU weiterbesteht.

- **Forcen beenden**  
Das Beenden der Online-Verbindung alleine stoppt den Vorgang des Forcens nicht! Um das Forcen zu beenden, müssen Sie auf den Befehl "Online > Forcen > Forcen beenden" klicken. Erst dann werden die in der aktiven Forcetabelle sichtbaren Variablen nicht mehr geforct.
- **Beenden des Forcens einzelner Variablen**  
Der Befehl "Online > Forcen > Forcen beenden" gilt immer für alle in der Forcetabelle angezeigten Variablen. Um das Forcen für einzelne Variablen zu beenden, müssen Sie das Optionskästchen für das Forcen dieser Variablen in der Forcetabelle deaktivieren und das Forcen über den Befehl "Online > Forcen > Alle forcen" erneut starten.

## Sicherheitsvorkehrungen beim Forcen von Variablen

### Sicherheitsvorkehrungen zum Forcen von Variablen

Da Sie mit der Funktionalität des Forcens permanent in den Prozess eingreifen, sind die folgenden Hinweise unbedingt zu beachten:



#### GEFAHR

##### Personen- und Sachschaden verhindern!

Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:

- Das Leben oder die Gesundheit von Personen gefährden kann.
- Schäden an der Maschine oder der gesamten Anlage verursachen kann.



#### VORSICHT

##### Personen- und Sachschäden verhindern!

- Bevor Sie die Funktion "Forcen" starten, sollten Sie sicherstellen, dass niemand gleichzeitig auf derselben CPU diese Funktion ebenfalls ausführt.
- Das Forcen kann nur durch Klicken auf das Symbol "Forcen beenden" oder über den Befehl "Online > Forcen > Forcen beenden" beendet werden. Das Schließen der aktiven Forcetablelle beendet das Forcen **nicht!**
- "Forcen" kann **nicht** rückgängig gemacht werden!
- Informieren Sie sich über die Unterschiede zwischen "Steuern von Variablen" (Seite 1552) und "Forcen von Variablen" (Seite 1576)".
- Falls eine CPU die Funktionalität des "Forcens" nicht unterstützt, so sind alle das Forcen betreffenden Symbole nicht anwählbar.
- Wenn die Funktion "Peripherieausgänge freischalten" auf einer CPU aktiv ist, ist das Forcen auf dieser CPU **nicht** möglich. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

## Variablen forcen auf "0"

### Einführung

Beim Forcen können Sie einzelne Variablen eines Anwenderprogramms mit festen Werten vorbelegen.



## Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 1578) informieren.



### GEFAHR

#### Personen- und Sachschaden verhindern!

Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:

- Das Leben oder die Gesundheit von Personen gefährden kann.
- Schäden an der Maschine oder der gesamten Anlage verursachen kann.

## Voraussetzung

- Eine Forcetabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist **nicht** aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

## Vorgehen

Um Variablen auf "0" zu forcen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Tragen Sie die gewünschte Adresse in die Forcetabelle ein.
3. Wählen Sie den Befehl "Online > Forcen > Forcen auf 0", um die ausgewählte Adresse mit dem vorgegebenen Wert zu forcen.
4. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

## Ergebnis

Die ausgewählte Adresse wird auf "0" geforct. Das gelbe Dreieck wird nicht mehr angezeigt. In der ersten Spalte wird z. B. ein rotes "F" eingeblendet, das anzeigt, dass das Forcen ausgeführt wird.

## Forcen beenden

Um das Forcen zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Klicken Sie auf den Befehl "Online > Forcen > Forcen beenden".
3. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

## Ergebnis

Das Forcen der ausgewählten Werte wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen ausgewählt ist, momentan aber nicht geforct wird.

---

### Hinweis

#### Beachten Sie beim Forcen:

- Forcen kann **nicht** rückgängig gemacht werden!
  - Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
  - Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.
- 


## Variablen forcen auf "1"

### Einführung

Beim Forcen können Sie einzelne Variablen eines Anwenderprogramms mit festen Werten vorbelegen.

### Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 1578) informieren.

|  |
|--|
|  <b>GEFAHR</b>  |
| <b>Personen- und Sachschaden verhindern!</b><br>Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung: <ul style="list-style-type: none"><li>• Das Leben oder die Gesundheit von Personen gefährden sowie</li><li>• Schäden an der Maschine oder der gesamten Anlage verursachen kann.</li></ul> |

### Voraussetzung

- Eine Forcetabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist **nicht** aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

## Vorgehen

Um Variablen auf "1" zu forcen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Tragen Sie die gewünschte Adresse in die Forcetabelle ein.
3. Wählen Sie den Befehl "Online > Forcen > Forcen auf 1", um die ausgewählte Adresse mit dem vorgegebenen Wert zu forcen.
4. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

## Ergebnis

Die ausgewählte Adresse wird auf "1" geforct. Das gelbe Dreieck wird nicht mehr angezeigt. In der ersten Spalte wird z. B. ein rotes "F" eingeblendet, das anzeigt, dass das Forcen ausgeführt wird.

## Forcen beenden

Um das Forcen zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Klicken Sie auf den Befehl "Online > Forcen > Forcen beenden".
3. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

## Ergebnis

Das Forcen der ausgewählten Werte wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen ausgewählt ist, momentan aber nicht geforct wird.

---

### Hinweis

#### Beachten Sie beim Forcen:

- Forcen kann **nicht** rückgängig gemacht werden!
  - Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
  - Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.
- 

## Variablen forcen "Alle forcen"


### Einführung

Beim Forcen können Sie einzelne Variablen eines Anwenderprogramms mit festen Werten vorbelegen.

Falls bereits ein Forcen läuft, so wird dieses durch den Befehl "Online > Forcen > Alle forcen" unterbrechungsfrei ersetzt. Dabei werden die nicht aktivierten geforcten Adressen nicht mehr geforct.

### Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 1578) informieren.

|   |
|---|
|  <b>GEFAHR</b>   |
| <b>Personen- und Sachschaden verhindern!</b>  |
| Beachten Sie, dass beim Ausführen der Funktion "Forcen" eine falsche Handlung:  |
| <ul style="list-style-type: none"><li>• Das Leben oder die Gesundheit von Personen gefährden kann.</li><li>• Schäden an der Maschine oder der gesamten Anlage verursachen kann.</li></ul> |

### Voraussetzung

- Eine Forcetabelle ist angelegt.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist **nicht** aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

### Vorgehen

Um Variablen mit dem Befehl "Online > Forcen > Alle forcen" zu forcen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Tragen Sie die gewünschten Adressen und Forcewerte in die Forcetabelle ein.
3. Wählen Sie die zu forcenden Adressen aus, indem Sie die Optionskästchen für das Forcen in der Spalte nach dem "Forcewert" aktivieren.  
Ein gelbes Dreieck erscheint hinter dem aktivierten Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen ausgewählt ist, momentan aber nicht geforct wird.
4. Wählen Sie den Befehl "Online > Forcen > Alle forcen", um die ausgewählten Adressen mit den vorgegebenen Werten zu forcen.
5. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

## Ergebnis

Die ausgewählten Adressen werden auf die vorgegebenen Werte geforct. Das gelbe Dreieck wird nicht mehr angezeigt. In der ersten Spalte wird z. B. ein rotes "F" eingeblendet, das anzeigt, dass das Forcen ausgeführt wird.

## Forcen beenden

Um das Forcen zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Klicken Sie auf den Befehl "Online > Forcen > Forcen beenden".
3. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

## Ergebnis

Das Forcen der ausgewählten Adressen wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen ausgewählt ist, momentan aber nicht geforct wird.

---

### Hinweis

#### Beachten Sie beim Forcen:

- Forcen kann **nicht** rückgängig gemacht werden!
  - Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
  - Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.
- 

## 9.4.4.10 Forcen von Variablen beenden

### Forcen aller Variablen beenden


#### Einführung

Beachten Sie die nachfolgenden Hinweise, bevor Sie das Forcen von Variablen beenden:

- Das Beenden des Forcens kann **nicht** rückgängig gemacht werden!
- Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
- Um das Forcen zu beenden, muss die geforcte Adresse in der aktiven Forcetabelle sichtbar sein.

### Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 1578) informieren.

|   |
|---|
|  <b>GEFAHR</b>   |
| <b>Personen- und Sachschaden verhindern!</b>  |
| Beachten Sie, dass beim Beenden der Funktion "Forcen" eine falsche Handlung:  |
| <ul style="list-style-type: none"><li>• Das Leben oder die Gesundheit von Personen gefährden kann.</li><li>• Schäden an der Maschine oder der gesamten Anlage verursachen kann.</li></ul> |

### Voraussetzung

- Eine Forcetabelle ist angelegt, in der Variablen geforct werden.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist nicht aktiviert auf der CPU, auf der geforct werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

### Vorgehen

Um das **Forcen aller Variablen** zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Wählen Sie den Befehl "Online > Forcen > Forcen beenden", um das Forcen der angezeigten Adressen zu beenden.
3. Bestätigen Sie den Dialog "Forcen beenden" mit "Ja".

### Ergebnis

Das Forcen aller Variablen wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen vorgemerkt ist, momentan aber nicht geforct wird.

## Forcen einzelner Variablen beenden


### Einführung

Beachten Sie die nachfolgenden Hinweise, bevor Sie das Forcen von Variablen beenden:

- Das Beenden des Forcens kann **nicht** rückgängig gemacht werden!
- Das Beenden der Online-Verbindung beendet **nicht** das Forcen!
- Um das Forcen zu beenden, muss die geforcete Adresse in der aktiven Forcetabelle sichtbar sein.

### Vorsicht beim Forcen von Variablen

Vor dem Forcen müssen Sie sich unbedingt über die Sicherheitsvorkehrungen beim Forcen von Variablen (Seite 1562) informieren.

|   |
|---|
|  <b>GEFAHR</b>   |
| <b>Personen- und Sachschaden verhindern!</b>  |
| Beachten Sie, dass beim Beenden der Funktion "Forcen" eine falsche Handlung:  |
| <ul style="list-style-type: none"><li>• Das Leben oder die Gesundheit von Personen gefährden kann.</li><li>• Schäden an der Maschine oder der gesamten Anlage verursachen kann.</li></ul> |

### Voraussetzung

- Eine Forcetabelle ist angelegt, in der Variablen geforcet werden.
- Eine Online-Verbindung zur CPU ist vorhanden.
- Die verwendete CPU ermöglicht das Forcen.
- Die Funktion "Peripherieausgänge freischalten" ist nicht aktiviert auf der CPU, auf der geforcet werden soll. Deaktivieren Sie diese Funktion, falls gewünscht, in der Beobachtungstabelle.

### Vorgehen

Um das **Forcen einzelner Variablen** zu beenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Forcetabelle.
2. Deaktivieren Sie das Optionskästchen für die Adressen, die nicht mehr geforcet werden sollen.
3. Wählen Sie erneut den Befehl "Online > Forcen".

## Ergebnis

Das Forcen der deaktivierten Adressen wird beendet. Das rote "F" in der ersten Spalte wird nicht mehr angezeigt, das gelbe Dreieck erscheint wieder hinter dem Optionskästchen und zeigt an, dass die Adresse zwar für das Forcen vorgemerkt ist, momentan aber nicht geforct wird.

## 9.5 Übergreifende Projektfunktionen nutzen

### 9.5.1 Importieren und Exportieren

#### 9.5.1.1 Grundlagen zum Importieren und Exportieren

## Einführung

Sie können PLC-Variablen tabellen in ein standardisiertes XLSX-Format exportieren und mit externen Tabelleneditoren weiterbearbeiten. Ebenso können Sie PLC-Variablen tabellen, die mit externen Tabelleneditoren erstellt wurden, in das TIA-Portal importieren.

## Überschreiben bestehender PLC-Variablen und Konstanten beim Import

Beim Import werden bereits bestehende Einträge überschrieben, wenn sie denselben Namen haben wie zu importierende Einträge.

## Verbindung zu bestehenden Objekten

Wenn im Projekt bereits Verweise auf PLC-Variablen oder Konstanten vorhanden sind, werden diese Verweise beim Import automatisch aktualisiert. Die Aktualisierung wird auf Basis des Namens der PLC-Variablen und Konstanten durchgeführt.

## Siehe auch

Format der Exportdatei (Seite 1587)

Exportieren von PLC-Variablen (Seite 1587)

Importieren von PLC-Variablen (Seite 1588)



### 9.5.1.2 Format der Exportdatei

#### Einführung

Beim Export von PLC-Variablen Tabellen wird ein standardisiertes XSLX-Format erzeugt, das Sie mit externen Tabelleneditoren bearbeiten können.

Dieses Format wird auch beim Import von Tabellen erwartet.

#### Format der Exportdatei

Der Name des Blatts lautet immer "PLC Tags". In diesem Blatt können die dargestellten Spalten enthalten sein. Die Reihenfolge der Spalten kann variieren. Es müssen nicht alle Spalten vorhanden sein. Fehlende Werte werden beim Import durch die Angabe <no value> gekennzeichnet.

Die Benennungen der Spaltentitel sind ebenfalls fest definiert und werden immer in Englisch erwartet.

Die folgende Tabelle gibt an, welche Inhalte in den einzelnen Spalten erwartet werden:

| Element         | Erläuterung  |
|-----------------|--|
| Name            | Name der Variablen   |
| Path            | Gruppe und Name der PLC-Variablen Tabelle  |
| Data Type       | Die Notation des Datentyps entspricht der in der PLC-Variablen Tabelle verwendeten Notation. |
| Logical Address | Die Adresse kann in deutscher oder internationaler Mnemonik angegeben sein.                  |
| Comment         | Frei wählbarer Kommentar   |
| Hmi Visible     | Als Wert wird TRUE oder FALSE erwartet.  |
| Hmi Accessible  | Als Wert wird TRUE oder FALSE erwartet.  |

#### Siehe auch

Grundlagen zum Importieren und Exportieren (Seite 1586)

Exportieren von PLC-Variablen (Seite 1587)

Importieren von PLC-Variablen (Seite 1588)

### 9.5.1.3 Exportieren von PLC-Variablen

#### Voraussetzung

Eine PLC-Variablen Tabelle ist geöffnet.

## Vorgehen

Um PLC-Variablen und Konstanten zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der PLC-Variablen-Tabelle auf die Schaltfläche "Exportieren".  
Der Dialog "Exportieren nach Excel" wird geöffnet.
2. Wählen Sie den Pfad aus, in dem die Exportdatei abgelegt werden soll.
3. Wählen Sie aus, ob Sie Variablen und/oder Konstanten exportieren wollen.
4. Klicken Sie auf die Schaltfläche "OK".

## Ergebnis

Die Exportdatei wird erzeugt. Fehler und Warnungen, die während des Exports aufgetreten sind, werden im Inspektorfenster im Register "Info" angezeigt.

## Siehe auch

Grundlagen zum Importieren und Exportieren (Seite 1586)

Format der Exportdatei (Seite 1587)

Importieren von PLC-Variablen (Seite 1588)

### 9.5.1.4 Importieren von PLC-Variablen

## Voraussetzung

Eine Tabelle existiert und entspricht den Formatvorgaben.

## Vorgehen

Um eine PLC-Variablen-Tabelle zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Tabelle "Alle Variablen".
2. Klicken Sie auf die Schaltfläche "Importieren".  
Der Dialog "Importieren aus Excel" wird geöffnet.
3. Wählen Sie aus, ob Sie PLC-Variablen und/oder Konstanten importieren wollen.
4. Wählen Sie die Tabelle aus, die Sie importieren möchten.
5. Klicken Sie auf die Schaltfläche "OK".

## Ergebnis

Die PLC-Variablen-Tabelle wird importiert.

Fehler und Warnungen, die während des Exports aufgetreten sind, werden im Inspektorfenster im Register "Info" angezeigt.

## Siehe auch

Grundlagen zum Importieren und Exportieren (Seite 1586)

Format der Exportdatei (Seite 1587)

Exportieren von PLC-Variablen (Seite 1587)

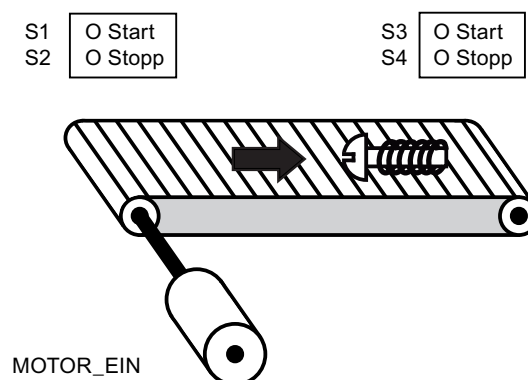
## 9.6 Programmierbeispiele

### 9.6.1 Programmierbeispiele KOP

#### 9.6.1.1 Beispiel für das Steuern eines Förderbands

##### Steuern eines Förderbands

Das folgende Bild zeigt ein Förderband, das elektrisch in Gang gesetzt werden kann. Am Anfang des Förderbands befinden sich zwei Druckschalter, S1 für START und S2 für STOPP. Am Ende des Förderbands befinden sich ebenfalls zwei Druckschalter, S3 für START und S4 für STOPP. Das Förderband kann von beiden Seiten aus gestartet oder gestoppt werden.



##### Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

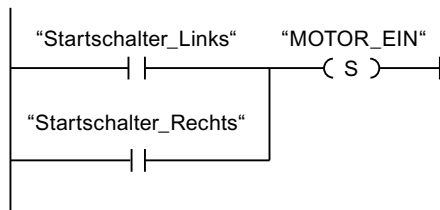
| Name                      | Datentyp | Beschreibung                                  |
|---------------------------|----------|---|
| Startschalter_Links (S1)  | BOOL     | Startschalter am linken Rand des Förderbands  |
| Stoppschalter_Links (S2)  | BOOL     | Stoppschalter am linken Rand des Förderbands  |
| Startschalter_Rechts (S3) | BOOL     | Startschalter am rechten Rand des Förderbands |

| Name                      | Datentyp | Beschreibung                                  |
|---------------------------|----------|---|
| Stoppschalter_Rechts (S4) | BOOL     | Stoppschalter am rechten Rand des Förderbands |
| MOTOR_EIN                 | BOOL     | Motor des Förderbands einschalten             |

Die folgenden Netzwerke zeigen die KOP-Programmierung zur Lösung der Aufgabe:

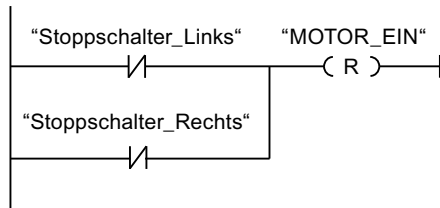
Netzwerk 1:

Der Motor des Förderbands wird eingeschaltet, wenn der Startschalter "S1" oder "S3" betätigt wird.



Netzwerk 2:

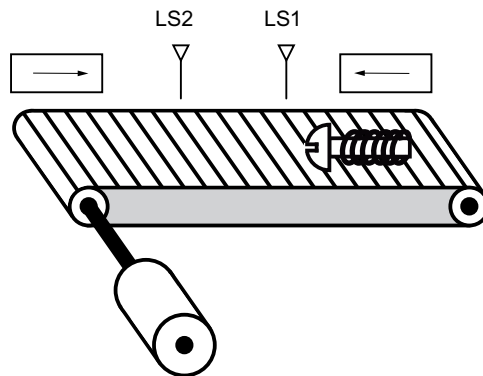
Der Motor des Förderbands wird ausgeschaltet, wenn der Stoppschalter "S2" oder "S4" betätigt wird.



### 9.6.1.2 Beispiel für das Erfassen der Richtung eines Förderbands

#### Erfassen der Richtung eines Förderbandes

Das folgende Bild zeigt ein Förderband, das mit zwei Lichtschranken (LS1, LS2) ausgestattet ist. Die Lichtschranken sollen feststellen, in welche Richtung sich ein Gegenstand auf dem Förderband bewegt.



#### Umsetzung

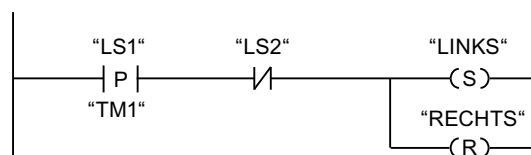
Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name   | Datentyp | Beschreibung                           |
|--------|----------|--|
| LS1    | BOOL     | Lichtschranke 1                        |
| LS2    | BOOL     | Lichtschranke 2                        |
| RECHTS | BOOL     | Anzeige bei einer Bewegung nach rechts |
| LINKS  | BOOL     | Anzeige bei einer Bewegung nach links  |
| TM1    | BOOL     | Flankenmerker 1                        |
| TM2    | BOOL     | Flankenmerker 2                        |

Die folgenden Netzwerke zeigen die KOP-Programmierung zur Lösung der Aufgabe:

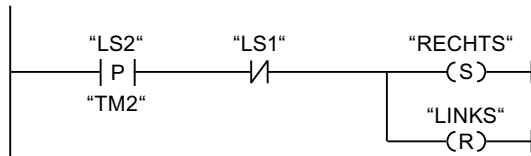
Netzwerk 1:

Wenn an der Lichtschranke "LS1" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Signalfanke) und gleichzeitig der Signalzustand an "LS2" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach links.



Netzwerk 2:

Wenn an der Lichtschranke "LS2" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Signalflanke) und gleichzeitig der Signalzustand an "LS1" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach rechts.

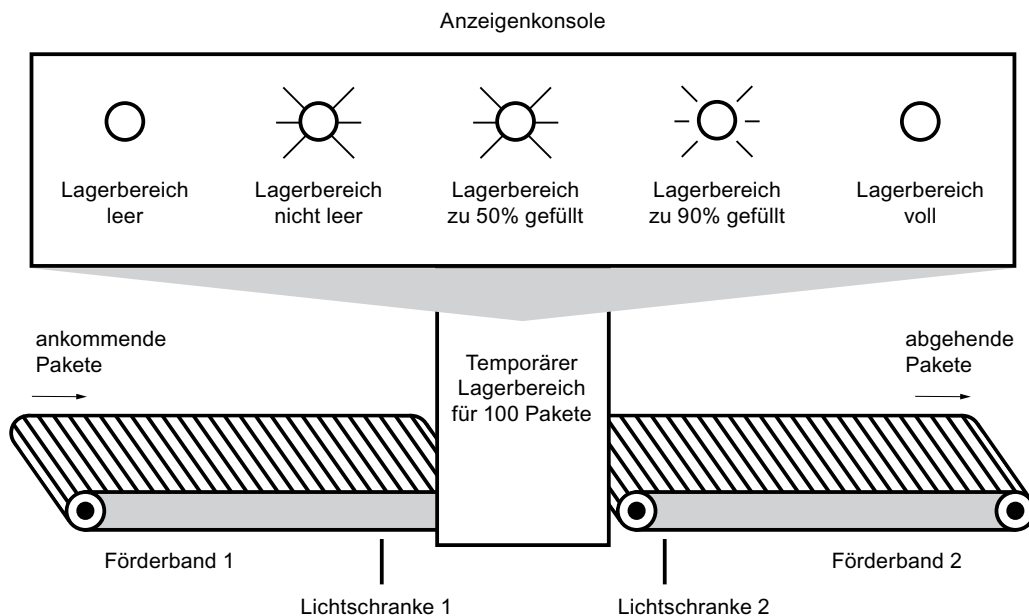


### 9.6.1.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs

#### Erfassen des Füllstands eines Lagerbereichs

Das folgende Bild zeigt ein System mit zwei Förderbändern und einem temporären Lagerbereich dazwischen. Förderband 1 transportiert die Pakete zum Lagerbereich. Eine Lichtschranke am Ende des Förderbandes 1 neben dem Lagerbereich ermittelt, wie viele Pakete in den Lagerbereich transportiert werden. Förderband 2 transportiert Pakete von diesem temporären Lagerbereich zu einer Laderampe, auf die sie zur Auslieferung beim Kunden auf LKW verladen werden. Eine Lichtschranke am Ausgang des Lagerbereichs ermittelt, wie viele Pakete aus dem Lagerbereich heraus zur Laderampe transportiert werden. Fünf Anzeigeleuchten zeigen an, wie weit der temporäre Lagerbereich gefüllt ist.

Bei einem neuen Start der Förderbänder wird der aktuelle Zählwert auf die Anzahl der im Lagerbereich vorhandenen Pakete gesetzt.



## Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name           | Datentyp | Beschreibung   |
|----------------|----------|--|
| LS1            | BOOL     | Lichtschanke 1   |
| LS2            | BOOL     | Lichtschanke 2   |
| RESET          | BOOL     | Zähler rücksetzen  |
| LOAD           | BOOL     | Zähler auf den Wert des Parameters "PV" setzen   |
| LAGERBESTAND   | INT      | Lagerbestand beim Neustart   |
| PAKETANZAHL    | INT      | Anzahl der Pakete im Lagerbereich (Aktueller Zählwert)   |
| LB_PAKETE      | BOOL     | Wird gesetzt, wenn der aktuelle Zählwert größer oder gleich dem Wert der Variablen "LAGERBESTAND" ist. |
| LAG_LEER       | BOOL     | Anzeigeleuchte: Lagerbereich leer  |
| LAG_NICHT_LEER | BOOL     | Anzeigeleuchte: Lagerbereich nicht leer  |
| LAG_50%_VOLL   | BOOL     | Anzeigeleuchte: Lagerbereich zu 50 % voll  |
| LAG_90%_VOLL   | BOOL     | Anzeigeleuchte: Lagerbereich zu 90 % voll  |
| LAG_VOLL       | BOOL     | Anzeigeleuchte: Lagerbereich voll  |
| VOLUME_50      | INT      | Vergleichswert: 50 Pakete  |
| VOLUME_90      | INT      | Vergleichswert: 90 Pakete  |
| VOLUME_100     | INT      | Vergleichswert: 100 Pakete   |

Die folgenden Netzwerke zeigen die KOP-Programmierung zur Aktivierung der Anzeigeleuchten:

Netzwerk 1:

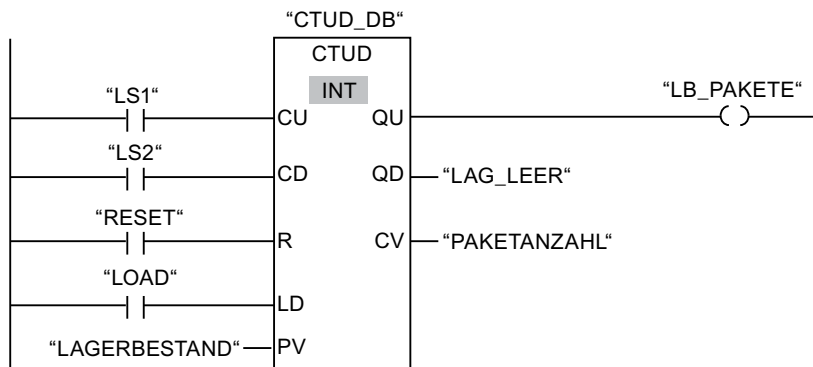
Wenn ein Paket zum Lagerbereich transportiert wird, wechselt der Signalzustand an "LS1" von "0" auf "1" (positive Signalfanke). Bei einer positiven Signalfanke an "LS1" wird der "Vorwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins erhöht.

Wenn ein Paket vom Lagerbereich zur Laderampe transportiert wird, wechselt der Signalzustand an "LS2" von "0" auf "1" (positive Signalfanke). Bei einer positiven Signalfanke an "LS2" wird der "Rückwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins verringert.

Wenn sich keine Pakete im Lagerbereich befinden ("PAKETANZAHL" = "0"), wird die Variable "LAG\_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich leer" eingeschaltet.

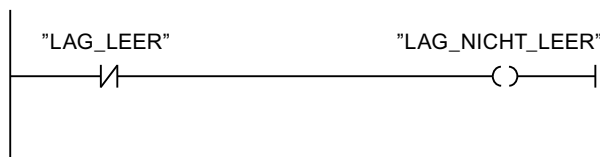
Der aktuelle Zählwert kann auf "0" zurückgesetzt werden, wenn die Variable "RESET" auf den Signalzustand "1" gesetzt wird.

Wenn die Variable "LOAD" auf den Signalzustand "1" gesetzt wird, wird der aktuelle Zählwert auf den Wert der Variablen "LAGERBESTAND" gesetzt. Solange der aktuelle Zählwert größer oder gleich dem Wert der Variablen "LAGERBESTAND" ist, liefert die Variable "LB\_PAKETE" den Signalzustand "1".



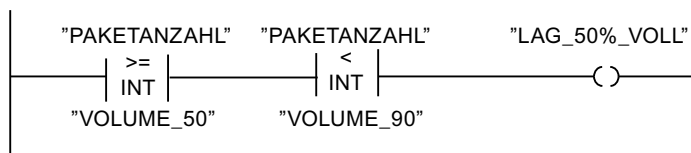
Netzwerk 2:

Solange sich Pakete im Lagerbereich befinden, wird die Variable "LAG\_NICHT\_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich nicht leer" eingeschaltet.



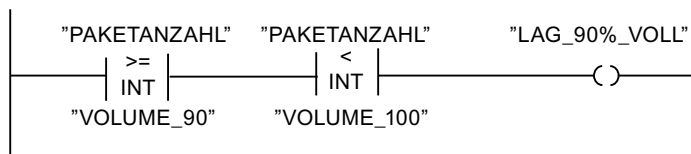
Netzwerk 3:

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 50 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 50 % gefüllt" ein.



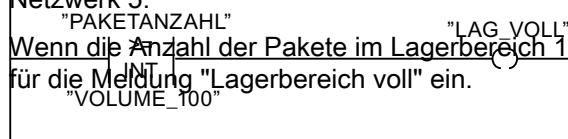
Netzwerk 4:

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 90 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 90 % gefüllt" ein.



Netzwerk 5:

Wenn die Anzahl der Pakete im Lagerbereich 100 erreicht, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich voll" ein.





### 9.6.1.4 Beispiel für das Kontrollieren der Raumtemperatur

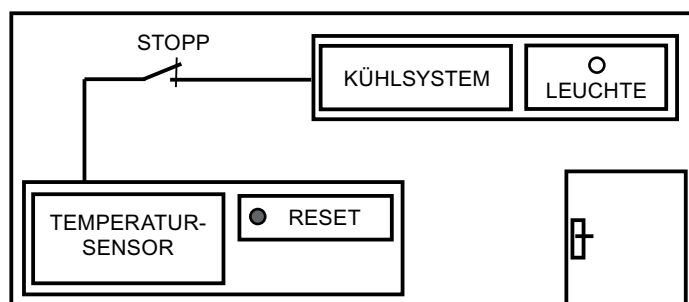
#### Kontrollieren der Raumtemperatur

In einem Kühlraum muss die Temperatur unter null Grad Celsius gehalten werden. Eventuelle Temperaturschwankungen werden mithilfe eines Sensors kontrolliert. Wenn die Temperatur über null Grad Celsius steigt, wird das Kühlsystem für eine voreingestellte Zeit eingeschaltet. Während des Kühlens leuchtet eine Anzeige "Kühlsystem gestartet".

Das Kühlsystem und die Anzeigeleuchte werden ausgeschaltet, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Sensor meldet eine Senkung der Temperatur unter null Grad Celsius.
- Die voreingestellte Kühlzeit ist abgelaufen.
- Der Druckschalter "STOPP" wird betätigt.

Wenn die voreingestellte Kühlzeit abgelaufen ist und die Temperatur im Kühlraum immer noch zu hoch ist, kann das Kühlsystem mithilfe des Druckschalters "RESET" erneut gestartet werden.



#### Umsetzung

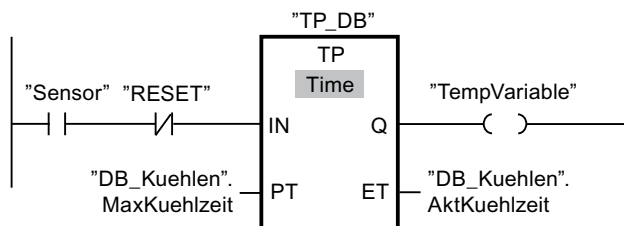
Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name         | Datentyp | Kommentar   |
|--------------|----------|---|
| Sensor       | BOOL     | Signal des Temperatursensors  |
| RESET        | BOOL     | Neustart  |
| STOPP        | BOOL     | Das Kühlsystem wird ausgeschaltet.  |
| MaxKuehlzeit | TIME     | Voreingestellte Kühlzeit<br>Diese Variable ist im Datenbaustein "DB_Kuehlen" definiert.     |
| AktKuehlzeit | TIME     | Aktuell abgelaufene Kühlzeit<br>Diese Variable ist im Datenbaustein "DB_Kuehlen" definiert. |
| Kuehlssystem | BOOL     | Das Kühlsystem wird eingeschaltet.  |

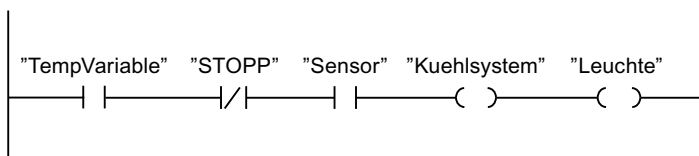
| Name         | Datentyp | Kommentar   |
|--------------|----------|---|
| Leuchte      | BOOL     | Die Anzeigeleuchte für die Meldung "Kühlsystem gestartet" wird eingeschaltet.     |
| TempVariable | BOOL     | Temporäre Variable<br>Diese Variable speichert den Signalzustand der IEC-Zeit TP. |

Das folgende Netzwerk zeigt die KOP-Programmierung zum Kontrollieren der Raumtemperatur:

Netzwerk 1:



Netzwerk 2:



Wenn die Temperatur im Kühlraum höher als null Grad Celsius wird, wechselt der Signalzustand am Operanden "Sensor" von "0" auf "1" (positive Signalfanke). Bei einer positiven Signalfanke am Eingang IN der Zeitfunktion wird die voreingestellte Kühlzeit gestartet und die "TempVariable" erhält den Signalzustand "1". Der Signalzustand "1" der "TempVariable" bewirkt im Netzwerk 2, dass das Kühlsystem sowie die Anzeigeleuchte eingeschaltet werden. Die Ausgänge "Sensor", "Kuehlsystem" und "Leuchte" müssen im Netzwerk 2 programmiert werden, da am Ausgang Q der Zeitfunktion nur eine Spule programmiert werden kann.

Sinkt die Temperatur im Kühlraum unter null Grad Celsius, wechselt der Signalzustand des Sensors wieder auf "0". Dadurch werden das Kühlsystem und die Anzeigeleuchte ausgeschaltet.

Wenn der Sensor keine Senkung der Temperatur meldet, werden das Kühlsystem und die Anzeigeleuchte spätestens nach dem Ablauf der voreingestellten Kühlzeit ausgeschaltet. Der Kühlprozess kann in diesem Fall mithilfe des Druckschalters "RESET" neu gestartet werden. Durch das Drücken und Loslassen des Druckschalters wird eine neue positive Signalfanke am Eingang IN erzeugt und dadurch das Kühlsystem neu gestartet.

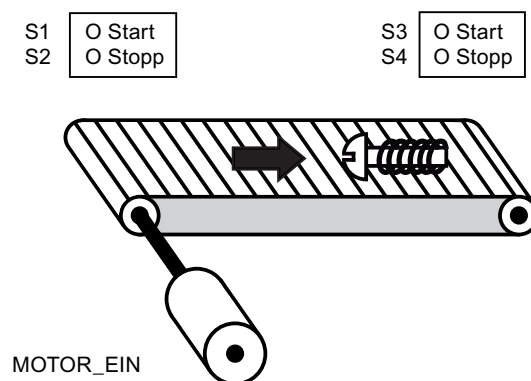
Das Kühlsystem und die Anzeigeleuchte können jederzeit mithilfe des Druckschalters "STOPP" ausgeschaltet werden.

## 9.6.2 Programmierbeispiele FUP

### 9.6.2.1 Beispiel für das Steuern eines Förderbands

#### Steuern eines Förderbands

Das folgende Bild zeigt ein Förderband, das elektrisch in Gang gesetzt werden kann. Am Anfang des Förderbands befinden sich zwei Druckschalter, S1 für START und S2 für STOPP. Am Ende des Förderbands befinden sich ebenfalls zwei Druckschalter, S3 für START und S4 für STOPP. Das Förderband kann von beiden Seiten aus gestartet oder gestoppt werden.



#### Umsetzung

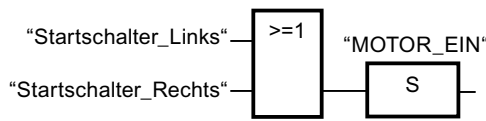
Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name                      | Datentyp | Beschreibung                                  |
|---------------------------|----------|---|
| Startschalter_Links (S1)  | BOOL     | Startschalter am linken Rand des Förderbands  |
| Stoppschalter_Links (S2)  | BOOL     | Stoppschalter am linken Rand des Förderbands  |
| Startschalter_Rechts (S3) | BOOL     | Startschalter am rechten Rand des Förderbands |
| Stoppschalter_Rechts (S4) | BOOL     | Stoppschalter am rechten Rand des Förderbands |
| MOTOR_EIN                 | BOOL     | Motor des Förderbands einschalten             |

Die folgenden Netzwerke zeigen die FUP-Programmierung zur Lösung der Aufgabe:

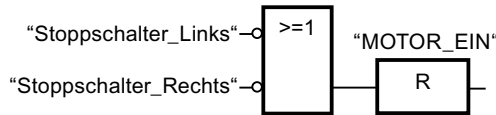
Netzwerk 1:

Der Motor des Förderbands wird eingeschaltet, wenn der Startschalter "S1" oder "S3" betätigt wird.



Netzwerk 2:

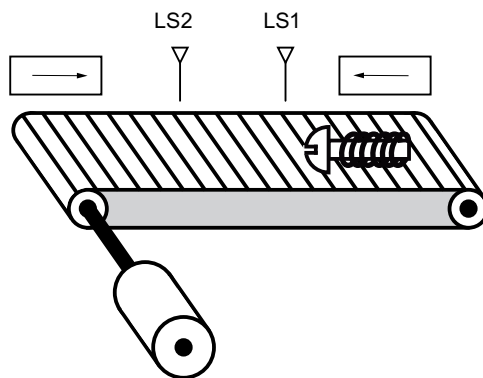
Der Motor des Förderbands wird ausgeschaltet, wenn der Stoppschalter "S2" oder "S4" betätigt wird.



### 9.6.2.2 Beispiel für das Erfassen der Richtung eines Förderbands

#### Erfassen der Richtung eines Förderbands

Das folgende Bild zeigt ein Förderband, das mit zwei Lichtschranken (LS1, LS2) ausgestattet ist. Die Lichtschranken sollen feststellen, in welche Richtung sich ein Gegenstand auf dem Förderband bewegt.



#### Umsetzung

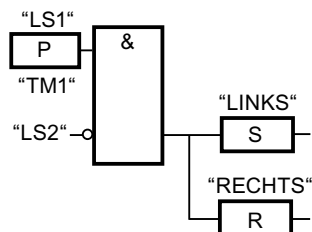
Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name   | Datentyp | Beschreibung                           |
|--------|----------|--|
| LS1    | BOOL     | Lichtschranke 1                        |
| LS2    | BOOL     | Lichtschranke 2                        |
| RECHTS | BOOL     | Anzeige bei einer Bewegung nach rechts |
| LINKS  | BOOL     | Anzeige bei einer Bewegung nach links  |
| TM1    | BOOL     | Flankenmerker 1                        |
| TM2    | BOOL     | Flankenmerker 2                        |

Die folgenden Netzwerke zeigen die FUP-Programmierung zur Lösung der Aufgabe:

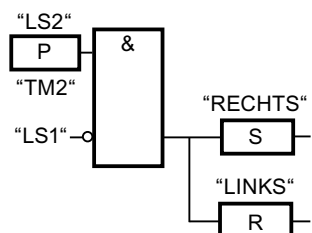
Netzwerk 1:

Wenn an der Lichtschranke "LS1" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Signalfanke) und gleichzeitig der Signalzustand an "LS2" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach links.



Netzwerk 2:

Wenn an der Lichtschranke "LS2" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Signalfanke) und gleichzeitig der Signalzustand an "LS1" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach rechts.

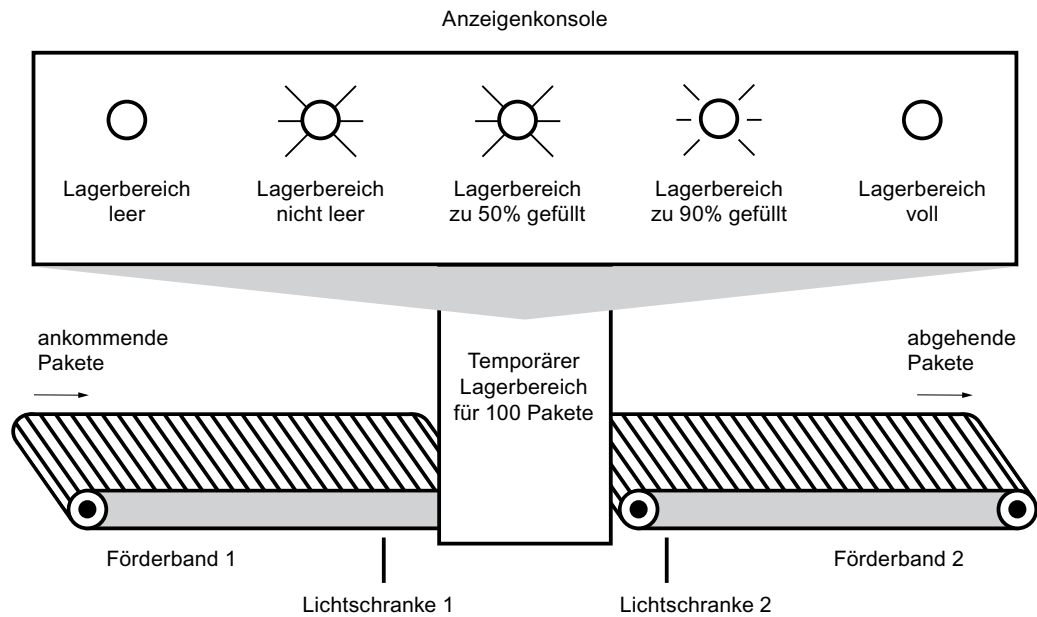


### 9.6.2.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs

#### Erfassen des Füllstands eines Lagerbereichs

Das folgende Bild zeigt ein System mit zwei Förderbändern und einem temporären Lagerbereich dazwischen. Förderband 1 transportiert die Pakete zum Lagerbereich. Eine Lichtschranke am Ende des Förderbands 1 neben dem Lagerbereich ermittelt, wie viele Pakete in den Lagerbereich transportiert werden. Förderband 2 transportiert Pakete von diesem temporären Lagerbereich zu einer Laderampe, auf die sie zur Auslieferung beim Kunden auf LKW verladen werden. Eine Lichtschranke am Ausgang des Lagerbereichs ermittelt, wie viele Pakete aus dem Lagerbereich heraus zur Laderampe transportiert werden. Fünf Anzeigeleuchten zeigen an, wie weit der temporäre Lagerbereich gefüllt ist.

Bei einem neuen Start der Förderbänder wird der aktuelle Zählwert auf die Anzahl der im Lagerbereich vorhandenen Pakete gesetzt.



### Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name           | Datentyp | Beschreibung   |
|----------------|----------|--|
| LS1            | BOOL     | Lichtschranke 1  |
| LS2            | BOOL     | Lichtschranke 2  |
| RESET          | BOOL     | Zähler rücksetzen  |
| LOAD           | BOOL     | Zähler auf den Wert des Parameters "ZW" setzen   |
| LAGERBESTAND   | INT      | Lagerbestand beim Neustart   |
| PAKETANZAHL    | INT      | Anzahl der Pakete im Lagerbereich (Aktueller Zählwert)   |
| LB_PAKETE      | BOOL     | Wird gesetzt, wenn der aktuelle Zählwert größer oder gleich dem Wert der Variablen "LAGERBESTAND" ist. |
| LAG_LEER       | BOOL     | Anzeigeleuchte: Lagerbereich leer  |
| LAG_NICHT_LEER | BOOL     | Anzeigeleuchte: Lagerbereich nicht leer  |
| LAG_50%_VOLL   | BOOL     | Anzeigeleuchte: Lagerbereich zu 50 % voll  |
| LAG_90%_VOLL   | BOOL     | Anzeigeleuchte: Lagerbereich zu 90 % voll  |
| LAG_VOLL       | BOOL     | Anzeigeleuchte: Lagerbereich voll  |
| VOLUME_50      | INT      | Vergleichswert: 50 Pakete  |

| Name       | Datentyp | Beschreibung               |
|------------|----------|----------------------------|
| VOLUME_90  | INT      | Vergleichswert: 90 Pakete  |
| VOLUME_100 | INT      | Vergleichswert: 100 Pakete |

Die folgenden Netzwerke zeigen die FUP-Programmierung zur Aktivierung der Anzeigeleuchten:

Netzwerk 1:

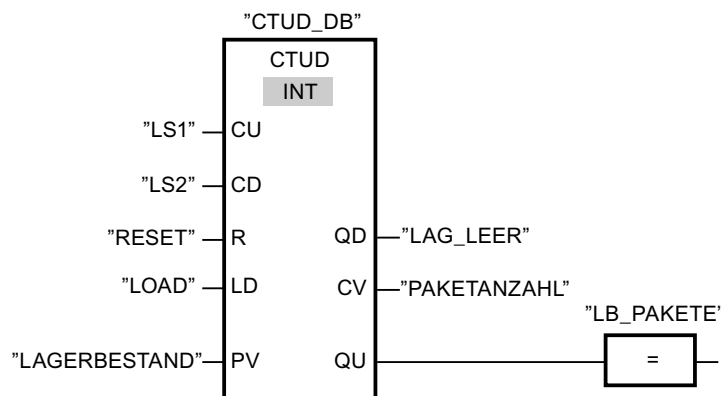
Wenn ein Paket zum Lagebereich transportiert wird, wechselt der Signalzustand an "LS1" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke an "LS1" wird der "Vorwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins erhöht.

Wenn ein Paket vom Lagerbereich zur Laderampe transportiert wird, wechselt der Signalzustand an "LS2" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke an "LS2" wird der "Rückwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins verringert.

Wenn sich keine Pakete im Lagerbereich befinden ("PAKETANZAHL" = "0"), wird die Variable "LAG\_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich leer" eingeschaltet.

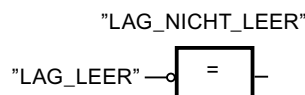
Der aktuelle Zählwert kann auf "0" zurückgesetzt werden, wenn die Variable "RESET" auf den Signalzustand "1" gesetzt wird.

Wenn die Variable "LOAD" auf den Signalzustand "1" gesetzt wird, wird der aktuelle Zählwert auf den Wert der Variablen "LAGERBESTAND" gesetzt. Solange der aktuelle Zählwert größer oder gleich dem Wert der Variablen "LAGERBESTAND" ist, liefert die Variable "LB\_PAKETE" den Signalzustand "1".



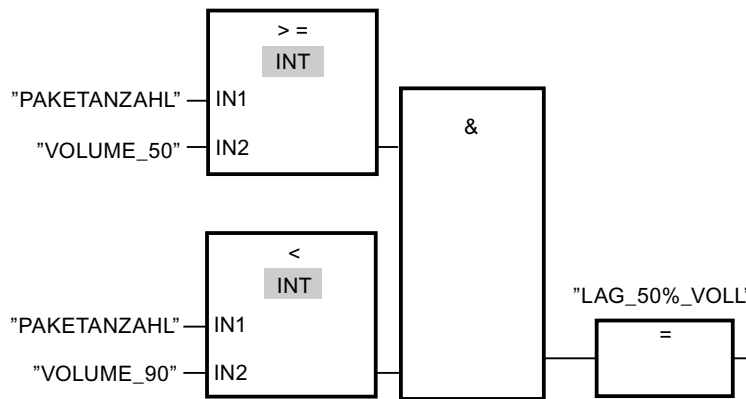
Netzwerk 2:

Solange sich Pakete im Lagerbereich befinden, wird die Variable "LAG\_NICHT\_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich nicht leer" eingeschaltet.



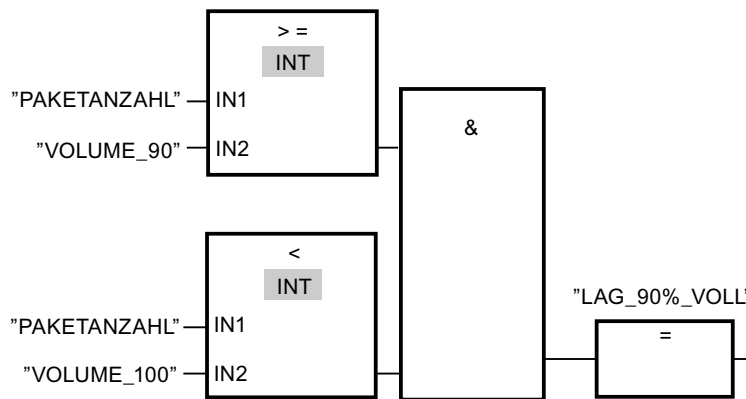
Netzwerk 3:

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 50 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 50 % gefüllt" ein.



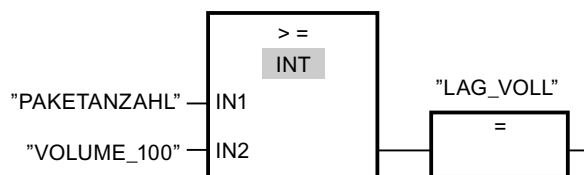
Netzwerk 4:

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 90 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 90 % gefüllt" ein.



Netzwerk 5:

Wenn die Anzahl der Pakete im Lagerbereich 100 erreicht, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich voll" ein.



### 9.6.2.4 Beispiel für das Kontrollieren der Raumtemperatur

#### Kontrollieren der Raumtemperatur

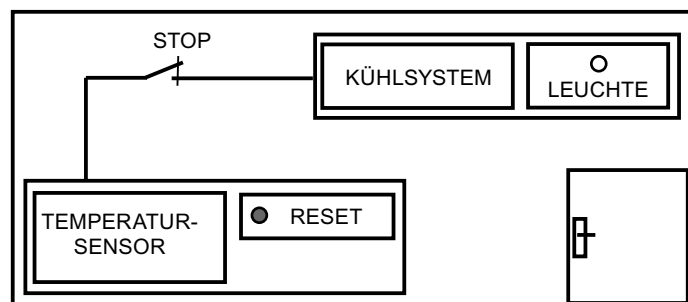
In einem Kühlraum muss die Temperatur unter null Grad Celsius gehalten werden. Eventuelle Temperaturschwankungen werden mithilfe eines Sensors kontrolliert. Wenn die Temperatur über null Grad Celsius steigt, wird das Kühlsystem für eine voreingestellte Zeit eingeschaltet. Während des Kühlens leuchtet eine Anzeige "Kühlsystem gestartet".



Das Kühlsystem und die Anzeigeleuchte werden ausgeschaltet, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Sensor meldet eine Senkung der Temperatur unter null Grad Celsius.
- Die voreingestellte Kühlzeit ist abgelaufen.
- Der Druckschalter "Stopp" wird betätigt.

Wenn die voreingestellte Kühlzeit abgelaufen ist und die Temperatur im Kühlraum immer noch zu hoch ist, kann das Kühlsystem mithilfe des Druckschalters "RESET" erneut gestartet werden.



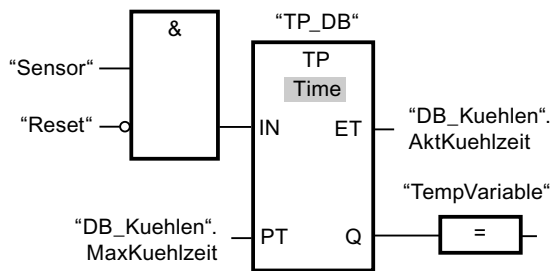
## Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

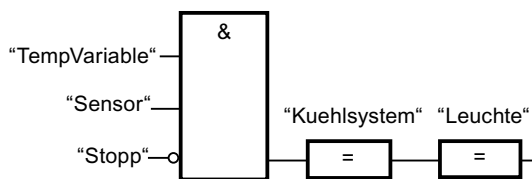
| Name         | Datentyp | Kommentar   |
|--------------|----------|---|
| Sensor       | BOOL     | Signal des Temperatursensors  |
| RESET        | BOOL     | Neustart  |
| STOPP        | BOOL     | Das Kühlsystem wird ausgeschaltet.  |
| MaxKuehlzeit | TIME     | Voreingestellte Kühlzeit<br>Diese Variable ist im Datenbaustein "DB_Kuehlen" definiert.     |
| AktKuehlzeit | TIME     | Aktuell abgelaufene Kühlzeit<br>Diese Variable ist im Datenbaustein "DB_Kuehlen" definiert. |
| Kuehlsystem  | BOOL     | Das Kühlsystem wird eingeschaltet.  |
| Leuchte      | BOOL     | Die Anzeigeleuchte für die Meldung "Kühlsystem gestartet" wird eingeschaltet.               |
| TempVariable | BOOL     | Temporäre Variable<br>Diese Variable speichert den Signalzustand der IEC-Zeit TP.           |

Das folgende Netzwerk zeigt die FUP-Programmierung zum Kontrollieren der Raumtemperatur:

Netzwerk 1:



Netzwerk 2:



Wenn die Temperatur im Kühlraum höher als null Grad Celsius wird, wechselt der Signalzustand am Operanden "Sensor" von "0" auf "1" (positive Signalflanke). Bei einer positiven Signalflanke am Eingang IN der Zeitfunktion wird die voreingestellte Kühlzeit gestartet und die "TempVariable" erhält den Signalzustand "1". Der Signalzustand "1" der "TempVariable" bewirkt im Netzwerk 2, dass das Kühlsystem sowie die Anzeigeleuchte eingeschaltet werden. Die Ausgänge "Sensor", "Kuehlsystem" und "Leuchte" müssen im Netzwerk 2 programmiert werden, da am Ausgang Q der Zeitfunktion nur eine Spule programmiert werden kann.

Sinkt die Temperatur im Kühlraum unter null Grad Celsius, wechselt der Signalzustand des Sensors wieder auf "0". Dadurch werden das Kühlsystem und die Anzeigeleuchte ausgeschaltet.

Wenn der Sensor keine Senkung der Temperatur meldet, werden das Kühlsystem und die Anzeigeleuchte spätestens nach dem Ablauf der voreingestellten Kühlzeit ausgeschaltet. Der Kühlprozess kann in diesem Fall mithilfe des Druckschalters "RESET" neu gestartet werden. Durch das Drücken und Loslassen des Druckschalters wird eine neue positive Signalflanke am Eingang IN erzeugt und dadurch das Kühlsystem neu gestartet.

Das Kühlsystem und die Anzeigeleuchte können jederzeit mithilfe des Druckschalters "STOPP" ausgeschaltet werden.

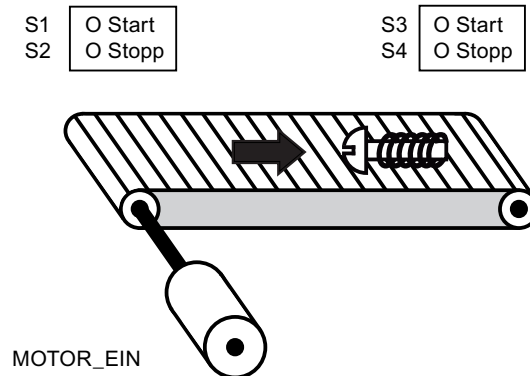
### 9.6.3 Programmierbeispiele AWL

#### 9.6.3.1 Beispiel für das Steuern eines Förderbands

##### Steuern eines Förderbands

Das folgende Bild zeigt ein Förderband, das elektrisch in Gang gesetzt werden kann. Am Anfang des Förderbands befinden sich zwei Druckschalter, S1 für START und S2 für STOPP.

Am Ende des Förderbands befinden sich ebenfalls zwei Druckschalter, S3 für START und S4 für STOPP. Das Förderband kann von beiden Seiten aus gestartet oder gestoppt werden.



## Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Operand                   | Deklaration | Datentyp | Beschreibung                                  |
|---------------------------|-------------|----------|---|
| Startschalter_Links (S1)  | Input       | BOOL     | Startschalter am linken Rand des Förderbands  |
| Stoppschalter_Links (S2)  | Input       | BOOL     | Stoppschalter am linken Rand des Förderbands  |
| Startschalter_Rechts (S3) | Input       | BOOL     | Startschalter am rechten Rand des Förderbands |
| Stoppschalter_Rechts (S4) | Input       | BOOL     | Stoppschalter am rechten Rand des Förderbands |
| MOTOR_ON/OFF              | Output      | BOOL     | Motor des Förderbands ein-/ausschalten        |

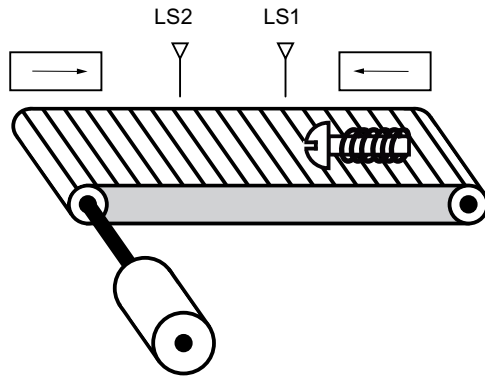
Das folgende AWL-Programm zeigt die Umsetzung der Aufgabe:

| AWL               | Erläuterung   |
|-------------------|---|
| O #S1             | // Startschalter S1 auf "1" abfragen.   |
| O #S3             | // Startschalter S3 auf "1" abfragen.   |
| S #"MOTOR_ON/OFF" | // Wenn einer der Startschalter (S1 oder S3) den Signalzustand "1" liefert, wird der Motor des Förderbands eingeschaltet. |
| O #S2             | // Stoppschalter S2 auf "1" abfragen.   |
| O #S4             | // Stoppschalter S4 auf "1" abfragen.   |
| R #"MOTOR_ON/OFF" | // Wenn einer der Stoppschalter (S2 oder S4) den Signalzustand "1" liefert, wird der Motor des Förderbands ausgeschaltet. |

### 9.6.3.2 Beispiel für das Erfassen der Richtung eines Förderbands

#### Erfassen der Richtung eines Förderbands

Das folgende Bild zeigt ein Förderband, das mit zwei Lichtschranken (LS1, LS2) ausgestattet ist. Die Lichtschranken sollen feststellen, in welche Richtung sich ein Gegenstand auf dem Förderband bewegt.



#### Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name   | Deklaration | Datentyp | Beschreibung                          |
|--------|-------------|----------|---------------------------------------|
| S1     | Input       | BOOL     | Lichtschranke 1                       |
| S2     | Input       | BOOL     | Lichtschranke 2                       |
| TM1    | Input       | BOOL     | Flankenmerker 1                       |
| TM2    | Input       | BOOL     | Flankenmerker 2                       |
| RECHTS | Output      | BOOL     | Anzeige für eine Bewegung nach rechts |
| LINKS  | Output      | BOOL     | Anzeige für eine Bewegung nach links  |

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

| AWL      | Erläuterung  |
|----------|--|
| U #S1    | // Lichtschranke "S1" auf "1" abfragen   |
| FP #TM1  | // Positive Flanke abfragen  |
| UN #S2   | // Lichtschranke "S2" auf "0" abfragen   |
| S #LINKS | // Wenn an der Lichtschranke "S1" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Flanke) und gleichzeitig der Signalzustand der Lichtschranke "S2" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach links.<br><br>// Die Anzeige für eine Bewegung nach links wird eingeschaltet. |
| U #S2    | // Lichtschranke "S2" auf "1" abfragen   |
| FP #TM2  | // Positive Flanke abfragen  |

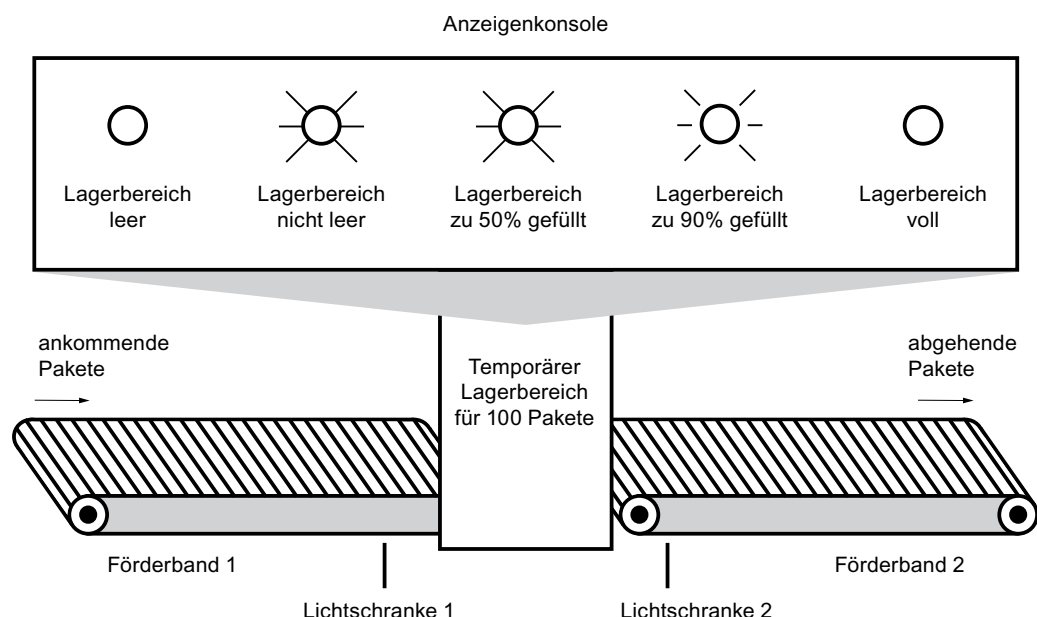
| AWL       | Erläuterung   |
|-----------|---|
| UN #S1    | // Lichtschranke "S1" auf "0" abfragen  |
| S #RECHTS | // Wenn an der Lichtschranke "S2" ein Wechsel des Signalzustands von "0" auf "1" auftritt (positive Flanke) und gleichzeitig der Signalzustand an "S1" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach rechts.<br><br>// Die Anzeige für eine Bewegung nach rechts wird eingeschaltet. |
| UN #S1    | // Lichtschranke "S1" auf "0" abfragen  |
| UN #S2    | // Lichtschranke "S2" auf "0" abfragen  |
| R #LINKS  | // Die Anzeige für eine Bewegung nach links wird ausgeschaltet, wenn der Signalzustand an den beiden Lichtschranken "0" ist.  |
| R #RECHTS | // Die Anzeige für eine Bewegung nach rechts wird ausgeschaltet, wenn der Signalzustand an den beiden Lichtschranken "0" ist.   |

### 9.6.3.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs

#### Erfassen des Füllstands eines Lagerbereichs

Das folgende Bild zeigt ein System mit zwei Förderbändern und einem temporären Lagerbereich dazwischen. Förderband 1 transportiert die Pakete zum Lagerbereich. Eine Lichtschranke am Ende des Förderbands 1 neben dem Lagerbereich ermittelt, wie viele Pakete in den Lagerbereich transportiert werden. Förderband 2 transportiert Pakete vom temporären Lagerbereich zu einer Laderampe, auf die sie zur Auslieferung beim Kunden auf LKW verladen werden. Eine Lichtschranke am Ausgang des Lagerbereichs ermittelt, wie viele Pakete aus dem Lagerbereich heraus zur Laderampe transportiert werden. Fünf Anzeigeleuchten zeigen an, wie weit der temporäre Lagerbereich gefüllt ist.

Bei einem neuen Start der Förderbänder wird der aktuelle Zählwert auf die Anzahl der im Lagerbereich vorhandenen Pakete gesetzt.



### Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name         | Datentyp | Adresse | Beschreibung   |
|--------------|----------|---------|--|
| PACKAGECOUNT | COUNTER  | Z1      | Anzahl der Pakete im Lagerbereich (Aktueller Zählwert) |

| Name           | Abschnitt | Datentyp | Beschreibung                             |
|----------------|-----------|----------|--|
| LS1            | Input     | BOOL     | Lichtschanke 1                           |
| LS2            | Input     | BOOL     | Lichtschanke 2                           |
| LAG_LEER       | Output    | BOOL     | Anzeigeleuchte: Lagerbereich leer        |
| LAG_NICHT_LEER | Output    | BOOL     | Anzeigeleuchte: Lagerbereich nicht leer  |
| LAG_50%_VOLL   | Output    | BOOL     | Anzeigeleuchte: Lagerbereich zu 50% voll |
| LAG_90%_VOLL   | Output    | BOOL     | Anzeigeleuchte: Lagerbereich zu 90% voll |
| LAG_VOLL       | Output    | BOOL     | Anzeigeleuchte: Lagerbereich voll        |

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

| AWL               | Erläuterung  |
|-------------------|--|
| U #LS1            | // Lichtschanke "LS1" auf "1" abfragen.  |
| ZV "PACKAGECOUNT" | // Bei einer positiven Flanke an der Lichtschanke "LS1" wird der Zählwert des Zählers "PACKAGECOUNT" um eins erhöht.     |
| U #LS2            | // Lichtschanke "LS2" auf "1" abfragen.  |
| ZR "PACKAGECOUNT" | // Bei einer positiven Flanke an der Lichtschanke "LS2" wird der Zählwert des Zählers "PACKAGECOUNT" um eins verringert. |
| UN "PACKAGECOUNT" | // Zählwert auf "0" abfragen.  |
| = #LAG_LEER       | // Bei einem Zählwert "0" wird die Anzeigeleuchte "Lagerbereich leer" eingeschaltet.                                     |
| U "PACKAGECOUNT"  | // Zählwert auf "1" abfragen.  |
| = #LAG_NICHT_LEER | // Bei einem Zählerstand größer "0" wird die Anzeigeleuchte "Lagerbereich nicht leer" eingeschaltet.                     |
| L 50              | // Vergleichswert "50" in Akkumulator 1 laden.   |
| L "PACKAGECOUNT"  | // Vergleichswert in Akkumulator 2 verschieben.  |
|                   | // Aktuellen Zählwert in Akkumulator 1 laden.  |
| <=I               | // Werte vergleichen   |
| = #"LAG_50%_VOLL" | // Bei einem Zählwert größer oder gleich "50" wird die Anzeigeleuchte "Lagerbereich zu 50% voll" eingeschaltet.          |
| L 90              | // Zählwert in Akkumulator 2 verschieben.  |
|                   | // Vergleichswert "90" in Akkumulator 1 laden.   |

| AWL                | Erläuterung   |
|--------------------|---|
| >=I                | // Werte vergleichen  |
| = # "LAG_90%_VOLL" | // Bei einem Zählwert größer oder gleich "90" wird die Anzeigeleuchte "Lagerbereich zu 90% voll" eingeschaltet. |
| L "PACKAGECOUNT"   | // Aktuellen Zählwert in Akkumulator 1 laden.   |
| L 100              | // Zählwert in Akkumulator 2 verschieben.   |
|                    | // Vergleichswert "100" in Akkumulator 1 laden.   |
| >=I                | // Werte vergleichen  |
| = #LAG_VOLL        | // Bei einem Zählwert größer oder gleich "100" wird die Anzeigeleuchte "Lagerbereich voll" eingeschaltet.       |

### 9.6.3.4 Beispiel für das Berechnen einer Gleichung

#### Berechnen einer Gleichung

Das folgende Programmbeispiel zeigt, wie Sie mit drei arithmetischen Operationen die folgende Gleichung berechnen können:

$$\text{RESULT} = ((A + B) \times C) / D$$

#### Umsetzung

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der PLC-Variablentabelle:

| Name   | Datentyp | Kommentar                 |
|--------|----------|---------------------------|
| A      | INT      | Erster Wert der Addition  |
| B      | INT      | Zweiter Wert der Addition |
| C      | INT      | Multiplikator             |
| D      | INT      | Divisor                   |
| RESULT | INT      | Endergebnis               |

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

| AWL   | Erläuterung  |
|-------|--|
| L "A" | // Wert des Operanden "A" in Akkumulator 1 laden       |
| L "B" | // Wert des Operanden "A" in Akkumulator 2 verschieben |
|       | // Wert des Operanden "B" in Akkumulator 1 laden       |
| +I    | // Werte der Akkumulatoren 1 und 2 addieren            |
|       | // Summe im Akkumulator 1 speichern                    |
| L "C" | // Summe in Akkumulator 2 verschieben                  |
|       | // Wert des Operanden "C" in Akkumulator 1 laden       |
| *I    | // Werte der Akkumulatoren 1 und 2 multiplizieren      |
|       | // Produkt im Akkumulator 1 speichern                  |
| L "D" | // Produkt in Akkumulator 2 verschieben                |
|       | // Wert des Operanden "D" in Akkumulator 1 laden       |

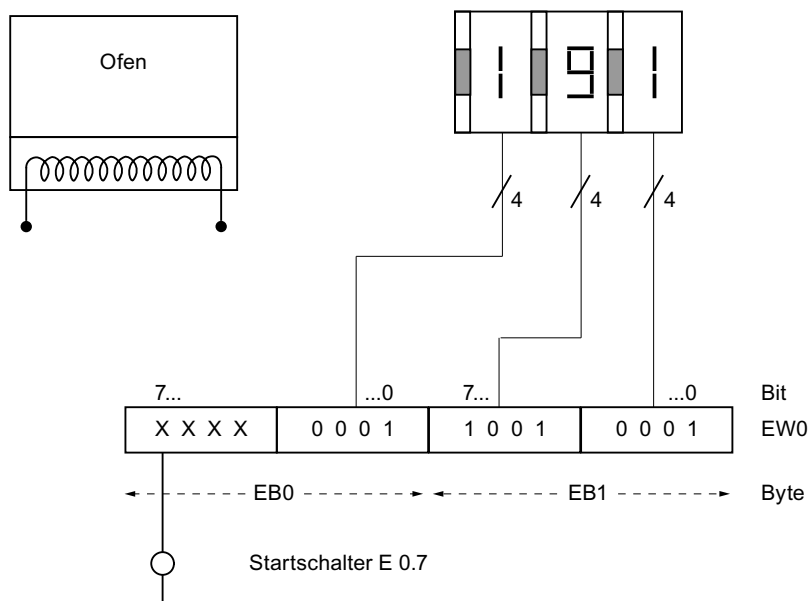
| AWL        | Erläuterung   |
|------------|---|
| /I         | // Wert des Akkumulators 2 durch den Wert des Akkumulators 1 dividieren |
|            | // Ergebnis im Akkumulator 1 speichern                                  |
| T "RESULT" | // Ergebnis in den Operanden "RESULT" übertragen                        |

### 9.6.3.5 Beispiel für das Heizen eines Ofens

#### Heizen eines Ofens

Das folgende Bild zeigt einen Ofen, der mit einem Startschalter eingeschaltet wird. Wenn der Startschalter gedrückt wird, wird der Heizvorgang angestoßen. Die Heizdauer wird durch digitale Vorwählschalter festgelegt. Die Angabe der Heizdauer erfolgt dabei in Sekunden im BCD-Format.

Digitale Vorwählschalter zum Einstellen der Heizdauer





## Umsetzung

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der PLC-Variablen-tabelle:

| Name     | Datentyp | Adresse | Kommentar   |
|----------|----------|---------|---|
| DURATION | WORD     | EW0     | Heizdauer in Sekunden <ul style="list-style-type: none"> <li>• E1.0 bis E1.3: Vorwählschalter für Einer</li> <li>• E1.4 bis E1.7: Vorwählschalter für Zehner</li> <li>• E0.0 bis E0.3: Vorwählschalter für Hunderter</li> </ul> |
| HEATING  | TIMER    | T1      | Zeit, die mit der voreingestellten Heizdauer gestartet wird.  |

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der Bausteinschnittstelle des Codebausteins:

| Name          | Abschnitt | Datentyp | Kommentar               |
|---------------|-----------|----------|-------------------------|
| START         | Input     | BOOL     | Startschalter           |
| START_HEATING | Output    | BOOL     | Beginn des Heizvorgangs |

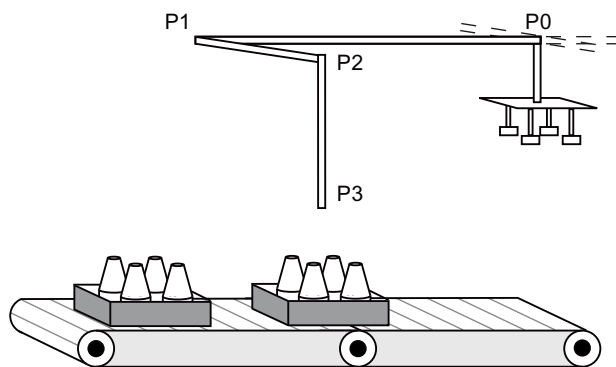
Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

| AWL              | Erläuterung  |
|------------------|--|
| U "HEATING"      | // Abfrage, ob die Zeit gestartet ist  |
| = #START_HEATING | // Heizvorgang anstoßen  |
| BEB              | // Bei VKE=1 Bearbeitung des Bausteins beenden   |
|                  | // Dadurch wird ein Neustart der Zeit "HEATING" verhindert, wenn der Startschalter gedrückt wird |
| L "DURATION"     | // Heizdauer in den Akkumulator 1 laden  |
| UW W#16#0FFF     | // Eingangsbits E0.4 bis E0.7 auf "0" zurücksetzen.  |
| OW W#16#4000     | // Sekunden in Bits E1.2 und E1.3 des Akkumulators 1 einstellen                                  |
| U #START         | // Startschalter auf "1" abfragen  |
| SV "HEATING"     | // Bei einer positiven Flanke am Startschalter Heizvorgang als verlängerten Impuls starten       |

### 9.6.3.6 Beispiel für eine Schrittkette

#### Schrittkette programmieren

Das folgende Bild zeigt eine Station zum Entpalettieren von Glasbehältern. Die Paletten werden auf einem Förderband zur Station transportiert. Wenn eine Palette mit Glasbehältern in der Station eintrifft, wird das Förderband angehalten und ein Greifer aus seiner Grundposition (P0) zur Position über der Palette (P2) bewegt. Befindet sich der Greifer über der Palette, werden die Greifklemmen geöffnet und der Greifer wird gesenkt. Die aktuelle Position des Greifers und der Zustand der Greiferklemmen werden durch Sensoren erfasst. Der beschriebene Ablauf der Greiferbewegung wird in diesem Beispiel durch eine Schrittkette umgesetzt. Die weiteren Schritte zum Entnehmen der Flaschen und deren Transport auf einem weiteren Förderband können zusätzlich programmiert werden.



#### Umsetzung

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der PLC-Variablen-tabelle:

| Name      | Datentyp | Kommentar  |
|-----------|----------|--|
| NUMBER    | INT      | Schrittnummer  |
| Tag_Error | BOOL     | Operand, der gesetzt wird, wenn die Schrittnummer größer 3 ist oder einer der Schritte nicht ausgeführt wurde. |

Die folgende Tabelle zeigt die Deklaration der verwendeten Operanden in der Bausteinschnittstelle des Codebausteins:

| Name         | Abschnitt | Datentyp | Kommentar                      |
|--------------|-----------|----------|--------------------------------|
| POS_0        | Input     | BOOL     | Greifer in Grundstellung (P0)  |
| POS_1        | Input     | BOOL     | Greifer in Position 1 (P1)     |
| POS_2        | Input     | BOOL     | Greifer in Position 2 (P2)     |
| GRIPPER_OPEN | Input     | BOOL     | Greiferklemmen offen           |
| OUT_POS_1    | Output    | BOOL     | Greifer zur Position 1 bewegen |

| Name        | Abschnitt | Datentyp | Kommentar                      |
|-------------|-----------|----------|--------------------------------|
| OUT_POS_2   | Output    | BOOL     | Greifer zur Position 2 bewegen |
| OUT_GRIPPER | Output    | BOOL     | Greiferklemmen öffnen          |
| OUT_POS_3   | Output    | BOOL     | Greifer zur Position 3 bewegen |

Das folgende AWL-Programm zeigt die Umsetzung des Beispiels:

| AWL                  | Erläuterung   |
|----------------------|---|
| L "NUMBER"           | // Schrittnummer in Akkumulator 1 laden.  |
| SPL END              | // Anfang der Sprungliste   |
| SPA POSITION_0       | // Bei einem Wert "0" im Akkumulator 1 zur Sprungmarke "POSITION_0" springen.                 |
| SPA POSITION_1       | // Bei einem Wert "1" im Akkumulator 1 zur Sprungmarke "POSITION_1" springen.                 |
| SPA POSITION_2       | // Bei einem Wert "2" im Akkumulator 1 zur Sprungmarke "POSITION_2" springen.                 |
| SPA POSITION_3       | // Bei einem Wert "3" im Akkumulator 1 zur Sprungmarke "POSITION_3" springen.                 |
| END: SPA ERROR       | // Ende der Sprungliste   |
|                      | // Bei einer Schrittnummer größer als 3 zur Sprungmarke "ERROR" springen.                     |
| POSITION_0: U #POS_0 | // Sprungmarke "POSITION_0"   |
| = #OUT_POS_1         | // Abfrage, ob der Greifer in Grundstellung (P0) ist.   |
| SPBN ERROR           | // Bei VKE "0" zur Sprungmarke "ERROR" springen.  |
| SPB NEXT             | // Bei VKE "1" zur Sprungmarke "NEXT" springen.   |
| POSITION_1: U #POS_1 | // Sprungmarke "POSITION_1"   |
| = #OUT_POS_2         | // Abfrage, ob der Greifer in der Position 1 (P1) ist.  |
| SPBN ERROR           | // Bei VKE "0" zur Sprungmarke "ERROR" springen.  |
| SPB NEXT             | // Bei VKE "1" zur Sprungmarke "NEXT" springen.   |
| POSITION_2: U #POS_2 | // Sprungmarke "POSITION_2"   |
| = #OUT_GRIPPER       | // Abfrage, ob der Greifer in der Position 2 (P2) ist.  |
| SPBN ERROR           | // Bei VKE "0" zur Sprungmarke "ERROR" springen.  |
| SPB NEXT             | // Bei VKE "1" zur Sprungmarke "NEXT" springen.   |
| POSITION_3: U #POS_3 | // Sprungmarke "POSITION_3"   |
| U #GRIPPER_OPEN      | // Abfrage, ob die Greiferklemmen offen sind  |
| = #OUT_POS_3         | // Bei erfüllter Bedingung Ausgang "OUT_POS_3" setzen und Greifer zur Position 3 (P3) senken. |
| SPBN ERROR           | // Bei VKE "0" zur Sprungmarke "ERROR" springen.  |
| SPB NEXT             | // Bei VKE "1" zur Sprungmarke "NEXT" springen.   |

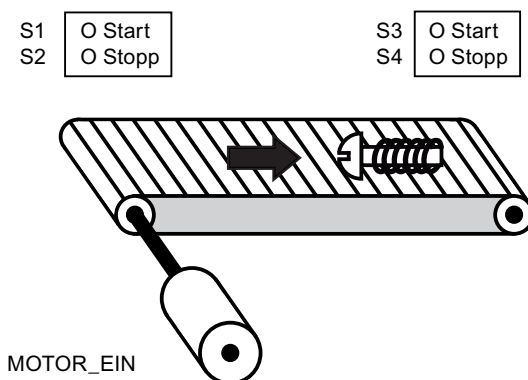
| AWL               | Erläuterung   |
|-------------------|---|
| NEXT: INC 1       | // Sprungmarke "NEXT"<br>// Schrittnummer im Akkumulator 1 um eins erhöhen.   |
| T "NUMBER"        | // Schrittnummer in den Operanden "NUMBER" transferieren.   |
| L 3               | // Aktuelle Schrittnummer in Akkumulator 2 verschieben.   |
| >I                | // Den Wert 3 in Akkumulator 1 laden.   |
| SPB RESET_NUMBER  | // Abfrage, ob die aktuelle Schrittnummer größer als 3 ist.   |
| BEA               | // Bei einem Abfrageergebnis "1" zur Sprungmarke "RESET_NUMBER" springen und die Programmbearbeitung dort fortsetzen. |
| RESET_NUMBER: L 0 | // Baustein beenden   |
| T "NUMBER"        | // Sprungmarke "RESET_NUMBER"<br>// Den Wert "0" in Akkumulator 1 laden.  |
| BEA               | // Operanden "NUMBER" (Schrittnummer) den Wert "0" zuweisen.  |
| ERROR: NOT        | // Baustein beenden   |
| = "Tag_Error"     | // Sprungmarke "ERROR"<br>// Negiertes VKE dem Operanden "Tag_Error" zuweisen.  |
| BEA               | // Baustein beenden   |

## 9.6.4 Programmierbeispiele SCL

### 9.6.4.1 Beispiel für das Steuern eines Förderbands

#### Steuern eines Förderbands

Das folgende Bild zeigt ein Förderband, das elektrisch in Gang gesetzt werden kann. Am Anfang des Förderbands befinden sich zwei Druckschalter, S1 für START und S2 für STOPP. Am Ende des Förderbands befinden sich ebenfalls zwei Druckschalter, S3 für START und S4 für STOPP. Das Förderband kann von beiden Seiten aus gestartet oder gestoppt werden.



## Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Operand                   | Deklaration | Datentyp | Beschreibung                                  |
|---------------------------|-------------|----------|---|
| Startschalter_Links (S1)  | Input       | BOOL     | Startschalter am linken Rand des Förderbands  |
| Stoppschalter_Links (S2)  | Input       | BOOL     | Stoppschalter am linken Rand des Förderbands  |
| Startschalter_Rechts (S3) | Input       | BOOL     | Startschalter am rechten Rand des Förderbands |
| Stoppschalter_Rechts (S4) | Input       | BOOL     | Stoppschalter am rechten Rand des Förderbands |
| MOTOR_ON                  | Output      | BOOL     | Motor des Förderbands einschalten             |
| MOTOR_OFF                 | Output      | BOOL     | Motor des Förderbands ausschalten             |

Das folgende SCL-Programm zeigt die Umsetzung der Aufgabe:

```

SCL
IF "Startschalter_Links" OR "Startschalter_Rechts" = 1 THEN 1 := "MOTOR_ON";
IF "Stoppschalter_Links" OR "Stoppschalter_Rechts" = 1 THEN 1 := "MOTOR_OFF";

```

Wenn der Startschalter "Startschalter\_Links" oder "Startschalter\_Rechts" betätigt wird, wird der Motor des Förderbands eingeschaltet. Wenn der Stoppschalter "Stoppschalter\_Links" oder "Stoppschalter\_Rechts" betätigt wird, wird der Motor des Förderbands ausgeschaltet.

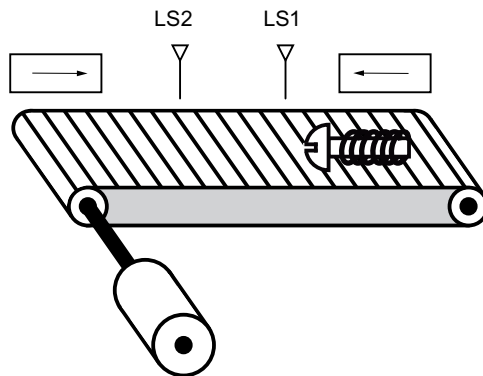
## Siehe auch

Logische Ausdrücke (Seite 1369)

### 9.6.4.2 Beispiel für das Erfassen der Richtung eines Förderbands

#### Erfassen der Richtung eines Förderbands

Das folgende Bild zeigt ein Förderband, das mit zwei Lichtschranken (LS1, LS2) ausgestattet ist. Die Lichtschranken sollen feststellen, in welche Richtung sich ein Gegenstand auf dem Förderband bewegt.



#### Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name   | Deklaration | Datentyp | Beschreibung                          |
|--------|-------------|----------|---------------------------------------|
| LS1    | Input       | BOOL     | Lichtschranke 1                       |
| LS2    | Input       | BOOL     | Lichtschranke 2                       |
| RECHTS | Output      | BOOL     | Anzeige für eine Bewegung nach rechts |
| LINKS  | Output      | BOOL     | Anzeige für eine Bewegung nach links  |

Das folgende SCL-Programm zeigt die Umsetzung des Beispiels:

```

SCL
IF "LS1" = 1 AND NOT "LS2" = 0 THEN 1 := "LINKS";
IF "LS2" = 1 AND NOT "LS1" = 0 THEN 1 := "RECHTS";
IF "LS2" = 0 THEN 0 := "RECHTS";
IF "LS1" = 0 THEN 0 := "LINKS";
    
```

Wenn an der Lichtschranke "LS1" der Signalzustand "1" auftritt und gleichzeitig der Signalzustand an der Lichtschranke "LS2" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach links. Wenn an der Lichtschranke "LS2" der Signalzustand "1" auftritt und gleichzeitig der Signalzustand an der Lichtschranke "LS1" "0" ist, dann bewegt sich der Gegenstand auf dem Förderband nach rechts. Die Anzeigen für eine Bewegung nach links oder rechts werden ausgeschaltet, wenn der Signalzustand an den beiden Lichtschranken "0" ist.

Siehe auch

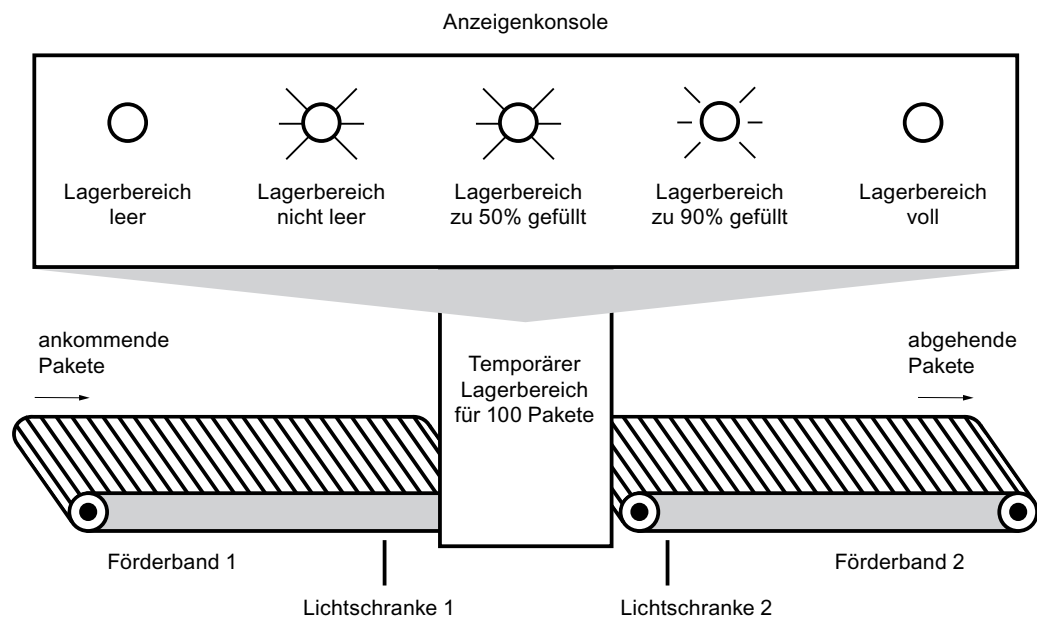
Logische Ausdrücke (Seite 1369)

### 9.6.4.3 Beispiel für das Erfassen des Füllstands eines Lagerbereichs

#### Erfassen des Füllstands eines Lagerbereichs

Das folgende Bild zeigt ein System mit zwei Förderbändern und einem temporären Lagerbereich dazwischen. Förderband 1 transportiert die Pakete zum Lagerbereich. Eine Lichtschranke am Ende des Förderbands 1 neben dem Lagerbereich ermittelt, wie viele Pakete in den Lagerbereich transportiert werden. Förderband 2 transportiert Pakete von diesem temporären Lagerbereich zu einer Laderampe, auf die sie zur Auslieferung beim Kunden auf LKW verladen werden. Eine Lichtschranke am Ausgang des Lagerbereichs ermittelt, wie viele Pakete aus dem Lagerbereich heraus zur Laderampe transportiert werden. Fünf Anzeigelampen zeigen an, wie weit der temporäre Lagerbereich gefüllt ist.

Bei einem neuen Start der Förderbänder wird der aktuelle Zählwert auf die Anzahl der im Lagerbereich vorhandenen Pakete gesetzt.



#### Umsetzung

Die folgende Tabelle zeigt die Definition der verwendeten Variablen:

| Name  | Deklaration | Datentyp | Beschreibung      |
|-------|-------------|----------|-------------------|
| LS1   | Input       | BOOL     | Lichtschranke 1   |
| LS2   | Input       | BOOL     | Lichtschranke 2   |
| RESET | Input       | BOOL     | Zähler rücksetzen |

| Name           | Deklaration | Datentyp | Beschreibung   |
|----------------|-------------|----------|--|
| LOAD           | Input       | BOOL     | Zähler auf den Wert des Parameters "ZW" setzen   |
| LAGERBESTAND   | Input       | INT      | Lagerbestand beim Neustart   |
| PAKETANZAHL    | Output      | INT      | Anzahl der Pakete im Lagerbereich (Aktueller Zählwert)   |
| LB_PAKETE      | Output      | BOOL     | Wird gesetzt, wenn der aktuelle Zählwert größer oder gleich dem Wert der Variablen "LAGERBESTAND" ist. |
| LAG_LEER       | Output      | BOOL     | Anzeigeleuchte: Lagerbereich leer  |
| LAG_NICHT_LEER | Output      | BOOL     | Anzeigeleuchte: Lagerbereich nicht leer  |
| LAG_50%_VOLL   | Output      | BOOL     | Anzeigeleuchte: Lagerbereich zu 50 % voll  |
| LAG_90%_VOLL   | Output      | BOOL     | Anzeigeleuchte: Lagerbereich zu 90 % voll  |
| LAG_VOLL       | Output      | BOOL     | Anzeigeleuchte: Lagerbereich voll  |
| VOLUME_50      | Input       | INT      | Vergleichswert: 50 Pakete  |
| VOLUME_90      | Input       | INT      | Vergleichswert: 90 Pakete  |
| VOLUME_100     | Input       | INT      | Vergleichswert: 100 Pakete   |

Das folgende SCL-Programm zeigt die Umsetzung des Beispiels:

Wenn ein Paket zum Lagerbereich transportiert wird, wechselt der Signalzustand an "LS1" von "0" auf "1" (positive Signalfanke). Bei einer positiven Signalfanke an "LS1" wird der "Vorwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins erhöht.

Wenn ein Paket vom Lagerbereich zur Laderampe transportiert wird, wechselt der Signalzustand an "LS2" von "0" auf "1" (positive Signalfanke). Bei einer positiven Signalfanke an "LS2" wird der "Rückwärts"-Zähler aktiviert und der aktuelle Zählwert von "PAKETANZAHL" um eins verringert.

Wenn sich keine Pakete im Lagerbereich befinden ("PAKETANZAHL" = "0"), wird die Variable "LAG\_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich leer" eingeschaltet.

Der aktuelle Zählwert kann auf "0" zurückgesetzt werden, wenn die Variable "RESET" auf den Signalzustand "1" gesetzt wird.

Wenn die Variable "LOAD" auf den Signalzustand "1" gesetzt wird, wird der aktuelle Zählwert auf den Wert der Variablen "LAGERBESTAND" gesetzt. Solange der aktuelle Zählwert größer



oder gleich dem Wert der Variablen "LAGERBESTAND" ist, liefert die Variable "LB\_PAKETE" den Signalzustand "1".

```
SCL
"CTUD_DB".CTUD(CU := "LS1",
               CD := "LS2",
               R  := "RESET",
               LD := "LOAD",
               PV := "LAGERBESTAND",
               QU := "LB_PAKETE",
               QD := "LAG_LEER",
               CV := "PAKETANZAHL");
```

Solange sich Pakete im Lagerbereich befinden, wird die Variable "LAG\_NICHT\_LEER" auf den Signalzustand "1" gesetzt und die Anzeigeleuchte "Lagerbereich nicht leer" eingeschaltet.

```
SCL
"LAG_NICHT_LEER" := NOT "LAG_LEER"
```

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 50 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 50 % gefüllt" ein.

```
SCL
IF "PAKETANZAHL" >= "VOLUME_50" THEN "LAG_50%_VOLL" := 1;
IF "PAKETANZAHL" <= "VOLUME_90" THEN "LAG_50%_VOLL" := 1;
```

Wenn die Anzahl der Pakete im Lagerbereich größer oder gleich 90 ist, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich zu 90 % gefüllt" ein.

```
SCL
IF "PAKETANZAHL" >= "VOLUME_90" THEN "LAG_90%_VOLL" := 1;
IF "PAKETANZAHL" < "VOLUME_100" THEN "LAG_90%_VOLL" := 1;
```

Wenn die Anzahl der Pakete im Lagerbereich 100 erreicht, schaltet sich die Anzeigeleuchte für die Meldung "Lagerbereich voll" ein.

```
SCL
IF "PAKETANZAHL" >= "VOLUME_100" THEN "LAG_VOLL" := 1;
```

## Siehe auch

Logische Ausdrücke (Seite 1369)

## 9.7 Referenzen

### 9.7.1 Allgemeine Parameter der Anweisungen

#### 9.7.1.1 Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen

##### Asynchron arbeitende Anweisungen

Bei asynchron arbeitenden Anweisungen erstreckt sich die Funktionsausführung über mehrere Aufrufe.

##### Identifikation des Auftrags

Falls Sie mit asynchron arbeitenden Anweisung die Auslösung eines Prozessalarms oder die Ausgabe von Steuerkommandos an DP-Slaves oder eine Datenübertragung oder den Abbruch einer nichtprojektierten Verbindung angestoßen haben und Sie diese Anweisung erneut aufrufen, bevor der laufende Auftrag beendet wurde, hängt das weitere Verhalten der Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt.

##### Parameter REQ

Der Eingangsparameter REQ (request) dient ausschließlich dem Anstoß des Auftrags:

- Wenn Sie die Anweisung zu einem Auftrag aufrufen, der derzeit nicht aktiviert ist, so stoßen Sie mit REQ = 1 den Auftrag an (Fall 1).
- Ist ein bestimmter Auftrag angestoßen und noch nicht abgeschlossen und Sie rufen die Anweisung zum gleichen Auftrag erneut auf (z. B. in einem Weckalarm-OB), so wird REQ durch die Anweisung nicht ausgewertet (Fall 2).

##### Parameter RET\_VAL und BUSY

Über die Ausgangsparameter RET\_VAL und BUSY wird der Zustand der Auftragsausführung angezeigt.

Beachten Sie den Hinweis in dem Abschnitt: Fehlerauswertung mit dem Ausgangsparameter RET\_VAL (Seite 1622)

- Im Fall 1 (Erstaufruf mit REQ=1) wird bei freien Systemressourcen und korrekter Versorgung der Eingangsparameter in RET\_VAL W#16#7001 eingetragen, und BUSY wird gesetzt.  
Falls die benötigten Systemressourcen momentan belegt sind oder ein Fehler in den Eingangsparametern vorliegt, wird in RET\_VAL der zugehörige Fehlercode eingetragen, und BUSY wird mit 0 beschrieben.
- Im Fall 2 (Zwischenaufruf) wird W#16#7002 in RET\_VAL eingetragen (Dies entspricht einer Warnung: Auftrag wird derzeit noch bearbeitet!), und BUSY wird gesetzt.
- Beim letzten Aufruf für einen Auftrag gilt:
  - Bei der Anweisung "DPNRM\_DG (Seite 2450)" wird bei fehlerfreier Datenübertragung in RET\_VAL die Anzahl gelieferter Daten in Bytes als positive Zahl eingetragen. BUSY wird in diesem Fall mit "0" beschrieben.  
Im Fehlerfall wird in RET\_VAL die Fehlerinformation eingetragen. BUSY dürfen Sie in diesem Fall nicht auswerten.
  - Bei allen anderen Anweisungen wird bei fehlerfreier Auftragsausführung in RET\_VAL "0" eingetragen, und BUSY wird mit "0" beschrieben. Im Fehlerfall wird in RET\_VAL der Fehlercode eingetragen, und BUSY wird mit "0" beschrieben.

---

**Hinweis**

Falls erster und letzter Aufruf zusammenfallen, gilt für RET\_VAL und BUSY das für den letzten Aufruf Gesagte.

---

## Überblick

Die folgende Tabelle gibt Ihnen einen Überblick über die oben beschriebenen Zusammenhänge. Sie zeigt insbesondere die möglichen Werte der Ausgangsparameter an, falls die Auftragsausführung nicht nach einem Anweisungs-Aufruf abgeschlossen ist.

---

**Hinweis**

Sie müssen in Ihrem Programm nach jedem Aufruf die relevanten Ausgangsparameter bewerten.

---

Zusammenhang zwischen Aufruf, REQ, RET\_VAL und BUSY bei einem "laufenden" Auftrag

| Lfd. Nr. des Aufrufs | Aufrufart      | REQ        | RET_VAL   | BUSY |
|----------------------|----------------|------------|---|------|
| 1                    | erster Aufruf  | 1          | W#16#7001                                       | 1    |
|                      |                |            | Fehlercode                                      | 0    |
| 2 bis (n - 1)        | Zwischenaufruf | irrelevant | W#16#7002                                       | 1    |
| n                    | letzter Aufruf | irrelevant | W#16#0000, falls keine Fehler aufgetreten sind. | 0    |
|                      |                |            | Fehlercode, falls Fehler aufgetreten sind.      | 0    |

### 9.7.1.2 Fehlerauswertung mit dem Ausgangsparameter RET\_VAL

#### Arten der Fehlerinformationen

Eine bearbeitete Anweisung zeigt Ihnen im Anwenderprogramm an, ob die CPU die Funktion der Anweisung erfolgreich ausführen konnte oder nicht.

Eine entsprechende Fehlerinformation erhalten Sie auf zwei Wegen:

- im BIE-Bit des Statusworts
- im Ausgangsparameter RET\_VAL (return value).

---

#### Hinweis

Sie sollten vor der Auswertung der anweisungsspezifischen Ausgangsparameter immer wie folgt vorgehen:

- Werten Sie als Erstes das BIE-Bit des Statusworts aus.
- Überprüfen Sie anschließend den Ausgangsparameter RET\_VAL.

Falls durch das BIE-Bit eine fehlerhafte Bearbeitung der Anweisung signalisiert wird oder in RET\_VAL ein allgemeiner Fehlercode steht, dürfen Sie die anweisungsspezifischen Ausgangsparameter nicht auswerten.

---

#### Fehlerinformationen im Rückgabewert

Eine Anweisung zeigt durch den Wert "0" im Binärergebnisbit (BIE) des Statusworts an, dass bei der Bearbeitung der Funktion ein Fehler aufgetreten ist. Einige Anweisungen stellen an einem Ausgangsparameter, der Rückgabewert (RET\_VAL) genannt wird, einen zusätzlichen Fehlercode zur Verfügung. Falls beim Ausgangsparameter RET\_VAL ein allgemeiner Fehler (Erklärung siehe unten) auftritt, so wird dies nur durch den Wert "0" im BIE-Bit des Statusworts angezeigt.

Der Rückgabewert ist vom Datentyp Ganzzahl (INT). Die Relation des Rückgabewerts zu dem Wert "0" zeigt an, ob während der Bearbeitung der Funktion ein Fehler aufgetreten ist.

| Bearbeitung der Anweisung durch die CPU | BIE | Rückgabewert               | Vorzeichen der Ganzzahl         |
|---|-----|----------------------------|---------------------------------|
| fehlerhaft                              | 0   | kleiner als "0"            | negativ (Vorzeichenbit ist "1") |
| fehlerfrei                              | 1   | größer als oder gleich "0" | positiv (Vorzeichenbit ist "0") |

#### Reagieren auf Fehlerinformationen

Bei den Fehlercodes in RET\_VAL wird unterschieden zwischen:

- einem allgemeinen Fehlercode, den alle Anweisungen ausgeben können
- einem spezifischen Fehlercode, den eine Anweisung abhängig von ihren spezifischen Funktionen ausgeben kann.

Sie können Ihr Programm so schreiben, dass es auf mögliche Fehler in der Bearbeitung der Anweisung reagiert. So können Sie Folgefehler vermeiden.

## Allgemeine und spezifische Fehlerinformationen

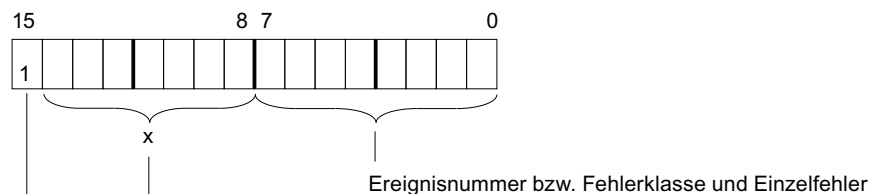
Der Rückgabewert (RET\_VAL) einer Anweisung stellt einen der beiden folgenden Fehlercodes zur Verfügung:

- Einen allgemeinen Fehlercode, der sich auf jede beliebige Anweisung beziehen kann.
- Einen spezifischen Fehlercode, der sich nur auf die jeweilige Anweisung bezieht.

Es handelt sich bei dem Datentyp des Ausgangsparameters RET\_VAL zwar um eine Ganzzahl (INT), doch die Fehlercodes der Anweisung werden nach hexadezimalen Werten gegliedert. Wenn Sie einen Rückgabewert auswerten und den Wert mit den Fehlercodes vergleichen, die in dieser Dokumentation aufgeführt sind, dann lassen Sie sich den Fehlercode im Hexadezimalformat anzeigen.

Das folgende Bild erläutert den Aufbau eines Fehlercodes einer Systemfunktion im Hexadezimalformat.

Fehlercode, z.B. W#16#8081



Wenn  $x = '0'$ , dann handelt es sich um einen spezifischen Fehlercode einer Anweisung. Den spezifischen Fehlercode finden Sie in den Beschreibungen der einzelnen Anweisungen.

Wenn  $x > '0'$ , dann handelt es sich um einen allgemeinen Fehlercode einer Anweisung. In diesem Fall ist  $x$  die Nummer des Anweisungs-Parameters, der den Fehler verursachte. Die möglichen allgemeinen Fehlercodes finden Sie in der folgenden Tabelle.

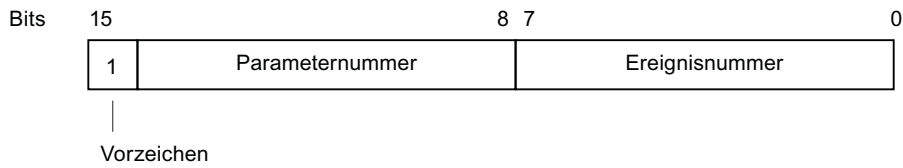
Vorzeichenbit = 1 zeigt an, dass ein Fehler aufgetreten ist.

## Allgemeine Fehlerinformationen

Der allgemeine Fehlercode zeigt Fehler an, die bei allen Anweisungen auftreten können. Ein allgemeiner Fehlercode besteht aus den beiden folgenden Nummern:

- Eine Parameternummer zwischen 1 und 111, wobei 1 den ersten Parameter, 2 den zweiten Parameter usw. der aufgerufenen Anweisung anzeigt.
- Eine Ereignisnummer zwischen 0 und 127. Die Ereignisnummer zeigt einen synchronen Fehler an.

Im Folgenden werden die Fehlercodes für allgemeine Fehler sowie Erläuterungen zu den Fehlern aufgeführt.



**Hinweis**

Falls in RET\_VAL ein allgemeiner Fehlercode eingetragen wurde, kann

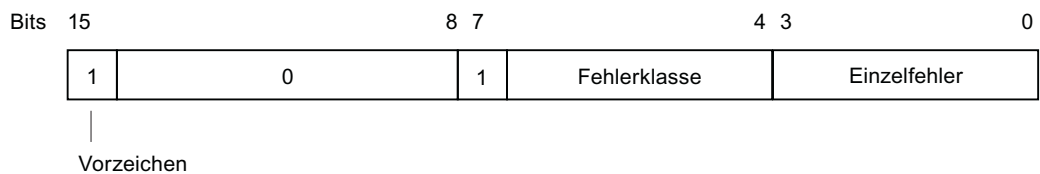
- die zur Anweisung gehörige Aktion angestoßen oder bereits vollständig durchgeführt worden sein.
- bei der Aktion ein anweisungsspezifischer Fehler aufgetreten sein. Aufgrund eines hinterher zusätzlich aufgetretenen allgemeinen Fehlers konnte der spezifische Fehler nicht mehr angezeigt werden.

**Spezifische Fehlerinformationen**

Einige Anweisungen besitzen einen Rückgabewert, der einen für die Anweisung spezifischen Fehlercode zur Verfügung stellt. Ein spezifischer Fehlercode zeigt Fehler an, die nur bei einzelnen Anweisung auftreten kann.

Ein spezifischer Fehlercode besteht aus den beiden folgenden Nummern:

- Eine Fehlerklasse zwischen 0 und 7.
- Ein Einzelfehler zwischen 0 und 15.



**Allgemeine Fehlercodes**

Die folgende Tabelle erläutert die allgemeinen Fehlercodes eines Rückgabewerts. Der Fehlercode wird im Hexadezimalformat gezeigt. Der Buchstabe x in jeder Codenummer dient nur als Platzhalter und stellt die Nummer des Parameters der Systemfunktion dar, die den Fehler verursacht hat.

Allgemeine Fehlercodes

| Fehlercode (W#16#...) | Erläuterung   |
|-----------------------|---|
| 8x7F                  | Interner Fehler<br>Dieser Fehlercode zeigt einen internen Fehler am Parameter x an. |
| 8x01                  | Unzulässige Syntaxkennung bei einem VARIANT-Parameter                               |
| 8x22                  | Bereichslängenfehler beim Lesen eines Parameters.                                   |

| Fehlercode (W#16#...) | Erläuterung   |
|-----------------------|---|
| 8x23                  | Bereichslängenfehler beim Schreiben eines Parameters.<br>Dieser Fehlercode zeigt an, dass sich der Parameter x vollständig oder teilweise außerhalb des Operandenbereichs befindet oder die Länge eines Bitfeldes bei einem VARIANT-Parameter nicht durch 8 teilbar ist.                                  |
| 8x24                  | Bereichsfehler beim Lesen eines Parameters.   |
| 8x25                  | Bereichsfehler beim Schreiben eines Parameters.<br>Dieser Fehlercode zeigt an, dass sich der Parameter x in einem Bereich befindet, der für die Systemfunktion unzulässig ist. Die Beschreibung der jeweiligen Funktion gibt die Bereiche an, die für die Funktion unzulässig sind.                       |
| 8x26                  | Der Parameter enthält eine zu große Nummer einer Zeitzelle.<br>Dieser Fehlercode zeigt an, dass die Zeitzelle, die in Parameter x angegeben wird, nicht vorhanden ist.  |
| 8x27                  | Der Parameter enthält eine zu große Nummer einer Zählerzelle (Nummernfehler des Zählers).<br>Dieser Fehlercode zeigt an, dass die Zählerzelle, die in Parameter x angegeben wird, nicht vorhanden ist.  |
| 8x28                  | Ausrichtungsfehler beim Lesen eines Parameters.   |
| 8x29                  | Ausrichtungsfehler beim Schreiben eines Parameters.<br>Dieser Fehlercode zeigt an, dass der Verweis auf den Parameter x ein Operand ist, dessen Bitadresse ungleich 0 ist.  |
| 8x30                  | Der Parameter befindet sich in dem schreibgeschützten Global-DB.  |
| 8x31                  | Der Parameter befindet sich in dem schreibgeschützten Instanz-DB.<br>Dieser Fehlercode zeigt an, dass der Parameter x sich in einem schreibgeschützten Datenbaustein befindet. Wenn der Datenbaustein von der Systemfunktion selbst geöffnet wurde, gibt die Systemfunktion immer den Wert W#16#8x30 aus. |
| 8x32                  | Der Parameter enthält eine zu große DB-Nummer (Nummernfehler des DB).   |
| 8x34                  | Der Parameter enthält eine zu große FC-Nummer (Nummernfehler der FC).   |
| 8x35                  | Der Parameter enthält eine zu große FB-Nummer (Nummernfehler des FB).<br>Dieser Fehlercode zeigt an, dass der Parameter x eine Bausteinnummer enthält, die größer ist als die maximal zulässige Bausteinnummer.   |
| 8x3A                  | Der Parameter enthält die Nummer eines DB, der nicht geladen ist.   |
| 8x3C                  | Der Parameter enthält die Nummer einer FC, die nicht geladen ist.   |
| 8x3E                  | Der Parameter enthält die Nummer eines FB, der nicht geladen ist.   |
| 8x42                  | Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter aus dem Peripheriebereich der Eingänge auslesen wollte.   |
| 8x43                  | Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter in den Peripheriebereich der Ausgänge schreiben wollte.   |
| 8x44                  | Fehler beim n-ten ( $n > 1$ ) Lesezugriff nach Auftreten eines Fehlers.   |
| 8x45                  | Fehler beim n-ten ( $n > 1$ ) Schreibzugriff nach Auftreten eines Fehlers.<br>Dieser Fehlercode zeigt an, dass der Zugriff auf den gewünschten Parameter verweigert wird.   |

## 9.7.2 Einfache Anweisungen

### 9.7.2.1 KOP

#### Bitverknüpfungen

--| |--: Schließerkontakt

#### Beschreibung

Die Aktivierung eines Schließerkontakts hängt vom Signalzustand des dazugehörigen Operanden ab. Wenn der Operand den Signalzustand "1" hat, wird der Schließerkontakt geschlossen und der Signalzustand des Eingangs auf den Ausgang übertragen.

Wenn der Operand den Signalzustand "0" hat, wird der Schließerkontakt nicht aktiviert und der Signalzustand am Ausgang der Anweisung auf "0" gesetzt.

In einer Reihenschaltung werden zwei oder mehrere Schließerkontakte bitweise durch UND verknüpft. Durch eine Reihenschaltung fließt Strom, wenn alle Kontakte geschlossen sind.

In einer Parallelschaltung werden Schließerkontakte durch ODER verknüpft. Durch eine Parallelschaltung fließt Strom, wenn einer der Kontakte geschlossen ist.

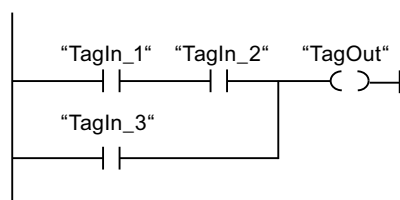
#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

| Parameter | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung                                  |
|-----------|-------------|----------|-----------------|---------------------|---|
|           |             |          | S7-1200         | S7-1500             |   |
| <Operand> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Operand, dessen Signalzustand abgefragt wird. |

#### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" haben den Signalzustand "1".
- Der Operand "TagIn\_3" hat den Signalzustand "1".



**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**---| / |---: Öffnerkontakt**

**Beschreibung**

Die Aktivierung eines Öffnerkontakts hängt vom Signalzustand des dazugehörigen Operanden ab. Wenn der Operand den Signalzustand "1" hat, wird der Öffnerkontakt geöffnet und der Signalzustand am Ausgang der Anweisung auf "0" zurückgesetzt.

Wenn der Operand den Signalzustand "0" hat, wird der Öffnerkontakt nicht aktiviert und der Signalzustand des Eingangs auf den Ausgang übertragen.

In einer Reihenschaltung werden zwei oder mehrere Öffnerkontakte bitweise durch UND verknüpft. Durch eine Reihenschaltung fließt Strom, wenn alle Kontakte geschlossen sind.

In einer Parallelschaltung werden Öffnerkontakte durch ODER verknüpft. Durch eine Parallelschaltung fließt Strom, wenn einer der Kontakte geschlossen ist.

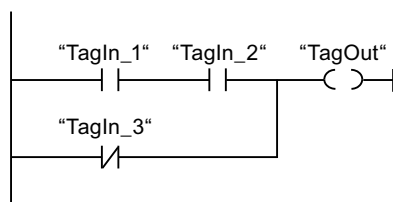
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung:

| Parameter | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung                                  |
|-----------|-------------|----------|-----------------|---------------------|---|
|           |             |          | S7-1200         | S7-1500             |   |
| <Operand> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Operand, dessen Signalzustand abgefragt wird. |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" haben den Signalzustand "1".
- Der Operand "TagIn\_3" hat den Signalzustand "0".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

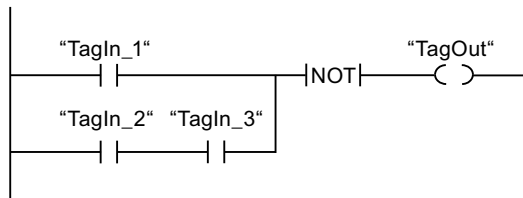
**--|NOT|--: VKE invertieren**

**Beschreibung**

Mit der Anweisung "VKE invertieren" invertieren Sie den Signalzustand des Verknüpfungsergebnisses (VKE). Wenn am Eingang der Anweisung der Signalzustand "1" ansteht, liefert der Ausgang der Anweisung den Signalzustand "0". Ist der Signalzustand am Eingang der Anweisung "0", liefert der Ausgang den Signalzustand "1".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Der Signalzustand an der Operanden "TagIn\_2" und "TagIn\_3" ist "1".

**--( )--: Zuweisung**

**Beschreibung**

Mit der Anweisung "Zuweisung" setzen Sie das Bit eines angegebenen Operanden. Wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule den Signalzustand "1" liefert, wird der angegebene Operand auf den Signalzustand "1" gesetzt. Wenn der Signalzustand am Eingang der Spule "0" ist, wird das Bit des angegebenen Operanden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Eingang der Spule wird direkt auf den Ausgang übertragen.

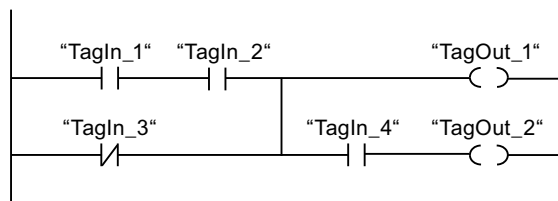
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                          |
|-----------|-------------|----------|-----------------|---------------------------------------|
| <Operand> | Output      | BOOL     | E, A, M, D, L   | Operand, dem das VKE zugewiesen wird. |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut\_1" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn\_3" ist "0".

Der Operand "TagOut\_2" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" sowie der Operand "TagIn\_4" liefern den Signalzustand "1"
- Der Signalzustand des Operanden "TagIn\_3" ist "0" und der des Operanden "TagIn\_4" ist "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## --( / )--: Zuweisung negieren

## Beschreibung

Mit der Anweisung "Zuweisung negieren" kehren Sie das Verknüpfungsergebnis (VKE) um und weisen es dem angegebenen Operanden zu. Wenn das VKE am Eingang der Spule "1" ist, wird der Operand zurückgesetzt. Wenn das VKE am Eingang der Spule "0" ist, wird der Operand auf den Signalzustand "1" gesetzt.

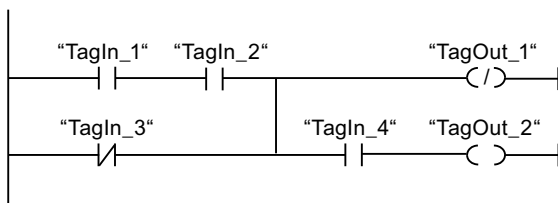
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung negieren":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                                   |
|-----------|-------------|----------|-----------------|--|
| <Operand> | Output      | BOOL     | E, A, M, D, L   | Operand, dem das negierte VKE zugewiesen wird. |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut\_1" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn\_3" ist "0".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### ---( R )---: Ausgang rücksetzen

#### Beschreibung

Mit der Anweisung "Ausgang rücksetzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Fließt Strom zur Spule (VKE = "1"), dann wird der angegebene Operand auf "0" zurückgesetzt. Bei einem VKE von "0" am Eingang der Spule (kein Signalfuss an der Spule), bleibt der Signalzustand des angegebenen Operanden unverändert.

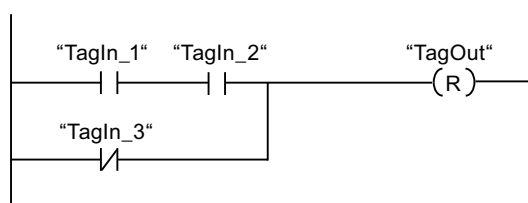
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang rücksetzen":

| Parameter | Deklaration | Datentyp | Speicherbereich |                        | Beschreibung                                   |
|-----------|-------------|----------|-----------------|------------------------|--|
|           |             |          | S7-1200         | S7-1500                |  |
| <Operand> | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L,<br>T, Z | Operand, der bei VKE = "1" zurückgesetzt wird. |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn\_3" ist "0".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ---( S )---: Ausgang setzen

### Beschreibung

Mit der Anweisung "Ausgang setzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "1".

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Fließt Strom zur Spule (VKE = "1"), dann wird der angegebene Operand auf "1" gesetzt. Bei einem VKE von "0" am Eingang der Spule (kein Signalfluss an der Spule), bleibt der Signalzustand des angegebenen Operanden unverändert.

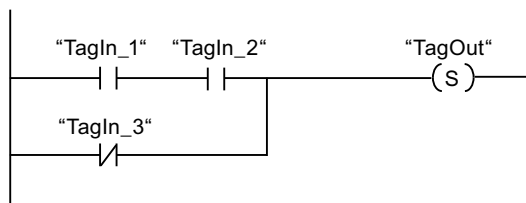
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                             |
|-----------|-------------|----------|-----------------|--|
| <Operand> | Output      | BOOL     | E, A, M, D, L   | Operand, der bei VKE = "1" gesetzt wird. |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Signalzustand des Operanden "TagIn\_3" ist "0".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### SET\_BF: Bitfeld setzen

#### Beschreibung

Mit der Anweisung "Bitfeld setzen" setzen Sie mehrere Bit ab einer bestimmten Adresse.

Die Anzahl der zu setzenden Bit bestimmen Sie durch den Wert von <Operand1>. Die Adresse des ersten zu setzenden Bit wird durch <Operand2> festgelegt. Wenn der Wert von <Operand1> größer als die Anzahl der Bit in einem selektierten Byte ist, werden die Bit des nächstfolgenden Byte gesetzt. Die Bit bleiben so lange gesetzt, bis sie explizit durch eine andere Anweisung zurückgesetzt werden.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Bei einem VKE von "0" am Eingang der Spule wird die Anweisung nicht ausgeführt.

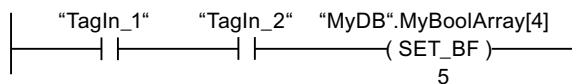
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld setzen":

| Parameter  | Deklaration | Datentyp | Speicherbereich  | Beschreibung                                |
|------------|-------------|----------|--|---|
| <Operand2> | Output      | BOOL     | E, A, M<br>Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL | Zeiger auf das erste Bit, das gesetzt wird. |
| <Operand1> | Input       | UINT     | Konstante  | Anzahl der zu setzenden Bit.                |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, werden 5 Bits ab der Adresse des Operanden "MyDB".MyBoolArray[4] gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## RESET\_BF: Bitfeld rücksetzen

### Beschreibung

Mit der Anweisung "Bitfeld rücksetzen" setzen Sie mehrere Bit ab einer bestimmten Adresse zurück.

Die Anzahl der zurückzusetzenden Bit bestimmen Sie durch den Wert von <Operand1>. Die Adresse des ersten zurückzusetzenden Bit wird durch <Operand2> festgelegt. Wenn der Wert von <Operand1> größer als die Anzahl der Bit in einem selektierten Byte ist, werden die Bit des nächstfolgenden Byte zurückgesetzt. Die Bit bleiben so lange zurückgesetzt, bis sie explizit durch eine andere Anweisung gesetzt werden.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Bei einem VKE von "0" am Eingang der Spule wird die Anweisung nicht ausgeführt.

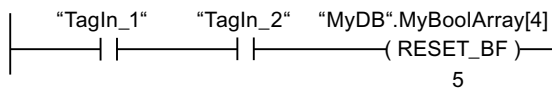
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld rücksetzen":

| Parameter  | Deklaration | Datentyp | Speicherbereich  | Beschreibung                                      |
|------------|-------------|----------|--|---|
| <Operand2> | Output      | BOOL     | E, A, M<br>Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL | Zeiger auf das erste Bit, das zurückgesetzt wird. |
| <Operand1> | Input       | UINT     | Konstante  | Anzahl der zurückzusetzenden Bit.                 |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, werden 5 Bit ab der Adresse des Operanden "MyDB".MyBoolArray[4] zurückgesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### SR: Flipflop setzen/rücksetzen

#### Beschreibung

Mit der Anweisung "Flipflop setzen/rücksetzen" setzen oder rücksetzen Sie das Bit eines angegebenen Operanden, abhängig vom Signalzustand an den Eingängen S und R1. Wenn der Signalzustand am Eingang S "1" und am Eingang R1 "0" ist, wird der angegebene Operand auf "1" gesetzt. Wenn der Signalzustand am Eingang S "0" und am Eingang R1 "1" ist, wird der angegebene Operand auf "0" zurückgesetzt.

Der Eingang R1 dominiert den Eingang S. Bei einem Signalzustand "1" an beiden Eingängen S und R1 wird der Signalzustand des angegebenen Operanden auf "0" zurückgesetzt.

Bei einem Signalzustand "0" an beiden Eingängen S und R1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.



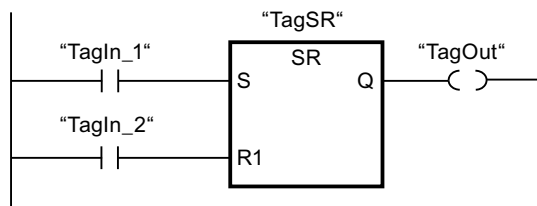
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop setzen/rücksetzen":

| Parameter | Deklaration | Datentyp | Speicherbereich |                        | Beschreibung   |
|-----------|-------------|----------|-----------------|------------------------|--|
|           |             |          | S7-1200         | S7-1500                |  |
| S         | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L          | Setzen<br>freigeben                                    |
| R1        | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L,<br>T, Z | Rücksetzen<br>freigeben                                |
| <Operand> | InOut       | BOOL     | E, A, M, D, L   | E, A, M, D, L          | Operand, der<br>gesetzt oder<br>zurückgesetzt<br>wird. |
| Q         | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L          | Signalzustand<br>des<br>Operanden                      |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Operanden "TagSR" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Der Operand "TagIn\_2" liefert den Signalzustand "0".

Die Operanden "TagSR" und "TagOut" werden zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn\_1" liefert den Signalzustand "0" und der Operand "TagIn\_2" liefert den Signalzustand "1".
- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## RS: Flipflop rücksetzen/setzen

### Beschreibung

Mit der Anweisung "Flipflop rücksetzen/setzen" rücksetzen oder setzen Sie das Bit eines angegebenen Operanden, abhängig vom Signalzustand an den Eingängen R und S1. Wenn der Signalzustand am Eingang R "1" und am Eingang S1 "0" ist, wird der angegebene Operand auf "0" zurückgesetzt. Wenn der Signalzustand am Eingang R "0" und am Eingang S1 "1" ist, wird der angegebene Operand auf "1" gesetzt.

Der Eingang S1 dominiert den Eingang R. Bei einem Signalzustand "1" an beiden Eingängen, R und S1, wird der Signalzustand des angegebenen Operanden auf "1" gesetzt.

Bei einem Signalzustand "0" an beiden Eingängen R und S1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

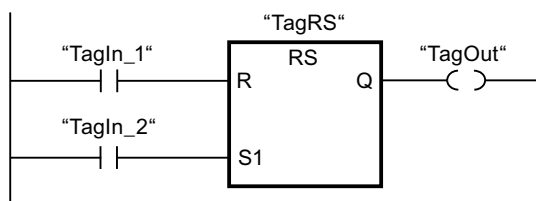
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop rücksetzen/setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung                                  |
|-----------|-------------|----------|-----------------|---------------------|---|
|           |             |          | S7-1200         | S7-1500             |   |
| R         | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Rücksetzen freigeben                          |
| S1        | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Setzen freigeben                              |
| <Operand> | InOut       | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Operand, der zurückgesetzt oder gesetzt wird. |
| Q         | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Signalzustand des Operanden                   |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Operanden "TagRS" und "TagOut" werden zurückgesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Der Operand "TagIn\_2" liefert den Signalzustand "0".

Die Operanden "TagRS" und "TagOut" werden gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn\_1" liefert den Signalzustand "0" und der Operand "TagIn\_2" liefert den Signalzustand "1".
- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## --|P|--: Operand auf positive Signalflanke abfragen

### Beschreibung

Mit der Anweisung "Operand auf positive Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "0" auf "1" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Wenn eine positive Signalflanke erfasst wird, liefert der Ausgang der Anweisung den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

---

### Hinweis

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

---

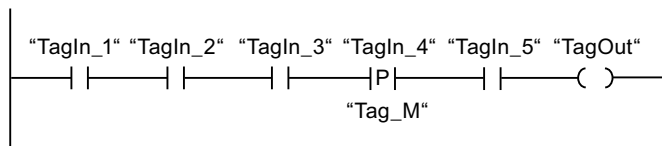
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf positive Signalflanke abfragen":

| Parameter  | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung   |
|------------|-------------|----------|-----------------|---------------------|--|
|            |             |          | S7-1200         | S7-1500             |  |
| <Operand1> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Abzufragendes Signal   |
| <Operand2> | InOut       | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird. |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1", "TagIn\_2" und "TagIn\_3" liefern den Signalzustand "1".
- Am Operanden "TagIn\_4" liegt eine steigende Flanke vor. Der Signalzustand der vorherigen Abfrage wird im Flankenmerker "Tag\_M" gespeichert.
- Der Signalzustand des Operanden "TagIn\_5" ist "1".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### --|N|--: Operand auf negative Signalflanke abfragen

### Beschreibung

Mit der Anweisung "Operand auf negative Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "1" auf "0" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker <Operand2> gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Wenn eine negative Signalflanke erfasst wird, liefert der Ausgang der Anweisung den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Den abzufragenden Operanden (<Operand1>) geben Sie im Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie im Operandenplatzhalter unterhalb der Anweisung an.

**Hinweis**

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

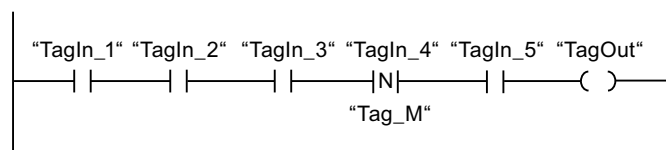
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf negative Signalflanke abfragen":

| Parameter      | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung   |
|----------------|-------------|----------|-----------------|---------------------|--|
|                |             |          | S7-1200         | S7-1500             |  |
| <Operand1<br>> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Abzufragendes Signal   |
| <Operand2<br>> | InOut       | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird. |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1", "TagIn\_2" und "TagIn\_3" liefern den Signalzustand "1".
- Am Operanden "TagIn\_4" liegt eine negative Signalflanke vor. Der Signalzustand der vorherigen Abfrage wird im Flankenmerker "Tag\_M" gespeichert.
- Der Signalzustand des Operanden "TagIn\_5" ist "1".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**--(P)--: Operand bei positiver Signalflanke setzen**

**Beschreibung**

Mit der Anweisung "Operand bei positiver Signalflanke setzen" setzen Sie einen angegebenen Operanden (<Operand1>), wenn eine Änderung im Verknüpfungsergebnis (VKE) von "0" auf "1" vorliegt. Die Anweisung vergleicht das aktuelle VKE mit dem VKE der vorherigen Abfrage, das in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Wenn eine positive Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den zu setzenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

---

**Hinweis**

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

---

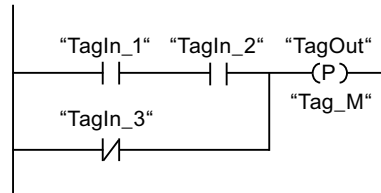
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand bei positiver Signalflanke setzen":

| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|------------|-------------|----------|-----------------|---|
| <Operand1> | Output      | BOOL     | E, A, M, D, L   | Operand, der bei einer positiven Flanke gesetzt wird. |
| <Operand2> | InOut       | BOOL     | E, A, M, D, L   | Flankenmerker   |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird für einen Programmzyklus gesetzt, wenn der Signalzustand am Eingang der Spule von "0" auf "1" wechselt (positive Signalflanke). In allen anderen Fällen führt der Operand "TagOut" den Signalzustand "0".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## --(N)--: Operand bei negativer Signalflanke setzen

### Beschreibung

Mit der Anweisung "Operand bei negativer Signalflanke setzen" setzen Sie einen angegebenen Operanden (<Operand1>), wenn eine Änderung im Verknüpfungsergebnis (VKE) von "1" auf "0" vorliegt. Die Anweisung vergleicht das aktuelle VKE mit dem VKE der vorherigen Abfrage, das in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Wenn eine negative Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den zu setzenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

### Hinweis

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

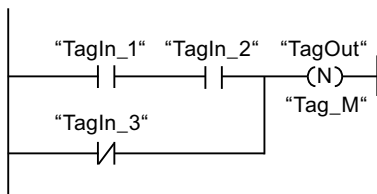
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand bei negativer Signalfanke setzen":

| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|------------|-------------|----------|-----------------|---|
| <Operand1> | Output      | BOOL     | E, A, M, D, L   | Operand, der bei einer negativen Flanke gesetzt wird. |
| <Operand2> | InOut       | BOOL     | E, A, M, D, L   | Flankenmerker   |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird für einen Programmzyklus gesetzt, wenn der Signalzustand am Eingang der Spule von "1" auf "0" wechselt (negative Signalfanke). In allen anderen Fällen führt der Operand "TagOut" den Signalzustand "0".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**P\_TRIG: VKE auf positive Signalfanke abfragen**

**Beschreibung**

Mit der Anweisung "VKE auf positive Signalfanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "0" auf "1" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalfanke vor.



Wenn eine positive Signalfanke erfasst wird, liefert der Ausgang der Anweisung den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

**Hinweis**

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

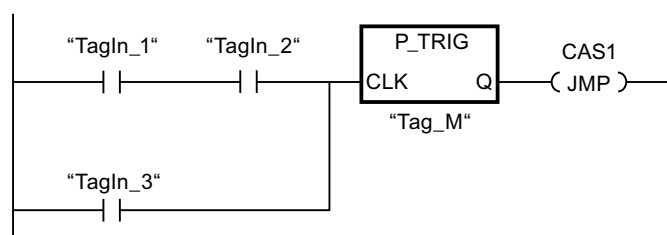
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf positive Signalfanke abfragen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| CLK       | Input       | BOOL     | E, A, M, D, L   | Aktuelles VKE  |
| <Operand> | InOut       | BOOL     | M, D            | Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird. |
| Q         | Output      | BOOL     | E, A, M, D, L   | Ergebnis der Flankenauswertung   |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Im Flankenmerker "Tag\_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

## N\_TRIG: VKE auf negative Signalflanke abfragen

### Beschreibung

Mit der Anweisung "VKE auf negative Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "1" nach "0" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Wenn eine negative Signalflanke erfasst wird, liefert der Ausgang der Anweisung den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

### Hinweis

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

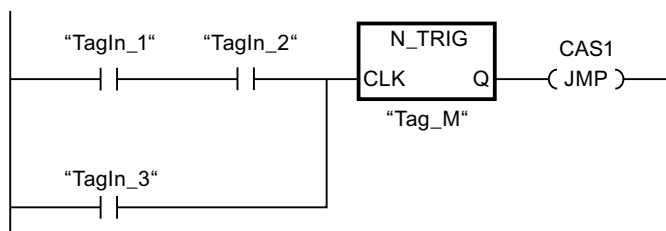
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf negative Signalflanke abfragen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| CLK       | Input       | BOOL     | E, A, M, D, L   | Aktuelles VKE  |
| <Operand> | InOut       | BOOL     | M, D            | Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird. |
| Q         | Output      | BOOL     | E, A, M, D, L   | Ergebnis der Flankenauswertung   |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Im Flankenmerker "Tag\_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## R\_TRIG: Variable bei positiver Signalflanke setzen

### Beschreibung

Mit der Anweisung "Variable bei positiver Signalflanke setzen" setzen Sie eine angegebene Variable im Instanz-DB, wenn eine Änderung im Verknüpfungsergebnis (VKE) von "0" auf "1" vorliegt. Die Anweisung vergleicht das aktuelle VKE am Eingang CLK mit dem VKE der vorherigen Abfrage, die im angegebenen Instanz-DB gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Wenn eine positive Signalflanke erfasst wird, wird die Variable im Instanz-DB auf den Signalzustand "1" gesetzt und der Ausgang Q liefert den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der Flankenmerker in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

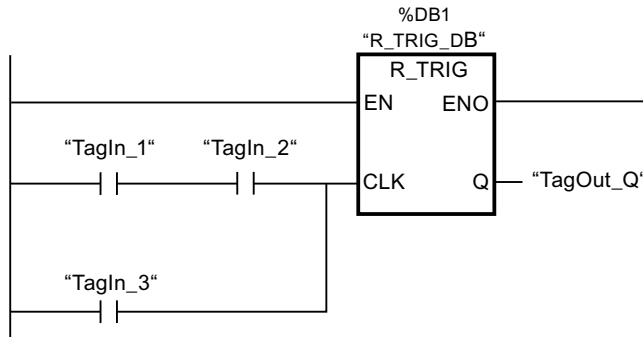
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Variable bei positiver Signalflanke setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung                                      |
|-----------|-------------|----------|------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L                | Freigabeeingang                                   |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang                                   |
| CLK       | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingehendes Signal, dessen Flanke abgefragt wird. |
| Q         | Output      | BOOL     | E, A, M, D, L                | Ergebnis der Flankenauswertung                    |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



In der Variablen im Instanz-DB "R\_TRIG\_DB" wird das VKE der vorherigen Abfrage gespeichert. Wenn in den Operanden "TagIn\_1" und "TagIn\_2" oder im Operanden "TagIn\_3" eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, liefert der Ausgang "TagOut\_Q" den Signalzustand "1".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### F\_TRIG: Variable bei negativer Signalflanke setzen

#### Beschreibung

Mit der Anweisung "Variable bei negativer Signalflanke setzen" setzen Sie eine angegebene Variable im Instanz-DB, wenn eine Änderung im Verknüpfungsergebnis (VKE) von "1" auf "0" vorliegt. Die Anweisung vergleicht das aktuelle VKE am Eingang CLK mit dem VKE der vorherigen Abfrage, die im angegebenen Instanz-DB gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Wenn eine negative Signalflanke erfasst wird, wird die Variable im Instanz-DB auf den Signalzustand "1" gesetzt und der Ausgang Q liefert den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der Flankenmerker in einem eigenen Datenbaustein (Einzelinstantz) oder als lokale Variable (Multiinstantz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

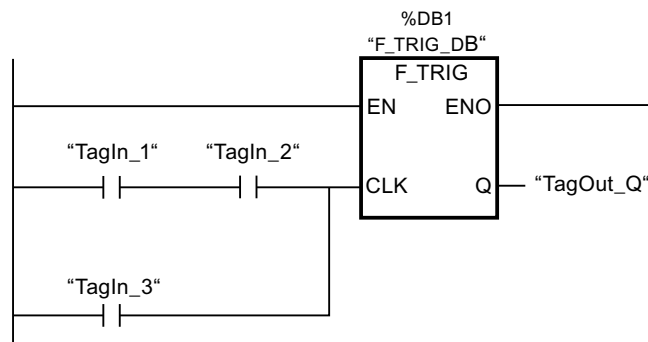
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Variable bei negativer Signalfanke setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung                                      |
|-----------|-------------|----------|------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L                | Freigabeeingang                                   |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang                                   |
| CLK       | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingehendes Signal, dessen Flanke abgefragt wird. |
| Q         | Output      | BOOL     | E, A, M, D, L                | Ergebnis der Flankenauswertung                    |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



In der Variablen im Instanz-DB "F\_TRIG\_DB" wird das VKE der vorherigen Abfrage gespeichert. Wenn in den Operanden "TagIn\_1" und "TagIn\_2" oder im Operanden "TagIn\_3" eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, liefert der Ausgang "TagOut\_Q" den Signalzustand "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Zeiten

### IEC-Zeiten

#### TP: Impuls erzeugen

#### Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für eine programmierte Zeitdauer. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalfanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Ausgang Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalfanke beeinflusst den Signalzustand am Ausgang Q nicht, solange die Zeitdauer PT läuft.

Am Ausgang ET können Sie den aktuellen Zeitwert abfragen. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

---

#### Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

#### Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TP\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME, TP\_LTIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

#### Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TP\_TIME oder TP\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME, TP\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstantz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Impuls erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

## Parameter

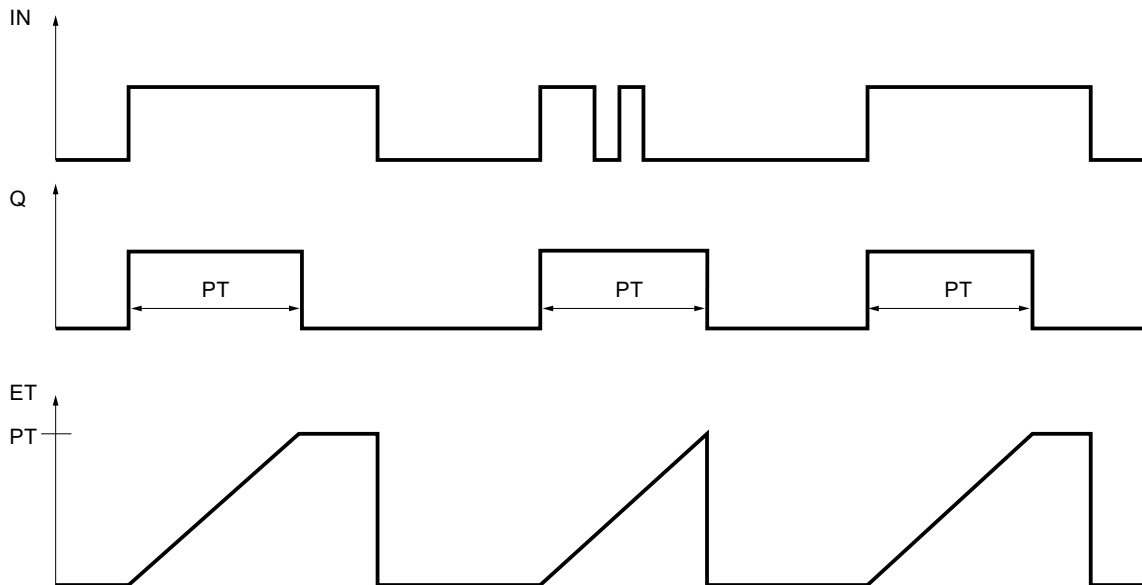
Die folgende Tabelle zeigt die Parameter der Anweisung "Impuls erzeugen":

| Parameter | Deklaration | Datentyp |             | Speicherbereich              |                                 | Beschreibung  |
|-----------|-------------|----------|-------------|------------------------------|---------------------------------|---|
|           |             | S7-1200  | S7-1500     | S7-1200                      | S7-1500                         |   |
| IN        | Input       | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Starteingang  |
| PT        | Input       | TIME     | TIME, LTIME | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Zeitdauer des Impulses<br>Der Wert am Parameter PT muss positiv sein. |
| Q         | Output      | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Impulsausgang   |
| ET        | Output      | TIME     | TIME, LTIME | E, A, M, D, L                | E, A, M, D, L, P                | Aktueller Zeitwert  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":



### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### TON: Einschaltverzögerung erzeugen

#### Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Ausgang Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Starteingang erfasst wird.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Ausgang ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.



---

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

---

#### **Hinweis**

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

### **Für CPU S7-1200**

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TON\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### **Für CPU S7-1500**

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TON\_TIME oder TON\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME, TON\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Einschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

### Parameter

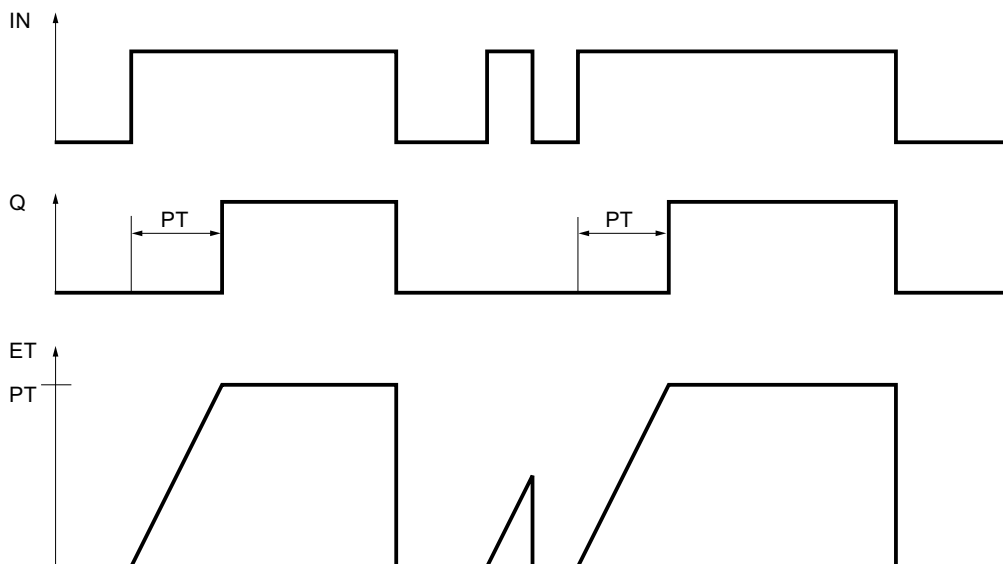
Die folgende Tabelle zeigt die Parameter der Anweisung "Einschaltverzögerung erzeugen":

| Parameter | Deklaration | Datentyp |             | Speicherbereich              |                                 | Beschreibung  |
|-----------|-------------|----------|-------------|------------------------------|---------------------------------|---|
|           |             | S7-1200  | S7-1500     | S7-1200                      | S7-1500                         |   |
| IN        | Input       | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Starteingang  |
| PT        | Input       | TIME     | TIME, LTIME | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Zeitdauer der Einschaltverzögerung<br>Der Wert am Parameter PT muss positiv sein. |
| Q         | Output      | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.                            |
| ET        | Output      | TIME     | TIME, LTIME | E, A, M, D, L                | E, A, M, D, L, P                | Aktueller Zeitwert  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":



**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**TOF: Ausschaltverzögerung erzeugen****Beschreibung**

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die programmierte Zeitdauer PT. Der Ausgang Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Eingang IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Ausgang Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Ausgang Q zurückgesetzt. Falls der Signalzustand am Eingang IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Ausgang Q bleibt weiterhin auf "1" gesetzt.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Ausgang ET solange auf dem aktuellen Wert stehen, bis der Eingang IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

---

**Hinweis**

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

**Für CPU S7-1200**

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TOF\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

**Für CPU S7-1500**

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TOF\_TIME oder TOF\_LTIMER, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME, TOF\_LTIMER, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelninstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Ausschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

### Parameter

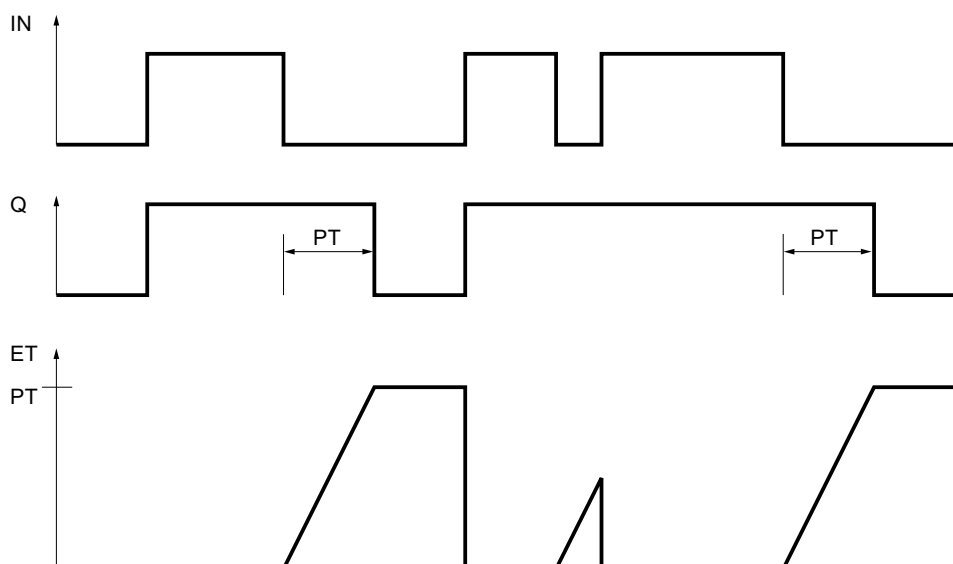
Die folgende Tabelle zeigt die Parameter der Anweisung "Ausschaltverzögerung erzeugen":

| Parameter | Deklaration | Datentyp |             | Speicherbereich              |                                 | Beschreibung  |
|-----------|-------------|----------|-------------|------------------------------|---------------------------------|---|
|           |             | S7-1200  | S7-1500     | S7-1200                      | S7-1500                         |   |
| IN        | Input       | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Starteingang  |
| PT        | Input       | TIME     | TIME, LTIME | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Zeitdauer der Ausschaltverzögerung<br>Der Wert am Parameter PT muss positiv sein. |
| Q         | Output      | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Ausgang, der nach dem Ablauf der Zeit PT zurückgesetzt wird.                      |
| ET        | Output      | TIME     | TIME, LTIME | E, A, M, D, L                | E, A, M, D, L, P                | Aktueller Zeitwert  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":



## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## TONR: Zeit akkumulieren

### Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines durch den Parameter PT gesetzten Zeitraums. Wenn der Signalzustand am Eingang IN von "0" auf "1" wechselt (positive Signalfanke), wird die Anweisung ausgeführt und die Zeitdauer PT gestartet. Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Eingang IN erfasst werden. Die akkumulierte Zeit wird am Ausgang ET ausgegeben und kann an diesem abgefragt werden. Wenn die Zeitdauer PT erreicht ist, liefert der Ausgang Q den Signalzustand "1". Der Parameter Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Parameter IN von "1" auf "0" wechselt (negative Signalfanke).

Der Eingang R setzt die Ausgänge ET und Q, unabhängig vom Signalzustand am Starteingang, zurück.

Jedem Aufruf der Anweisung "Zeit akkumulieren" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

**Für CPU S7-1200**

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TONR\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

**Für CPU S7-1500**

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TONR\_TIME oder TONR\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME, TONR\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

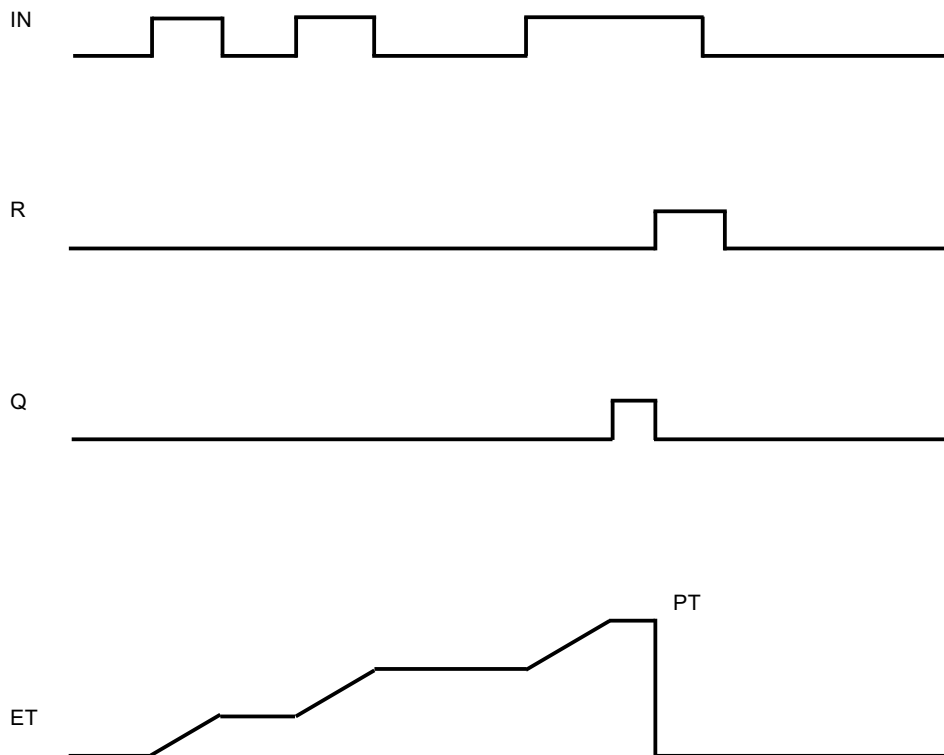
| Parameter | Deklaration | Datentyp |             | Speicherbereich              |                                 | Beschreibung  |
|-----------|-------------|----------|-------------|------------------------------|---------------------------------|---|
|           |             | S7-1200  | S7-1500     | S7-1200                      | S7-1500                         |   |
| IN        | Input       | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Starteingang  |
| R         | Input       | BOOL     | BOOL        | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Rücksetzeingang   |
| PT        | Input       | TIME     | TIME, LTIME | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Maximale Dauer der Zeiterfassung<br>Der Wert am Parameter PT muss positiv sein. |

| Parameter | Deklaration | Datentyp |             | Speicherbereich |                  | Beschreibung   |
|-----------|-------------|----------|-------------|-----------------|------------------|--|
|           |             | S7-1200  | S7-1500     | S7-1200         | S7-1500          |  |
| Q         | Output      | BOOL     | BOOL        | E, A, M, D, L   | E, A, M, D, L, P | Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird. |
| ET        | Output      | TIME     | TIME, LTIME | E, A, M, D, L   | E, A, M, D, L, P | Akkumulierte Zeit                                      |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit akkumulieren":



### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ---( TP )---: Zeit als Impuls starten

### Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Impuls. Die IEC-Zeit wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) wechselt. Die IEC-Zeit läuft mit der angegebenen Zeitdauer ab, unabhängig vom weiteren Verlauf des VKE. Auch die Erfassung einer neuen positiven Signalfanke beeinflusst den Ablauf der IEC-Zeit nicht. Solange die IEC-Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn die IEC-Zeit abgelaufen ist, liefert der Zeitstatus den Signalzustand "0".

---

#### Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC\_TIMER-Struktur aktualisiert.

---

### Für CPU S7-1200

Die Anweisung "Zeit als Impuls starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER oder TP\_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_LTIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### Für CPU S7-1500

Die Anweisung "Zeit als Impuls starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER, IEC\_LTIMER, TP\_TIME oder TP\_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME, TP\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente Q der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe eines Schließerkontakts auf "1" oder eines Öffnerkontakts auf "0" abfragen. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

Die Ausführung der Anweisung "Zeit als Impuls starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.



## Parameter

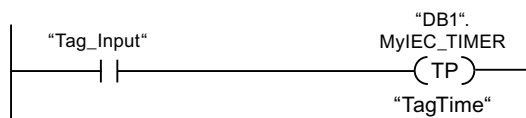
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

| Parameter   | Deklaration | Datentyp              |   | Speicherbereich                 | Beschreibung                           |
|-------------|-------------|-----------------------|---|---------------------------------|--|
|             |             | S7-1200               | S7-1500   |                                 |  |
| <Zeitdauer> | Input       | TIME                  | TIME, LTIME                                       | E, A, M, D, L<br>oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft. |
| <IEC-Zeit>  | InOut       | IEC_TIMER,<br>TP_TIME | IEC_TIMER,<br>IEC_LTIMER,<br>TP_TIME,<br>TP_LTIME | D, L                            | IEC-Zeit, die gestartet wird.          |

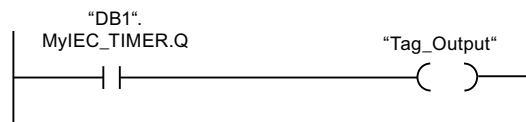
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Anweisung "Zeit als Impuls starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag\_Input" von "0" auf "1" wechselt. Die Zeit "DB1".MyIEC\_TIMER wird für die Zeitdauer gestartet, die im Operanden "TagTime" gespeichert ist.



Solange die Zeit "DB1".MyIEC\_TIMER läuft, liefert der Zeitstatus ("DB1".MyIEC\_TIMER.Q) den Signalzustand "1" und der Operand "Tag\_Output" wird gesetzt. Wenn die IEC-Zeit abgelaufen ist, wechselt der Signalzustand des Zeitstatus auf "0" und der Operand "Tag\_Output" wird zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ---( TON )---: Zeit als Einschaltverzögerung starten

### Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Einschaltverzögerung. Die IEC-Zeit wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) wechselt. Die IEC-Zeit läuft mit der angegebenen Zeitdauer ab. Der Ausgang liefert den Signalzustand "1", wenn das

VKE am Eingang der Anweisung den Signalzustand "1" führt. Wenn das VKE vor dem Ablauf der Zeit auf "0" wechselt, wird die laufende IEC-Zeit zurückgesetzt. Die Abfrage des Zeitstatus auf "1" liefert in diesem Fall den Signalzustand "0". Mit der Erfassung der nächsten positiven Signalfanke am Eingang der Anweisung startet die IEC-Zeit wieder.

---

#### Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC\_TIMER-Struktur aktualisiert.

---

### Für CPU S7-1200

Die Anweisung "Zeit als Einschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER oder TON\_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### Für CPU S7-1500

Die Anweisung "Zeit als Einschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER, IEC\_LTIMER, TON\_TIME oder TON\_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME, TON\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe eines Schließerkontakts auf "1" oder eines Öffnerkontakts auf "0" abfragen. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

Die Ausführung der Anweisung "Zeit als Einschaltverzögerung starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

## Parameter

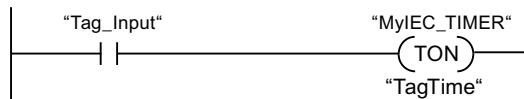
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

| Parameter   | Deklaration | Datentyp               |   | Speicherbereich                 | Beschreibung                           |
|-------------|-------------|------------------------|---|---------------------------------|--|
|             |             | S7-1200                | S7-1500   |                                 |  |
| <Zeitdauer> | Input       | TIME                   | TIME, LTIME   | E, A, M, D, L<br>oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft. |
| <IEC-Zeit>  | InOut       | IEC_TIMER,<br>TON_TIME | IEC_TIMER,<br>IEC_LTIMER,<br>TON_TIME,<br>TON_LTIME | D, L                            | IEC-Zeit, die gestartet wird.          |

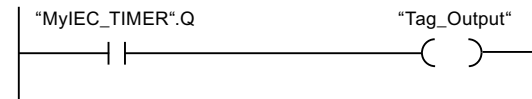
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Anweisung "Zeit als Einschaltverzögerung starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag\_Input" von "0" auf "1" wechselt. Die Zeit "MyIEC\_TIMER" wird für die Zeitdauer gestartet, die im Operanden "TagTime" gespeichert ist.



Wenn die Zeit "MyIEC\_TIMER" abgelaufen ist und der Operand "Tag\_Input" den Signalzustand "1" führt, liefert die Abfrage des Zeitstatus ("MyIEC\_TIMER".Q) den Signalzustand "1" und der Operand "Tag\_Output" wird gesetzt. Wenn der Signalzustand des Operanden "Tag\_Input" auf "0" wechselt, liefert die Abfrage des Zeitstatus den Signalzustand "0" und der Operand "Tag\_Output" wird zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ---( TOF )---: Zeit als Ausschaltverzögerung starten

## Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Ausschaltverzögerung. Die Abfrage des Zeitstatus auf "1" liefert den Signalzustand "1", wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung den

Signalzustand "0" führt. Wenn das VKE von "1" auf "0" (negative Signalfanke) wechselt, wird die IEC-Zeit mit der angegebenen Zeitdauer gestartet. Solange die IEC-Zeit läuft, bleibt der Zeitstatus auf den Signalzustand "1". Wenn die Zeit abgelaufen ist und das VKE am Eingang der Anweisung den Signalzustand "0" führt, wird der Zeitstatus auf den Signalzustand "0" gesetzt. Wechselt das VKE vor dem Ablauf der Zeit auf "1", wird die laufende IEC-Zeit zurückgesetzt und der Zeitstatus bleibt auf dem Signalzustand "1".

---

#### Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablaufebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC\_TIMER-Struktur aktualisiert.

---

### Für CPU S7-1200

Die Anweisung "Zeit als Ausschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER oder TOF\_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### Für CPU S7-1500

Die Anweisung "Zeit als Ausschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER, IEC\_LTIMER, TOF\_TIME oder TOF\_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME, TOF\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe eines Schließerkontakts auf "1" oder eines Öffnerkontakts auf "0" abfragen. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

Die Ausführung der Anweisung "Zeit als Ausschaltverzögerung starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

## Parameter

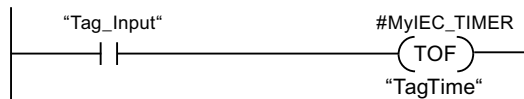
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

| Parameter   | Deklaration | Datentyp               |   | Speicherbereich                 | Beschreibung                           |
|-------------|-------------|------------------------|---|---------------------------------|--|
|             |             | S7-1200                | S7-1500   |                                 |  |
| <Zeitdauer> | Input       | TIME                   | TIME, LTIME   | E, A, M, D, L<br>oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft. |
| <IEC-Zeit>  | InOut       | IEC_TIMER,<br>TOF_TIME | IEC_TIMER,<br>IEC_LTIMER,<br>TOF_TIME,<br>TOF_LTIME | D, L                            | IEC-Zeit, die gestartet wird.          |

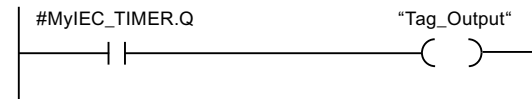
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Anweisung "Zeit als Ausschaltverzögerung starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag\_Input" von "1" auf "0" wechselt. Die Zeit #MyIEC\_TIMER wird für die Zeitdauer gestartet, die im Operanden "TagTime" gespeichert ist.



Solange die Zeit #MyIEC\_TIMER läuft, liefert die Abfrage des Zeitstatus (#MyIEC\_TIMER.Q) den Signalzustand "1" und der Operand "Tag\_Output" wird gesetzt. Wenn die Zeit abgelaufen ist und der Operand "Tag\_Input" den Signalzustand "0" führt, liefert die Abfrage des Zeitstatus den Signalzustand "0". Wechselt der Signalzustand des Operanden "Tag\_Input" vor dem Ablauf der Zeit #MyIEC\_TIMER auf "1", wird die Zeit zurückgesetzt. Beim Signalzustand "1" des Operanden "Tag\_Input" liefert die Abfrage des Zeitstatus den Signalzustand "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ---( TONR )---: Zeit akkumulieren

### Beschreibung

Mit der Anweisung "Zeit akkumulieren" erfassen Sie, wie lange das Signal am Eingang der Anweisung "1" ist. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) von

"0" auf "1" (positive Signalflanke) wechselt. Solange das VKE "1" ist, wird die Zeit erfasst. Wechselt das VKE auf "0", wird die Anweisung angehalten. Wechselt das VKE erneut auf "1", wird die Zeiterfassung fortgesetzt. Die Abfrage des Zeitstatus auf "1" liefert den Signalzustand "1", wenn die erfasste Zeit den Wert der angegebenen Zeitdauer überschreitet und das VKE am Eingang der Spule "1" führt.

Der Zeitstatus und die aktuell abgelaufene Zeit können durch die Anweisung "Zeit rücksetzen" auf "0" zurückgesetzt werden.

---

#### Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablauebene befinden, da jede Abfrage der Ausgänge Q oder ET die IEC\_TIMER-Struktur aktualisiert.

---

### Für CPU S7-1200

Die Anweisung "Zeit akkumulieren" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER oder TONR\_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### Für CPU S7-1500

Die Anweisung "Zeit akkumulieren" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER, IEC\_LTIMER, TONR\_TIME oder TONR\_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME, TONR\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe eines Schließerkontakts auf "1" oder eines Öffnerkontakts auf "0" abfragen. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

| Parameter   | Deklaration | Datentyp                |   | Speicherbereich                 | Beschreibung                           |
|-------------|-------------|-------------------------|---|---------------------------------|--|
|             |             | S7-1200                 | S7-1500   |                                 |  |
| <Zeitdauer> | Input       | TIME                    | TIME, LTIME   | E, A, M, D, L<br>oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft. |
| <IEC-Zeit>  | InOut       | IEC_TIMER,<br>TONR_TIME | IEC_TIMER,<br>IEC_LTIMER,<br>TONR_TIME,<br>TONR_LTIME | D, L                            | IEC-Zeit, die gestartet wird.          |

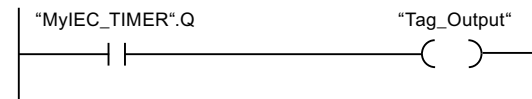
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Anweisung "Zeit akkumulieren" wird bei einer positiven Signalflanke im VKE ausgeführt. Solange der Operand "Tag\_Input" den Signalzustand "1" liefert, wird die Zeit erfasst.



Wenn die erfasste Zeit den Wert des Operanden "TagTime" überschreitet, liefert die Abfrage des Zeitstatus ("MyIEC\_TIMER".Q) den Signalzustand "1" und der Operand "Tag\_Output" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ---( RT )---: Zeit rücksetzen

### Beschreibung

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine IEC-Zeit auf "0" zurück. Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Spule "1" ist. Fließt Strom in die Spule (VKE ist "1"), werden die Strukturkomponenten der Zeit im angegebenen Datenbaustein auf "0" zurückgesetzt. Wenn das VKE am Eingang der Anweisung "0" ist, bleibt die Zeit unverändert.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Eingang der Spule wird direkt auf den Ausgang der Spule übertragen.

Der Anweisung "Zeit rücksetzen" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung und nicht bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage der Daten ist nur gleich vom Aufruf der Anweisung bis zum nächsten Aufruf der Anweisung.

### Parameter

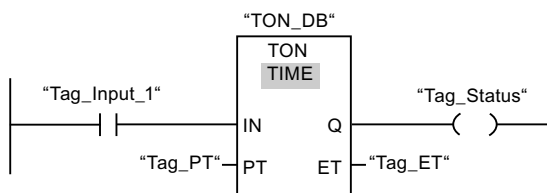
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit rücksetzen":

| Parameter  | Deklaration | Datentyp  |  | Speicherbereich | Beschreibung                      |
|------------|-------------|---|--|-----------------|-----------------------------------|
|            |             | S7-1200   | S7-1500  |                 |                                   |
| <IEC-Zeit> | Output      | IEC_TIMER,<br>TP_TIME,<br>TON_TIME,<br>TOF_TIME,<br>TONR_TIME | IEC_TIMER,<br>IEC_LTIMER,<br>TP_TIME,<br>TP_LTIME,<br>TON_TIME,<br>TON_LTIME,<br>TOF_TIME,<br>TOF_LTIME,<br>TONR_TIME,<br>TONR_LTIME | D, L            | IEC-Zeit, die zurückgesetzt wird. |

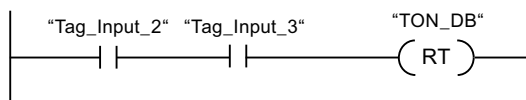
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "Tag\_Input\_1" von "0" auf "1" wechselt, wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON\_DB" abgelegte Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag\_PT" vorgegeben ist.



Wenn die Operanden "Tag\_Input\_2" und "Tag\_Input\_3" den Signalzustand "1" liefern, wird die Anweisung "Zeit rücksetzen" ausgeführt und die im Datenbaustein "TON\_DB" abgelegte Zeit zurückgesetzt.



**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**---( PT )---: Zeitdauer laden****Beschreibung**

Mit der Anweisung "Zeitdauer laden" stellen Sie die Zeitdauer einer IEC-Zeit ein. Die Anweisung wird in jedem Zyklus ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung den Signalzustand "1" führt. Die Anweisung schreibt die angegebene Zeitdauer in die Struktur der angegebenen IEC-Zeit.

**Hinweis**

Wenn die angegebene IEC-Zeit während der Ausführung der Anweisung läuft, überschreibt die Anweisung die aktuelle Zeitdauer der angegebenen IEC-Zeit. Dadurch kann sich der Zeitstatus der IEC-Zeit ändern.

Der Anweisung "Zeitdauer laden" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht bei einem Aufruf der Anweisung und bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

**Parameter**

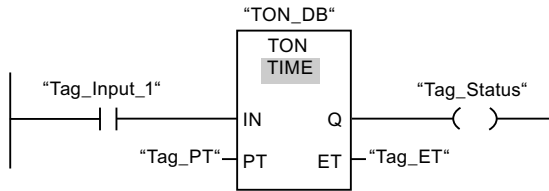
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitdauer laden":

| Parameter   | Deklaration | Datentyp  |  | Speicherbereich                 | Beschreibung                            |
|-------------|-------------|---|--|---------------------------------|---|
|             |             | S7-1200   | S7-1500  |                                 |   |
| <Zeitdauer> | Input       | TIME  | TIME, LTIME  | E, A, M, D, L<br>oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft.  |
| <IEC-Zeit>  | Output      | IEC_TIMER,<br>TP_TIME,<br>TON_TIME,<br>TOF_TIME,<br>TONR_TIME | IEC_TIMER,<br>IEC_LTIMER,<br>TP_TIME,<br>TP_LTIME,<br>TON_TIME,<br>TON_LTIME,<br>TOF_TIME,<br>TOF_LTIME,<br>TONR_TIME,<br>TONR_LTIME | D, L                            | IEC-Zeit, deren Dauer eingestellt wird. |

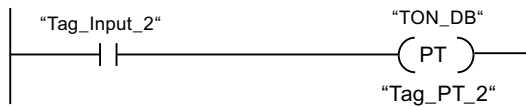
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "Tag\_Input\_1" von "0" auf "1" wechselt, wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON\_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag\_PT" vorgegeben ist.



Wenn der Operand "Tag\_Input\_2" den Signalzustand "1" führt, wird die Anweisung "Zeitdauer laden" ausgeführt. Die Anweisung schreibt die Zeitdauer "Tag\_PT\_2" in den Instanz-Datenbaustein "TON\_DB" und überschreibt dabei den Wert des Operanden "Tag\_PT" innerhalb des Datenbausteins. Dadurch kann sich der Signalzustand des Zeitstatus bei der nächsten Abfrage bzw. beim Zugriff auf "MyTimer".Q oder "MyTimer".ET ändern.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**SIMATIC-Zeiten**

**S\_IMPULS: Zeit als Impuls parametrieren und starten**

**Beschreibung**

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn der Signalzustand am Eingang S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten. In diesem Fall ist der Signalzustand am Ausgang Q "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als Impuls parametrieren und starten" benötigt für die Flankenbewertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusbeginn andere Werte als am Zyklusende liefert.

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

## Parameter

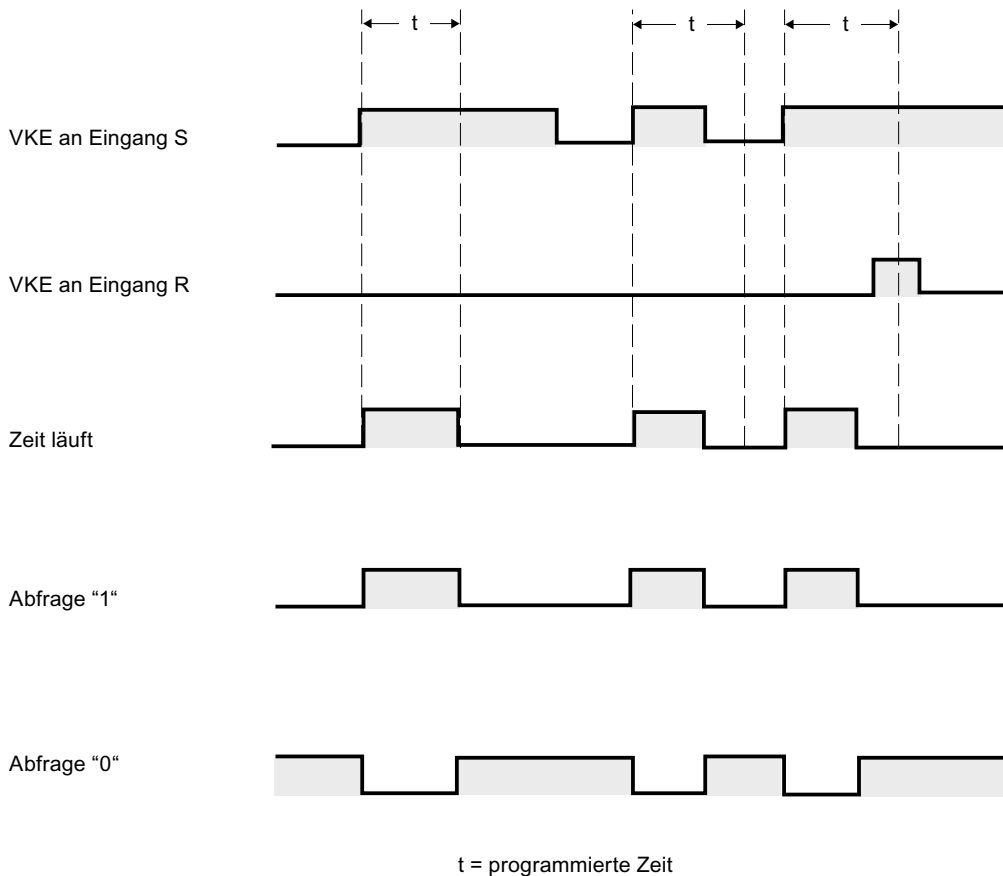
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, D, L                         | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Voreingestellter Zeitwert   |
| R         | Input       | BOOL         | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L                         | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

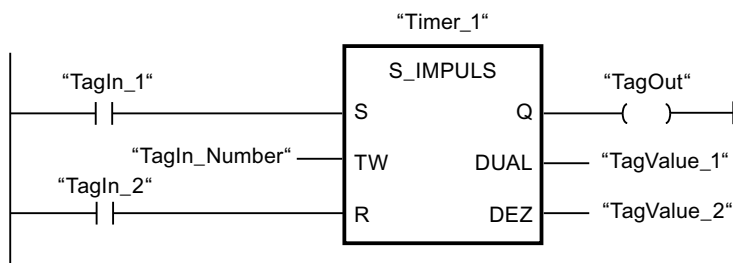
### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":



### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, solange der Operand "TagIn\_1" den Signalzustand "1" liefert. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit "Timer\_1" angehalten. Der Operand "TagOut" wird in diesem Fall auf "0" zurückgesetzt.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft und der Operand "TagIn\_1" den Signalzustand "1" hat. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## S\_VIMP: Zeit als verlängerten Impuls parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Solange die Zeit läuft, liefert der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Ausgang Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der am Eingang TW programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als verlängerten Impuls parametrieren und starten" benötigt für die Flankenbewertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusbeginn andere Werte als am Zyklusende liefert.

---

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

## Parameter

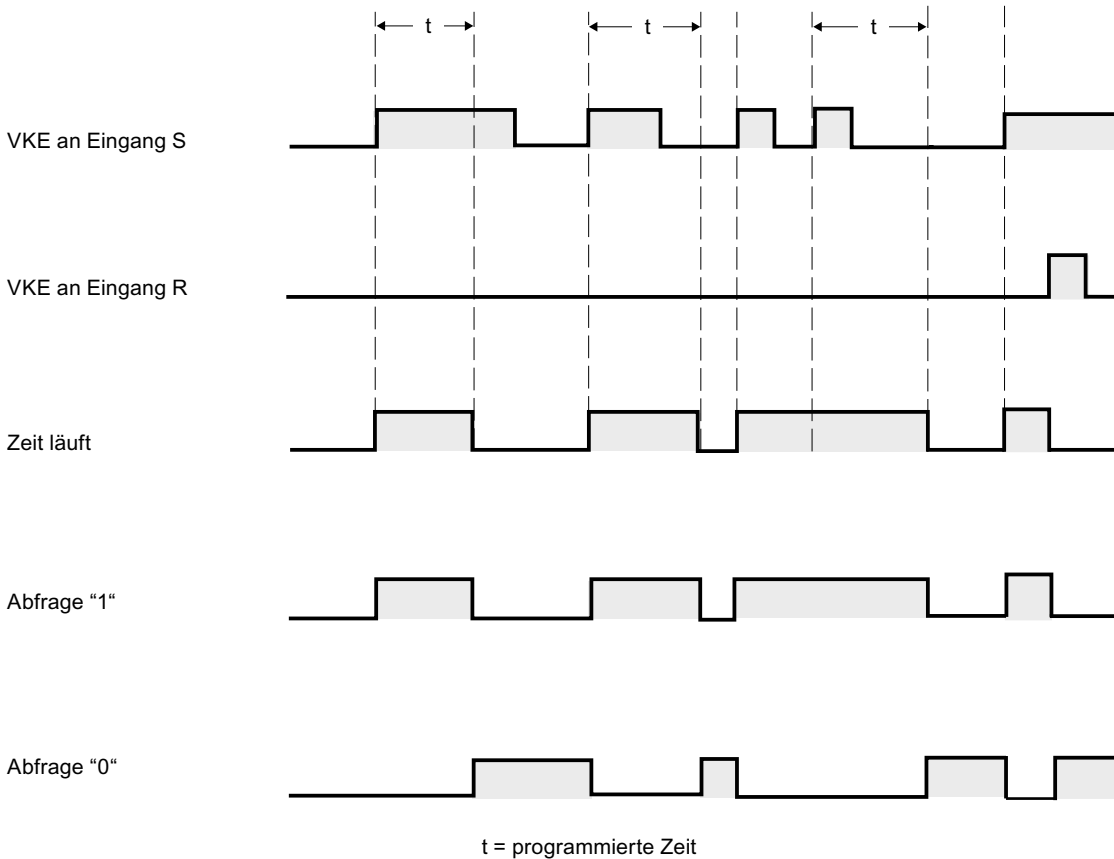
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, D, L                         | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Voreingestellter Zeitwert   |
| R         | Input       | BOOL         | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L                         | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

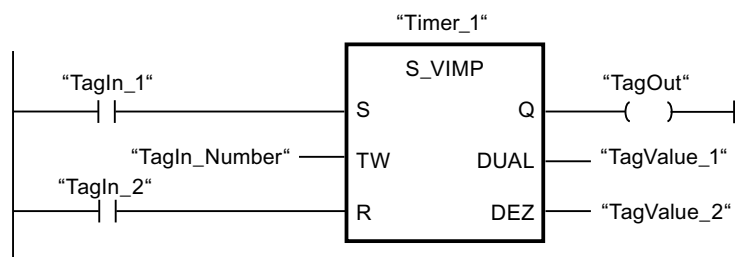
## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, ohne von einer fallenden Flanke am Eingang S beeinträchtigt zu werden. Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## S\_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn die Zeit ordnungsgemäß abgelaufen ist und am Eingang S der Signalzustand "1" noch anliegt, liefert der Ausgang Q den Signalzustand "1". Wechselt der Signalzustand am Eingang S von "1" auf "0", während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Ausgang Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Eingang R zurückgesetzt, auch wenn die Zeit nicht läuft und das VKE am Eingang S "1" ist.

Die Zeit der Anweisung geben Sie in den Platzhalter oberhalb der Box an. Die Zeit muss mit dem Datentyp TIMER deklariert sein.

Die Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

---

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".



## Parameter

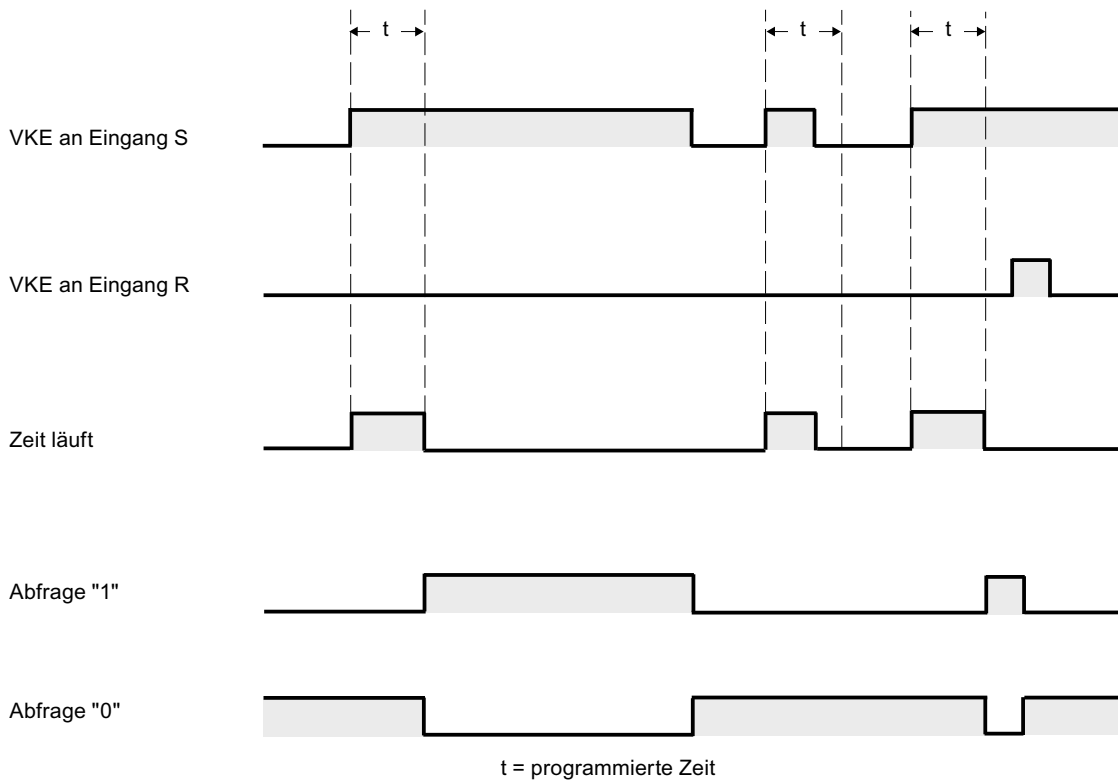
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, D, L                         | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Voreingestellter Zeitwert   |
| R         | Input       | BOOL         | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L                         | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

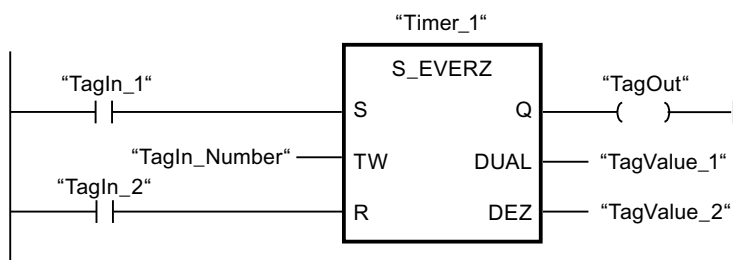
### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":



### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "1" ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Der Operand "TagOut" liefert in diesem Fall den Signalzustand "0".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**S\_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten****Beschreibung**

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Wenn die Zeit abgelaufen ist, liefert der Ausgang Q den Signalzustand "1" unabhängig vom Signalzustand am Eingang S. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer (TW) neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Starteingang S. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

---

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

### Parameter

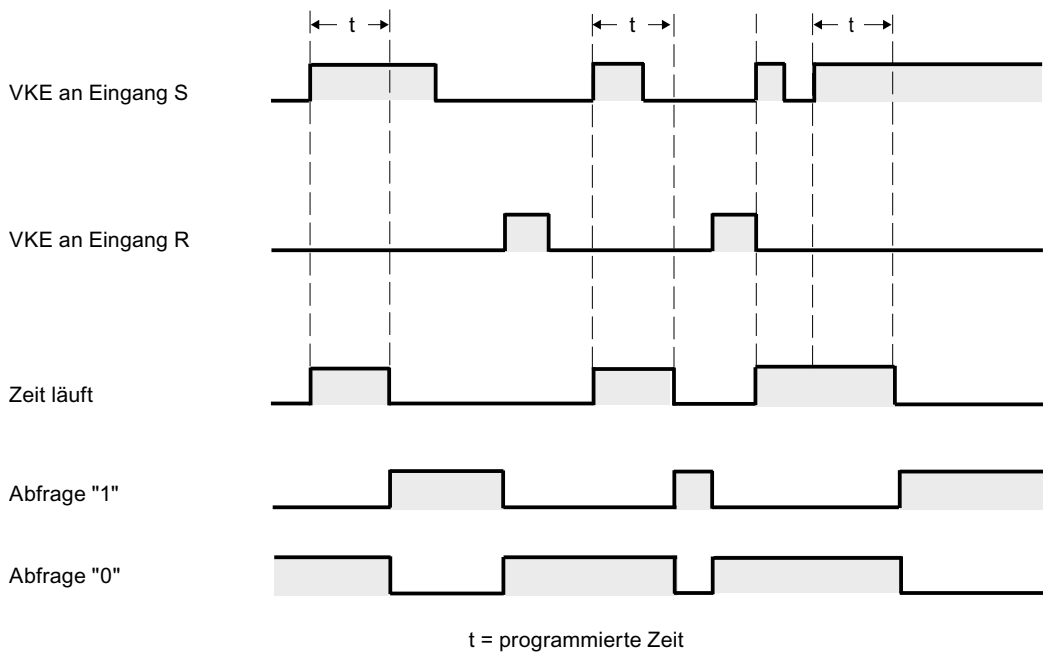
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, D, L                         | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Voreingestellter Zeitwert   |
| R         | Input       | BOOL         | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L                         | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

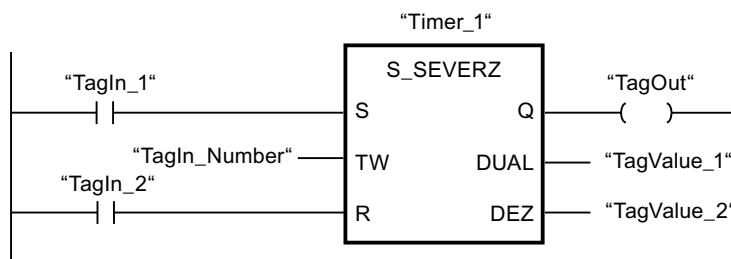
### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, auch wenn der Signalzustand des Operanden "TagIn\_1" auf "0" wechselt. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## S\_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab. Solange die Zeit läuft oder der Eingang S den Signalzustand "1" liefert, führt der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist und der Signalzustand am Eingang S "0" ist, wird der Ausgang Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit angehalten. Erst wenn eine negative Signalflanke am Eingang S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" benötigt für die Flankenbewertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

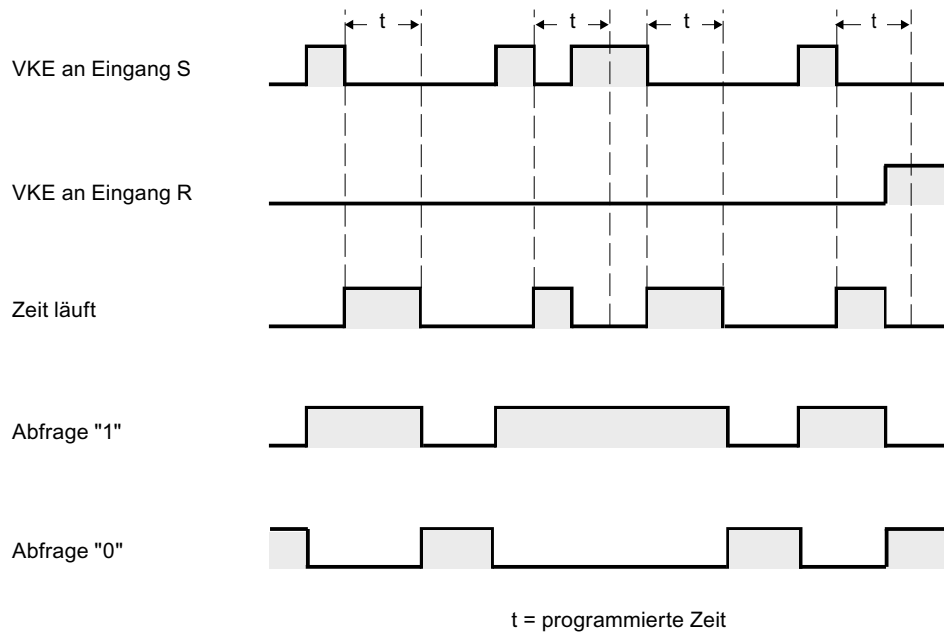
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, D, L                         | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Voreingestellter Zeitwert   |
| R         | Input       | BOOL         | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L                         | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

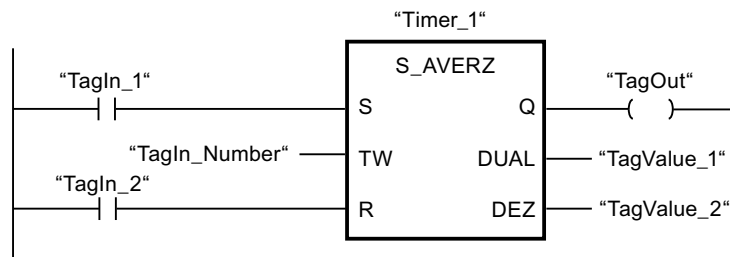
## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "1" auf "0" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Der Operand "TagOut" wird auf "1" gesetzt, wenn die Zeit läuft oder der Operand "TagIn\_1" den Signalzustand "0" liefert. Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", während die Zeit läuft, wird die Zeit zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

---( SI ): Zeit als Impuls starten

**Beschreibung**

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft solange mit der angegebenen Zeitdauer, wie das VKE den Signalzustand "1" liefert. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "1" auf "0", bevor der Zeitwert abgelaufen ist, wird die Zeit angehalten. In diesem Fall ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

| Parameter   | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-------------|-------------|--------------|------------------------------|--|
| <Zeitdauer> | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer, mit der die Zeit abläuft.   |
| <Zeit>      | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".





Die Anweisung "Zeit als verlängerten Impuls starten" benötigt für die Flankenbewertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

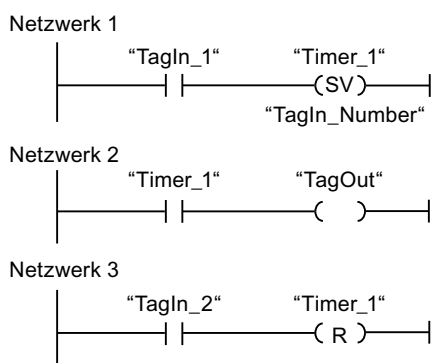
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls starten":

| Parameter   | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-------------|-------------|--------------|------------------------------|--|
| <Zeitdauer> | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer, mit der die Zeit abläuft.   |
| <Zeit>      | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, ohne von einer negativen Flanke im VKE beeinflusst zu werden. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**---( SE ): Zeit als Einschaltverzögerung starten****Beschreibung**

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, solange das VKE "1" ist. Wenn die Zeit abgelaufen ist und das VKE noch den Signalzustand "1" hat, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn das VKE von "1" auf "0" wechselt, während die Zeit läuft, wird die Zeit angehalten. Die Abfrage des Zeitstatus auf "1" ergibt in diesem Fall den Signalzustand "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Einschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

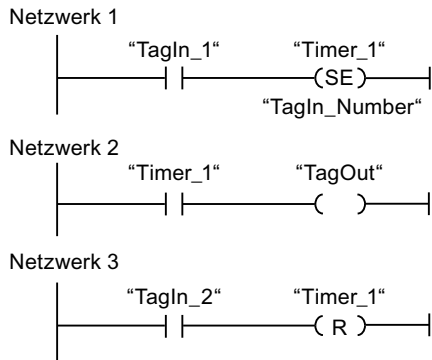
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

| Parameter   | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-------------|-------------|--------------|------------------------------|--|
| <Zeitdauer> | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer, mit der die Zeit abläuft.   |
| <Zeit>      | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Wenn die Zeit abgelaufen ist und das VKE den Signalzustand "1" hat, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Wenn der Signalzustand des Operanden "TagIn\_2" "1" ist, dann wird die Zeit "Timer\_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**---( SS ): Zeit als speichernde Einschaltverzögerung starten**

**Beschreibung**

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Wenn die Zeit abgelaufen ist, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Nach dem Ablauf der Zeit ist ein Neustart der Zeit erst dann möglich, wenn diese explizit zurückgesetzt wurde.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis an. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als speichernde Einschaltverzögerung starten" benötigt für die Flankenbewertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

## Parameter

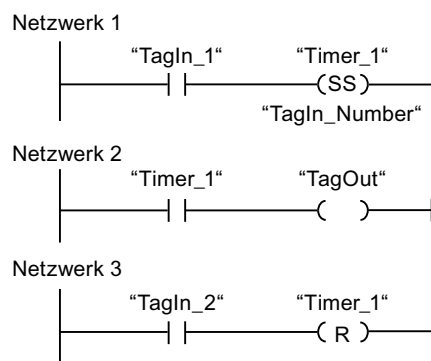
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung starten":

| Parameter   | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-------------|-------------|--------------|------------------------------|--|
| <Zeitdauer> | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer, mit der die Zeit abläuft.   |
| <Zeit>      | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet.

Wenn der Signalzustand des Operanden "TagIn\_2" "1" ist, dann wird die Zeit "Timer\_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**---( SA ): Zeit als Ausschaltverzögerung starten**

**Beschreibung**

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (Negative Signalfanke) erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn das VKE von "0" auf "1" wechselt, während die Zeit läuft, wird die Zeit zurückgesetzt. Die Zeit wird immer dann neu gestartet, wenn das VKE von "1" auf "0" wechselt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Ausschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

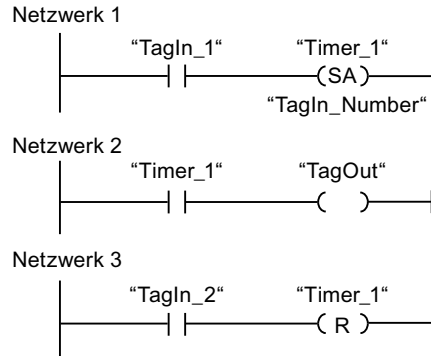
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

| Parameter   | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-------------|-------------|--------------|------------------------------|--|
| <Zeitdauer> | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer, mit der die Zeit abläuft.   |
| <Zeit>      | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "1" auf "0" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Solange die Zeit läuft, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn\_2" "1" ist, dann wird die Zeit "Timer\_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Zähler

### IEC-Zähler

### CTU: Vorwärts zählen

## Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Ausgang CV hoch. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins erhöht. Bei der ersten Ausführung der Anweisung ist der aktuelle Zählwert am Ausgang CV auf Null gestellt. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Eingang CU keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Der Signalzustand am Ausgang Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0".

Der Wert am Ausgang CV wird auf Null zurückgesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CU keine Wirkung auf die Anweisung.

**Hinweis**

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

**Für CPU S7-1200**

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTU_SINT / CTU_USINT</li> <li>• CTU_INT / CTU_UINT</li> <li>• CTU_DINT / CTU_UDINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> |

**Für CPU S7-1500**

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTU_SINT / CTU_USINT</li> <li>• CTU_INT / CTU_UINT</li> <li>• CTU_DINT / CTU_UDINT</li> <li>• CTU_LINT / CTU_ULINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTU oder IEC\_COUNTER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der



Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Ausführung der Anweisung "Vorwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

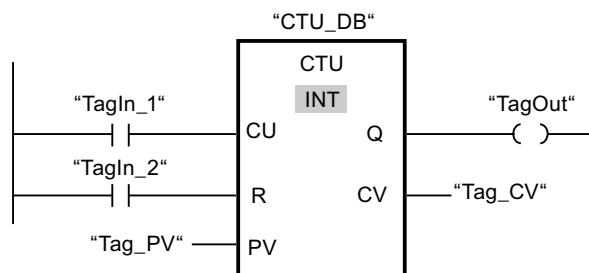
| Parameter | Deklaration | Datentyp               | Speicherbereich                       | Beschreibung                              |
|-----------|-------------|------------------------|---------------------------------------|---|
| CU        | Input       | BOOL                   | E, A, M, D, L oder Konstante          | Zähleingang                               |
| R         | Input       | BOOL                   | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang                           |
| PV        | Input       | Ganzzahlen             | E, A, M, D, L, P oder Konstante       | Wert, bei dem der Ausgang Q gesetzt wird. |
| Q         | Output      | BOOL                   | E, A, M, D, L                         | Zählerstatus                              |
| CV        | Output      | Ganzzahlen, CHAR, DATE | E, A, M, D, L, P                      | Aktueller Zählwert                        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag\_CV" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des Datentyps (INT = 32767) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "TagOut" übernommen. Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag\_PV" ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**CTD: Rückwärts zählen**

**Beschreibung**

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Ausgang CV runter. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins verringert. Bei der ersten Ausführung der Anweisung wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Eingang CD keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0".

Der Wert am Ausgang CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Eingang LD auf "1" wechselt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CD keine Wirkung auf die Anweisung.

**Hinweis**

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

**Für CPU S7-1200**

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTD_SINT / CTD_USINT</li> <li>• CTD_INT / CTD_UINT</li> <li>• CTD_DINT / CTD_UDINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> |

## Für CPU S7-1500

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTD_SINT / CTD_USINT</li> <li>• CTD_INT / CTD_UINT</li> <li>• CTD_DINT / CTD_UDINT</li> <li>• CTD_LINT / CTD_ULINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTD oder IEC\_COUNTER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelnanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Ausführung der Anweisung "Rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

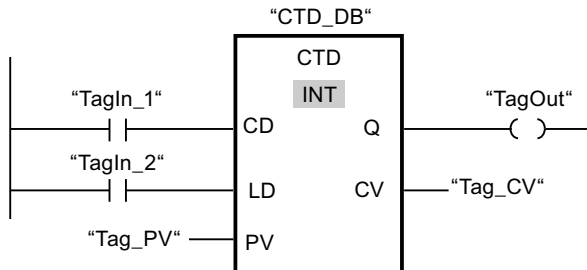
| Parameter | Deklaration | Datentyp               | Speicherbereich                       | Beschreibung                              |
|-----------|-------------|------------------------|---------------------------------------|---|
| CD        | Input       | BOOL                   | E, A, M, D, L oder Konstante          | Zähleingang                               |
| LD        | Input       | BOOL                   | E, A, M, D, L, T, Z, P oder Konstante | Ladeingang                                |
| PV        | Input       | Ganzzahlen             | E, A, M, D, L, P oder Konstante       | Wert, bei dem der Ausgang Q gesetzt wird. |
| Q         | Output      | BOOL                   | E, A, M, D, L                         | Zählerstatus                              |
| CV        | Output      | Ganzzahlen, CHAR, DATE | E, A, M, D, L, P                      | Aktueller Zählwert                        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Anweisung "Rückwärts zählen" ausgeführt und der Wert am Ausgang "Tag\_CV" um eins verringert. Bei jeder weiteren positiven Signalfanke wird der Zählwert verringert, bis der untere Grenzwert des angegebenen Datentyps (INT = -32768) erreicht ist.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**CTUD: Vorwärts und rückwärts zählen**

**Beschreibung**

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Ausgang CV hoch und runter. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalfanke), wird der aktuelle Zählwert um eins erhöht und am Ausgang CV abgelegt. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalfanke), wird der Zählwert am Ausgang CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalfanke vorliegt, bleibt der aktuelle Zählwert am Ausgang CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalfanke nicht mehr hoch gezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang LD auf "1" wechselt, wird der Zählwert am Ausgang CV auf den Wert des Parameters PV gesetzt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Eingängen CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Ausgang QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QU "0".

Am Ausgang QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QD "0".

---

### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

### Für CPU S7-1200

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTUD_SINT / CTUD_USINT</li> <li>• CTUD_INT / CTUD_UINT</li> <li>• CTUD_DINT / CTUD_UDINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> |

### Für CPU S7-1500

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTUD_SINT / CTUD_USINT</li> <li>• CTUD_INT / CTUD_UINT</li> <li>• CTUD_DINT / CTUD_UDINT</li> <li>• CTUD_LINT / CTUD_ULINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTUD oder IEC\_COUNTER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelninstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Ausführung der Anweisung "Vorwärts und rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts und rückwärts zählen":

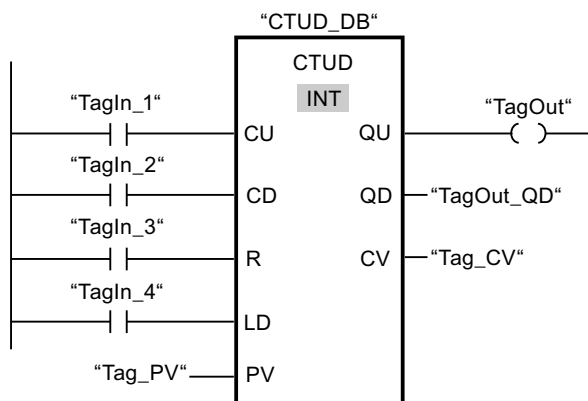
| Parameter | Deklaration | Datentyp               | Speicherbereich                       | Beschreibung                               |
|-----------|-------------|------------------------|---------------------------------------|--|
| CU        | Input       | BOOL                   | E, A, M, D, L oder Konstante          | Vorwärtszähleingang                        |
| CD        | Input       | BOOL                   | E, A, M, D, L oder Konstante          | Rückwärtszähleingang                       |
| R         | Input       | BOOL                   | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang                            |
| LD        | Input       | BOOL                   | E, A, M, D, L, T, Z, P oder Konstante | Ladeeingang                                |
| PV        | Input       | Ganzzahlen             | E, A, M, D, L, P oder Konstante       | Wert, bei dem der Ausgang QU gesetzt wird. |
| QU        | Output      | BOOL                   | E, A, M, D, L                         | Status des Vorwärtszählers                 |
| QD        | Output      | BOOL                   | E, A, M, D, L                         | Status des Rückwärtszählers                |
| CV        | Output      | Ganzzahlen, CHAR, DATE | E, A, M, D, L, P                      | Aktueller Zählwert                         |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand am Eingang "TagIn\_1" oder am Eingang "TagIn\_2" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Vorwärts und rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn\_1" vorliegt, wird der aktuelle Zählwert um eins erhöht und am Ausgang "Tag\_CV" abgelegt. Wenn eine positive Signalflanke am Eingang "TagIn\_2" vorliegt, wird der Zählwert um eins verringert und am Ausgang "Tag\_CV" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Eingang CU so lange erhöht, bis er den oberen Grenzwert von 32767 erreicht. Bei einer positiven Signalflanke am Eingang CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert von INT = -32768 erreicht.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert am Eingang "Tag\_PV" ist. In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

Der Ausgang "TagOut\_QD" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "TagOut\_QD" den Signalzustand "0".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## SIMATIC-Zähler

### ZAEHLER: Parametrieren und vorwärts/rückwärts zählen

## Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ

BCD-codiert ausgegeben. Wenn in einem Programmzyklus an den Eingängen ZV und ZR eine positive Signalfanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalfanke nicht mehr inkrementiert. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE an den Eingängen ZV und ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV, ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

**Hinweis**

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und vorwärts/rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts/rückwärts zählen":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung  |
|-----------|-------------|----------|------------------------------------|---|
| <Zähler>  | InOut/Input | COUNTER  | Z                                  | Zähler der Anweisung<br>Die Anzahl der Zähler ist von der CPU abhängig. |
| ZV        | Input       | BOOL     | E, A, M, D, L                      | Vorwärtszähleingang   |
| ZR        | Input       | BOOL     | E, A, M, D, L, T, Z oder Konstante | Rückwärtszähleingang  |
| S         | Input       | BOOL     | E, A, M, D, L, T, Z oder Konstante | Eingang zum Voreinstellen des Zählers                                   |
| ZW        | Input       | WORD     | E, A, M, D, L, Z oder Konstante    | Voreingestellter Zählwert (C#0 bis C#999)                               |
| R         | Input       | BOOL     | E, A, M, D, L, T, Z oder Konstante | Rücksetzeingang   |

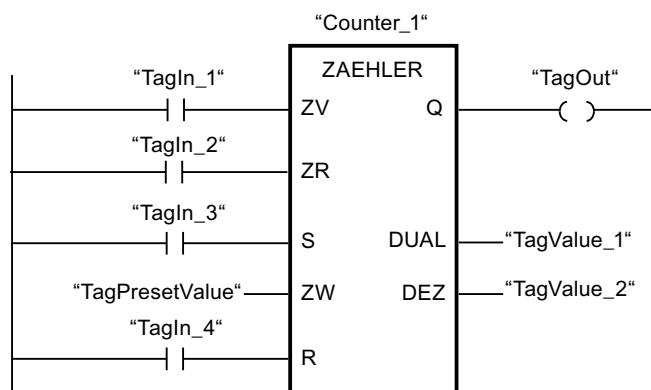


| Parameter | Deklaration | Datentyp           | Speicherbereich | Beschreibung                     |
|-----------|-------------|--------------------|-----------------|----------------------------------|
| DUAL      | Output      | WORD, S5TIME, DATE | E, A, M, D, L   | Aktueller Zählwert (Hexadezimal) |
| DEZ       | Output      | WORD, S5TIME, DATE | E, A, M, D, L   | Aktueller Zählwert (BCD-Format)  |
| Q         | Output      | BOOL               | E, A, M, D, L   | Status des Zählers               |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand am Eingang "TagIn\_1" oder am Eingang "TagIn\_2" von "0" auf "1" wechselt (positive Signalfanke), wird die Anweisung "Parametrieren und vorwärts/rückwärts zählen" ausgeführt. Wenn eine positive Signalfanke am Eingang "TagIn\_1" vorliegt und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalfanke am Eingang "TagIn\_2" vorliegt und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand am Eingang "TagIn\_3" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn\_4" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue\_1" und BCD-codiert im Operanden "TagValue\_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Z\_VORW: Parametrieren und vorwärts zählen

### Beschreibung

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZV "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

---

### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

Die Anweisung "Parametrieren und vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts zählen":

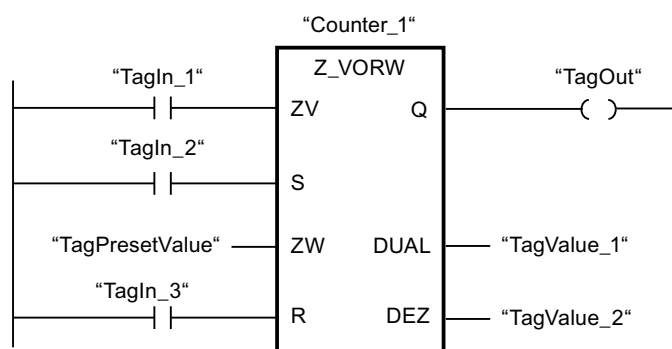
| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung  |
|-----------|-------------|----------|------------------------------------|---|
| <Zähler>  | InOut/Input | COUNTER  | Z                                  | Zähler der Anweisung<br>Die Anzahl der Zähler ist von der CPU abhängig. |
| ZV        | Input       | BOOL     | E, A, M, D, L                      | Vorwärtszähleingang   |
| S         | Input       | BOOL     | E, A, M, D, L, T, Z oder Konstante | Eingang zum Voreinstellen des Zählers                                   |
| ZW        | Input       | WORD     | E, A, M, D, L, Z oder Konstante    | Voreingestellter Zählwert (C#0 bis C#999)                               |
| R         | Input       | BOOL     | E, A, M, D, L, T, Z oder Konstante | Rücksetzeingang   |

| Parameter | Deklaration | Datentyp           | Speicherbereich | Beschreibung                     |
|-----------|-------------|--------------------|-----------------|----------------------------------|
| DUAL      | Output      | WORD, S5TIME, DATE | E, A, M, D, L   | Aktueller Zählwert (Hexadezimal) |
| DEZ       | Output      | WORD, S5TIME, DATE | E, A, M, D, L   | Aktueller Zählwert (BCD-Format)  |
| Q         | Output      | BOOL               | E, A, M, D, L   | Status des Zählers               |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand am Eingang "TagIn\_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "TagIn\_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn\_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue\_1" und BCD-codiert im Operanden "TagValue\_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Z\_RUECK: Parametrieren und rückwärts zählen

### Beschreibung

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

**Hinweis**

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

**Parameter**

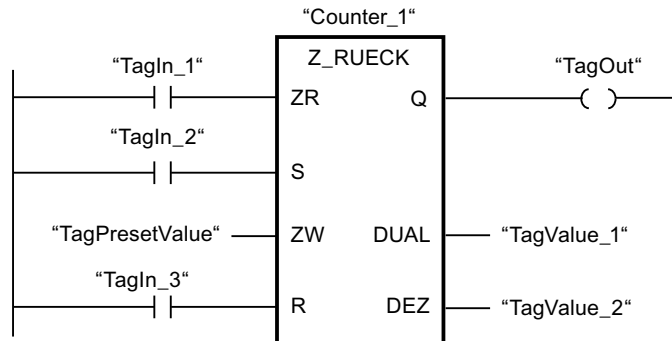
Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und rückwärts zählen":

| Parameter | Deklaration | Datentyp           | Speicherbereich                    | Beschreibung  |
|-----------|-------------|--------------------|------------------------------------|---|
| <Zähler>  | InOut/Input | COUNTER            | Z                                  | Zähler der Anweisung<br>Die Anzahl der Zähler ist von der CPU abhängig. |
| ZR        | Input       | BOOL               | E, A, M, D, L oder Konstante       | Rückwärtszähleingang  |
| S         | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Eingang zum Voreinstellen des Zählers                                   |
| ZW        | Input       | WORD               | E, A, M, D, L, Z oder Konstante    | Voreingestellter Zählwert (C#0 bis C#999)                               |
| R         | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD, S5TIME, DATE | E, A, M, D, L                      | Aktueller Zählwert (Hexadezimal)  |
| DEZ       | Output      | WORD, S5TIME, DATE | E, A, M, D, L                      | Aktueller Zählwert (BCD-Format)   |
| Q         | Output      | BOOL               | E, A, M, D, L                      | Status des Zählers  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand am Eingang "TagIn\_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand am Eingang "TagIn\_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn\_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue\_1" und BCD-codiert im Operanden "TagValue\_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ---( SZ ): Zähleranfangswert setzen

### Beschreibung

Mit der Anweisung "Zähleranfangswert setzen" stellen Sie den Wert eines Zählers ein. Die Anweisung wird ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung von "0" auf "1" wechselt. Wenn die Anweisung ausgeführt wird, wird der Zähler auf den angegebenen Zählwert gesetzt.

Die Anweisung "Zähleranfangswert setzen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

### Parameter

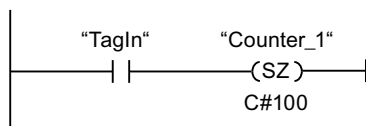
Die folgende Tabelle zeigt die Parameter der Anweisung "Zähleranfangswert setzen":

| Parameter  | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|------------|-------------|----------|------------------------------|---|
| <Zählwert> | Input       | WORD     | E, A, M, D, L oder Konstante | Wert, mit dem der Zähler im BCD-Format voreingestellt wird. (C#0 bis C#999) |
| <Zähler>   | InOut/Input | COUNTER  | Z                            | Zähler, der voreingestellt wird.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn" von "0" auf "1" wechselt, wird der Zähler "Counter\_1" mit dem Wert "100" voreingestellt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### ---( ZV ): Vorwärts zählen

#### Beschreibung

Mit der Anweisung "Vorwärts zählen" erhöhen Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke im Verknüpfungsergebnis (VKE) vorliegt. Der Zählwert kann so lange erhöht werden, bis der Grenzwert "999" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Die Anweisung "Vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

## Parameter

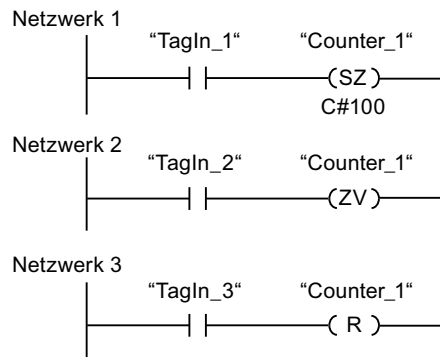
Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                     |
|-----------|-------------|----------|-----------------|----------------------------------|
| <Zähler>  | InOut/Input | COUNTER  | Z               | Zähler, dessen Wert erhöht wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" (positive Signalfanke), wird der Zähler "Counter\_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter\_1" wird um eins erhöht, wenn der Signalzustand des Operanden "TagIn\_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn\_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter\_1" auf "0" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ---( ZR ): Rückwärts zählen

### Beschreibung

Mit der Anweisung "Rückwärts zählen" verringern Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalfanke im Verknüpfungsergebnis (VKE) vorliegt. Der Zählwert kann so lange verringert werden, bis der Grenzwert "0" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalfanke nicht mehr verändert.

Die Anweisung "Rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Parameter**

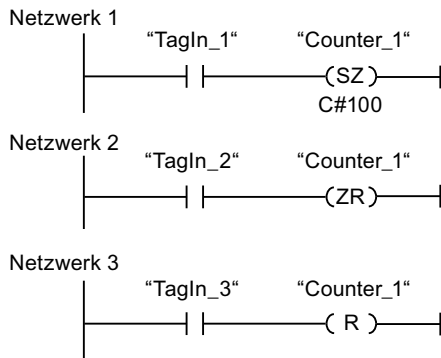
Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                         |
|-----------|-------------|----------|-----------------|--------------------------------------|
| <Zähler>  | InOut/Input | COUNTER  | Z               | Zähler, dessen Wert verringert wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter\_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter\_1" wird um eins verringert, wenn der Signalzustand des Operanden "TagIn\_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn\_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter\_1" auf "0" zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**Vergleicher**

**CMP ==: Gleich**

**Beschreibung**

Mit der Anweisung "Gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) gleich dem zweiten Vergleichswert (<Operand2>) ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die



Anweisung des VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| <Operand1>   | <Operand2>  | VKE der Anweisung |
|--------------|-------------|-------------------|
| 'AA'         | 'AA'        | 1                 |
| 'Hallo Welt' | 'HalloWelt' | 0                 |
| 'AA'         | 'aa'        | 0                 |

Mit der Anweisung "Gleich" können Sie auch einzelne Zeichen einer Zeichenkette (STRING) vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Bei aktivierter IEC-Prüfung müssen die zu vergleichenden Operanden vom gleichen Datentyp sein. Wenn die IEC-Prüfung nicht aktiviert ist, muss die Breite (Länge) der Operanden gleich sein. Bei Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

---

### Hinweis

#### Vergleich von Gleitpunktzahlen

Wenn Sie die Datentypen REAL oder LREAL vergleichen wollen, dann verwenden Sie statt der Anweisung "CMP ==: Gleich" die Anweisung "IN\_RANGE: Wert innerhalb Bereich".

---

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gleich":

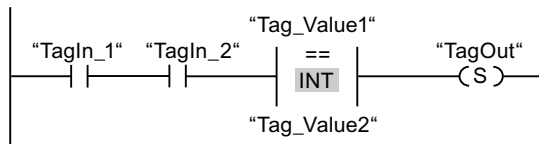
| Parameter  | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung           |
|------------|-------------|--|--|---------------------------------|------------------------|
|            |             | S7-1200  | S7-1500  |                                 |                        |
| <Operand1> | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| <Operand2> | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" = "Tag\_Value2" ist.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## CMP <>: Ungleich

### Beschreibung

Mit der Anweisung "Ungleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) ungleich dem zweiten Vergleichswert (<Operand2>) ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| <Operand1>   | <Operand2>  | VKE der Anweisung |
|--------------|-------------|-------------------|
| 'AA'         | 'aa'        | 1                 |
| 'Hallo Welt' | 'HalloWelt' | 1                 |
| 'AA'         | 'AA'        | 0                 |

Mit der Anweisung "Ungleich" können Sie auch einzelne Zeichen einer Zeichenkette (STRING) vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Bei aktivierter IEC-Prüfung müssen die zu vergleichenden Operanden vom gleichen Datentyp sein. Wenn die IEC-Prüfung nicht aktiviert ist, muss die Breite (Länge) der Operanden gleich sein. Beim Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ungleich":

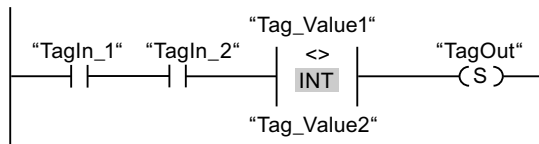
| Parameter  | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung           |
|------------|-------------|--|--|---------------------------------|------------------------|
|            |             | S7-1200  | S7-1500  |                                 |                        |
| <Operand1> | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| <Operand2> | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" <> "Tag\_Value2" ist.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

**CMP >=: Größer gleich****Beschreibung**

Mit der Anweisung "Größer gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) größer oder gleich dem zweiten Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| <Operand1>   | <Operand2>   | VKE der Anweisung |
|--------------|--------------|-------------------|
| 'BB'         | 'AA'         | 1                 |
| 'AAA'        | 'AA'         | 1                 |
| 'Hallo Welt' | 'Hallo Welt' | 1                 |
| 'Hallo Welt' | 'HalloWelt'  | 0                 |
| 'AA'         | 'aa'         | 0                 |
| 'AAA'        | 'a'          | 0                 |

Mit der Anweisung "Größer gleich" können Sie auch einzelne Zeichen einer Zeichenkette (STRING) vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am <Operand1> größer (jünger) oder gleich dem Zeitpunkt am <Operand2> ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer gleich":

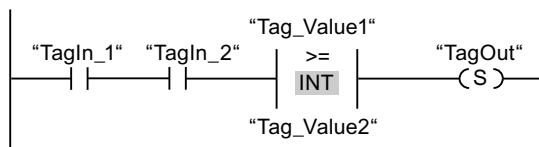
| Parameter  | Deklaration | Datentyp  |   | Speicherbereich                 | Beschreibung           |
|------------|-------------|---|---|---------------------------------|------------------------|
|            |             | S7-1200   | S7-1500   |                                 |                        |
| <Operand1> | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| <Operand2> | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" >= "Tag\_Value2" ist.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 1592)

## CMP <=: Kleiner gleich

### Beschreibung

Mit der Anweisung "Kleiner gleich" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) kleiner oder gleich dem zweiten Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die kürzere Zeichenkette als kleiner.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| <Operand1>   | <Operand2>   | VKE der Anweisung |
|--------------|--------------|-------------------|
| 'AA'         | 'aa'         | 1                 |
| 'AAA'        | 'a'          | 1                 |
| 'Hallo Welt' | 'Hallo Welt' | 1                 |
| 'HalloWelt'  | 'Hallo Welt' | 0                 |
| 'BB'         | 'AA'         | 0                 |
| 'AAA'        | 'AA'         | 0                 |

Mit der Anweisung "Kleiner gleich" können Sie auch einzelne Zeichen einer Zeichenkette (STRING) vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am <Operand1> kleiner (älter) oder gleich dem Zeitpunkt am <Operand2> ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner gleich":

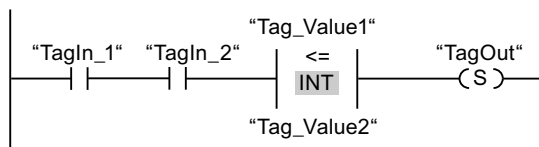
| Parameter  | Deklaration | Datentyp  |   | Speicherbereich                 | Beschreibung           |
|------------|-------------|---|---|---------------------------------|------------------------|
|            |             | S7-1200   | S7-1500   |                                 |                        |
| <Operand1> | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| <Operand2> | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" <= "Tag\_Value2" ist.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)



**CMP >: Größer****Beschreibung**

Mit der Anweisung "Größer" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) größer als der zweite Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| <Operand1> | <Operand2> | VKE der Anweisung |
|------------|------------|-------------------|
| 'BB'       | 'AA'       | 1                 |
| 'AAA'      | 'AA'       | 1                 |
| 'AA'       | 'aa'       | 0                 |
| 'AAA'      | 'a'        | 0                 |

Mit der Anweisung "Größer" können Sie auch einzelne Zeichen einer Zeichenkette (STRING) vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am <Operand1> größer (jünger) als der Zeitpunkt am <Operand2> ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer":

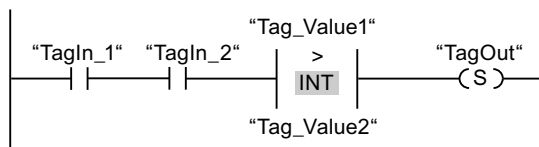
| Parameter  | Deklaration | Datentyp  |   | Speicherbereich                 | Beschreibung           |
|------------|-------------|---|---|---------------------------------|------------------------|
|            |             | S7-1200   | S7-1500   |                                 |                        |
| <Operand1> | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| <Operand2> | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" > "Tag\_Value2" ist.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## CMP <: Kleiner

### Beschreibung

Mit der Anweisung "Kleiner" fragen Sie ab, ob der erste Vergleichswert (<Operand1>) kleiner als der zweite Vergleichswert (<Operand2>) ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0". Das VKE der Anweisung wird mit dem VKE des gesamten Strompfads folgendermaßen verknüpft:

- Durch UND, wenn die Vergleichsanweisung in Reihe geschaltet ist.
- Durch ODER, wenn die Vergleichsanweisung parallel geschaltet ist.

Den ersten Vergleichswert (<Operand1>) geben Sie am Operandenplatzhalter oberhalb der Anweisung an. Den zweiten Vergleichswert (<Operand2>) geben Sie am Operandenplatzhalter unterhalb der Anweisung an.

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die kürzere Zeichenkette als kleiner.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| <Operand1> | <Operand2> | VKE der Anweisung |
|------------|------------|-------------------|
| 'AA'       | 'aa'       | 1                 |
| 'AAA'      | 'a'        | 1                 |
| 'BB'       | 'AA'       | 0                 |
| 'AAA'      | 'AA'       | 0                 |

Mit der Anweisung "Kleiner" können Sie auch einzelne Zeichen einer Zeichenkette (STRING) vergleichen. Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am <Operand1> kleiner (älter) als der Zeitpunkt am <Operand2> ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner":

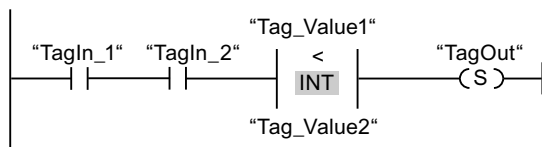
| Parameter  | Deklaration | Datentyp  |   | Speicherbereich                 | Beschreibung           |
|------------|-------------|---|---|---------------------------------|------------------------|
|            |             | S7-1200   | S7-1500   |                                 |                        |
| <Operand1> | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| <Operand2> | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???"> der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" < "Tag\_Value2" ist.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 1592)

## IN\_RANGE: Wert innerhalb Bereich

### Beschreibung

Mit der Anweisung "Wert innerhalb Bereich" fragen Sie ab, ob der Wert am Eingang VAL innerhalb eines bestimmten Wertebereichs liegt.

Die Grenzen des Wertebereichs legen Sie durch die Eingänge MIN und MAX fest. Die Anweisung "Wert innerhalb Bereich" vergleicht den Wert am Eingang VAL mit den Werten der Eingänge MIN und MAX und führt das Ergebnis auf den Boxausgang. Wenn der Wert am Eingang VAL den Vergleich  $MIN \leq VAL$  oder  $VAL \leq MAX$  erfüllt, liefert der Boxausgang den Signalzustand "1". Bei nicht erfülltem Vergleich steht am Boxausgang der Signalzustand "0".

Wenn der Boxeingang den Signalzustand "0" führt, wird die Anweisung "Wert innerhalb Bereich" nicht bearbeitet.

Für die Bearbeitung der Vergleichsfunktion wird vorausgesetzt, dass die zu vergleichenden Werte vom gleichen Datentyp sind und der Boxeingang beschaltet ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert innerhalb Bereich":

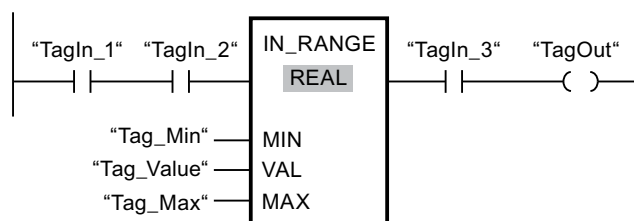
| Parameter  | Deklaration | Datentyp                     | Speicherbereich              | Beschreibung                        |
|------------|-------------|------------------------------|------------------------------|-------------------------------------|
| Boxeingang | Input       | BOOL                         | E, A, M, D, L                | Ergebnis der vorherigen Verknüpfung |
| MIN        | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L oder Konstante | Untere Grenze des Wertebereichs     |
| VAL        | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L oder Konstante | Vergleichswert                      |
| MAX        | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L oder Konstante | Obere Grenze des Wertebereichs      |
| Boxausgang | Output      | BOOL                         | E, A, M, D, L                | Ergebnis des Vergleichs             |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden unter "Siehe auch":

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Wert des Operanden "Tag\_Value" liegt innerhalb des Wertebereichs, der durch die aktuellen Werte der Operanden "Tag\_Min" und "Tag\_Max" vorgegeben ist (MIN <= VAL oder VAL <= MAX).
- Der Operand "TagIn\_3" liefert den Signalzustand "1".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**OUT\_RANGE: Wert außerhalb Bereich**

**Beschreibung**

Mit der Anweisung "Wert außerhalb Bereich" fragen Sie ab, ob der Wert am Eingang VAL außerhalb eines bestimmten Wertebereichs liegt.

Die Grenzen des Wertebereichs legen Sie durch die Eingänge MIN und MAX fest. Die Anweisung "Wert außerhalb Bereich" vergleicht den Wert am Eingang VAL mit den Werten der Eingänge MIN und MAX und führt das Ergebnis auf den Boxausgang. Wenn der Wert am Eingang VAL den Vergleich MIN > VAL oder VAL > MAX erfüllt, liefert der Boxausgang den Signalzustand "1". Der Boxausgang liefert auch den Signalzustand "1", wenn ein angegebener Operand vom Datentyp REAL einen ungültigen Wert aufweist.

Der Boxausgang liefert den Signalzustand "0", wenn der Wert am Eingang VAL die Bedingung MIN > VAL oder VAL > MAX nicht erfüllt.

Wenn der Boxeingang den Signalzustand "0" führt, wird die Anweisung "Wert außerhalb Bereich" nicht bearbeitet.

Für die Bearbeitung der Vergleichsfunktion wird vorausgesetzt, dass die zu vergleichenden Werte vom gleichen Datentyp sind und der Boxeingang beschaltet ist.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert außerhalb Bereich":

| Parameter  | Deklaration | Datentyp                     | Speicherbereich              | Beschreibung                        |
|------------|-------------|------------------------------|------------------------------|-------------------------------------|
| Boxeingang | Input       | BOOL                         | E, A, M, D, L                | Ergebnis der vorherigen Verknüpfung |
| MIN        | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L oder Konstante | Untere Grenze des Wertebereichs     |
| VAL        | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L oder Konstante | Vergleichswert                      |

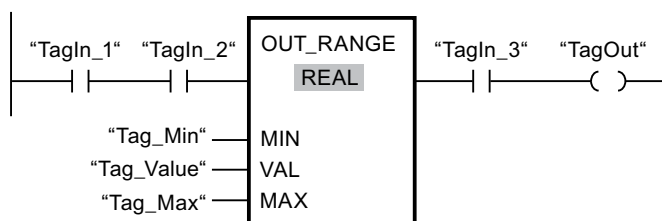
| Parameter  | Deklaration | Datentyp                        | Speicherbereich                 | Beschreibung                      |
|------------|-------------|---------------------------------|---------------------------------|-----------------------------------|
| MAX        | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Obere Grenze des<br>Wertebereichs |
| Boxausgang | Output      | BOOL                            | E, A, M, D, L                   | Ergebnis des<br>Vergleichs        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Wert des Operanden "Tag\_Value" liegt außerhalb des Wertebereichs, der durch die Werte der Operanden "Tag\_Min" und "Tag\_Max" vorgegeben ist ( $\text{MIN} > \text{VAL}$  oder  $\text{VAL} > \text{MAX}$ ).
- Der Operand "TagIn\_3" liefert den Signalzustand "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ----I OK I----: Gültigkeit prüfen

### Beschreibung

Mit der Anweisung "Gültigkeit prüfen" fragen Sie ab, ob der Wert eines Operanden (<Operand>) eine gültige Gleitpunktzahl ist. Die Abfrage wird bei jedem Programmzyklus gestartet, wenn der Eingang der Anweisung den Signalzustand "1" hat.

Der Ausgang der Anweisung hat den Signalzustand "1", wenn der Wert des Operanden zum Zeitpunkt der Abfrage eine gültige Gleitpunktzahl ist und der Eingang der Anweisung den Signalzustand "1" hat. In allen anderen Fällen hat der Ausgang der Anweisung "Gültigkeit prüfen" den Signalzustand "0".

Die Anweisung "Gültigkeit prüfen" können Sie in Verbindung mit dem EN-Mechanismus nutzen. Wenn Sie die Anweisungsbox an einen Freigabeeingang EN anschließen, wird der Freigabeeingang nur bei einer positiven Abfrage der Wertgültigkeit gesetzt. Durch diese

Funktionalität können Sie sicherstellen, dass eine Anweisung nur dann aktiviert wird, wenn der Wert des angegebenen Operanden eine gültige Gleitpunktzahl ist.

### Parameter

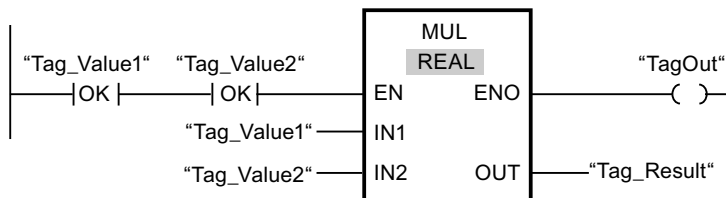
Die folgende Tabelle zeigt die Parameter der Anweisung "Gültigkeit prüfen":

| Parameter | Deklaration | Datentyp         | Speicherbereich | Beschreibung              |
|-----------|-------------|------------------|-----------------|---------------------------|
| <Operand> | Input       | Gleitpunktzahlen | E, A, M, D, L   | Wert, der abgefragt wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Werte der Operanden "Tag\_Value1" und "Tag\_Value2" gültige Gleitpunktzahlen aufweisen, wird die Anweisung "Multiplizieren" (MUL) aktiviert und der Ausgang ENO gesetzt. Bei der Bearbeitung der Anweisung "Multiplizieren" (MUL) wird der Wert des Operanden "Tag\_Value1" mit dem Wert des Operanden "Tag\_Value2" multipliziert. Das Multiplikationsergebnis wird dann in den Operanden "Tag\_Result" gespeichert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden die Ausgänge ENO und "TagOut" auf den Signalzustand "1" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### ----I NOT\_OK I----: Ungültigkeit prüfen

#### Beschreibung

Mit der Anweisung "Ungültigkeit prüfen" fragen Sie ab, ob der Wert eines Operanden (<Operand>) eine ungültige Gleitpunktzahl ist. Die Abfrage wird bei jedem Programmzyklus gestartet, wenn der Eingang der Anweisung den Signalzustand "1" hat.

Der Ausgang der Anweisung hat den Signalzustand "1", wenn der Wert des Operanden zum Zeitpunkt der Abfrage eine ungültige Gleitpunktzahl ist und der Eingang der Anweisung den Signalzustand "1" hat. In allen anderen Fällen hat der Ausgang der Anweisung "Ungültigkeit prüfen" den Signalzustand "0".



## Parameter

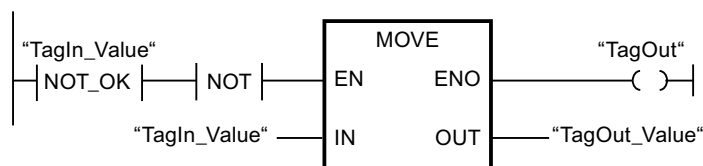
Die folgende Tabelle zeigt die Parameter der Anweisung "Ungültigkeit prüfen":

| Parameter | Deklaration | Datentyp         | Speicherbereich | Beschreibung              |
|-----------|-------------|------------------|-----------------|---------------------------|
| <Operand> | Input       | Gleitpunktzahlen | E, A, M, D, L   | Wert, der abgefragt wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Wert des Operanden "TagIn\_Value" eine ungültige Gleitpunktzahl ist, wird die Anweisung "Wert kopieren" (MOVE) nicht ausgeführt. Der Ausgang "TagOut" wird auf den Signalzustand "0" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Mathematische Funktionen

### CALCULATE: Berechnen

#### Beschreibung

Mit der Anweisung "Berechnen" definieren und führen Sie, abhängig vom gewählten Datentyp, einen Ausdruck zum Berechnen mathematischer Operationen oder komplexer Verknüpfungen aus.

Den Datentyp der Anweisung wählen Sie aus der Klappliste "<???" der Anweisungsbox. Abhängig vom ausgewählten Datentyp können Sie die Funktionalität bestimmter Anweisungen kombinieren, um eine komplexe Berechnung auszuführen. Die Angabe des zu berechnenden Ausdrucks erfolgt über einen Dialog, den Sie über das Symbol am rechten oberen Rand der Anweisungsbox öffnen können. Im Ausdruck können Namen der Eingangsparameter und die Syntax der Anweisungen enthalten sein. Die Angabe von Operandennamen oder -adressen ist nicht zulässig.

Die folgende Tabelle zeigt die Anweisungen, die, abhängig vom gewählten Datentyp, im Ausdruck der Anweisung "Berechnen" kombiniert werden, ausgeführt werden können:

| Datentyp   | Anweisung  | Syntax   | Beispiel  |
|--|--|----------|---|
| Bitfolgen  | AND: UND verknüpfen  | AND      | IN1 AND IN2 OR IN3  |
|  | OR: ODER verknüpfen  | OR       |   |
|  | XOR: EXCLUSIV ODER verknüpfen                              | XOR      |   |
|  | INV: Einerkomplement erzeugen                              | NOT      |   |
|  | SWAP: Anordnung ändern <sup>1)</sup>                       | SWAP     |   |
| Ganzzahlen   | ADD: Addieren  | +        | (IN1 + IN2) * IN3;<br>(ABS(IN2))*(ABS(IN1))   |
|  | SUB: Subtrahieren  | -        |   |
|  | MUL: Multiplizieren  | *        |   |
|  | DIV: Dividieren  | /        |   |
|  | MOD: Divisionsrest gewinnen                                | MOD      |   |
|  | INV: Einerkomplement erzeugen                              | NOT      |   |
|  | NEG: Zweierkomplement erzeugen                             | -(in1)   |   |
|  | ABS: Absolutwert bilden                                    | ABS( )   |   |
| Gleitpunktzahlen                                   | ADD: Addieren  | +        | ((SIN(IN2)*SIN(IN2)+<br>SIN(IN3)*SIN(IN3))/<br>IN3;<br>SQR(SIN(IN2))+<br>SQR(COS(IN3)))/IN2 |
|  | SUB: Subtrahieren  | -        |   |
|  | MUL: Multiplizieren  | *        |   |
|  | DIV: Dividieren  | /        |   |
|  | EXPT: Potenzieren  | **       |   |
|  | ABS: Absolutwert bilden                                    | ABS( )   |   |
|  | SQR: Quadrat bilden  | SQR( )   |   |
|  | SQRT: Quadratwurzel bilden                                 | SQRT( )  |   |
|  | LN: Natürlichen Logarithmus bilden                         | LN( )    |   |
|  | EXP: Exponentialwert bilden                                | EXP( )   |   |
|  | FRAC: Nachkommastellen ermitteln                           | FRAC( )  |   |
|  | SIN: Sinuswert bilden                                      | SIN( )   |   |
|  | COS: Cosinuswert bilden                                    | COS( )   |   |
|  | TAN: Tangenswert bilden                                    | TAN( )   |   |
|  | ASIN: Arcussinuswert bilden                                | ASIN( )  |   |
|  | ACOS: Arcuscosinuswert bilden                              | ACOS( )  |   |
|  | ATAN: Arcustangenswert bilden                              | ATAN( )  |   |
|  | NEG: Zweierkomplement erzeugen                             | -(in1)   |   |
|  | TRUNC: Ganzzahl erzeugen                                   | TRUNC( ) |   |
|  | ROUND: Zahl runden   | ROUND( ) |   |
|  | CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen   | CEIL( )  |   |
|  | FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen | FLOOR( ) |   |
| <sup>1)</sup> Nicht für den Datentyp BYTE möglich. |  |          |   |

Die Anweisungsbox enthält im Grundzustand mindestens 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert.

Die Werte der Eingänge werden zur Ausführung des angegebenen Ausdrucks verwendet. Im Ausdruck müssen dabei nicht alle definierten Eingänge verwendet werden. Das Ergebnis der Anweisung wird an den Ausgang OUT übertragen.

Wenn Sie im Ausdruck Eingänge verwenden, die in der Box nicht verfügbar sind, werden diese automatisch eingefügt. Voraussetzung dafür ist, dass die Nummerierung der neu zu definierenden Eingänge im Ausdruck lückenlos ist. Sie können zum Beispiel den Eingang IN4 im Ausdruck nicht verwenden, wenn der Eingang IN3 nicht definiert ist.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis oder ein Zwischenergebnis der Anweisung "Berechnen" liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.
- Bei der Ausführung einer der im Ausdruck angegebenen Anweisungen ist ein Fehler aufgetreten.

## Parameter

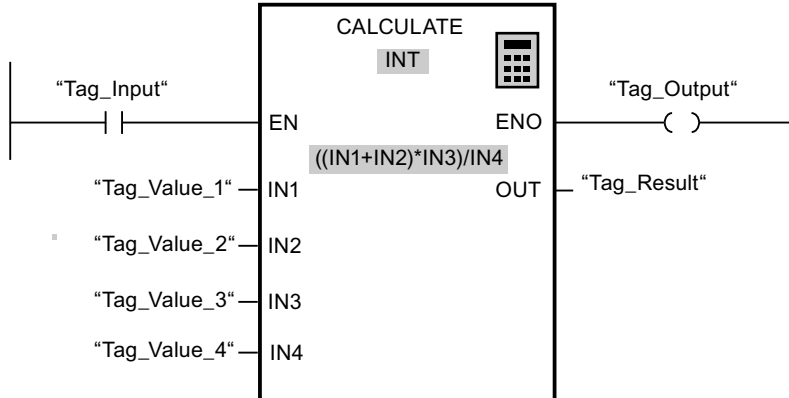
Die folgende Tabelle zeigt die Parameter der Anweisung "Berechnen":

| Parameter | Deklaration | Datentyp                                      | Speicherbereich                    | Beschreibung                                     |
|-----------|-------------|---|------------------------------------|--|
| EN        | Input       | BOOL  | E, A, M, D, L                      | Freigabeeingang                                  |
| ENO       | Output      | BOOL  | E, A, M, D, L                      | Freigabeausgang                                  |
| IN1       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Erster verfügbarer Eingang                       |
| IN2       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Zweiter verfügbarer Eingang                      |
| INn       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Zusätzlich eingefügte Eingänge                   |
| OUT       | Output      | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P                   | Ausgang, an den das Endergebnis übertragen wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand     | Wert |
|-----------|-------------|------|
| IN1       | Tag_Value_1 | 4    |
| IN2       | Tag_Value_2 | 4    |
| IN3       | Tag_Value_3 | 3    |
| IN4       | Tag_Value_4 | 2    |
| OUT       | Tag_Result  | 12   |

Wenn der Eingang "Tag\_Input" den Signalzustand "1" liefert, wird die Anweisung "Berechnen" ausgeführt. Der Wert des Operanden "Tag\_Value\_1" wird mit dem Wert des Operanden "Tag\_Value\_2" addiert. Die Summe wird mit dem Wert des Operanden "Tag\_Value\_3" multipliziert. Das Produkt wird durch den Wert des Operanden "Tag\_Value\_4" dividiert. Der Quotient wird als Endergebnis in den Operanden "Tag\_Result" am Ausgang OUT der Anweisung übertragen. Wenn keine Fehler während der Bearbeitung der einzelnen Anweisungen auftreten, werden der Freigabeausgang ENO und der Operand "Tag\_Output" auf "1" gesetzt.

**Siehe auch**

- Anweisung "Berechnen" verwenden (Seite 1302)
- Übersicht über die gültigen Datentypen (Seite 1087)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)
- Eingänge und Ausgänge entfernen (Seite 1311)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ADD: Addieren

### Beschreibung

Mit der Anweisung "Addieren" addieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragt die Summe am Ausgang OUT ab ( $OUT := IN1 + IN2$ ).

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter addiert. Die Summe wird am Ausgang OUT gespeichert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Addieren":

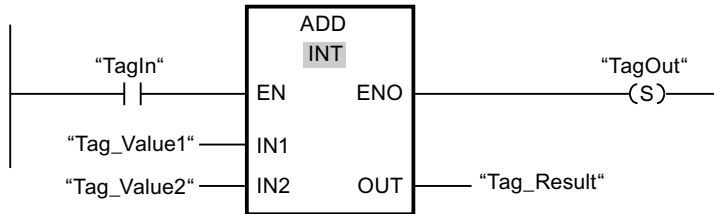
| Parameter | Deklaration | Datentyp                        | Speicherbereich                    | Beschreibung  |
|-----------|-------------|---------------------------------|------------------------------------|---|
| EN        | Input       | BOOL                            | E, A, M, D, L                      | Freigabeeingang                                       |
| ENO       | Output      | BOOL                            | E, A, M, D, L                      | Freigabeausgang                                       |
| IN1       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Erster Summand  |
| IN2       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Zweiter Summand                                       |
| INn       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Optionale<br>Eingangswerte,<br>die addiert<br>werden. |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P                   | Summe   |

Aus der Klappliste "<???"> der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weiter Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Addieren" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" addiert. Das Ergebnis der Addition wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Eingänge und Ausgänge entfernen (Seite 1311)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Auswahl eines Datentyps (Seite 1294)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

**SUB: Subtrahieren**

**Beschreibung**

Mit der Anweisung "Subtrahieren" ziehen Sie den Wert am Eingang IN2 vom Wert am Eingang IN1 ab und fragen die Differenz am Ausgang OUT ab (OUT := IN1-IN2).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Subtrahieren":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung    |
|-----------|-------------|----------|-----------------|-----------------|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang |

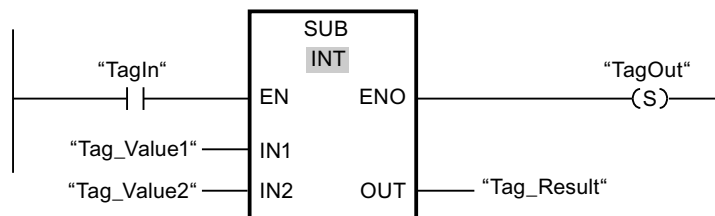
| Parameter | Deklaration | Datentyp                        | Speicherbereich                    | Beschreibung |
|-----------|-------------|---------------------------------|------------------------------------|--------------|
| IN1       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Minuend      |
| IN2       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Subtrahend   |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P                   | Differenz    |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Subtrahieren" ausgeführt. Der Wert des Operanden "Tag\_Value2" wird vom Wert des Operanden "Tag\_Value1" subtrahiert. Das Ergebnis der Subtraktion wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Auswahl eines Datentyps (Seite 1294)

## MUL: Multiplizieren

### Beschreibung

Mit der Anweisung "Multiplizieren" multiplizieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und das Produkt am Ausgang OUT abfragen ( $OUT := IN1 * IN2$ ).

Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter multipliziert. Das Produkt wird am Ausgang OUT gespeichert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Eingang EN liefert den Signalzustand "0".
- Das Ergebnis liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplizieren":

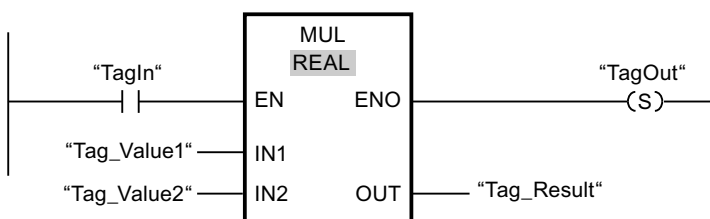
| Parameter | Deklaration | Datentyp                        | Speicherbereich                    | Beschreibung   |
|-----------|-------------|---------------------------------|------------------------------------|--|
| EN        | Input       | BOOL                            | E, A, M, D, L                      | Freigabeeingang  |
| ENO       | Output      | BOOL                            | E, A, M, D, L                      | Freigabeausgang  |
| IN1       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Multiplikator  |
| IN2       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Multiplikand   |
| INn       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Optionale<br>Eingangswerte,<br>die multipliziert<br>werden können. |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P                   | Produkt  |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Multiplizieren" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" multipliziert. Das Multiplikationsergebnis wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.



**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Eingänge und Ausgänge entfernen (Seite 1311)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Auswahl eines Datentyps (Seite 1294)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

**DIV: Dividieren****Beschreibung**

Mit der Anweisung "Dividieren" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Quotienten am Ausgang OUT ab ( $OUT := IN1/IN2$ ).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Dividieren":

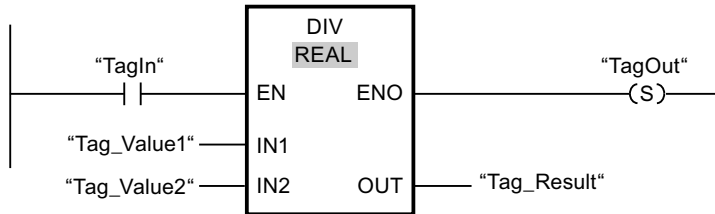
| Parameter | Deklaration | Datentyp                        | Speicherbereich                    | Beschreibung           |
|-----------|-------------|---------------------------------|------------------------------------|------------------------|
| EN        | Input       | BOOL                            | E, A, M, D, L                      | Freigabeeingang        |
| ENO       | Output      | BOOL                            | E, A, M, D, L                      | Freigabeausgang        |
| IN1       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Dividend               |
| IN2       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Divisor                |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P                   | Wert des<br>Quotienten |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Dividieren" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird durch den Wert des Operanden "Tag\_Value2" dividiert. Das Ergebnis der Division wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Auswahl eines Datentyps (Seite 1294)

**MOD: Divisionsrest gewinnen**

**Beschreibung**

Mit der Anweisung "Divisionsrest gewinnen" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Divisionsrest am Ausgang OUT ab.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Divisionsrest gewinnen":

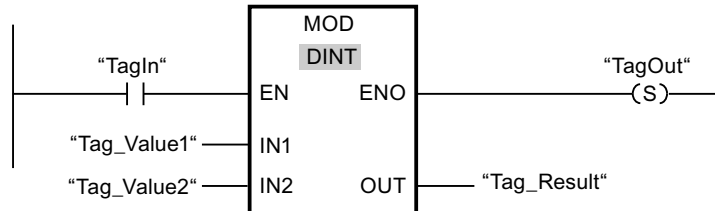
| Parameter | Deklaration | Datentyp   | Speicherbereich                 | Beschreibung    |
|-----------|-------------|------------|---------------------------------|-----------------|
| EN        | Input       | BOOL       | E, A, M, D, L                   | Freigabeeingang |
| ENO       | Output      | BOOL       | E, A, M, D, L                   | Freigabeausgang |
| IN1       | Input       | Ganzzahlen | E, A, M, D, L, P oder Konstante | Dividend        |
| IN2       | Input       | Ganzzahlen | E, A, M, D, L, P oder Konstante | Divisor         |
| OUT       | Output      | Ganzzahlen | E, A, M, D, L, P                | Divisionsrest   |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Divisionsrest gewinnen" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird durch den Wert des Operanden "Tag\_Value2" dividiert. Der Divisionsrest wird im Operanden "Tag\_Result" gespeichert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Auswahl eines Datentyps (Seite 1294)

## NEG: Zweierkomplement erzeugen

### Beschreibung

Mit der Anweisung "Zweierkomplement erzeugen" wechseln Sie das Vorzeichen des Werts am Eingang IN und fragen das Ergebnis am Ausgang OUT ab. Wenn z. B. am Eingang IN ein positiver Wert ansteht, wird am Ausgang OUT das negative Äquivalent dieses Wertes ausgegeben.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zweierkomplement erzeugen":

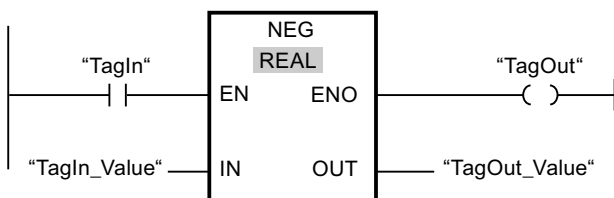
| Parameter | Deklaration | Datentyp                          |   | Speicherbereich                 | Beschreibung                       |
|-----------|-------------|-----------------------------------|---|---------------------------------|------------------------------------|
|           |             | S7-1200                           | S7-1500                                 |                                 |                                    |
| EN        | Input       | BOOL                              | BOOL                                    | E, A, M, D, L                   | Freigabeeingang                    |
| ENO       | Output      | BOOL                              | BOOL                                    | E, A, M, D, L                   | Freigabeausgang                    |
| IN        | Input       | SINT, INT, DINT, Gleitpunktzahlen | SINT, INT, DINT, LINT, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | Eingangswert                       |
| OUT       | Output      | SINT, INT, DINT, Gleitpunktzahlen | SINT, INT, DINT, LINT, Gleitpunktzahlen | E, A, M, D, L, P                | Zweierkomplement des Eingangswerts |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Zweierkomplement erzeugen" ausgeführt. Das Vorzeichen des Werts am Eingang "TagIn\_Value" wird gewechselt und das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## INC: Inkrementieren

### Beschreibung

Mit der Anweisung "Inkrementieren" ändern Sie den Wert des Operanden am Parameter IN/OUT auf den nächst höheren Wert und fragen das Ergebnis ab. Die Bearbeitung der Anweisung "Inkrementieren" wird nur gestartet, wenn der Signalzustand am Freigabeeingang EN "1" ist. Wenn kein Überlauf während der Bearbeitung auftritt, führt der Freigabeausgang ENO ebenfalls den Signalzustand "1".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Inkrementieren":

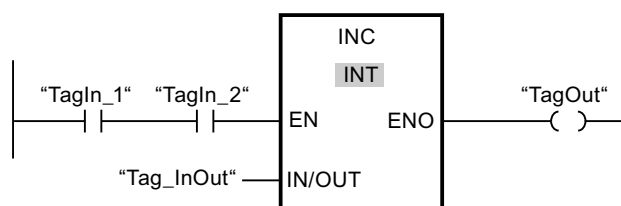
| Parameter | Deklaration | Datentyp   | Speicherbereich | Beschreibung                  |
|-----------|-------------|------------|-----------------|-------------------------------|
| EN        | Input       | BOOL       | E, A, M, D, L   | Freigabeeingang               |
| ENO       | Output      | BOOL       | E, A, M, D, L   | Freigabeausgang               |
| IN/OUT    | InOut       | Ganzzahlen | E, A, M, D, L   | Wert, der inkrementiert wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird der Wert des Operanden "Tag\_InOut" um eins erhöht und der Ausgang "TagOut" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## DEC: Dekrementieren

### Beschreibung

Mit der Anweisung "Dekrementieren" ändern Sie den Wert des Operanden am Parameter IN/OUT auf den nächst niedrigeren Wert und fragen das Ergebnis ab. Die Bearbeitung der Anweisung "Dekrementieren" wird gestartet, wenn der Signalzustand am Freigabeeingang EN "1" ist. Wenn der Wertebereich des ausgewählten Datentyps während der Bearbeitung nicht unterschritten wird, führt der Ausgang ENO ebenfalls den Signalzustand "1".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dekrementieren":

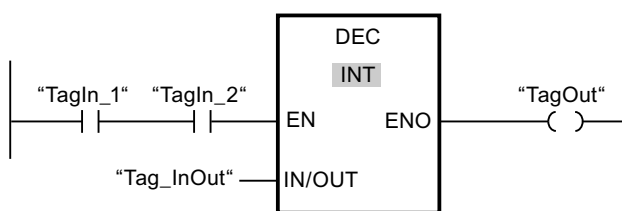
| Parameter | Deklaration | Datentyp   | Speicherbereich | Beschreibung                  |
|-----------|-------------|------------|-----------------|-------------------------------|
| EN        | Input       | BOOL       | E, A, M, D, L   | Freigabeeingang               |
| ENO       | Output      | BOOL       | E, A, M, D, L   | Freigabeausgang               |
| IN/OUT    | InOut       | Ganzzahlen | E, A, M, D, L   | Wert, der dekrementiert wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird der Wert des Operanden "Tag\_InOut" um eins verringert und der Ausgang "TagOut" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ABS: Absolutwert bilden

### Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag des Wertes, der am Eingang IN angegeben ist. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Absolutwert bilden":

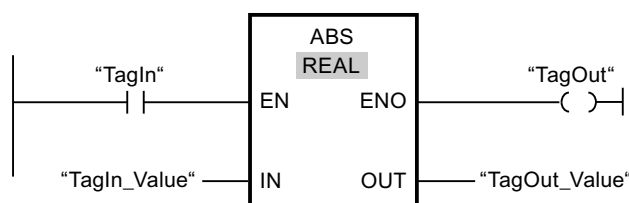
| Parameter | Deklaration | Datentyp                          |   | Speicherbereich                 | Beschreibung                  |
|-----------|-------------|-----------------------------------|---|---------------------------------|-------------------------------|
|           |             | S7-1200                           | S7-1500                                 |                                 |                               |
| EN        | Input       | BOOL                              | BOOL                                    | E, A, M, D, L                   | Freigabeeingang               |
| ENO       | Output      | BOOL                              | BOOL                                    | E, A, M, D, L                   | Freigabeausgang               |
| IN        | Input       | SINT, INT, DINT, Gleitpunktzahlen | SINT, INT, DINT, LINT, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | Eingangswert                  |
| OUT       | Output      | SINT, INT, DINT, Gleitpunktzahlen | SINT, INT, DINT, LINT, Gleitpunktzahlen | E, A, M, D, L, P                | Absolutwert des Eingangswerts |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert   |
|-----------|--------------|--------|
| IN        | TagIn_Value  | -6.234 |
| OUT       | TagOut_Value | 6.234  |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Absolutwert bilden" ausgeführt. Die Anweisung berechnet den Absolutbetrag des Wertes am Eingang "TagIn\_Value" und gibt das Ergebnis am Ausgang "TagOut\_Value" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**MIN: Minimum ermitteln**

**Beschreibung**

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und schreiben den kleinsten Wert in den Ausgang OUT. Die Anzahl der Eingänge ist an der Anweisungsbox durch zusätzliche Eingänge erweiterbar. Die Eingänge werden in der Box aufsteigend nummeriert.

Für die Ausführung der Anweisung müssen mindestens zwei und können maximal 100 Eingänge angegeben werden.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

| Parameter | Deklaration | Datentyp |         | Speicherbereich | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|-----------------|
|           |             | S7-1200  | S7-1500 |                 |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | Freigabeausgang |



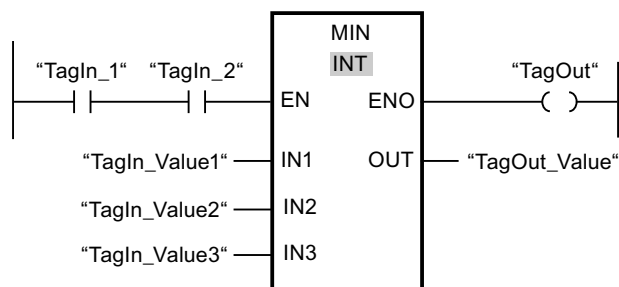
| Parameter | Deklaration | Datentyp   |   | Speicherbereich                 | Beschreibung  |
|-----------|-------------|--|---|---------------------------------|---|
|           |             | S7-1200  | S7-1500   |                                 |   |
| IN1       | Input       | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Erster Eingangswert   |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE    | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Zweiter Eingangswert  |
| INn       | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE    | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Zusätzlich eingefügte Eingänge, deren Werte verglichen werden |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE    | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | Ergebnis  |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert  |
|-----------|--------------|-------|
| IN1       | TagIn_Value1 | 12222 |
| IN2       | TagIn_Value2 | 14444 |
| IN3       | TagIn_Value3 | 13333 |
| OUT       | TagOut_Value | 12222 |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" haben, wird die Anweisung "Minimum ermitteln" ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("TagIn\_Value1") in den Ausgang "TagOut\_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**MAX: Maximum ermitteln**

**Beschreibung**

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und schreiben den größten Wert in den Ausgang OUT. Die Anzahl der Eingänge ist an der Anweisungsbox durch zusätzliche Eingänge erweiterbar. Die Eingänge werden in der Box aufsteigend nummeriert.

Für die Ausführung der Anweisung müssen mindestens zwei und können maximal 100 Eingänge angegeben werden.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

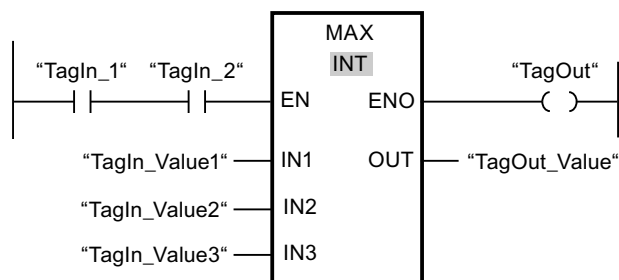
| Parameter | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung  |
|-----------|-------------|--|--|---------------------------------|---|
|           |             | S7-1200  | S7-1500  |                                 |   |
| EN        | Input       | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeeingang   |
| ENO       | Output      | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeausgang   |
| IN1       | Input       | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen,<br>Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Erster Eingangswert   |
| IN2       | Input       | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen,<br>Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Zweiter Eingangswert  |
| INn       | Input       | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen,<br>Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Zusätzlich eingefügte Eingänge, deren Werte verglichen werden |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen,<br>Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | Ergebnis  |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert  |
|-----------|--------------|-------|
| IN1       | TagIn_Value1 | 12222 |
| IN2       | TagIn_Value2 | 14444 |
| IN3       | TagIn_Value3 | 13333 |
| OUT       | TagOut_Value | 14444 |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" haben, wird die Anweisung "Maximum ermitteln" ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("TagIn\_Value2") in den Ausgang "TagOut\_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)
- Eingänge und Ausgänge entfernen (Seite 1311)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**LIMIT: Limitieren**

**Beschreibung**

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung  $MN \leq IN \leq MX$  erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis undefiniert und der Freigabeausgang ENO ist "0".

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Eingang MN ist größer als der Wert am Eingang MX.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Limitieren":

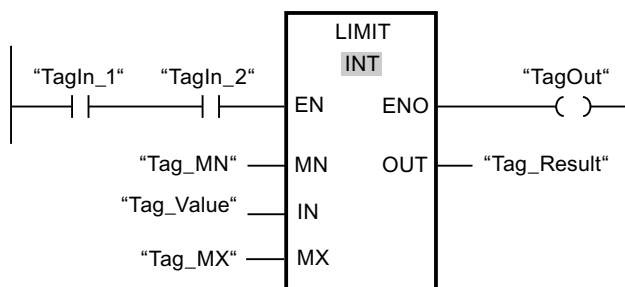
| Parameter | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung    |
|-----------|-------------|--|--|---------------------------------|-----------------|
|           |             | S7-1200  | S7-1500  |                                 |                 |
| EN        | Input       | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeeingang |
| ENO       | Output      | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeausgang |
| MN        | Input       | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen,<br>Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Untergrenze     |
| IN        | Input       | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen,<br>Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Eingangswert    |
| MX        | Input       | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen,<br>Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Obergrenze      |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen,<br>Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | Ergebnis        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|-------|
| MN        | Tag_MN     | 12000 |
| IN        | Tag_Value  | 8000  |
| MX        | Tag_MX     | 16000 |
| OUT       | Tag_Result | 12000 |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" haben, wird die Anweisung "Limitieren" ausgeführt. Der Wert des Operanden "Tag\_Value" wird mit den Werten der Operanden "Tag\_MN" und "Tag\_MX" verglichen. Da der Wert des Operanden "Tag\_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag\_MN" in den Ausgang "Tag\_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**SQR: Quadrat bilden**

**Beschreibung**

Mit der Anweisung "Quadrat bilden" quadrieren Sie den am Eingang IN stehenden Wert und fragen das Ergebnis am Ausgang OUT ab.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft.

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadrat bilden":

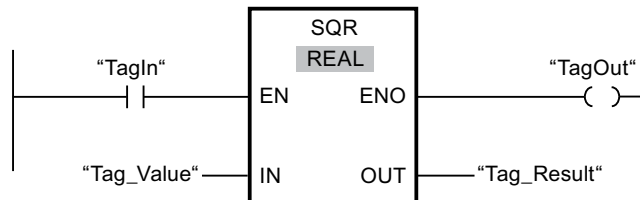
| Parameter | Deklaration | Datentyp         | Speicherbereich                 | Beschreibung               |
|-----------|-------------|------------------|---------------------------------|----------------------------|
| EN        | Input       | BOOL             | E, A, M, D, L                   | Freigabeeingang            |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | Freigabeausgang            |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | Eingangswert               |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | Quadrat des Eingangswertes |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert |
|-----------|------------|------|
| IN        | Tag_Value  | 5.0  |
| OUT       | Tag_Result | 25.0 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Quadrat bilden" ausgeführt. Die Anweisung quadriert den Wert des Operanden "Tag\_Value" und gibt das Ergebnis am Ausgang "Tag\_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## SQRT: Quadratwurzel bilden

### Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem am Eingang IN stehenden Wert die Quadratwurzel und fragen das Ergebnis am Ausgang OUT ab. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück. Wenn der Wert am Eingang IN "0" ist, ist das Ergebnis auch "0".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadratwurzel bilden":

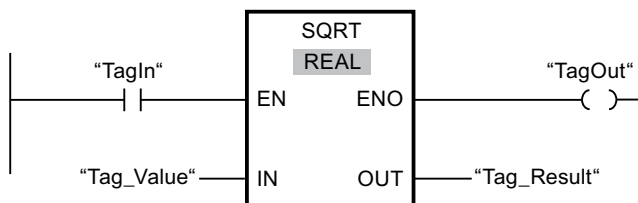
| Parameter | Deklaration | Datentyp         | Speicherbereich                    | Beschreibung                          |
|-----------|-------------|------------------|------------------------------------|---------------------------------------|
| EN        | Input       | BOOL             | E, A, M, D, L                      | Freigabeeingang                       |
| ENO       | Output      | BOOL             | E, A, M, D, L                      | Freigabeausgang                       |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Eingangswert                          |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                   | Quadratwurzel<br>des<br>Eingangswerts |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert |
|-----------|------------|------|
| IN        | Tag_Value  | 25.0 |
| OUT       | Tag_Result | 5.0  |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Quadratwurzel bilden" ausgeführt. Die Anweisung zieht aus dem Wert des Operanden "Tag\_Value" die Quadratwurzel und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)



## LN: Natürlichen Logarithmus bilden

### Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem am Eingang IN stehenden Wert den natürlichen Logarithmus zur Basis e ( $e = 2.718282$ ). Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Natürlichen Logarithmus bilden":

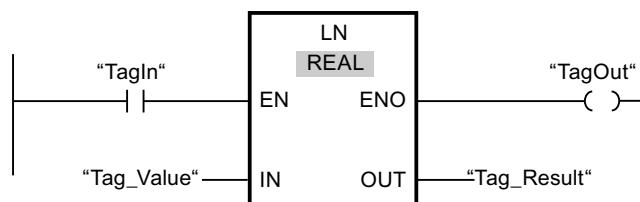
| Parameter | Deklaration | Datentyp         | Speicherbereich                    | Beschreibung                              |
|-----------|-------------|------------------|------------------------------------|---|
| EN        | Input       | BOOL             | E, A, M, D, L                      | Freigabeeingang                           |
| ENO       | Output      | BOOL             | E, A, M, D, L                      | Freigabeausgang                           |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Eingangswert                              |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                   | Natürlicher Logarithmus des Eingangswerts |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Natürlichen Logarithmus bilden" ausgeführt. Die Anweisung bildet den natürlichen Logarithmus des Wertes am Eingang "Tag\_Value" und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn

keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**EXP: Exponentialwert bilden**

**Beschreibung**

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2.718282) und dem am Eingang IN angegebenen Wert. Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden (OUT = e<sup>IN</sup>).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Exponentialwert bilden":

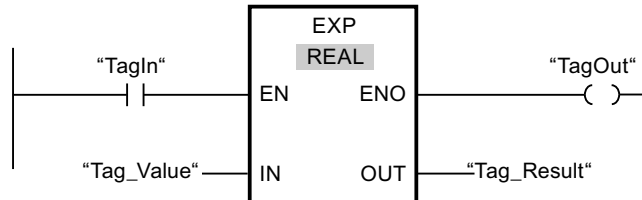
| Parameter | Deklaration | Datentyp         | Speicherbereich                    | Beschreibung                               |
|-----------|-------------|------------------|------------------------------------|--|
| EN        | Input       | BOOL             | E, A, M, D, L                      | Freigabeeingang                            |
| ENO       | Output      | BOOL             | E, A, M, D, L                      | Freigabeausgang                            |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Eingangswert                               |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                   | Exponentialwert<br>des<br>Eingangswerts IN |

Aus der Klappliste "<???">" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Exponentialwert bilden" ausgeführt. Die Anweisung bildet die Potenz aus der Basis e und dem Wert des Operanden "Tag\_Value" und gibt das Ergebnis am Ausgang "Tag\_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## SIN: Sinuswert bilden

### Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sinuswert bilden":

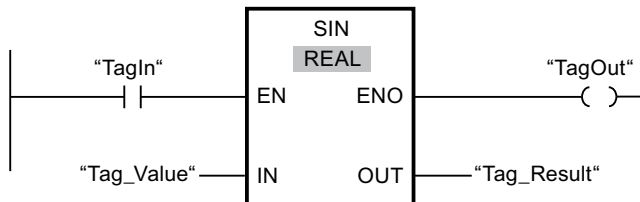
| Parameter | Deklaration | Datentyp         | Speicherbereich                    | Beschreibung                  |
|-----------|-------------|------------------|------------------------------------|-------------------------------|
| EN        | Input       | BOOL             | E, A, M, D, L                      | Freigabeeingang               |
| ENO       | Output      | BOOL             | E, A, M, D, L                      | Freigabeausgang               |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Größe des Winkels im Bogenmaß |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                   | Sinus des angegebenen Winkels |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | +1,570796 ( $\pi/2$ ) |
| OUT       | Tag_Result | 1.0                   |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Sinuswert bilden" ausgeführt. Die Anweisung berechnet den Sinus des am Eingang "Tag\_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### COS: Cosinuswert bilden

#### Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Cosinuswert bilden":

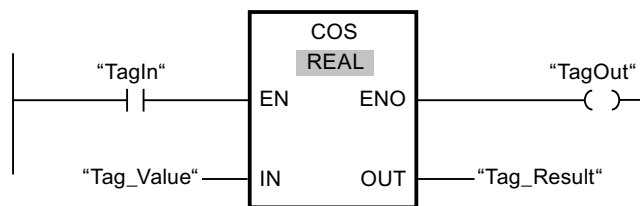
| Parameter | Deklaration | Datentyp         | Speicherbereich                    | Beschreibung                    |
|-----------|-------------|------------------|------------------------------------|---------------------------------|
| EN        | Input       | BOOL             | E, A, M, D, L                      | Freigabeeingang                 |
| ENO       | Output      | BOOL             | E, A, M, D, L                      | Freigabeausgang                 |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Größe des Winkels im Bogenmaß   |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                   | Cosinus des angegebenen Winkels |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | +1.570796 ( $\pi/2$ ) |
| OUT       | Tag_Result | 0                     |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Cosinuswert bilden" ausgeführt. Die Anweisung berechnet den Cosinus des am Eingang "Tag\_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## TAN: Tangenswert bilden

### Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Tangenswert bilden":

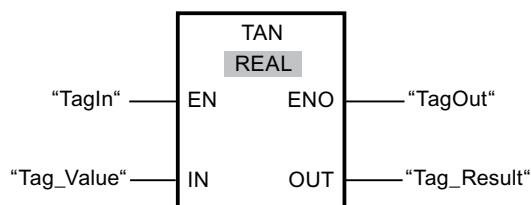
| Parameter | Deklaration | Datentyp         | Speicherbereich                 | Beschreibung                    |
|-----------|-------------|------------------|---------------------------------|---------------------------------|
| EN        | Input       | BOOL             | E, A, M, D, L                   | Freigabeeingang                 |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | Freigabeausgang                 |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | Größe des Winkels im Bogenmaß   |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | Tangens des angegebenen Winkels |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                |
|-----------|------------|---------------------|
| IN        | Tag_Value  | +3.141593 ( $\pi$ ) |
| OUT       | Tag_Result | 0                   |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Tangenswert bilden" ausgeführt. Die Anweisung berechnet den Tangens des am Eingang "Tag\_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ASIN: Arcussinuswert bilden

### Beschreibung

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Sinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von  $-\pi/2$  bis  $+\pi/2$  liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcussinuswert bilden":

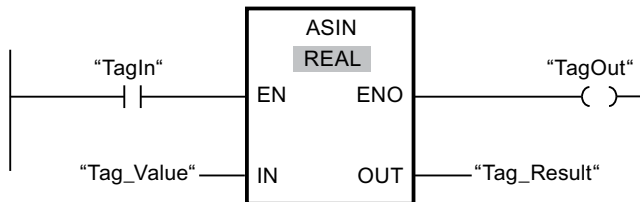
| Parameter | Deklaration | Datentyp         | Speicherbereich                    | Beschreibung                  |
|-----------|-------------|------------------|------------------------------------|-------------------------------|
| EN        | Input       | BOOL             | E, A, M, D, L                      | Freigabeeingang               |
| ENO       | Output      | BOOL             | E, A, M, D, L                      | Freigabeausgang               |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Sinuswert                     |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                   | Größe des Winkels im Bogenmaß |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | 1.0                   |
| OUT       | Tag_Result | +1.570796 ( $\pi/2$ ) |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcussinuswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Sinuswert am Eingang "Tag\_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**ACOS: Arcuscocinuswert bilden**

**Beschreibung**

Mit der Anweisung "Arcuscocinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von 0 bis  $+\pi$  liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcuscosinuswert bilden":

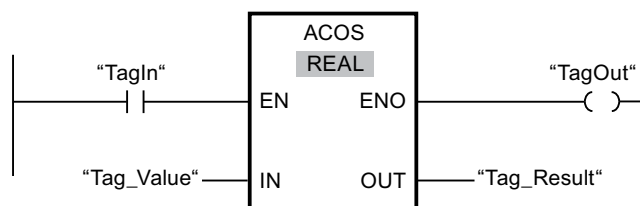
| Parameter | Deklaration | Datentyp         | Speicherbereich                    | Beschreibung                  |
|-----------|-------------|------------------|------------------------------------|-------------------------------|
| EN        | Input       | BOOL             | E, A, M, D, L                      | Freigabeeingang               |
| ENO       | Output      | BOOL             | E, A, M, D, L                      | Freigabeausgang               |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Cosinuswert                   |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                   | Größe des Winkels im Bogenmaß |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | 0                     |
| OUT       | Tag_Result | +1.570796 ( $\pi/2$ ) |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcuscosinuswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Cosinuswert am Eingang "Tag\_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ATAN: Arcustangenswert bilden

### Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Tangenswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von  $-\pi/2$  bis  $+\pi/2$  liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcustangenswert bilden":

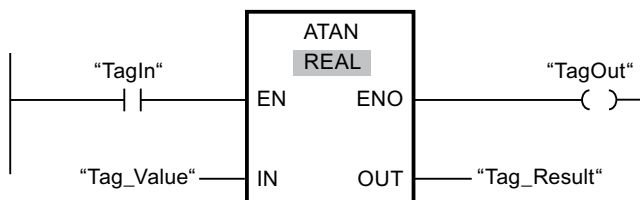
| Parameter | Deklaration | Datentyp         | Speicherbereich                 | Beschreibung                  |
|-----------|-------------|------------------|---------------------------------|-------------------------------|
| EN        | Input       | BOOL             | E, A, M, D, L                   | Freigabeeingang               |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | Freigabeausgang               |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | Tangenswert                   |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | Größe des Winkels im Bogenmaß |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | 1.0                   |
| OUT       | Tag_Result | +0.785398 ( $\pi/4$ ) |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcustangenswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Tangenswert am Eingang "Tag\_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## FRAC: Nachkommastellen ermitteln

### Beschreibung

Mit der Anweisung "Nachkommastellen ermitteln" ermitteln Sie die Nachkommastellen des Wertes am Eingang IN. Das Ergebnis der Abfrage wird am Ausgang OUT abgelegt und kann an diesem abgefragt werden. Wenn am Eingang IN z. B. der Wert 123,4567 ansteht, liefert der Ausgang OUT den Wert 0,4567.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Nachkommastellen ermitteln":

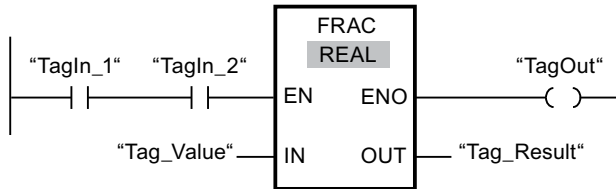
| Parameter | Deklaration | Datentyp         | Speicherbereich                 | Beschreibung                                    |
|-----------|-------------|------------------|---------------------------------|---|
| EN        | Input       | BOOL             | E, A, M, D, L                   | Freigabeeingang                                 |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | Freigabeausgang                                 |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | Wert, dessen Nachkommastellen ermittelt werden. |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | Nachkommastellen des Werts am Eingang IN        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|-------|
| IN        | Tag_Value  | 2.555 |
| OUT       | Tag_Result | 0.555 |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Nachkommastellen ermitteln" gestartet. Die Nachkommastellen aus dem Wert des Operanden "Tag\_Value" werden in den Operanden "Tag\_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Ausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**EXPT: Potenzieren**

**Beschreibung**

Mit der Anweisung "Potenzieren" potenzieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2. Das Ergebnis der Anweisung wird am Ausgang OUT abgelegt und kann an diesem abgefragt werden ( $OUT = IN1^{IN2}$ ).

Der Eingang IN1 kann nur mit gültigen Gleitpunktzahlen belegt werden. Zur Belegung des Eingangs IN2 sind auch Ganzzahlen zulässig.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung der Anweisung auf, z. B. es tritt einen Überlauf auf.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Potenzieren":

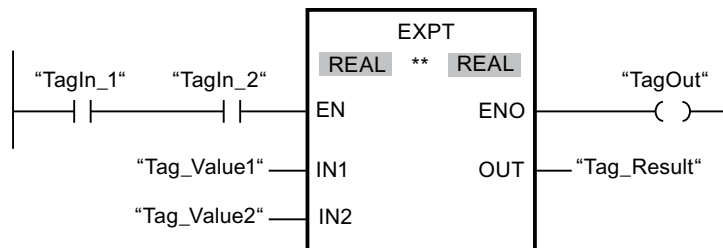
| Parameter | Deklaration | Datentyp                        | Speicherbereich                    | Beschreibung                                      |
|-----------|-------------|---------------------------------|------------------------------------|---|
| EN        | Input       | BOOL                            | E, A, M, D, L                      | Freigabeeingang                                   |
| ENO       | Output      | BOOL                            | E, A, M, D, L                      | Freigabeausgang                                   |
| IN1       | Input       | Gleitpunktzahlen                | E, A, M, D, L, P<br>oder Konstante | Basiswert   |
| IN2       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P<br>oder Konstante | Wert, mit dem der<br>Basiswert<br>potenziert wird |
| OUT       | Output      | Gleitpunktzahlen                | E, A, M, D, L, P                   | Ergebnis  |

Aus der Klappliste "<???"> der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Potenzieren" gestartet. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" potenziert. Das Ergebnis wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**Verschieben**

**MOVE: Wert kopieren**

**Beschreibung**

Mit der Anweisung "Wert kopieren" übertragen Sie den Inhalt des Operanden am Eingang IN zum Operanden am Ausgang OUT1. Die Übertragung erfolgt dabei immer in Richtung aufsteigender Adresse.

Die folgende Tabelle zeigt die möglichen Übertragungen für die CPU-Familie S7-1200:

| Quelle (IN)                              | Ziel (OUT1)                |   |
|--|----------------------------|---|
|  | Mit IEC-Prüfung            | Ohne IEC-Prüfung  |
| BYTE                                     | BYTE, WORD, DWORD          | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD, CHAR       |
| WORD                                     | WORD, DWORD                | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD, CHAR       |
| DWORD                                    | DWORD                      | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, REAL, TIME, DATE, TOD, CHAR |
| SINT                                     | SINT                       | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| USINT                                    | USINT, UINT, UDINT         | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| INT                                      | INT                        | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| UINT                                     | UINT, UDINT                | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| DINT                                     | DINT                       | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| UDINT                                    | UDINT                      | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| REAL                                     | REAL                       | DWORD, REAL   |
| LREAL                                    | LREAL                      | LREAL   |
| TIME                                     | TIME                       | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME                        |
| DATE                                     | DATE                       | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, DATE                        |
| TOD                                      | TOD                        | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TOD                         |
| DTL                                      | DTL                        | DTL   |
| CHAR                                     | CHAR                       | BYTE, WORD, DWORD, CHAR, Zeichen einer Zeichenkette <sup>1)</sup>                   |
| Zeichen einer Zeichenkette <sup>1)</sup> | Zeichen einer Zeichenkette | CHAR, Zeichen einer Zeichenkette  |
| ARRAY <sup>2)</sup>                      | ARRAY                      | ARRAY   |
| STRUCT                                   | STRUCT                     | STRUCT  |

| Quelle (IN)        | Ziel (OUT1)        |                    |
|--------------------|--------------------|--------------------|
|                    | Mit IEC-Prüfung    | Ohne IEC-Prüfung   |
| PLC-Datentyp (UDT) | PLC-Datentyp (UDT) | PLC-Datentyp (UDT) |
| IEC_TIMER          | IEC_TIMER          | IEC_TIMER          |
| IEC_SCOUNTER       | IEC_SCOUNTER       | IEC_SCOUNTER       |
| IEC_USCOUNTER      | IEC_USCOUNTER      | IEC_USCOUNTER      |
| IEC_COUNTER        | IEC_COUNTER        | IEC_COUNTER        |
| IEC_UCOUNTER       | IEC_UCOUNTER       | IEC_UCOUNTER       |
| IEC_DCOUNTER       | IEC_DCOUNTER       | IEC_DCOUNTER       |
| IEC_UDCOUNTER      | IEC_UDCOUNTER      | IEC_UDCOUNTER      |

Die folgende Tabelle zeigt die möglichen Übertragungen für die CPU-Familie S7-1500:

| Quelle (IN) | Ziel (OUT1)               |   |
|-------------|---------------------------|---|
|             | Mit IEC-Prüfung           | Ohne IEC-Prüfung  |
| BYTE        | BYTE, WORD, DWORD, LWORD  | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR         |
| WORD        | WORD, DWORD, LWORD        | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, S5TIME, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR |
| DWORD       | DWORD, LWORD              | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR   |
| LWORD       | LWORD                     | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LREAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR  |
| SINT        | SINT                      | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |
| USINT       | USINT, UINT, UDINT, ULINT | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |
| INT         | INT                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |
| UINT        | UINT, UDINT, ULINT        | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |
| DINT        | DINT                      | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |

| Quelle (IN)                              | Ziel (OUT1)                |   |
|--|----------------------------|---|
|  | Mit IEC-Prüfung            | Ohne IEC-Prüfung  |
| UDINT                                    | UDINT, ULINT               | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD |
| LINT                                     | LINT                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD |
| ULINT                                    | ULINT                      | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD |
| REAL                                     | REAL                       | DWORD, REAL   |
| LREAL                                    | LREAL                      | LWORD, LREAL  |
| S5TIME                                   | S5TIME                     | WORD, S5TIME  |
| TIME                                     | TIME                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME                              |
| LTIME                                    | LTIME                      | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTIME                             |
| DATE                                     | DATE                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, DATE                              |
| DT                                       | DT                         | DT  |
| LDT                                      | LDT                        | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LDT                               |
| TOD                                      | TOD                        | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TOD                               |
| LTOD                                     | LTOD                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTOD                              |
| DTL                                      | DTL                        | DTL   |
| CHAR                                     | CHAR                       | BYTE, WORD, DWORD, LWORD, CHAR, Zeichen einer Zeichenkette <sup>1)</sup>                                      |
| Zeichen einer Zeichenkette <sup>1)</sup> | Zeichen einer Zeichenkette | CHAR, Zeichen einer Zeichenkette  |
| ARRAY <sup>2)</sup>                      | ARRAY                      | ARRAY   |
| STRUCT                                   | STRUCT                     | STRUCT  |
| COUNTER                                  | COUNTER, WORD, INT         | WORD, DWORD, INT, UINT, DINT, UDINT   |
| TIMER                                    | TIMER, WORD, INT           | WORD, DWORD, INT, UINT, DINT, UDINT   |
| PLC-Datentyp (UDT)                       | PLC-Datentyp (UDT)         | PLC-Datentyp (UDT)  |
| IEC_TIMER                                | IEC_TIMER                  | IEC_TIMER   |
| IEC_LTIMER                               | IEC_LTIMER                 | IEC_LTIMER  |
| IEC_SCOUNTER                             | IEC_SCOUNTER               | IEC_SCOUNTER  |
| IEC_USCOUNTER                            | IEC_USCOUNTER              | IEC_USCOUNTER   |
| IEC_COUNTER                              | IEC_COUNTER                | IEC_COUNTER   |
| IEC_UCOUNTER                             | IEC_UCOUNTER               | IEC_UCOUNTER  |



| Quelle (IN)       | Ziel (OUT1)     |                  |
|-------------------|-----------------|------------------|
|                   | Mit IEC-Prüfung | Ohne IEC-Prüfung |
| IEC_DCOUNT<br>R   | IEC_DCOUNTER    | IEC_DCOUNTER     |
| IEC_UDCOUN<br>TER | IEC_UDCOUNTER   | IEC_UDCOUNTER    |
| IEC_LCOUNT<br>R   | IEC_LCOUNTER    | IEC_LCOUNTER     |
| IEC_ULCOUN<br>TER | IEC_ULCOUNTER   | IEC_ULCOUNTER    |

<sup>1)</sup> Mit der Anweisung "Wert kopieren" können Sie auch einzelne Zeichen einer Zeichenkette (STRING) zu Operanden vom Datentyp CHAR übertragen. Die Nummer des zu übertragenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" übertragen. Die Übertragung vom Operanden des Datentyps CHAR zu einzelnen Zeichen einer Zeichenkette ist ebenfalls möglich. Sie können auch ein bestimmtes Zeichen einer Zeichenkette durch das Zeichen einer anderen Zeichenkette ersetzen.

<sup>2)</sup> Das Übertragen von ganzen Feldern (ARRAY) ist nur dann möglich, wenn die Feldkomponenten der Operanden am Eingang IN und am Ausgang OUT1 vom gleichen Datentyp sind.

Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 überschreitet, gehen die höherwertigen Bits des Quellwerts verloren. Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 unterschreitet, werden die höherwertigen Bits des Zielwerts mit Nullen überschrieben.

Die Anweisungsbox enthält im Grundzustand 1 Ausgang (OUT1). Die Anzahl der Ausgänge ist erweiterbar. Die eingefügten Ausgänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung wird der Inhalt des Operanden am Eingang IN zu allen verfügbaren Ausgängen übertragen. Die Anweisungsbox kann nicht erweitert werden, wenn strukturierte Datentypen (DTL, STRUCT, ARRAY) oder Zeichen einer Zeichenkette (STRING) übertragen werden.

Zum Kopieren von Operanden des Datentyps ARRAY können auch die Anweisungen "Bereich kopieren" (MOVE\_BLK) und "Bereich ununterbrechbar kopieren" (UMOVE\_BLK) verwendet werden. Operanden des Datentyps STRING können mit der Anweisung "Zeichenkette verschieben" (S\_MOVE) kopiert werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert kopieren":

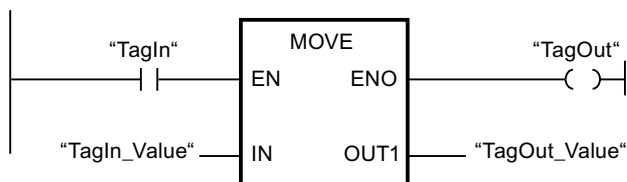
| Parameter | Deklaration | Datentyp |         | Speicherbereich | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|-----------------|
|           |             | S7-1200  | S7-1500 |                 |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | Freigabeausgang |

| Parameter | Deklaration | Datentyp   |  | Speicherbereich              | Beschreibung                                   |
|-----------|-------------|--|--|------------------------------|--|
|           |             | S7-1200  | S7-1500  |                              |  |
| IN        | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, DATE, TIME, TOD, DTL, CHAR, STRUCT, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT) | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, DATE, DT, LDT, S5TIME, TIME, LTIME, TOD, LTOD, DTL, CHAR, STRUCT, ARRAY, TIMER, COUNTER, IEC-Datentypen, PLC-Datentyp (UDT) | E, A, M, D, L oder Konstante | Quellwert                                      |
| OUT1      | Output      | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, DATE, TIME, TOD, DTL, CHAR, STRUCT, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT) | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, DATE, DT, LDT, S5TIME, TIME, LTIME, TOD, LTOD, DTL, CHAR, STRUCT, ARRAY, TIMER, COUNTER, IEC-Datentypen, PLC-Datentyp (UDT) | E, A, M, D, L                | Operand, in den der Quellwert übertragen wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0011 1111 1010 1111 |
| OUT1      | TagOut_Value | 0011 1111 1010 1111 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Wert kopieren" ausgeführt. Die Anweisung kopiert die Inhalte des Operanden "TagIn\_Value" zum Operanden "TagOut\_Value" und setzt den Ausgang "TagOut" auf den Signalzustand "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Eingänge und Ausgänge entfernen (Seite 1311)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

MOVE\_BLK: Bereich kopieren (Seite 1769)

UMOVE\_BLK: Bereich ununterbrechbar kopieren (Seite 1775)

S\_MOVE: Zeichenkette verschieben (Seite 2364)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## FieldRead: Feld lesen

### Beschreibung

Mit der Anweisung "Feld lesen" lesen Sie eine bestimmte Komponente aus dem am Eingang MEMBER angegebenen Feld aus und übertragen deren Inhalt in die Variable am Ausgang VALUE. Den Index der zu lesenden Feldkomponente legen Sie am Eingang INDEX fest. Am Eingang MEMBER geben Sie die erste Komponente des Felds an, aus dem gelesen wird.

Die Datentypen der Feldkomponente am Eingang MEMBER und der Variablen am Ausgang VALUE müssen mit dem Datentyp der Anweisung "Feld lesen" übereinstimmen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die am Eingang INDEX angegebene Feldkomponente ist in dem am Eingang MEMBER angegebenen Feld nicht definiert.
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Feld lesen":

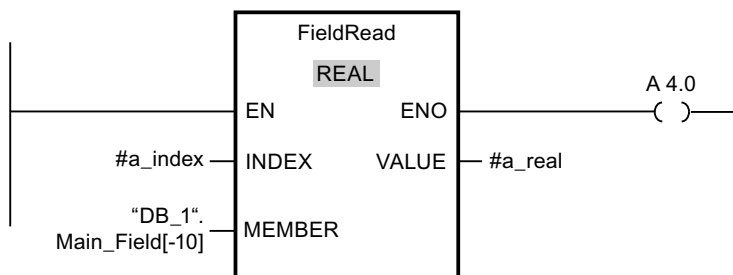
| Parameter | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung   |
|-----------|-------------|--|--|---------------------------------|--|
|           |             | S7-1200  | S7-1500  |                                 |  |
| EN        | Input       | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeeingang  |
| ENO       | Output      | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeausgang  |
| INDEX     | Input       | DINT   | DINT   | E, A, M, D, L, P oder Konstante | Index der Feldkomponente, deren Inhalt ausgelesen wird.        |
| MEMBER    | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD und CHAR als Komponente einer ARRAY-Variablen | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD und CHAR als Komponente einer ARRAY-Variablen | D, L                            | Erste Komponente des Felds, aus dem gelesen wird.              |
| VALUE     | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD und CHAR                                      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD und CHAR                                      | E, A, M, D, L, P                | Operand, in den der Inhalt der Feldkomponente übertragen wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Variable               | Wert   |
|-----------|------------------------|--|
| INDEX     | a_index                | 4  |
| MEMBER    | "DB_1".Main_Field[-10] | Erste Komponente des Felds "Main_Field[-10..10] of REAL" im Datenbaustein "DB_1" |
| VALUE     | a_real                 | Komponente mit Index 4 des Felds "Main_Field[-10..10] of REAL"                   |

Die Feldkomponente mit Index 4 wird aus dem Feld "Main\_Field[-10..10] of REAL" ausgelesen und in die Variable "a\_real" geschrieben. Die zu lesende Feldkomponente wird durch den Wert am Eingang INDEX festgelegt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## FieldWrite: Feld schreiben

### Beschreibung

Mit der Anweisung "Feld schreiben" übertragen Sie den Inhalt der Variablen am Eingang VALUE in eine bestimmte Komponente des Felds am Ausgang MEMBER. Den Index der Feldkomponente, die beschrieben wird, legen Sie durch den Wert am Eingang INDEX fest. Am Ausgang MEMBER geben Sie die erste Komponente des Felds an, in das geschrieben wird.

Die Datentypen der am Ausgang MEMBER angegebenen Feldkomponente und der Variablen am Eingang VALUE müssen mit dem Datentyp der Anweisung "Feld schreiben" übereinstimmen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die am Eingang INDEX angegebene Feldkomponente ist in dem am Ausgang MEMBER angegebenen Feld nicht definiert.
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Feld schreiben":

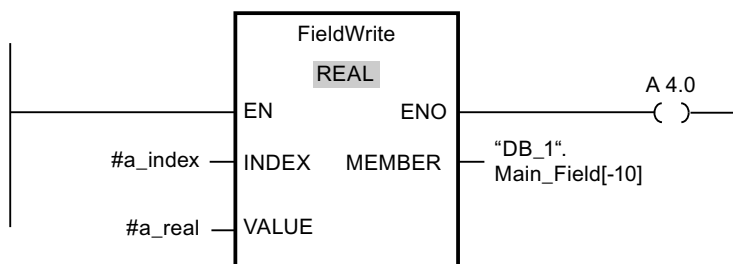
| Parameter | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung  |
|-----------|-------------|--|--|---------------------------------|---|
|           |             | S7-1200  | S7-1500  |                                 |   |
| EN        | Input       | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeeingang   |
| ENO       | Output      | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeausgang   |
| INDEX     | Input       | DINT   | DINT   | E, A, M, D, L, P oder Konstante | Index der Feldkomponente, die mit dem Inhalt von VALUE beschrieben wird.      |
| VALUE     | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD und CHAR                                      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD und CHAR                                      | E, A, M, D, L, P oder Konstante | Operand, dessen Inhalt kopiert wird.  |
| MEMBER    | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD und CHAR als Komponente einer ARRAY-Variablen | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD und CHAR als Komponente einer ARRAY-Variablen | D, L                            | Erste Komponente des Felds, in welches der Inhalt von VALUE geschrieben wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand                | Wert   |
|-----------|------------------------|--|
| INDEX     | a_index                | 4  |
| VALUE     | a_real                 | 10.54  |
| MEMBER    | "DB_1".Main_Field[-10] | Erste Komponente des Felds "Main_Field[-10..10] of REAL" im Datenbaustein "DB_1" |

Der Wert "10.54" der Variablen "a\_real" wird in die Feldkomponente mit Index 4 des Felds "Main\_Field[-10..10] of REAL" geschrieben. Der Index der Feldkomponente, in die der Inhalt der Variablen "a\_real" übertragen wird, wird durch den Wert am Eingang INDEX festgelegt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## MOVE\_BLK: Bereich kopieren

### Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Eingang COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein Array of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

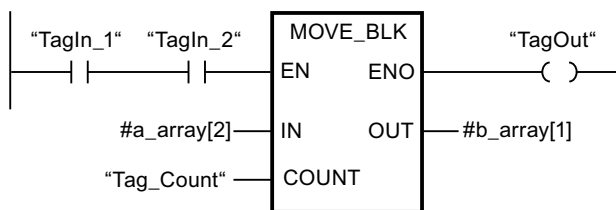
| Parameter         | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung  |
|-------------------|-------------|--|--|---------------------------------|---|
|                   |             | S7-1200  | S7-1500  |                                 |   |
| EN                | Input       | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeeingang   |
| ENO               | Output      | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeausgang   |
| IN <sup>1)</sup>  | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, CHAR, TOD | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | D, L                            | Das erste Element des Quellbereichs, das kopiert wird                                   |
| COUNT             | Input       | USINT, UINT, UDINT   | USINT, UINT, UDINT, ULINT  | E, A, M, D, L, P oder Konstante | Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden.        |
| OUT <sup>1)</sup> | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, CHAR, TOD | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | D, L                            | Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden |

<sup>1)</sup> Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:



| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Bereich kopieren" ausgeführt. Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a\_array und kopiert deren Inhalte in die Ausgangsvariable #b\_array, beginnend bei dem zweiten Element. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## MOVE\_BLK\_VARIANT: Bereich kopieren

### Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Sie können Elemente eines Arrays in ein anderes Array des selben Datentyps kopieren. Die Größen (Anzahl der Elemente) von Quell- und Ziel-Array dürfen unterschiedlich sein. Sie können mehrere Elemente innerhalb eines Arrays umkopieren oder einzelne Elemente kopieren.

Zum Zeitpunkt der Bausteinerstellung, wenn Sie die Anweisung verwenden, muss der Array noch nicht bekannt sein, da die Quelle und das Ziel per VARIANT übergeben werden.

Die Zählung an den Parametern SRC\_INDEX und DEST\_INDEX beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des Arrays.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als zur Verfügung gestellt werden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

| Parameter | Deklaration | Datentyp                                  | Speicherbereich              | Beschreibung   |
|-----------|-------------|---|------------------------------|--|
| EN        | Input       | BOOL                                      | E, A, M, D, L                | Freigabeeingang  |
| ENO       | Output      | BOOL                                      | E, A, M, D, L                | Freigabeausgang  |
| SRC       | Input       | VARIANT<br>(Array oder einzelnes Element) | E, A, D, L                   | Das erste Element des Quellbereichs, das kopiert wird  |
| COUNT     | Input       | UDINT                                     | E, A, M, D, L oder Konstante | Gibt die Anzahl der Elemente, die kopiert werden, an.<br><br>Geben Sie dem Parameter COUNT den Wert „1“, wenn am Parameter SRC oder am Parameter DEST kein Array angegeben ist.  |
| SRC_INDEX | Input       | UDINT                                     | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• Wenn am Parameter SRC ein Array angegeben ist, dann gibt der Parameter SRC_INDEX das erste Element am Parameter SRC an, das kopiert werden soll.</li> <li>• Wenn am Parameter SRC kein Array angegeben ist, dann geben Sie dem Parameter SRC_INDEX den Wert „0“.</li> </ul> |

| Parameter  | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|------------|-------------|----------|------------------------------|---|
| DEST_INDEX | Input       | UDINT    | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• Wenn am Parameter DEST ein Array angegeben ist, dann gibt der Parameter DEST_INDEX das erste Element am Parameter DEST an, das kopiert werden soll.</li> <li>• Wenn am Parameter DEST kein Array angegeben ist, dann geben Sie dem Parameter DEST_INDEX den Wert „0“.</li> </ul> |
| DEST       | InOut       | VARIANT  | E, A, D, L                   | Das erste Element des Quellbereichs, das kopiert wird   |
| RET_VAL    | Output      | INT      | E, A, M, D, L                | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameter RET\_VAL:

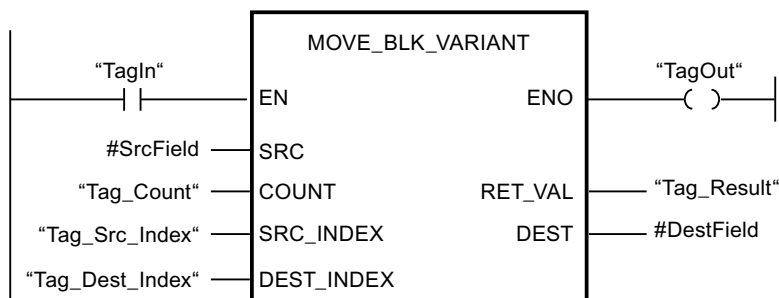
| Fehlercode*<br>(W#16#...) | Erläuterung                           |
|---------------------------|---------------------------------------|
| 0000                      | Kein Fehler                           |
| 8082                      | Datentypen stimmen nicht überein      |
| 80B4                      | Datentypen stimmen nicht überein      |
| 8151                      | Codeerzeugungsfehler am Parameter SRC |
| 8152                      | Codeerzeugungsfehler am Parameter SRC |
| 8153                      | Codeerzeugungsfehler am Parameter SRC |
| 8251                      | Ungültiger Wert am Parameter COUNT    |
| 8254                      | Ungültiger Wert am Parameter COUNT    |
| 8281                      | Ungültiger Wert am Parameter COUNT    |

| Fehlercode*<br>(W#16#...) | Erläuterung  |
|---------------------------|--|
| 8282                      | Ungültiger Wert am Parameter COUNT   |
| 8382                      | Der Wert am Parameter SRC_INDEX befindet sich außerhalb der Grenzwerte des VARIANT         |
| 8383                      | Der Wert am Parameter SRC_INDEX befindet sich außerhalb des oberen Grenzwerts des Arrays.  |
| 8482                      | Der Wert am Parameter DEST_INDEX befindet sich außerhalb der Grenzwerte des VARIANT        |
| 8483                      | Der Wert am Parameter DEST_INDEX befindet sich außerhalb des oberen Grenzwerts des Arrays. |
| 8534                      | Der Parameter DEST ist schreibgeschützt  |
| 8551                      | Codeerzeugungsfehler am Parameter DEST   |
| 8552                      | Codeerzeugungsfehler am Parameter DEST   |
| 8553                      | Codeerzeugungsfehler am Parameter DEST   |
| 85A2                      | Der Parameter DEST ist schreibgeschützt  |

\*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand       | Wert  |
|-----------|---------------|---|
| SRC       | #SrcField     | Der lokale Operand #SrcField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekanntes, UDT. |
| COUNT     | Tag_Count     | 3   |
| SRC_INDEX | Tag_Src_Index | 4   |

| Parameter  | Operand        | Wert   |
|------------|----------------|--|
| DEST_INDEX | Tag_Dest_Index | 2  |
| DEST       | #DestField     | Der lokale Operand #DestField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekanntes, UDT. |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Bereich kopieren" ausgeführt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## UMOVE\_BLK: Bereich ununterbrechbar kopieren

### Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich) ununterbrechbar. Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

### Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein Array of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

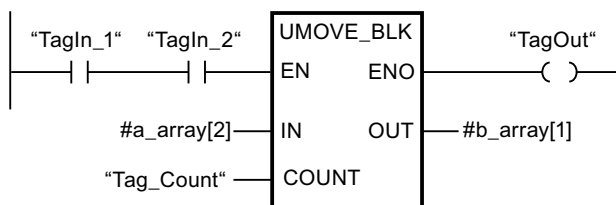
| Parameter         | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung  |
|-------------------|-------------|--|--|---------------------------------|---|
|                   |             | S7-1200  | S7-1500  |                                 |   |
| EN                | Input       | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeeingang   |
| ENO               | Output      | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeausgang   |
| IN <sup>1)</sup>  | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, CHAR, TOD | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | D, L                            | Das erste Element des Quellbereichs, das kopiert wird                                   |
| COUNT             | Input       | USINT, UINT, UDINT   | USINT, UINT, UDINT, ULINT  | E, A, M, D, L, P oder Konstante | Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden.        |
| OUT <sup>1)</sup> | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, CHAR, TOD | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | D, L                            | Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden |

<sup>1)</sup> Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Bereich ununterbrechbar kopieren" ausgeführt. Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a\_array und kopiert deren Inhalte in die Ausgangsvariable #b\_array, beginnend bei dem zweiten Element. Der Kopiervorgang kann durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Ausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## FILL\_BLK: Bereich befüllen

### Beschreibung

Mit der Anweisung "Bereich befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN selektiert und so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein Array of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

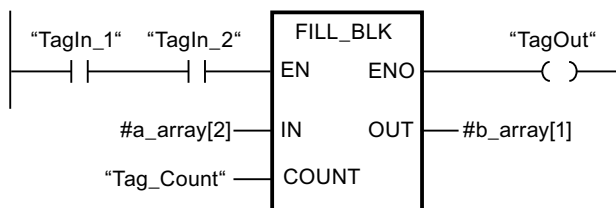
| Parameter         | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung                                  |
|-------------------|-------------|--|--|---------------------------------|---|
|                   |             | S7-1200  | S7-1500  |                                 |   |
| EN                | Input       | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeeingang                               |
| ENO               | Output      | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeausgang                               |
| IN <sup>1)</sup>  | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD, CHAR | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | E, A, M, D, L, P oder Konstante | Element, mit dem der Zielbereich befüllt wird |
| COUNT             | Input       | USINT, UINT, UDINT   | USINT, UINT, UDINT, ULINT  | E, A, M, D, L, P oder Konstante | Anzahl der Kopierwiederholungen               |
| OUT <sup>2)</sup> | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD, CHAR | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | D, L                            | Adresse im Zielbereich, ab der befüllt wird   |

<sup>1)</sup> Die angegebenen Datentypen können auch als Elemente einer ARRAY-Struktur verwendet werden.  
<sup>2)</sup> Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:



| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Bereich befüllen" ausgeführt. Die Anweisung kopiert drei Mal das dritte Element (#a\_array[2]) der Variablen #a\_array in die Ausgangsvariable #b\_array. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden der Freigabeausgang ENO und "TagOut" auf den Signalzustand "1" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## UFILL\_BLK: Bereich ununterbrechbar befüllen

### Beschreibung

Mit der Anweisung "Bereich ununterbrechbar befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN ununterbrechbar. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN selektiert und so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

---

### Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar befüllen" erhöhen.

---

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein Array of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der ARRAY-Struktur überschritten wird. Wird die Bytegrenze der ARRAY-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Mit der Anweisung "Bereich ununterbrechbar befüllen" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar befüllen":

| Parameter         | Deklaration | Datentyp   |  | Speicherbereich                 | Beschreibung                                   |
|-------------------|-------------|--|--|---------------------------------|--|
|                   |             | S7-1200  | S7-1500  |                                 |  |
| EN                | Input       | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeeingang                                |
| ENO               | Output      | BOOL   | BOOL   | E, A, M, D, L                   | Freigabeausgang                                |
| IN <sup>1)</sup>  | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, CHAR, TOD | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | E, A, M, D, L, P oder Konstante | Element, mit dem der Zielbereich befüllt wird. |
| COUNT             | Input       | USINT, UINT, UDINT   | USINT, UINT, UDINT, ULINT  | E, A, M, D, L, P oder Konstante | Anzahl der Kopierwiederholungen                |
| OUT <sup>2)</sup> | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, CHAR, TOD | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | D, L                            | Adresse im Zielbereich, ab der befüllt wird.   |

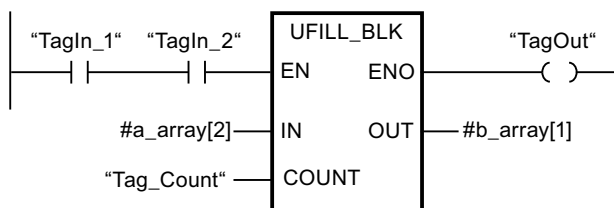
<sup>1)</sup> Die angegebenen Datentypen können auch als Elemente einer ARRAY-Struktur verwendet werden.

<sup>2)</sup> Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Bereich ununterbrechbar befüllen" ausgeführt. Die Anweisung kopiert drei Mal das dritte Element (#a\_array[2]) der Variablen #a\_array in die Ausgangsvariable #b\_array. Der Kopiervorgang kann dabei durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden der Freigabeausgang ENO und "TagOut" auf den Signalzustand "1" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

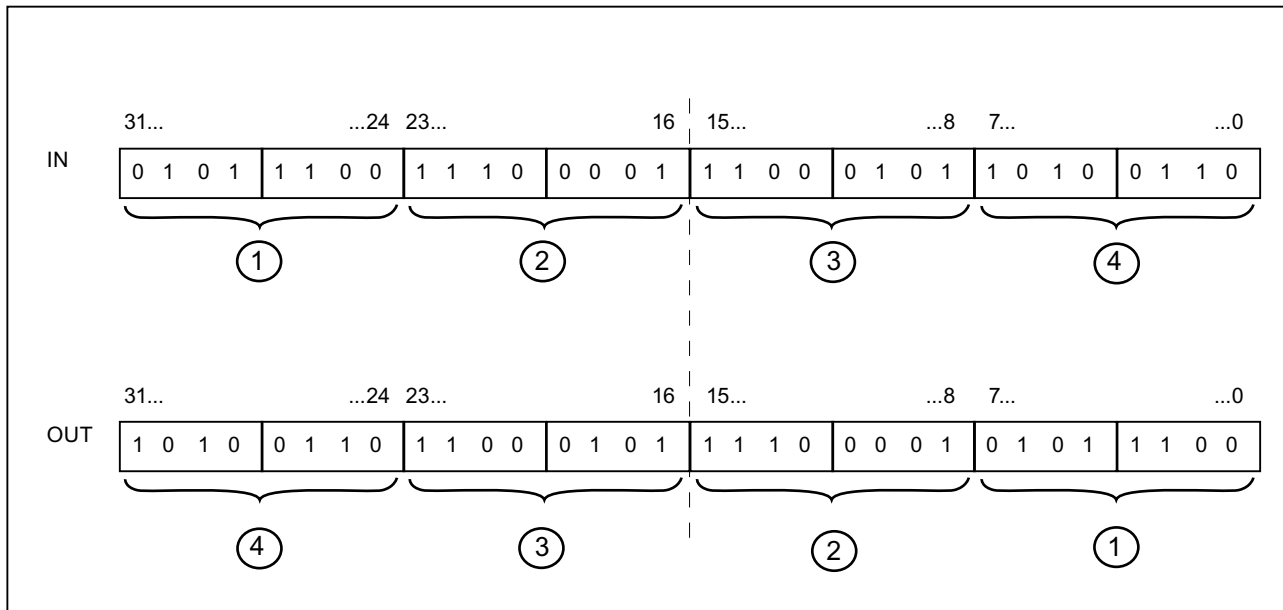
Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

### SWAP: Anordnung ändern

#### Beschreibung

Mit der Anweisung "Anordnung ändern" ändern Sie die Anordnung der Bytes am Eingang IN und fragt das Ergebnis am Ausgang OUT ab.

Das folgende Bild zeigt, wie die Bytes eines Operanden vom Datentyp DWORD mit der Anweisung "Anordnung ändern" vertauscht werden:



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anordnung ändern":

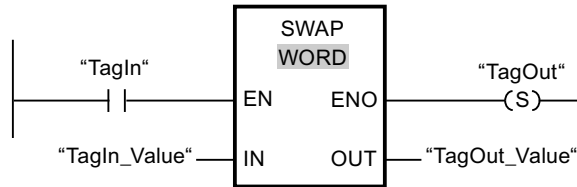
| Parameter | Deklaration | Datentyp    |                    | Speicherbereich                 | Beschreibung                             |
|-----------|-------------|-------------|--------------------|---------------------------------|--|
|           |             | S7-1200     | S7-1500            |                                 |  |
| EN        | Input       | BOOL        | BOOL               | E, A, M, D, L                   | Freigabeeingang                          |
| ENO       | Output      | BOOL        | BOOL               | E, A, M, D, L                   | Freigabeausgang                          |
| IN        | Input       | WORD, DWORD | WORD, DWORD, LWORD | E, A, M, D, L oder, P Konstante | Operand, dessen Bytes vertauscht werden. |
| OUT       | Output      | WORD, DWORD | WORD, DWORD, LWORD | E, A, M, D, L, P                | Ergebnis                                 |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0000 1111 0101 0101 |
| OUT       | TagOut_Value | 0101 0101 0000 1111 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Anordnung ändern" ausgeführt. Die Anordnung der Bytes wird geändert und im Operanden "TagOut\_Value" abgelegt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## Array-Datenbausteine

### ReadFromArrayDB: Aus Array-Datenbaustein lesen

#### Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein lesen" lesen Sie Daten aus einem Array-DB.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des Arrays.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein lesen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, aus dem gelesen wird  |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element, das gelesen wird  |
| VALUE     | InOut       | VARIANT  | E, A, M, D, L   | Wert, der gelesen wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Parameter RET\_VAL**

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...) | Erläuterung  |
|---------------------------|--|
| 0000                      | Kein Fehler  |
| 80B4                      | Datentypen stimmen nicht überein   |
| 80B5                      | Kopiervorgang wurde abgebrochen  |
| 8131                      | Der Datenbaustein existiert nicht  |
| 8132                      | Der Datenbaustein ist zu kurz  |
| 8133                      | Der Datenbaustein liegt im Ladespeicher  |
| 8134                      | Der Datenbaustein ist schreibgeschützt   |
| 8135                      | Der Datenbaustein ist kein Array-DB  |
| 8282                      | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350                      | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352                      | Codeerzeugungsfehler   |
| 8353                      | Codeerzeugungsfehler   |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

**WriteToArrayDB: In Array-Datenbaustein schreiben****Beschreibung**

Mit der Anweisung "In Array-Datenbaustein schreiben" schreiben Sie Daten in einen Array-DB.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein schreiben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, in den geschrieben wird   |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element im DB, in das geschrieben wird   |
| VALUE     | Input       | VARIANT  | E, A, M, D, L   | Wert, der geschrieben wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 80B4   | Datentypen stimmen nicht überein   |
| 80B5   | Kopiervorgang wurde abgebrochen  |
| 8131   | Der Datenbaustein existiert nicht  |
| 8132   | Der Datenbaustein ist zu kurz  |
| 8133   | Der Datenbaustein liegt im Ladespeicher  |
| 8134   | Der Datenbaustein ist schreibgeschützt   |
| 8135   | Der Datenbaustein ist kein Array-DB  |
| 8282   | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350   | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352   | Codeerzeugungsfehler   |
| 8353   | Codeerzeugungsfehler   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

### ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen

#### Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen" lesen Sie Daten aus einem Array-DB im Ladespeicher.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Wenn der Array-DB mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalfanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalfanke erfasst wird, ist die Anweisung beendet. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1" und innerhalb dieses Zyklus wird der



ausgelesene Wert am Parameter VALUE ausgegeben. Bei allen anderen Programmzyklen wird der Wert am Parameter VALUE nicht verändert.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang   |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang   |
| REQ       | Input       | BOOL     | E, A, M, D, L   | REQ = "1": Mit dem Auslesen des Array-DBs beginnen                  |
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, aus dem gelesen wird                                 |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element, das gelesen wird   |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | BUSY = "1": Das Auslesen des Array-DBs ist noch nicht abgeschlossen |
| DONE      | Output      | BOOL     | E, A, M, D, L   | DONE = "1": Die Anweisung wurde erfolgreich ausgeführt              |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| VALUE     | InOut       | VARIANT  | E, A, M, D, L   | Wert, der gelesen wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...) | Erläuterung  |
|---------------------------|--|
| 0000                      | Kein Fehler  |
| 80B4                      | Datentypen stimmen nicht überein   |
| 80B5                      | Kopiervorgang wurde abgebrochen  |
| 8131                      | Der Datenbaustein existiert nicht  |
| 8132                      | Der Datenbaustein ist zu kurz  |
| 8133                      | Der Datenbaustein liegt nicht im Ladespeicher                                  |
| 8134                      | Der Datenbaustein ist schreibgeschützt   |
| 8135                      | Der Datenbaustein ist kein Array-DB  |
| 8282                      | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350                      | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352                      | Codeerzeugungsfehler   |
| 8353                      | Codeerzeugungsfehler   |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben

### Beschreibung

Mit der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben" schreiben Sie Daten in einen Array-DB im Ladespeicher.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Wenn der Array-DB mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalflanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalflanke erfasst wird, ist die Anweisung beendet und der Wert des Parameters VALUE wird in den Datenbaustein geschrieben. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang                                       |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang                                       |
| REQ       | Input       | BOOL     | E, A, M, D, L   | REQ = "1": Mit dem Schreiben in den Array-DB beginnen |
| DB        | Input       | DB_ANY   | D               | Datenbaustein, in den geschrieben wird                |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element im DB, in das geschrieben wird                |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| VALUE     | Input       | VARIANT  | E, A, M, D, L   | Wert, der geschrieben wird   |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | BUSY = "1": Das Schreiben in den Array-DB ist noch nicht abgeschlossen   |
| DONE      | Output      | BOOL     | E, A, M, D, L   | DONE = "1": Die Anweisung wurde erfolgreich ausgeführt   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...) | Erläuterung  |
|---------------------------|--|
| 0000                      | Kein Fehler  |
| 80B4                      | Datentypen stimmen nicht überein   |
| 80B5                      | Kopiervorgang wurde abgebrochen  |
| 8131                      | Der Datenbaustein existiert nicht  |
| 8132                      | Der Datenbaustein ist zu kurz  |
| 8133                      | Der Datenbaustein liegt nicht im Ladespeicher                                  |
| 8134                      | Der Datenbaustein ist schreibgeschützt   |
| 8135                      | Der Datenbaustein ist kein Array-DB  |
| 8282                      | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350                      | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352                      | Codeerzeugungsfehler   |
| 8353                      | Codeerzeugungsfehler   |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

**Weitere**

**BLKMOV: Bereich kopieren**

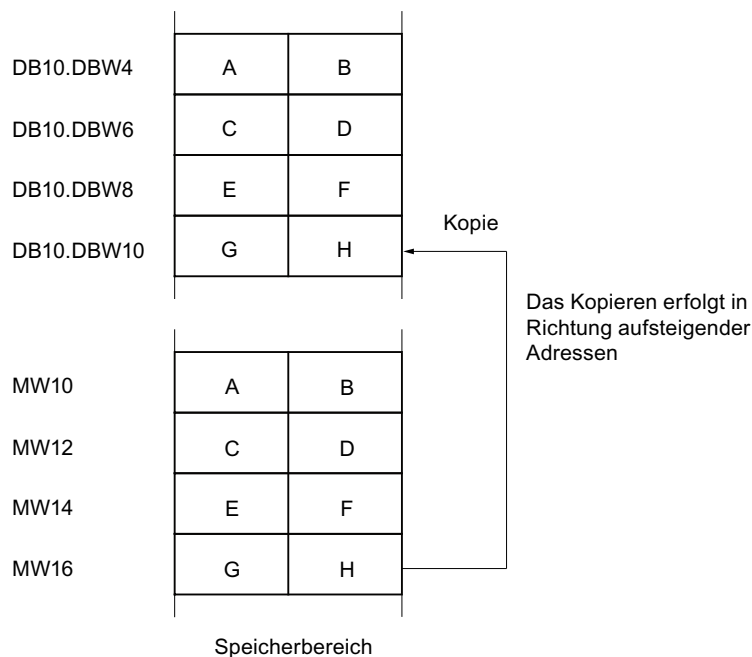
**Beschreibung**

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

**Hinweis**

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:



### Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

### Unterbrechbarkeit

Es gibt keine Begrenzung der Schachtelungstiefe.

### Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

### Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

### Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung   |
|-----------|-------------|----------|------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L    | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L    | Freigabeausgang  |
| SRCBLK    | Input       | VARIANT  | E, A, M, D, L    | Angabe des Speicherbereichs, der kopiert wird (Quellbereich).  |
| RET_VAL   | Output      | INT      | E, A, M, D, L, P | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |
| DSTBLK    | Output      | VARIANT  | E, A, M, D, L    | Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).  |

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode* (W#16#...)       | Erläuterung  |
|------------------------------|--|
| 0000                         | Kein Fehler  |
| 8091                         | Die zulässige Schachtelungstiefe wurde überschritten   |
| 8092                         | Die Anweisung kann nicht ausgeführt werden, weil ein angegebener Datenbaustein schreibgeschützt, nicht ablauffähig oder nicht geladen ist. |
| allgemeine Fehlerinformation | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"   |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## UBLKMOV: Bereich ununterbrechbar kopieren

### Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der

Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

---

#### Hinweis

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

---

### Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

### Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

### Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich



geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung   |
|-----------|-------------|----------|------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L    | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L    | Freigabeausgang  |
| SRCBLK    | Input       | VARIANT  | E, A, M, D, L    | Angabe des Speicherbereichs, der kopiert wird (Quellbereich).  |
| RET_VAL   | Output      | INT      | E, A, M, D, L, P | Fehlerinformation:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |
| DSTBLK    | Output      | VARIANT  | E, A, M, D, L    | Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).  |

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)    | Erläuterung   |
|------------------------------|---|
| 0000                         | Kein Fehler   |
| 8091                         | Der Quellbereich liegt in einem nicht ablaufrelevanten Datenbaustein. |
| allgemeine Fehlerinformation | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"                    |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

## FILL: Bereich befüllen

### Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

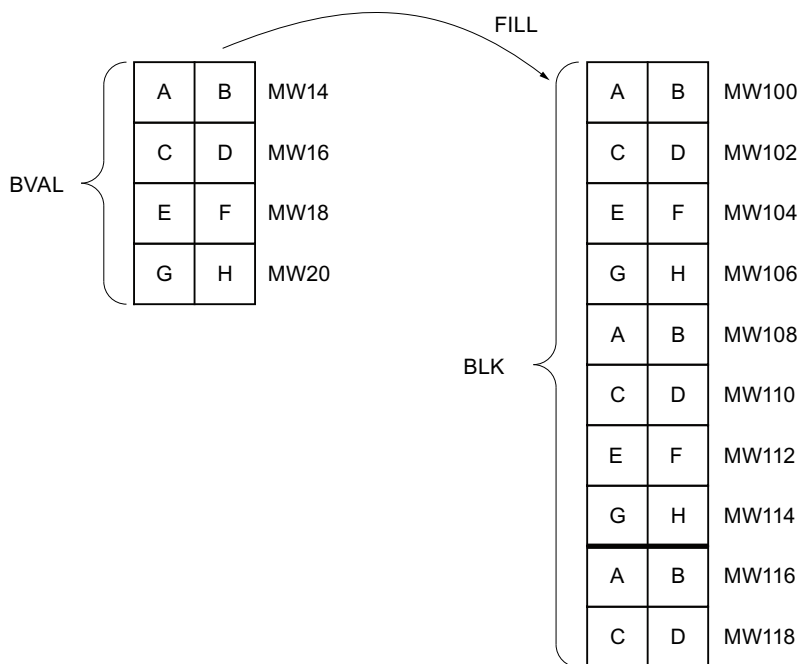
Den Quell- und Zielbereich definieren Sie durch VARIANT.

### Hinweis

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

Für Bausteine mit dem Attribut "optimierter Bausteinzugriff" können Sie die Anweisung "FILL\_BLK: Bereich befüllen" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:



Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbelegt werden.

### Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

## Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

## Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrisierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss er absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| BVAL      | Input       | VARIANT  | E, A, M, D, L   | Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.                                 |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |
| BLK       | Output      | VARIANT  | E, A, M, D, L   | Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.  |

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| allgemeine Fehlerinformation   | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen" |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

### Umwandler

#### CONVERT: Wert konvertieren

##### Beschreibung

Mit der Anweisung "Wert konvertieren" lesen Sie den Inhalt des Parameters IN und konvertieren ihn entsprechend den in der Anweisungsbox ausgewählten Datentypen. Der konvertierte Wert wird am Ausgang OUT ausgegeben.

Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.
- Am Eingang IN ist ein Operand vom Datentyp BYTE, WORD, DWORD oder LWORD angegeben, dessen hochwertigstes Bit gesetzt ist. Am Ausgang OUT ist eine vorzeichenbehaftete Ganzzahl (SINT, INT, DINT, LINT) angegeben, die die gleiche Bitlänge wie der Operand am Eingang IN hat.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert konvertieren":

| Parameter | Deklaration | Datentyp  | Speicherbereich                    | Beschreibung                   |
|-----------|-------------|---|------------------------------------|--------------------------------|
| EN        | Input       | BOOL  | E, A, M, D, L                      | Freigabeeingang                |
| ENO       | Output      | BOOL  | E, A, M, D, L                      | Freigabeausgang                |
| IN        | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>CHAR, BCD16,<br>BCD32 | E, A, M, D, L, P<br>oder Konstante | Wert, der<br>konvertiert wird. |
| OUT       | Output      | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>CHAR, BCD16,<br>BCD32 | E, A, M, D, L, P                   | Ergebnis der<br>Konvertierung  |

Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Bitfolgen (BYTE, WORD, DWORD, LWORD) sind in der Anweisungsbox nicht auswählbar. Wenn Sie einen Operanden vom Datentyp BYTE, WORD, DWORD oder LWORD an einem Parameter der Anweisung angeben, wird der Wert des Operanden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird in diesem Fall als USINT, WORD als UINT, DWORD als UDINT und LWORD als LINT interpretiert.

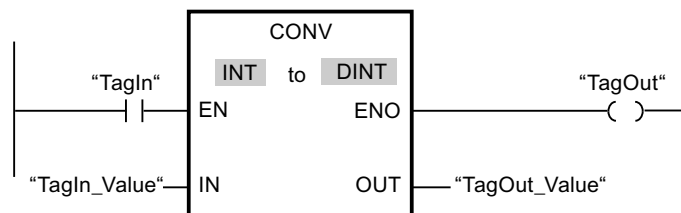
### Hinweis

Für CPU S7-1500 gilt: Es können die Datentypen DWORD und LWORD ausgewählt werden, wenn als IN-Datentyp REAL oder LREAL ausgewählt wurde.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird der Inhalt des Operanden "TagIn\_Value" gelesen und in eine Ganzzahl (32 Bit) umgewandelt. Das Ergebnis wird im Operanden "TagOut\_Value" abgelegt. Der Ausgang "TagOut" wird auf "1" gesetzt, wenn die Anweisung fehlerfrei ausgeführt wurde.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Explizite Konvertierung von CHAR (Seite 1178)

**ROUND: Zahl runden**

**Beschreibung**

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt sie in eine Ganzzahl vom Datentyp DINT um. Liegt der Eingangswert zwischen einer geraden und einer ungeraden Zahl, wird die gerade Zahl gewählt. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zahl runden":

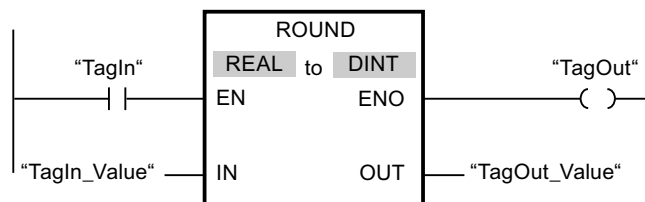
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 | Beschreibung                     |
|-----------|-------------|------------------------------|---------------------------------|----------------------------------|
| EN        | Input       | BOOL                         | E, A, M, D, L                   | Freigabeeingang                  |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | Freigabeausgang                  |
| IN        | Input       | Gleitpunktzahlen             | E, A, M, D, L, P oder Konstante | Eingangswert, der gerundet wird. |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | Ergebnis der Rundung             |

Aus der Klappliste "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert       |             |
|-----------|--------------|------------|-------------|
| IN        | TagIn_Value  | 1.50000000 | -1.50000000 |
| OUT       | TagOut_Value | 2          | -2          |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Zahl runden" ausgeführt. Die Gleitpunktzahl am Eingang "TagIn\_Value" wird zur nächsten geraden Ganzzahl gerundet und am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

### Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst höhere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Der Ausgangswert kann dabei größer oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen":

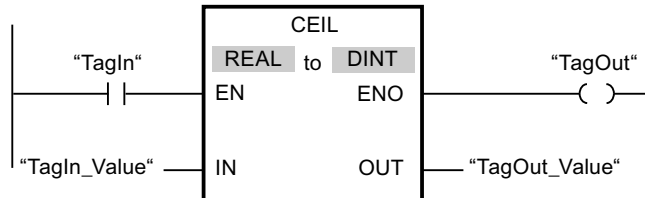
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 | Beschreibung                         |
|-----------|-------------|------------------------------|---------------------------------|--------------------------------------|
| EN        | Input       | BOOL                         | E, A, M, D, L                   | Freigabeeingang                      |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | Freigabeausgang                      |
| IN        | Input       | Gleitpunktzahlen             | E, A, M, D, L, P oder Konstante | Eingangswert                         |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | Ergebnis mit nächst höherer Ganzzahl |

Aus der Klappliste "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert       |             |
|-----------|--------------|------------|-------------|
| IN        | TagIn_Value  | 0.50000000 | -0.50000000 |
| OUT       | TagOut_Value | 1          | 0           |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" ausgeführt. Die Gleitpunktzahl am Eingang "TagIn\_Value" wird zur nächst höheren Ganzzahl gerundet und am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**FLOOR: Aus Gleitpunktzahl nächst niedrigere Ganzzahl erzeugen**

**Beschreibung**

Mit der Anweisung "Aus Gleitpunktzahl nächst niedrigere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst niedrigere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und an diesem abgefragt werden. Der Ausgangswert kann dabei kleiner oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen":

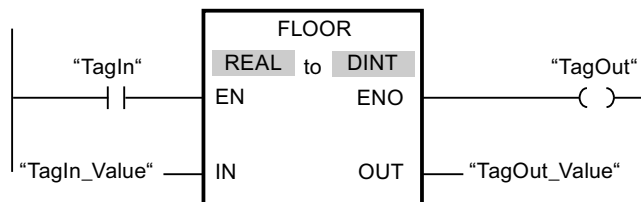
| Parameter | Deklaration | Datentyp                        | Speicherbereich                    | Beschreibung                                   |
|-----------|-------------|---------------------------------|------------------------------------|--|
| EN        | Input       | BOOL                            | E, A, M, D, L                      | Freigabeeingang                                |
| ENO       | Output      | BOOL                            | E, A, M, D, L                      | Freigabeausgang                                |
| IN        | Input       | Gleitpunktzahlen                | E, A, M, D, L, P<br>oder Konstante | Eingangswert                                   |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L, P                   | Ergebnis mit<br>nächst niedrigerer<br>Ganzzahl |

Aus der Klappliste "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert       |             |
|-----------|--------------|------------|-------------|
| IN        | TagIn_Value  | 0.50000000 | -0.50000000 |
| OUT       | TagOut_Value | 0          | -1          |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" ausgeführt. Die Gleitpunktzahl am Eingang "TagIn\_Value" wird zur nächst niedrigeren Ganzzahl gerundet und am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## TRUNC: Ganzzahl erzeugen

### Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Wert am Eingang IN eine Ganzzahl. Der Wert am Eingang IN wird als Gleitpunktzahl interpretiert. Die Anweisung selektiert nur den ganzzahligen Anteil der Gleitpunktzahl und gibt diesen ohne die Nachkommastellen am Ausgang OUT aus.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Eingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ganzzahl erzeugen":

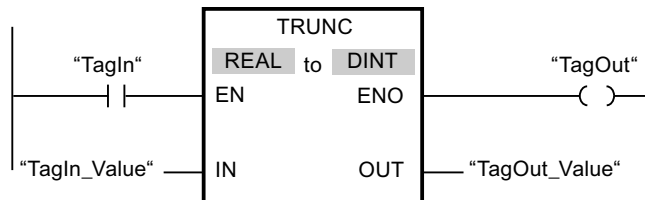
| Parameter | Deklaration | Datentyp                     | Speicherbereich              | Beschreibung                        |
|-----------|-------------|------------------------------|------------------------------|-------------------------------------|
| EN        | Input       | BOOL                         | E, A, M, D, L                | Freigabeeingang                     |
| ENO       | Output      | BOOL                         | E, A, M, D, L                | Freigabeausgang                     |
| IN        | Input       | Gleitpunktzahlen             | E, A, M, D, L oder Konstante | Eingangswert                        |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L                | Ganzzahliger Teil des Eingangswerts |

Aus der Klappliste "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert       |             |
|-----------|--------------|------------|-------------|
| IN        | TagIn_Value  | 1.50000000 | -1.50000000 |
| OUT       | TagOut_Value | 1          | -1          |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Ganzzahl erzeugen" ausgeführt. Der ganzzahlige Anteil der Gleitpunktzahl am Eingang "TagIn\_Value" wird in eine Ganzzahl umgewandelt und am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

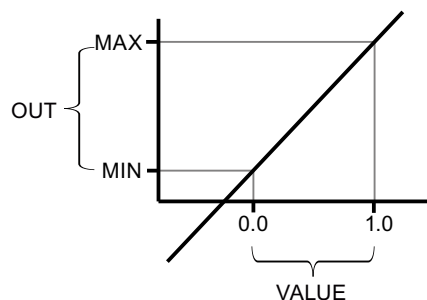
Übersicht über die gültigen Datentypen (Seite 1087)

## SCALE\_X: Skalieren

### Beschreibung

Mit der Anweisung "Skalieren" skalieren Sie den Wert am Eingang VALUE, indem Sie diesen auf einem bestimmten Wertebereich abbilden. Bei der Ausführung der Anweisung "Skalieren" wird der Gleitpunktwert am Eingang VALUE auf den Wertebereich skaliert, der durch die Parameter MIN und MAX definiert wurde. Das Ergebnis der Skalierung ist eine Ganzzahl, die am Ausgang OUT abgelegt wird.

Das folgende Bild zeigt beispielhaft, wie Werte skaliert werden können:



Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

$$\text{OUT} = [\text{VALUE} * (\text{MAX} - \text{MIN})] + \text{MIN}$$

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang MIN ist größer oder gleich dem Wert am Eingang MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Es tritt ein Überlauf auf.
- Der Wert am Eingang VALUE ist NaN (Not a number = Ergebnis einer ungültigen Rechenoperation).

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

| Parameter | Deklaration | Datentyp                        | Speicherbereich                 | Beschreibung   |
|-----------|-------------|---------------------------------|---------------------------------|--|
| EN        | Input       | BOOL                            | E, A, M, D, L                   | Freigabeeingang  |
| ENO       | Output      | BOOL                            | E, A, M, D, L                   | Freigabeausgang  |
| MIN       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Untere Grenze<br>des Wertebereichs   |
| VALUE     | Input       | Gleitpunktzahlen                | E, A, M, D, L oder<br>Konstante | Wert, der skaliert<br>wird.<br>Bei Angabe einer<br>Konstante<br>müssen Sie diese<br>deklarieren. |
| MAX       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Obere Grenze des<br>Wertebereichs  |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L                   | Ergebnis der<br>Skalierung   |

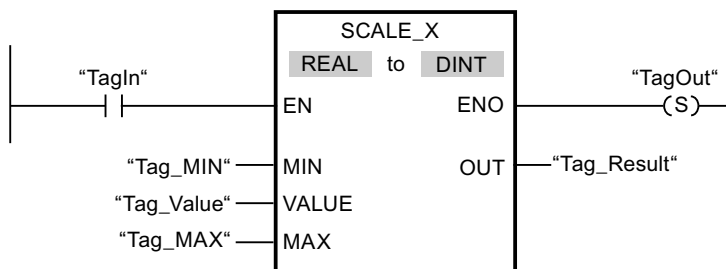
Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert |
|-----------|------------|------|
| MIN       | Tag_MIN    | 10   |
| VALUE     | Tag_Value  | 0.5  |
| MAX       | Tag_MAX    | 30   |
| OUT       | Tag_Result | 20   |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Skalieren" ausgeführt. Der Wert am Eingang "Tag\_Value" wird auf den Wertebereich skaliert, der durch die Werte an den Eingängen "Tag\_MIN" und "Tag\_MAX" definiert wurde. Das Ergebnis wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

NORM\_X: Normieren (Seite 1807)

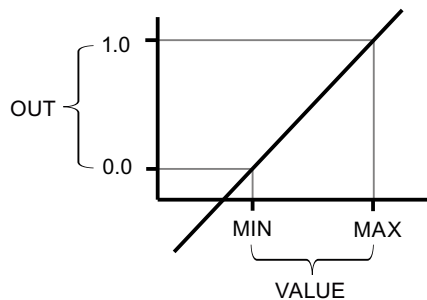
Konstanten deklarieren (Seite 1202)

## NORM\_X: Normieren

### Beschreibung

Mit der Anweisung "Normieren" normieren Sie den Wert der Variablen am Eingang VALUE, indem Sie diesen auf einer linearen Skala abbilden. Mit den Parametern MIN und MAX definieren Sie die Grenzen eines Wertebereichs, der auf die Skala gespiegelt wird. Abhängig von der Lage des zu normierenden Werts in diesem Wertebereich wird das Ergebnis berechnet und am Ausgang OUT als Gleitpunktzahl abgelegt. Wenn der zu normierende Wert gleich dem Wert am Eingang MIN ist, liefert der Ausgang OUT den Wert "0.0". Wenn der zu normierende Wert gleich dem Wert am Eingang MAX ist, liefert der Ausgang OUT den Wert "1.0".

Das folgende Bild zeigt beispielhaft, wie Werte normiert werden können:



Die Anweisung "Normieren" arbeitet mit der folgenden Gleichung:

$$OUT = (VALUE - MIN) / (MAX - MIN)$$

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang MIN ist größer oder gleich dem Wert am Eingang MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Der Wert am Eingang VALUE ist NaN (Ergebnis einer ungültigen Rechenoperation).

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Normieren":

| Parameter   | Deklaration | Datentyp                        | Speicherbereich                 | Beschreibung                       |
|---|-------------|---------------------------------|---------------------------------|------------------------------------|
| EN  | Input       | BOOL                            | E, A, M, D, L                   | Freigabeeingang                    |
| ENO   | Output      | BOOL                            | E, A, M, D, L                   | Freigabeausgang                    |
| MIN <sup>1)</sup>   | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Untere Grenze<br>des Wertebereichs |
| VALUE <sup>1)</sup>   | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Wert, der normiert<br>wird.        |
| MAX <sup>1)</sup>   | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Obere Grenze des<br>Wertebereichs  |
| OUT   | Output      | Gleitpunktzahlen                | E, A, M, D, L                   | Ergebnis der<br>Normierung         |
| <sup>1)</sup> Bei Verwendung von Konstanten an diesen drei Parametern genügt es, wenn Sie eine davon deklarieren. |             |                                 |                                 |                                    |

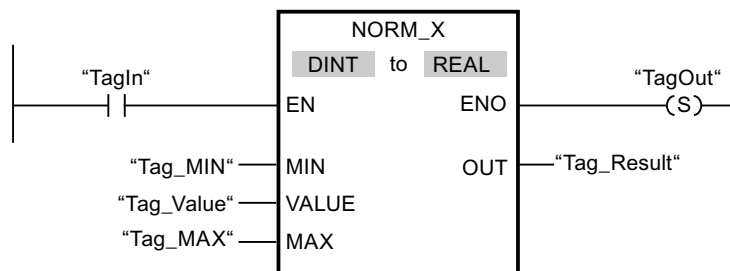
Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert |
|-----------|------------|------|
| MIN       | Tag_MIN    | 10   |
| VALUE     | Tag_Value  | 20   |
| MAX       | Tag_MAX    | 30   |
| OUT       | Tag_Result | 0.5  |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Normieren" ausgeführt. Der Wert am Eingang "Tag\_Value" wird dem Wertebereich zugeordnet, der durch die Werte an den Eingängen "Tag\_MIN" und "Tag\_MAX" definiert wurde. Entsprechend dem definierten Wertebereich wird der Variablenwert am Eingang "Tag\_Value" normiert. Das Ergebnis wird am Ausgang "Tag\_Result" als Gleitpunktzahl abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCALE\_X: Skalieren (Seite 1805)

Konstanten deklarieren (Seite 1202)

Weitere

SCALE: Skalieren

Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO\_LIM und HI\_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

$$OUT = [((FLOAT (IN) - K1)/(K2-K1)) * (HI\_LIM-LO\_LIM)] + LO\_LIM$$

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI\_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO\_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO\_LIM > HI\_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung                    |
|-----------|-------------|----------|------------------------------------|---------------------------------|
| EN        | Input       | BOOL     | E, A, M, D, L                      | Freigabeeingang                 |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang                 |
| IN        | Input       | INT      | E, A, M, D, L, P<br>oder Konstante | Eingabewert, der skaliert wird. |
| HI_LIM    | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | Oberer Grenzwert                |
| LO_LIM    | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | Unterer Grenzwert               |



| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| BIPOLAR   | Input       | BOOL     | E, A, M, D, L oder Konstante | Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen:<br>1: Bipolar<br>0: Unipolar |
| OUT       | Output      | REAL     | E, A, M, D, L, P             | Ergebnis der Anweisung   |
| RET_VAL   | Output      | WORD     | E, A, M, D, L, P             | Fehlerinformationen  |

### Parameter RET\_VAL

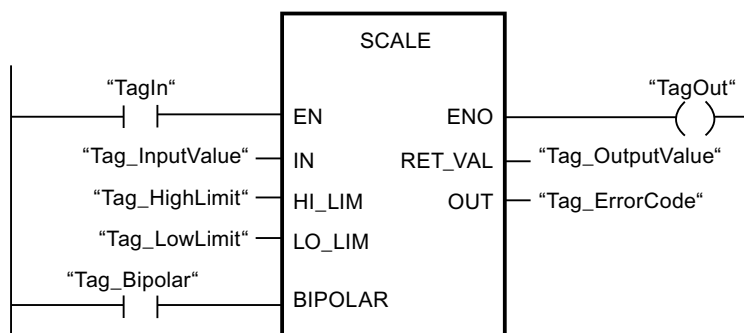
Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode* (W#16#...)       | Erläuterung  |
|------------------------------|--|
| 0000                         | Kein Fehler  |
| 0008                         | Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1" |
| allgemeine Fehlerinformation | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"   |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand         | Wert        |
|-----------|-----------------|-------------|
| IN        | Tag_InputValue  | 22          |
| HI_LIM    | Tag_HighLimit   | 100.0       |
| LO_LIM    | Tag_LowLimit    | 0.0         |
| BIPOLAR   | Tag_Bipolar     | 1           |
| OUT       | Tag_OutputValue | 50.03978588 |
| RET_VAL   | Tag_ErrorCode   | W#16#0000   |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

**UNSCALE: Deskalieren**

**Beschreibung**

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandeln sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO\_LIM und HI\_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

$$OUT = [ ((IN-LO\_LIM)/(HI\_LIM-LO\_LIM)) * (K2-K1) ] + K1$$

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert des oberen Grenzwert (HI\_LIM) ist, wird das Ergebnis der Anweisung auf den Wert der Konstante "K2" gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante des unteren Grenzwert (LO\_LIM) ist, wird das Ergebnis der Anweisung auf den Wert der Konstante "K1" gesetzt und ein Fehler ausgegeben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deskalieren":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung   |
|-----------|-------------|----------|------------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L                      | Freigabeeingang  |
| ENO       | Input       | BOOL     | E, A, M, D, L                      | Freigabeausgang  |
| IN        | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | Eingabewert, der in einen ganzzahligen Wert deskaliert wird.   |
| HI_LIM    | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | Oberer Grenzwert   |
| LO_LIM    | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | Unterer Grenzwert  |
| BIPOLAR   | Input       | BOOL     | E, A, M, D, L oder<br>Konstante    | Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen:<br>1: Bipolar<br>0: Unipolar |
| OUT       | Output      | INT      | E, A, M, D, L, P                   | Ergebnis der Anweisung   |
| RET_VAL   | Output      | WORD     | E, A, M, D, L, P                   | Fehlerinformationen  |

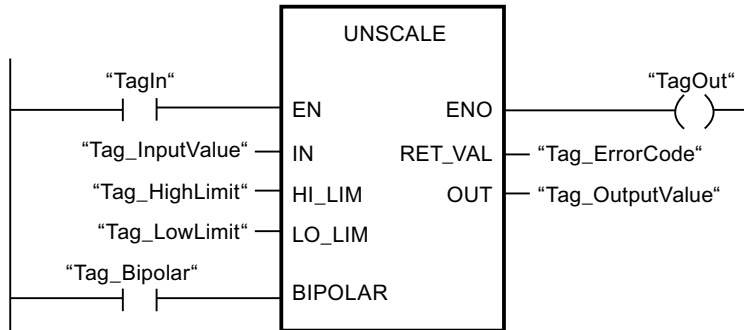
## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)   | Erläuterung  |
|---|--|
| 0000  | Kein Fehler  |
| 0008  | Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM). |
| allgemeine Fehlerinformationen  | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"   |
| *Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand         | Wert        |
|-----------|-----------------|-------------|
| IN        | Tag_InputValue  | 50.03978588 |
| HI_LIM    | Tag_HighLimit   | 100.0       |
| LO_LIM    | Tag_LowLimit    | 0.0         |
| BIPOLAR   | Tag_Bipolar     | 1           |
| OUT       | Tag_OutputValue | 22          |
| RET_VAL   | Tag_ErrorCode   | W#16#0000   |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

**Programmsteuerung**

---( JMP ): Springen bei VKE = 1

**Beschreibung**

Mit der Anweisung "Springen bei VKE = 1" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisung angegeben.

Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

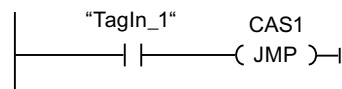
Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn die Bedingung am Eingang der Anweisung nicht erfüllt ist (VKE = 0), wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

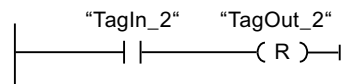
## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

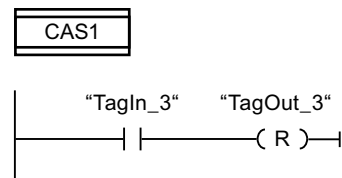
Netzwerk 1



Netzwerk 2



Netzwerk 3



Wenn der Operand "TagIn\_1" den Signalzustand "1" liefert, wird die Anweisung "Springen bei VKE = 1" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn\_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut\_3" zurückgesetzt.

## ---( JMPN ): Springen bei VKE = 0

### Beschreibung

Mit der Anweisung "Springen bei VKE = 0" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort, wenn das Verknüpfungsergebnis am Eingang der Anweisung "0" ist. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisung angegeben.

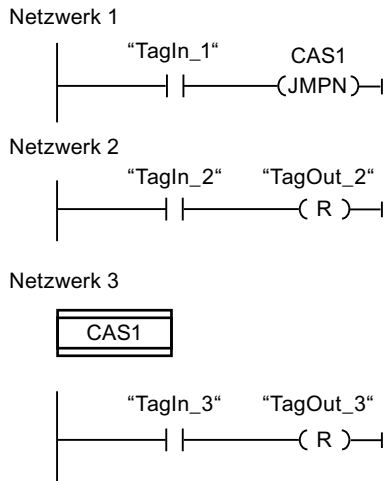
Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "0" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn das Verknüpfungsergebnis am Eingang der Anweisung "1" ist, wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn\_1" den Signalzustand "0" liefert, wird die Anweisung "Springen bei VKE = 0" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn\_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut\_3" zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**LABEL: Sprungmarke**

**Beschreibung**

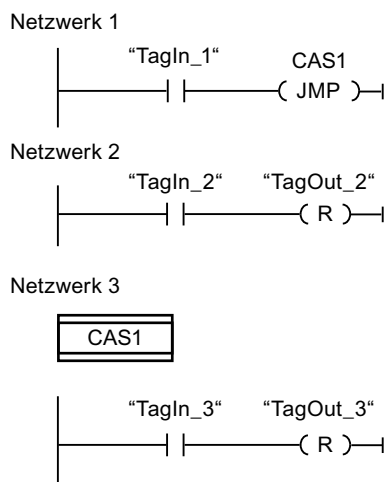
Mit einer Sprungmarke kennzeichnen Sie ein Zielnetzwerk, in dem bei einem ausgeführten Sprung die Programmbearbeitung fortgesetzt wird.

Die Sprungmarke und die Anweisung, in der die Sprungmarke als Sprungziel angegeben ist, müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb des Bausteins nur einmal vergeben werden. Sie können maximal 32 Sprungmarken bei Verwendung einer CPU S7-1200 und maximal 256 Sprungmarken bei Verwendung einer CPU S7-1500 deklarieren.

In einem Netzwerk kann nur eine Sprungmarke platziert werden. Jede Sprungmarke kann von mehreren Stellen angesprungen werden.

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn\_1" den Signalzustand "1" liefert, wird die Anweisung "Springen bei VKE = 1" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn\_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut\_3" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

## JMP\_LIST: Sprungliste definieren

### Beschreibung

Mit der Anweisung "Sprungliste definieren" definieren Sie mehrere bedingte Sprünge und setzen die Programmbearbeitung, abhängig vom Wert des Parameters K, in einem bestimmten Netzwerk fort.

Die Sprünge definieren Sie durch Sprungmarken (LABEL), die Sie an den Ausgängen der Anweisungsbox angeben. Die Anzahl der Ausgänge ist in der Anweisungsbox erweiterbar. Sie können maximal 32 Ausgänge bei Verwendung einer CPU S7-1200 und maximal 256 Ausgänge bei Verwendung einer CPU S7-1500 deklarieren.

Die Nummerierung der Ausgänge beginnt mit dem Wert "0" und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. An den Ausgängen der Anweisung können nur Sprungmarken angegeben werden. Die Angabe von Anweisungen oder Operanden ist nicht zulässig.

Durch den Wert des Parameters K geben Sie die Nummer des Ausgangs und damit die Sprungmarke an, bei der die Programmbearbeitung fortgesetzt wird. Wenn der Wert am Parameter K größer als die Anzahl der verfügbaren Ausgänge ist, wird die Programmbearbeitung im nächsten Netzwerk des Bausteins fortgesetzt.

Die Anweisung "Sprungliste definieren" wird nur ausgeführt, wenn der Signalzustand am Freigabeeingang EN "1" ist.

**Parameter**

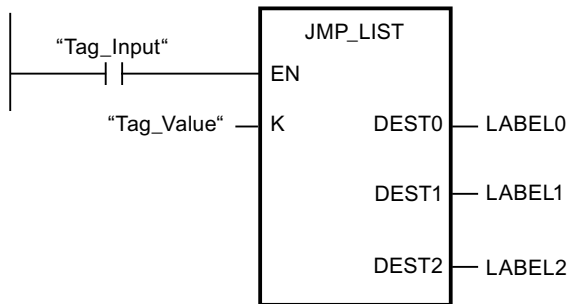
Die folgende Tabelle zeigt die Parameter der Anweisung "Sprungliste definieren":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L                | Freigabeeingang  |
| K         | Input       | UINT     | E, A, M, D, L oder Konstante | Gibt die Nummer des Ausgangs und damit den Sprung an, der ausgeführt wird. |
| DEST0     | -           | -        | -                            | Erste Sprungmarke  |
| DEST1     | -           | -        | -                            | Zweite Sprungmarke   |
| DESTn     | -           | -        | -                            | Optionale Sprungmarken   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand/Sprungmarke | Wert   |
|-----------|---------------------|--|
| K         | Tag_Value           | 1  |
| Dest0     | LABEL0              | Sprung in das Netzwerk, das mit der Sprungmarke "LABEL0" gekennzeichnet ist. |



| Parameter | Operand/Sprungmarke | Wert   |
|-----------|---------------------|--|
| Dest1     | LABEL1              | Sprung in das Netzwerk, das mit der Sprungmarke "LABEL1" gekennzeichnet ist. |
| Dest2     | LABEL2              | Sprung in das Netzwerk, das mit der Sprungmarke "LABEL2" gekennzeichnet ist. |

Wenn der Operand "Tag\_Input" den Signalzustand "1" liefert, wird die Anweisung "Sprungliste definieren" ausgeführt. Die Programmbearbeitung wird entsprechend dem Wert des Operanden "Tag\_Value" in dem Netzwerk fortgesetzt, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

## SWITCH: Sprungverteiler

### Beschreibung

Mit der Anweisung "Sprungverteiler" definieren Sie mehrere Programmsprünge, die abhängig vom Ergebnis einer oder mehrer Vergleichsanweisungen ausgeführt werden.

Den zu vergleichenden Wert geben Sie am Parameter K an. Dieser Wert wird mit den Werten verglichen, die die einzelnen Eingänge liefern. Die Art des Vergleichs wählen Sie für jeden Eingang einzeln aus. Die Verfügbarkeit der verschiedenen Vergleichsanweisungen hängt dabei vom Datentyp der Anweisung ab.

Die folgende Tabelle zeigt die Vergleichsanweisungen, die abhängig vom gewählten Datentyp verfügbar sind:

| Datentyp  |  | Anweisung      | Syntax |
|---|--|----------------|--------|
| S7-1200   | S7-1500  |                |        |
| Bitfolgen   | Bitfolgen  | Gleich         | ==     |
|   |  | Ungleich       | <>     |
| Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE, TOD | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME, DATE,<br>TOD, LTOD, LDT | Gleich         | ==     |
|   |  | Ungleich       | <>     |
|   |  | Größer gleich  | >=     |
|   |  | Kleiner gleich | <=     |
|   |  | Größer         | >      |
|   |  | Kleiner        | <      |

Den Datentyp der Anweisung wählen Sie aus der Klappliste "<???">" der Anweisungsbox. Wenn Sie eine Vergleichsanweisung auswählen und der Datentyp der Anweisung noch nicht

festgelegt ist, werden in der Klappliste "<???" nur die Datentypen aufgelistet, die für die ausgewählte Vergleichsanweisung zulässig sind.

Die Bearbeitung der Anweisung beginnt mit dem ersten Vergleich und wird so lange ausgeführt, bis eine Vergleichsbedingung erfüllt ist. Ist eine Vergleichsbedingung erfüllt, werden die nachfolgenden Vergleichsbedingungen nicht berücksichtigt. Wenn keine der angegebenen Vergleichsbedingungen erfüllt ist, wird der Sprung am Ausgang ELSE ausgeführt. Wenn am Ausgang ELSE kein Programmsprung definiert ist, wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

Die Anzahl der Ausgänge ist in der Anweisungsbox erweiterbar. Die Nummerierung der Ausgänge beginnt mit dem Wert "0" und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. An den Ausgängen der Anweisung geben Sie Sprungmarken (LABEL) an. Die Angabe von Anweisungen oder Operanden ist an den Ausgängen der Anweisung nicht zulässig.

Zu jedem zusätzlichen Ausgang wird automatisch auch ein Eingang eingefügt. Der an einem Ausgang programmierte Sprung wird ausgeführt, wenn die Vergleichsbedingung des entsprechenden Eingangs erfüllt ist.

### Parameter

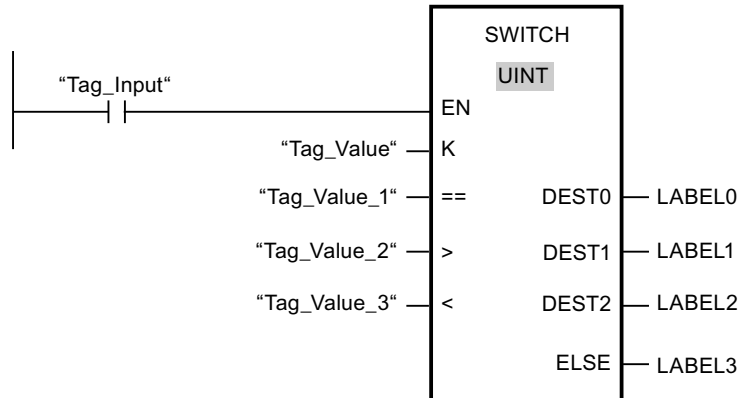
Die folgende Tabelle zeigt die Parameter der Anweisung "Sprungverteiler":

| Parameter         | Deklaration | Datentyp   |  | Speicherbereich              | Beschreibung  |
|-------------------|-------------|--|--|------------------------------|---|
|                   |             | S7-1200  | S7-1500  |                              |   |
| EN                | Input       | BOOL   | BOOL   | E, A, M, D, L                | Freigabeeingang   |
| K                 | Input       | UINT   | UINT   | E, A, M, D, L oder Konstante | Gibt den zu vergleichenden Wert an.   |
| <Vergleichswerte> | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, DATE, TOD, LTOD, LDT | E, A, M, D, L oder Konstante | Eingangswerte, mit denen der Wert des Parameters K verglichen wird.                     |
| DEST0             | -           | -  | -  | -                            | Erste Sprungmarke   |
| DEST1             | -           | -  | -  | -                            | Zweite Sprungmarke  |
| DEST(n)           | -           | -  | -  | -                            | Optionale Sprungmarken (n = 2 bis 99)   |
| ELSE              | -           | -  | -  | -                            | Programmsprung, der ausgeführt wird, wenn keine der Vergleichsbedingungen erfüllt wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand/Sprungmarke | Wert  |
|-----------|---------------------|---|
| K         | Tag_Value           | 23  |
| ==        | Tag_Value_1         | 20  |
| >         | Tag_Value_2         | 21  |
| <         | Tag_Value_3         | 19  |
| Dest 0    | LABEL0              | Sprung zur Sprungmarke "LABEL0", wenn der Wert des Parameters K gleich 20 ist.      |
| Dest 1    | LABEL1              | Sprung zur Sprungmarke "LABEL1", wenn der Wert des Parameters K größer 21 ist.      |
| Dest 2    | LABEL2              | Sprung zur Sprungmarke "LABEL2", wenn der Wert des Parameters K kleiner als 19 ist. |
| ELSE      | LABEL 3             | Sprung zur Sprungmarke "LABEL3", wenn keine der Vergleichsbedingungen erfüllt ist.  |

Wenn der Operand "Tag\_Input" auf den Signalzustand "1" wechselt, wird die Anweisung "Sprungverteiler" ausgeführt. Die Programmbearbeitung wird in dem Netzwerk fortgesetzt, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

--(RET): Zurück springen

**Beschreibung**

Mit der Anweisung "Zurück springen" beenden Sie die Bearbeitung eines Bausteins. Hierdurch ergeben sich drei Arten, wie die Bearbeitung eines Bausteins beendet werden kann:

- Ohne Aufruf der Anweisung "Zurück springen"  
Nach der Ausführung des letzten Netzwerks wird der Baustein verlassen. Das ENO der Aufruffunktion wird auf den Signalzustand "1" gesetzt.
- Aufruf der Anweisung "Zurück springen" mit Vorverknüpfung (siehe Beispiel)  
Hat der linke Anschluss den Signalzustand "1", wird der Baustein verlassen. Das ENO der Aufruffunktion entspricht dem Operanden.
- Aufruf der Anweisung "Zurück springen" ohne Vorverknüpfung  
Der Baustein wird verlassen. Das ENO der Aufruffunktion entspricht dem Operanden.

---

**Hinweis**

In einem Netzwerk darf nur eine springende Spule verwendet werden ("Zurück springen", "Springen bei VKE = 1", "Springen bei VKE = 0").

---

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "Zurück springen" "1" ist, wird die Programmbearbeitung im aktuell aufgerufenen Baustein beendet und im aufrufenden Baustein (z. B. im aufrufenden OB) nach der Aufruffunktion fortgesetzt. Der Status (ENO) der Aufruffunktion wird durch den Parameter der Anweisung bestimmt. Dieser kann die folgenden Werte annehmen:

- RLO
- TRUE/FALSE
- <Operand>

Um die Werte des Parameters einzustellen, doppelklicken Sie auf die Anweisung und wählen Sie in der Klappliste den entsprechenden Wert aus.

Die folgende Tabelle zeigt den Status der Aufruffunktion, wenn die Anweisung "Zurück springen" in einem Netzwerk innerhalb des aufgerufenen Bausteins programmiert ist:

| VKE | Parameterwert | ENO der Aufruffunktion  |
|-----|---------------|---|
| 1   | RLO           | 1   |
|     | TRUE          | 1   |
|     | FALSE         | 0   |
|     | <Operand>     | <Operand>   |
| 0   | RLO           | Die Programmbearbeitung wird im nächsten Netzwerk des aufgerufenen Bausteins fortgesetzt. |
|     | TRUE          |   |
|     | FALSE         |   |
|     | <Operand>     |   |

Wenn ein OB beendet wird, dann wird durch das Ablaufebensystem ein anderer Baustein ausgewählt und gestartet bzw. weiter ausgeführt:

- Wenn der Programmzyklus-OB beendet wurde, wird dieser wieder gestartet.
- Wenn ein OB beendet wird, der einen anderen Baustein unterbrochen hat (z. B. ein Alarm-OB), so wird der unterbrochene Baustein (z. B. Programmzyklus-OB) ausgeführt.

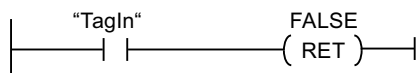
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zurück springen":

| Parameter                                    | Deklaration | Datentyp | Speicherbereich | Beschreibung                                |
|--|-------------|----------|-----------------|---|
| Status der aufrufenden Funktion bei VKE = 1: |             |          |                 |   |
| RLO  | -           | -        |                 | Wird auf den Signalzustand des VKE gesetzt. |
| TRUE   | -           | -        |                 | 1   |
| FALSE  | -           | -        |                 | 0   |
| <Operand>                                    | Input       | BOOL     | E, A, M, D, L   | Signalzustand des angegebenen Operanden     |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Zurück springen" ausgeführt. Die Programmbearbeitung wird im aufgerufenen Baustein beendet und im aufrufenden Baustein fortgesetzt. Der Freigabeausgang ENO der Aufruffunktion wird auf den Signalzustand "0" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Laufzeitsteuerung

### ENDIS\_PW: Passwort-Legitimierung einschränken und freigeben

#### Beschreibung

Mit der Anweisung "Passwort-Legitimierung einschränken und freigeben" legen Sie fest, ob für die CPU eine Legitimation erlaubt ist oder nicht. Sie können legitimierte Verbindungen verhindern, auch wenn das korrekte Passwort bekannt ist. Wenn Sie die Anweisung aufrufen

und der Parameter REQ den Signalzustand "0" hat, wird lediglich der momentan eingestellte Zustand an den Ausgangsparametern angezeigt, aber keine Einstellung verändert. Wenn der Parameter REQ den Signalzustand "1" hat, wird der Signalzustand von den Eingangsparametern (F\_PWD, FULL\_PWD, R\_PWD, HMI\_PWD) übernommen. FALSE bedeutet, dass die Legitimation per Passwort nicht erlaubt ist, TRUE bedeutet, das Passwort ist verwendbar. Das Sperren bzw. Freigeben der Passwörter kann einzeln erlaubt oder untersagt werden. Es können z. B. alle Passwörter nicht erlaubt sein, außer dem Failsafe-Passwort. So können Sie die Zugriffsmöglichkeiten auf eine kleine Anwendergruppe begrenzen. Die Ausgangsparameter (F\_PWD\_ON, FULL\_PWD\_ON, R\_PWD\_ON, HMI\_PWD\_ON) zeigen immer den aktuellen Status der Passwortverwendung, unabhängig vom Parameter REQ.

Die gleiche Einstellung kann im Frontpanel der CPU vorgenommen werden und die CPU speichert die jeweils letzte Einstellung.

Um ein ungewolltes Aussperren zu verhindern, kann bei einer S7-1500-CPU der Schutz durch ein Umlegen des Betriebsartenschalters nach STOP außer Kraft gesetzt werden. Der Schutz wird nach dem Umlegen des Betriebsartenschalters in RUN automatisch wieder eingestellt, ohne dass die Anweisung "Passwort-Legitimierung einschränken und freigeben" erneut aufgerufen werden muss oder im Frontpanel weitere Aktionen notwendig sind.

Die Anweisung muss immer komplett durchführbar sein, d. h. es muss z. B. der Parameter F\_PWD immer den Signalzustand "0" haben, damit die Einstellungen abgespeichert werden können.

Unter den folgenden Bedingungen können gesperrte Passwörter wieder zugelassen werden:

- Die CPU wurde auf die Werkseinstellung zurückgesetzt.
- Das Frontpanel der S7-1500-CPU unterstützt einen Dialog, mit dessen Hilfe Sie zum passenden Menü navigieren und die Passwörter wieder zulassen können.
- Beim Aufruf der Anweisung "Passwort-Legitimierung einschränken und freigeben" hat der Eingangsparameter des gewünschten Passworts den Signalzustand "1".
- Den Betriebsartenschalter auf STOP setzen. Die Einschränkung der Passwort-Legitimierung wird wieder eingerichtet, sobald der Schalter wieder nach RUN bewegt wird.
- Das Stecken eines Transfermoduls in eine S7-1200-CPU.

---

#### Hinweis

Die Anweisung "Passwort-Legitimierung einschränken und freigeben" sperrt den Zugriff auf HMI Panels, wenn das HMI Passwort nicht freigegeben ist.

---

#### Hinweis


Bereits bestehende und legitimierte Verbindungen behalten ihre Zugriffs-Rechte und können über die Anweisung "Passwort-Legitimierung einschränken und freigeben" nicht eingeschränkt werden.

---

### Vorgehensweise bei verlorenem Passwort bei einer S7-1200-CPU

Wenn Sie das Passwort für eine passwortgeschützte S7-1200-CPU verloren haben, löschen Sie das passwortgeschützte Programm mit einer leeren Übertragungskarte. Die leere

Übertragungskarte löscht den internen Ladespeicher der CPU. Dann können Sie ein neues Anwenderprogramm aus STEP 7 Basic in die CPU laden.

|  |
|--|
|  <b>WARNUNG</b>   |
| <b>Übertragungskarte stecken</b>   |
| Wenn Sie eine Übertragungskarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann. |

Sie müssen die Übertragungskarte ziehen, bevor Sie die CPU in RUN versetzen.

### Auswirkungen der Passwortverwendung auf die Betriebsarten

Die folgende Tabelle zeigt, welche Auswirkungen die Passwortverwendung durch die Anweisung "Passwort-Legitimierung einschränken und freigeben" auf die Betriebsarten und die entsprechenden Anwenderaktionen hat.

| Aktion  | Passwortschutz durch die Anweisung  |
|---|---|
| Grundzustand nach <ul style="list-style-type: none"> <li>• Betriebsartenschalter auf STOP</li> <li>• Speicher manuell zurücksetzen (PG, Schalter, Änderung des MC (Motion Control))</li> <li>• Zurücksetzen auf Werkseinstellung</li> </ul> | Nicht aktiviert<br>(Keine Einschränkungen)  |
| Grundzustand nach NETZ-EIN  | Aktiviert (wenn vor NETZ-AUS eine Sperre aktiviert war)<br>Die Möglichkeit, Passwörter nicht zu erlauben, ist remanent. |
| Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch das Beenden der Anweisung, einen Fehler oder Kommunikation) oder STOP -> ANLAUF/ RUN/HALT   | Aktiviert<br>Die Passwörter dürfen weiterhin nicht verwendet werden.  |

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Passwort-Legitimierung einschränken und freigeben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung    |
|-----------|-------------|----------|-----------------|-----------------|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang |

| Parameter   | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-------------|-------------|----------|-----------------|---|
| REQ         | Input       | BOOL     | E, A, M, D, L   | Wenn der Parameter REQ den Signalzustand "0" führt, wird der aktuell eingestellte Signalzustand der Passwörter abgefragt.   |
| F_PWD       | Input       | BOOL     | E, A, M, D, L   | Lese-/Schreibzugriff inklusive Failsafe <ul style="list-style-type: none"> <li>• F_PWD = "0": Passwort nicht erlauben</li> <li>• F_PWD = "1": Passwort erlauben</li> </ul>            |
| FULL_PWD    | Input       | BOOL     | E, A, M, D, L   | Lese-/Schreibzugriff <ul style="list-style-type: none"> <li>• FULL_PWD = "0": Passwort nicht erlauben</li> <li>• FULL_PWD = "1": Passwort erlauben</li> </ul>                         |
| R_PWD       | Input       | BOOL     | E, A, M, D, L   | Lesezugriff <ul style="list-style-type: none"> <li>• R_PWD = "0": Passwort nicht erlauben</li> <li>• R_PWD = "1": Passwort erlauben</li> </ul>  |
| HMI_PWD     | Input       | BOOL     | E, A, M, D, L   | HMI-Zugriff <ul style="list-style-type: none"> <li>• HMI_PWD = "0": Passwort nicht erlauben</li> <li>• HMI_PWD = "1": Passwort erlauben</li> </ul>                                    |
| F_PWD_ON    | Output      | BOOL     | E, A, M, D, L   | Status Lese-/Schreibzugriff inklusive Failsafe <ul style="list-style-type: none"> <li>• F_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• F_PWD_ON = "1": Passwort erlaubt</li> </ul> |
| FULL_PWD_ON | Output      | BOOL     | E, A, M, D, L   | Status Lese-/Schreibzugriff <ul style="list-style-type: none"> <li>• FULL_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• FULL_PWD_ON = "1": Passwort erlaubt</li> </ul>              |



| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|------------|-------------|----------|-----------------|---|
| R_PWD_ON   | Output      | BOOL     | E, A, M, D, L   | Status Lesezugriff <ul style="list-style-type: none"> <li>• R_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• R_PWD_ON = "1": Passwort erlaubt</li> </ul>     |
| HMI_PWD_ON | Output      | BOOL     | E, A, M, D, L   | Status HMI-Zugriff <ul style="list-style-type: none"> <li>• HMI_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• HMI_PWD_ON = "1": Passwort erlaubt</li> </ul> |
| RET_VAL    | Output      | WORD     | E, A, M, D, L   | Fehlerinformation   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 8090   | Die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird nicht unterstützt |
| 80D0   | Das Passwort für Failsafe ist nicht konfiguriert   |
| 80D1   | Der Lese-/Schreibzugriff ist nicht konfiguriert  |
| 80D2   | Der Lesezugriff ist nicht konfiguriert   |
| 80D3   | Der HMI-Zugriff ist nicht konfiguriert   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## RE\_TRIGR: Zyklusüberwachungszeit neu starten

### Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zykluszeitüberwachung der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann unabhängig von der Priorität in allen Bausteinen aufgerufen werden.

Wenn die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm, einem Diagnosealarm oder einem Weckalarm, aufgerufen wird, wird die Anweisung nicht ausgeführt und der Freigabeausgang ENO auf den Signalzustand "0" gesetzt.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann max. 10-mal in einem Programmzyklus aufgerufen werden.

#### **Parameter**

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter.

#### **Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

#### **STP: Programm beenden**

##### **Beschreibung**

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOPP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOPP hängen von der CPU-Konfiguration ab.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wechselt die CPU in den Betriebszustand STOPP und die Programmbearbeitung wird beendet. Der Signalzustand am Ausgang der Anweisung wird nicht ausgewertet.

Wenn das VKE am Eingang der Anweisung "0" ist, wird die Anweisung nicht ausgeführt.

##### **Parameter**

Die Anweisung "Programm beenden" hat keine Parameter.

##### **Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

#### **GET\_ERROR: Fehler lokal abfragen**

##### **Beschreibung**

Mit der Anweisung "Fehler lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Wenn das System während der Bausteinbearbeitung Fehler meldet, wird zum ersten aufgetretenen Fehler eine ausführliche Information im Operanden am Ausgang ERROR gespeichert.

Am Ausgang ERROR können nur Operanden vom Systemdatentyp "ErrorStruct" angegeben werden. Der Systemdatentyp "ErrorStruct" gibt die genaue Struktur vor, in der die Informationen zum aufgetretenen Fehler gespeichert werden. Mithilfe weiterer Anweisungen können Sie diese Struktur auswerten und eine entsprechende Reaktion programmieren. Wenn der erste aufgetretene Fehler behoben ist, gibt die Anweisung die Fehlerinformation zum nächsten aufgetretenen Fehler aus.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler lokal abfragen":

| Parameter | Deklaration | Datentyp    | Speicherbereich | Beschreibung        |
|-----------|-------------|-------------|-----------------|---------------------|
| ERROR     | Output      | ErrorStruct | D, L            | Fehlerinformationen |

### Datentyp "ErrorStruct"

Die folgende Tabelle zeigt die Struktur des Datentyps "ErrorStruct":

| Strukturkomponente | Datentyp | Beschreibung   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
|--------------------|----------|--|------|--------|--------|-----|-----|-----|---|--|--|--|--|--|---|--|--|--|--|--------|---|--|--|------|--|--|---|----------|-------|--|--------|--|---|--|--|------|--|--------|---|--|--|------|--------|--------|
| ERROR_ID           | WORD     | Fehler-ID  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| FLAGS              | BYTE     | Zeigt, ob der Fehler während eines Bausteinaufrufs aufgetreten ist.<br>16#01: Fehler während eines Bausteinaufrufs<br>16#00: Kein Fehler während eines Bausteinaufrufs   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| REACTION           | BYTE     | Voreingestellte Reaktion:<br>0: Ignorieren (Schreibfehler),<br>1: Mit dem Ersatzwert "0" fortsetzen (Lesefehler),<br>2: Anweisung überspringen (Systemfehler)  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| CODE_ADDRESS       | CREF     | Informationen zur Adresse und Art des Bausteins  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| BLOCK_TYPE         | BYTE     | Bausteinart, in dem der Fehler aufgetreten ist:<br>1: OB<br>2: FC<br>3: FB   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| CB_NUMBER          | UINT     | Nummer des Codebausteins   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| OFFSET             | UDINT    | Verweis auf den internen Speicher  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| MODE               | BYTE     | Zugriffsart: Abhängig von der Zugriffsart können die folgenden Informationen ausgegeben werden:  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
|                    |          | <table border="1"> <thead> <tr> <th>Mode</th> <th>(A)</th> <th>(B)</th> <th>(C)</th> <th>(D)</th> <th>(E)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>1</td> <td></td> <td></td> <td></td> <td></td> <td>Offset</td> </tr> <tr> <td>2</td> <td></td> <td></td> <td>Area</td> <td></td> <td></td> </tr> <tr> <td>3</td> <td>Location</td> <td>Scope</td> <td></td> <td>Number</td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td>Area</td> <td></td> <td>Offset</td> </tr> <tr> <td>5</td> <td></td> <td></td> <td>Area</td> <td>DB no.</td> <td>Offset</td> </tr> </tbody> </table> | Mode | (A)    | (B)    | (C) | (D) | (E) | 0 |  |  |  |  |  | 1 |  |  |  |  | Offset | 2 |  |  | Area |  |  | 3 | Location | Scope |  | Number |  | 4 |  |  | Area |  | Offset | 5 |  |  | Area | DB no. | Offset |
| Mode               | (A)      | (B)  | (C)  | (D)    | (E)    |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| 0                  |          |  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| 1                  |          |  |      |        | Offset |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| 2                  |          |  | Area |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| 3                  | Location | Scope  |      | Number |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| 4                  |          |  | Area |        | Offset |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |
| 5                  |          |  | Area | DB no. | Offset |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |

| Strukturkomponente      |           | Datentyp | Beschreibung  |                |                       |      |        |        |
|-------------------------|-----------|----------|---|----------------|-----------------------|------|--------|--------|
|                         |           |          | 6   | PtrNo./<br>Acc |                       | Area | DB no. | Offset |
|                         |           |          | 7   | PtrNo./<br>Acc | Slot<br>No./<br>Scope | Area | DB no. | Offset |
| OPERAND_NUMBER          |           | UINT     | Operandnummer des Maschinenbefehls  |                |                       |      |        |        |
| POINTER_NUMBER_LOCATION |           | UINT     | (A) Interner Zeiger   |                |                       |      |        |        |
| SLOT_NUMBER_SCOPE       |           | UINT     | (B) Ablagebereich im internen Speicher  |                |                       |      |        |        |
| DATA_ADDRESS            |           | NREF     | Informationen zur Adresse eines Operanden   |                |                       |      |        |        |
|                         | AREA      | BYTE     | (C) Speicherbereich:<br>L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE<br>E: 16#81<br>A: 16#82<br>M: 16#83<br>DB: 16#84, 85, 8A, 8B<br>Bereichsverletzungen bei einer direkt editierbaren<br>Variablen vom Datentyp DINT: 16#04 |                |                       |      |        |        |
|                         | DB_NUMBER | UINT     | (D) Nummer des Datenbausteins   |                |                       |      |        |        |
|                         | OFFSET    | UDINT    | (E) Relative Adresse des Operanden  |                |                       |      |        |        |

**Strukturkomponente "ERROR\_ID"**

Die folgende Tabelle zeigt die Werte, die an der Strukturkomponente "ERROR\_ID" ausgegeben werden können:

| ID*<br>(Hexadezimal) | ID*<br>(Dezimal) | Beschreibung   |
|----------------------|------------------|--|
| 0                    | 0                | Kein Fehler  |
| 2503                 | 9475             | Ungültiger Zeiger                                      |
| 2520                 | 9504             | Ungültiger STRING                                      |
| 2522                 | 9506             | Lesefehler: Operand außerhalb des gültigen Bereichs    |
| 2523                 | 9507             | Schreibfehler: Operand außerhalb des gültigen Bereichs |
| 2524                 | 9508             | Lesefehler: Ungültiger Operand                         |
| 2525                 | 9509             | Schreibfehler: Ungültiger Operand                      |
| 2528                 | 9512             | Lesefehler: Datenausrichtung                           |
| 2529                 | 9513             | Schreibfehler: Datenausrichtung                        |
| 252C                 | 9516             | Ungültiger Zeiger                                      |
| 2530                 | 9520             | Schreibfehler: Datenbaustein                           |
| 2533                 | 9523             | Unzulässig verwendeter Zeiger                          |
| 2538                 | 9528             | Zugriffsfehler: DB nicht vorhanden                     |
| 2539                 | 9529             | Zugriffsfehler: Falscher DB verwendet                  |
| 253A                 | 9530             | Globaler Datenbaustein existiert nicht                 |
| 253C                 | 9532             | Fehlerhafte Angabe oder die Funktion existiert nicht   |

| ID*<br>(Hexadezimal)  | ID*<br>(Dezimal) | Beschreibung   |
|---|------------------|--|
| 253D  | 9533             | Systemfunktion existiert nicht   |
| 253E  | 9534             | Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht                    |
| 253F  | 9535             | Systembaustein existiert nicht   |
| 2550  | 9552             | Zugriffsfehler: DB nicht vorhanden   |
| 2551  | 9553             | Zugriffsfehler: Falscher DB verwendet  |
| 2575  | 9589             | Fehler in der Programmschachtelungstiefe   |
| 2577  | 9591             | Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert. |
| 2576  | 9590             | Fehler in der Lokaldatenverteilung   |
| 25A0  | 9632             | Interner Fehler im TP  |
| 25A1  | 9633             | Variable ist schreibgeschützt  |
| 25A2  | 9634             | Ungültiger Zahlenwert der Variable   |
| 2942  | 10562            | Lesefehler: Eingang  |
| 2943  | 10563            | Schreibfehler: Ausgang   |
| *Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                  |  |

Der Freigabeausgang ENO der Anweisung "Fehler lokal abfragen" wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und eine Fehlerinformation vorliegt. Wenn eine dieser Bedingungen nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung "Fehler lokal abfragen" nicht beeinflusst.

Die Anweisung "Fehler lokal abfragen" kann auch eingesetzt werden, um an den aufrufenden Baustein eine Meldung über den Fehlerzustand weiterzuleiten. Dafür muss die Anweisung im letzten Netzwerk des aufgerufenen Bausteins positioniert werden.

---

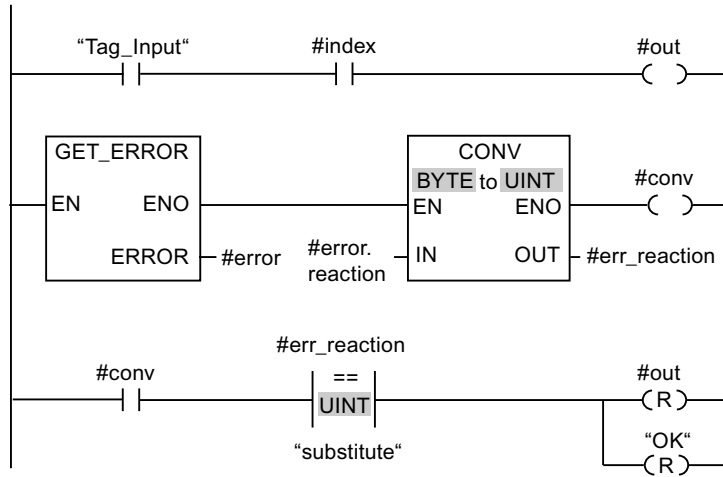
#### Hinweis

Die Anweisung "Fehler lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn "Fehler lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

---

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn ein Fehler auftritt, liefert die Anweisung "Fehler lokal abfragen" die Fehlerinformation an die lokal angelegte Struktur "#error" am Ausgang ERROR. Die Fehlerinformation wird konvertiert und mithilfe der Vergleichsanweisung "Gleich" ausgewertet. Als erster Vergleichswert wird der Anweisung die Information zur Fehlerart zugewiesen. Als zweiter Vergleichswert wird im Operanden "substitute" der Wert "1" angegeben. Wenn es sich um einen Lesefehler handelt, ist die Bedingung der Vergleichsanweisung erfüllt. Die Ausgänge "#out" und "OK" werden in diesem Fall zurückgesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zur Fehlerbehandlung (Seite 1403)
- Grundlagen der lokalen Fehlerbehandlung (Seite 1405)
- Prioritäten bei der Fehlerausgabe (Seite 1406)
- Lokale Fehlerbehandlung für einen Baustein aktivieren (Seite 1407)

**GET\_ERR\_ID: Fehler-ID lokal abfragen**

**Beschreibung**

Mit der Anweisung "Fehler-ID lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Wenn das System während der Bausteinbearbeitung Fehler meldet, wird die Fehler-ID des ersten aufgetretenen Fehlers in der Variablen am Ausgang ID gespeichert. Am Ausgang ID können nur Operanden vom Datentyp WORD angegeben werden. Wenn der erste aufgetretene Fehler behoben ist, gibt die Anweisung die Fehler-ID des nächsten aufgetretenen Fehlers aus.

Der Ausgang der Anweisung "Fehler-ID lokal abfragen" wird nur gesetzt, wenn der Eingang der Anweisung den Signalzustand "1" liefert und eine Fehlerinformation vorliegt. Wenn eine

dieser Bedingungen nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung "Fehler-ID lokal abfragen" nicht beeinflusst.

Die Anweisung "Fehler-ID lokal abfragen" kann auch eingesetzt werden, um an den aufrufenden Baustein eine Meldung über den Fehlerzustand weiterzuleiten. Dafür muss die Anweisung im letzten Netzwerk des aufgerufenen Bausteins positioniert werden.

---

#### Hinweis

Die Anweisung "Fehler-ID lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn die Anweisung "Fehler-ID lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

---

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler-ID lokal abfragen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung |
|-----------|-------------|----------|-----------------|--------------|
| ID        | Output      | WORD     | E, A, M, D, L   | Fehler-ID    |

## Parameter ID

Die folgende Tabelle zeigt die Werte, die am Parameter ID ausgegeben werden können:

| ID*<br>(Hexadezimal) | ID*<br>(Dezimal) | Beschreibung  |
|----------------------|------------------|---|
| 0                    | 0                | Kein Fehler   |
| 2503                 | 9475             | Ungültiger Zeiger   |
| 2520                 | 9504             | Ungültiger STRING   |
| 2522                 | 9506             | Lesefehler: Operand außerhalb des gültigen Bereichs           |
| 2523                 | 9507             | Schreibfehler: Operand außerhalb des gültigen Bereichs        |
| 2524                 | 9508             | Lesefehler: Ungültiger Operand                                |
| 2525                 | 9509             | Schreibfehler: Ungültiger Operand                             |
| 2528                 | 9512             | Lesefehler: Datenausrichtung                                  |
| 2529                 | 9513             | Schreibfehler: Datenausrichtung                               |
| 252C                 | 9516             | Ungültiger Zeiger   |
| 2530                 | 9520             | Schreibfehler: Datenbaustein                                  |
| 2533                 | 9523             | Unzulässig verwendeter Zeiger                                 |
| 2538                 | 9528             | Zugriffsfehler: DB nicht vorhanden                            |
| 2539                 | 9529             | Zugriffsfehler: Falscher DB verwendet                         |
| 253A                 | 9530             | Globaler Datenbaustein existiert nicht                        |
| 253C                 | 9532             | Fehlerhafte Angabe oder die Funktion existiert nicht          |
| 253D                 | 9533             | Systemfunktion existiert nicht                                |
| 253E                 | 9534             | Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht |

| ID*<br>(Hexadezimal)   | ID*<br>(Dezimal) | Beschreibung   |
|--|------------------|--|
| 253F   | 9535             | Systembaustein existiert nicht   |
| 2550   | 9552             | Zugriffsfehler: DB nicht vorhanden   |
| 2551   | 9553             | Zugriffsfehler: Falscher DB verwendet  |
| 2575   | 9589             | Fehler in der Programmschachtelungstiefe   |
| 2576   | 9590             | Fehler in der Lokaldatenverteilung   |
| 2577   | 9591             | Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert. |
| 25A0   | 9632             | Interner Fehler im TP  |
| 25A1   | 9633             | Variable ist schreibgeschützt  |
| 25A2   | 9634             | Ungültiger Zahlenwert der Variable   |
| 2942   | 10562            | Lesefehler: Eingang  |
| 2943   | 10563            | Schreibfehler: Ausgang   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                  |  |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zur Fehlerbehandlung (Seite 1403)
- Grundlagen der lokalen Fehlerbehandlung (Seite 1405)
- Prioritäten bei der Fehlerausgabe (Seite 1406)
- Lokale Fehlerbehandlung für einen Baustein aktivieren (Seite 1407)

**INIT\_RD: Alle remanenten Daten zurücksetzen**

**Beschreibung**

Mit der Anweisung "Alle remanenten Daten zurücksetzen" setzen Sie gleichzeitig alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler zurück. Die Anweisung kann nur innerhalb eines Anlauf-OBS ausgeführt werden, da die Ausführung die Programmzyklusdauer überschreiten würde.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Alle remanenten Daten zurücksetzen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung    |
|-----------|-------------|----------|-----------------|-----------------|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang |



| Parameter | Deklaration | Datentyp | Speicherbereich                       | Beschreibung   |
|-----------|-------------|----------|---------------------------------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L, T, Z<br>oder Konstante | Wenn der Eingang REQ den Signalzustand "1" führt, werden alle remanenten Daten zurückgesetzt.  |
| RET_VAL   | Output      | INT      | E, A, M, D, L                         | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter RET\_VAL

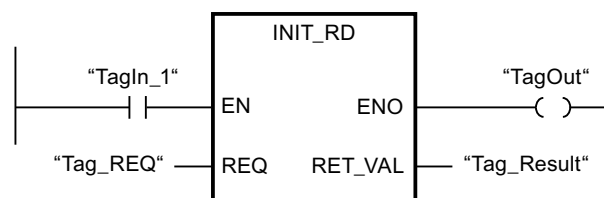
Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)    | Erläuterung   |
|------------------------------|---|
| 0000                         | Kein Fehler   |
| 80B5                         | Die Anweisung kann nicht ausgeführt werden, da sie nicht innerhalb eines Anlauf-OBs programmiert wurde. |
| allgemeine Fehlerinformation | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"  |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "Tag\_REQ" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler werden zurückgesetzt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1".

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

**WAIT: Zeitverzögerung programmieren**

**Beschreibung**

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten bis zu 32767 Mikrosekunden ( $\mu$ s) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung "Zeitverzögerung programmieren".

Die Ausführung der Anweisung kann durch höherpriorie Ereignisse unterbrochen werden.

Die Anweisung "Zeitverzögerung programmieren" liefert keine Fehlerinformationen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitverzögerung programmieren":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung                                       |
|-----------|-------------|----------|------------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L                      | Freigabeeingang                                    |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang                                    |
| WT        | Input       | INT      | E, A, M, D, L, P<br>oder Konstante | Verzögerungszeit<br>in Mikrosekunden<br>( $\mu$ s) |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**Wortverknüpfungen**

**AND: UND verknüpfen**

**Beschreibung**

Mit der Anweisung "UND verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch UND und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch UND verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits der angegebenen Werte ausgeführt.

Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch UND verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", nur wenn beide zu verknüpfende Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND verknüpfen":

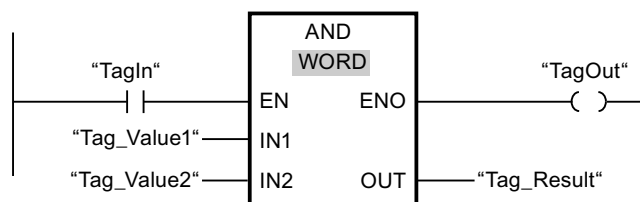
| Parameter | Deklaration | Datentyp  | Speicherbereich                 | Beschreibung                                    |
|-----------|-------------|-----------|---------------------------------|---|
| EN        | Input       | BOOL      | E, A, M, D, L                   | Freigabeeingang                                 |
| ENO       | Output      | BOOL      | E, A, M, D, L                   | Freigabeausgang                                 |
| IN1       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | Erster Wert der Verknüpfung                     |
| IN2       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | Zweiter Wert der Verknüpfung                    |
| INn       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | Weitere Eingänge, deren Werte verknüpft werden. |
| OUT       | Output      | Bitfolgen | E, A, M, D, L, P                | Ergebnis der Anweisung                          |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                |
|-----------|------------|---------------------|
| IN1       | Tag_Value1 | 0101 0101 0101 0101 |
| IN2       | Tag_Value2 | 0000 0000 0000 1111 |
| OUT       | Tag_Result | 0000 0000 0000 0101 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "UND verknüpfen" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" durch UND verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag\_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)
- Eingänge und Ausgänge entfernen (Seite 1311)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**OR: ODER verknüpfen**

**Beschreibung**

Mit der Anweisung "ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle Bits der angegebenen Variablen ausgeführt.

Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch ODER verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER verknüpfen":

| Parameter | Deklaration | Datentyp  | Speicherbereich                 | Beschreibung                |
|-----------|-------------|-----------|---------------------------------|-----------------------------|
| EN        | Input       | BOOL      | E, A, M, D, L                   | Freigabeeingang             |
| ENO       | Output      | BOOL      | E, A, M, D, L                   | Freigabeausgang             |
| IN1       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | Erster Wert der Verknüpfung |

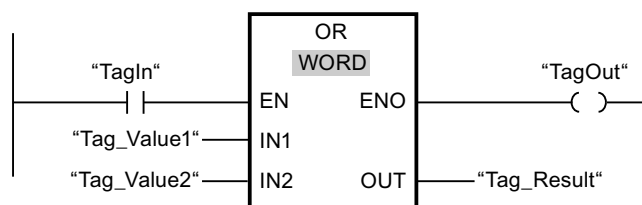
| Parameter | Deklaration | Datentyp  | Speicherbereich                 | Beschreibung                                    |
|-----------|-------------|-----------|---------------------------------|---|
| IN2       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | Zweiter Wert der Verknüpfung                    |
| INn       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | Weitere Eingänge, deren Werte verknüpft werden. |
| OUT       | Output      | Bitfolgen | E, A, M, D, L, P                | Ergebnis der Anweisung                          |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                |
|-----------|------------|---------------------|
| IN1       | Tag_Value1 | 0101 0101 0101 0101 |
| IN2       | Tag_Value2 | 0000 0000 0000 1111 |
| OUT       | Tag_Result | 0101 0101 0101 1111 |

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung "ODER verknüpfen" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" durch ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag\_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## XOR: EXKLUSIV ODER verknüpfen

### Beschreibung

Mit der Anweisung "EXKLUSIV ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch EXKLUSIV ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch EXKLUSIV ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits des angegebenen Werts ausgeführt.

Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch EXKLUSIV ODER verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", wenn eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "1" oder "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER verknüpfen":

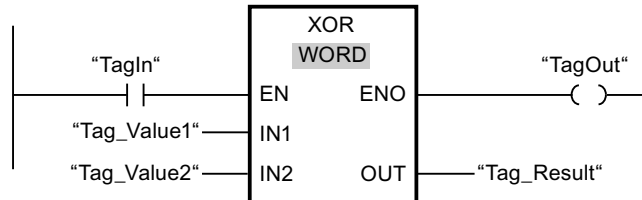
| Parameter | Deklaration | Datentyp  | Speicherbereich                    | Beschreibung                                    |
|-----------|-------------|-----------|------------------------------------|---|
| EN        | Input       | BOOL      | E, A, M, D, L                      | Freigabeeingang                                 |
| ENO       | Output      | BOOL      | E, A, M, D, L                      | Freigabeausgang                                 |
| IN1       | Input       | Bitfolgen | E, A, M, D, L, P<br>oder Konstante | Erster Wert der Verknüpfung                     |
| IN2       | Input       | Bitfolgen | E, A, M, D, L, P<br>oder Konstante | Zweiter Wert der Verknüpfung                    |
| INn       | Input       | Bitfolgen | E, A, M, D, L, P<br>oder Konstante | Weitere Eingänge, deren Werte verknüpft werden. |
| OUT       | Output      | Bitfolgen | E, A, M, D, L, P                   | Ergebnis der Anweisung                          |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                |
|-----------|------------|---------------------|
| IN1       | Tag_Value1 | 0101 0101 0101 0101 |
| IN2       | Tag_Value2 | 0000 0000 0000 1111 |
| OUT       | Tag_Result | 0101 0101 0101 1010 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "EXKLUSIV ODER verknüpfen" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" durch EXKLUSIV ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag\_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in KOP-Elemente einfügen (Seite 1309)

Eingänge und Ausgänge entfernen (Seite 1311)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## INV: Einerkomplement erzeugen

### Beschreibung

Mit der Anweisung "Einerkomplement erzeugen" invertieren Sie den Signalzustand der Bits am Eingang IN. Bei der Bearbeitung der Anweisung wird der Wert am Eingang IN mit einer Hexadezimalen Schablone (W#16#FFFF für 16 Bit-Zahlen oder DW#16#FFFF FFFF für 32 Bit-Zahlen) durch EXKLUSIV ODER verknüpft. Dadurch wird der Signalzustand der einzelnen Bits umgekehrt und am Ausgang OUT ausgegeben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einerkomplement erzeugen":

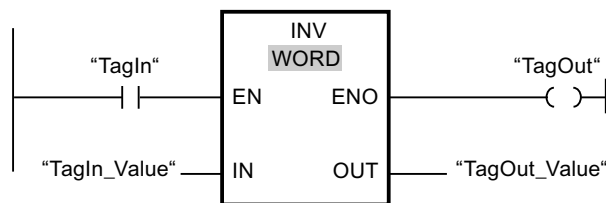
| Parameter | Deklaration | Datentyp                 | Speicherbereich                    | Beschreibung                                   |
|-----------|-------------|--------------------------|------------------------------------|--|
| EN        | Input       | BOOL                     | E, A, M, D, L                      | Freigabeeingang                                |
| ENO       | Output      | BOOL                     | E, A, M, D, L                      | Freigabeausgang                                |
| IN        | Input       | Bitfolgen,<br>Ganzzahlen | E, A, M, D, L, P<br>oder Konstante | Eingangswert                                   |
| OUT       | Output      | Bitfolgen,<br>Ganzzahlen | E, A, M, D, L, P                   | Einerkomplement<br>des Wertes am<br>Eingang IN |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert      |         |
|-----------|--------------|-----------|---------|
| IN        | TagIn_Value  | W#16#000F | W#16#7E |
| OUT       | TagOut_Value | W#16#FFF0 | W#16#81 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Einerkomplement erzeugen" ausgeführt. Die Anweisung kehrt den Signalzustand der einzelnen Bits am Eingang "TagIn\_Value" um und schreibt das Ergebnis in den Ausgang "TagOut\_Value". Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)



## DECO: Decodieren

### Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert am Eingang IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Eingang IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Decodieren":

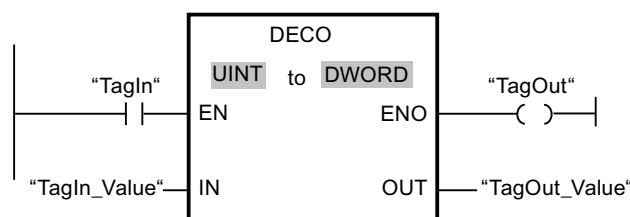
| Parameter | Deklaration | Datentyp  | Speicherbereich                    | Beschreibung   |
|-----------|-------------|-----------|------------------------------------|--|
| EN        | Input       | BOOL      | E, A, M, D, L                      | Freigabeeingang  |
| ENO       | Output      | BOOL      | E, A, M, D, L                      | Freigabeausgang  |
| IN        | Input       | UINT      | E, A, M, D, L, P<br>oder Konstante | Position des Bits<br>im Ausgangswert,<br>das gesetzt wird. |
| OUT       | Output      | Bitfolgen | E, A, M, D, L, P                   | Ausgangswert   |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

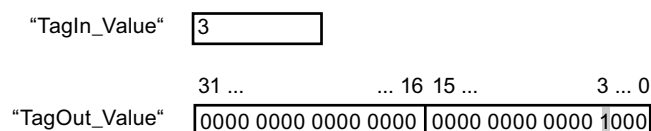
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:



Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung "Decodieren" ausgeführt. Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "TagIn\_Value" am Eingang ab und setzt das dritte Bit im Wert des Operanden "TagOut\_Value" am Ausgang.

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**ENCO: Encodieren**

**Beschreibung**

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben sie am Ausgang OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Eingang IN und schreibt dessen Bitnummer in die Variable am Ausgang OUT.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Encodieren":

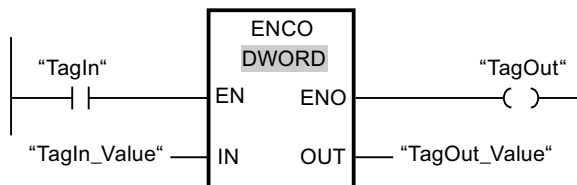
| Parameter | Deklaration | Datentyp  | Speicherbereich                 | Beschreibung    |
|-----------|-------------|-----------|---------------------------------|-----------------|
| EN        | Input       | BOOL      | E, A, M, D, L                   | Freigabeeingang |
| ENO       | Output      | BOOL      | E, A, M, D, L                   | Freigabeausgang |
| IN        | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | Eingangswert    |
| OUT       | Output      | INT       | E, A, M, D, L, P                | Ausgangswert    |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

31 ...                      ... 16 15 ...                      3 ... 0  
 "TagIn\_Value"    

|                     |                     |
|---------------------|---------------------|
| 0000 1111 0000 0101 | 0000 1001 0000 1000 |
|---------------------|---------------------|

  
 "TagOut\_Value"    

|   |
|---|
| 3 |
|---|

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung "Encodieren" ausgeführt. Die Anweisung selektiert das niederwertigste gesetzte Bit am Eingang "TagIn\_Value" und schreibt die Bitposition "3" in die Variable am Ausgang "TagOut\_Value".

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## SEL: Selektieren

### Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Eingang G) einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Alle Variablen an allen Parametern müssen den gleichen Datentyp haben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Selektieren":

| Parameter | Deklaration | Datentyp   |   | Speicherbereich                 |                                    | Beschreibung        |
|-----------|-------------|--|---|---------------------------------|------------------------------------|---------------------|
|           |             | S7-1200  | S7-1500   | S7-1200                         | S7-1500                            |                     |
| EN        | Input       | BOOL   | BOOL  | E, A, M, D, L                   | E, A, M, D, L                      | Freigabeeingang     |
| ENO       | Output      | BOOL   | BOOL  | E, A, M, D, L                   | E, A, M, D, L                      | Freigabeausgang     |
| G         | Input       | BOOL   | BOOL  | E, A, M, D, L oder Konstante    | E, A, M, D, L, T, Z oder Konstante | Schalter            |
| IN0       | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, TOD, CHAR, DATE | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, LDT, CHAR, DATE | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante    | Erster Eingangswert |

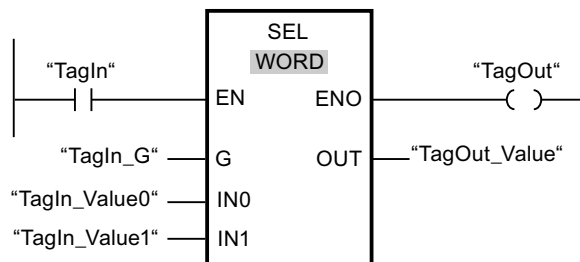
| Parameter | Deklaration | Datentyp   |   | Speicherbereich                 |                                 | Beschreibung         |
|-----------|-------------|--|---|---------------------------------|---------------------------------|----------------------|
|           |             | S7-1200  | S7-1500   | S7-1200                         | S7-1500                         |                      |
| IN1       | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, TOD, CHAR, DATE | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, LDT, CHAR, DATE | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter Eingangswert |
| OUT       | Output      | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, TOD, CHAR, DATE | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, Zeiten, TOD, LTOD, LDT, CHAR, DATE | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis             |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert      |         |
|-----------|--------------|-----------|---------|
| G         | TagIn_G      | 0         | 1       |
| IN0       | TagIn_Value0 | W#16#0000 | W#16#4C |
| IN1       | TagIn_Value1 | W#16#FFFF | W#16#5E |
| OUT       | TagOut_Value | W#16#0000 | W#16#5E |

Wenn der Operand "TagIn" den Signalzustand "1" hat, wird die Anweisung "Selektieren" ausgeführt. Abhängig vom Signalzustand am Eingang "TagIn\_G" wird der Wert am Eingang "TagIn\_Value0" oder "TagIn\_Value1" selektiert und in den Ausgang "TagOut\_Value" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, hat der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**MUX: Multiplexen****Beschreibung**

Mit der Anweisung "Multiplexen" kopieren Sie den Inhalt eines ausgewählten Eingangs in den Ausgang OUT. Die Anzahl der auswählbaren Eingänge der Anweisungsbox ist erweiterbar. Sie können maximal 32 Eingänge deklarieren.

Die Eingänge werden in der Box automatisch nummeriert. Die Nummerierung fängt mit IN0 an und wird mit jedem neuen Eingang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Eingang, dessen Inhalt in den Ausgang OUT kopiert wird. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Eingänge ist, wird der Inhalt des Parameters ELSE in den Ausgang OUT kopiert und dem Freigabeausgang ENO den Signalzustand "0" zugewiesen.

Die Anweisung "Multiplexen" kann nur ausgeführt werden, wenn die Variablen an allen Eingängen sowie am Ausgang OUT vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, für den nur Ganzzahlen angegeben werden können.

Der Freigabeausgang ENO wird zurückgesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeingang EN liefert den Signalzustand "0".
- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Eingänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplexen":

| Parameter | Deklaration | Datentyp |         | Speicherbereich | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|-----------------|
|           |             | S7-1200  | S7-1500 |                 |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | Freigabeingang  |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | Freigabeausgang |

| Parameter | Deklaration | Datentyp  |  | Speicherbereich                 | Beschreibung  |
|-----------|-------------|---|--|---------------------------------|---|
|           |             | S7-1200   | S7-1500  |                                 |   |
| K         | Input       | Ganzzahlen  | Ganzzahlen   | E, A, M, D, L, P oder Konstante | Gibt den Eingang an, dessen Inhalt kopiert wird. <ul style="list-style-type: none"> <li>• Wenn K = 0 =&gt; Parameter IN0</li> <li>• Wenn K = 1 =&gt; Parameter IN1, etc.</li> </ul> |
| IN0       | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Erster Eingangswert   |
| IN1       | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Zweiter Eingangswert  |
| INn       | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Optionale Eingangswerte   |

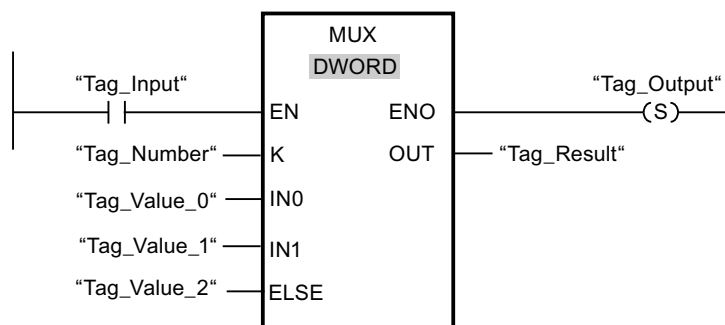
| Parameter | Deklaration | Datentyp  |  | Speicherbereich                 | Beschreibung                                 |
|-----------|-------------|---|--|---------------------------------|--|
|           |             | S7-1200   | S7-1500  |                                 |  |
| ELSE      | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Gibt den Wert an, der bei K > n kopiert wird |
| OUT       | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | Ausgang, in den der Wert kopiert wird        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand     | Wert           |
|-----------|-------------|----------------|
| K         | Tag_Number  | 1              |
| IN0       | Tag_Value_0 | DW#16#00000000 |
| IN1       | Tag_Value_1 | DW#16#003E4A7D |
| ELSE      | Tag_Value_2 | DW#16#FFFF0000 |
| OUT       | Tag_Result  | DW#16#003E4A7D |

Wenn der Operand "Tag\_Input" den Signalzustand "1" liefert, wird die Anweisung "Multiplexen" ausgeführt. Entsprechend dem Wert des Operanden "Tag\_Number" wird der Wert am Eingang

"Tag\_Value\_1" kopiert und dem Operanden am Ausgang "Tag\_Result" zugewiesen. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden der Freigabeausgang ENO und "Tag\_Output" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**DEMUX: Demultiplexen**

**Beschreibung**

Mit der Anweisung "Demultiplexen" kopieren Sie den Inhalt des Eingangs IN in einen ausgewählten Ausgang. Die Anzahl der auswählbaren Ausgänge ist in der Anweisungsbox erweiterbar. Die Ausgänge werden in der Box automatisch nummeriert. Die Nummerierung fängt bei OUT0 an und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Ausgang, in welchen der Inhalt des Eingangs IN kopiert wird. Die anderen Ausgänge werden nicht verändert. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Ausgänge ist, wird der Inhalt des Eingangs IN in den Parameter ELSE und dem Freigabeausgang ENO der Signalzustand "0" zugewiesen.

Die Anweisung "Demultiplexen" kann nur ausgeführt werden, wenn die Variablen am Eingang IN sowie an allen Ausgängen vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, an welchem nur Ganzzahlen angegeben werden können.

Der Freigabeausgang ENO wird zurückgesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Ausgänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Demultiplexen":

| Parameter | Deklaration | Datentyp |         | Speicherbereich | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|-----------------|
|           |             | S7-1200  | S7-1500 |                 |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | Freigabeausgang |



| Parameter | Deklaration | Datentyp  |  | Speicherbereich                 | Beschreibung   |
|-----------|-------------|---|--|---------------------------------|--|
|           |             | S7-1200   | S7-1500  |                                 |  |
| K         | Input       | Ganzzahlen  | Ganzzahlen   | E, A, M, D, L, P oder Konstante | Gibt den Ausgang an, in welchen der Eingangswert (IN) kopiert wird. <ul style="list-style-type: none"> <li>• Wenn K = 0 =&gt; Parameter OUT0</li> <li>• Wenn K = 1 =&gt; Parameter OUT1, etc.</li> </ul> |
| IN        | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | Eingangswert   |
| OUT0      | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | Erster Ausgang   |
| OUT1      | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | Zweiter Ausgang  |

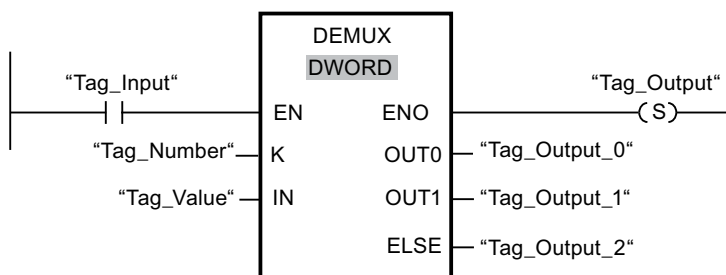
| Parameter | Deklaration | Datentyp  |  | Speicherbereich  | Beschreibung  |
|-----------|-------------|---|--|------------------|---|
|           |             | S7-1200   | S7-1500  |                  |   |
| OUTn      | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P | Optionale Ausgänge  |
| ELSE      | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P | Ausgang, in welchen der Eingangswert (IN) bei K > n kopiert wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den verfügbaren Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Tabelle 9-24 Eingangswerte der Anweisung "Demultiplexen" vor dem Ausführen des Netzwerks

| Parameter | Operand    | Werte         |                |
|-----------|------------|---------------|----------------|
| K         | Tag_Number | 1             | 4              |
| IN        | Tag_Value  | DW#16#FFFFFFF | DW#16#003E4A7D |

Tabelle 9-25 Ausgangswerte der Anweisung "Demultiplexen" nach dem Ausführen des Netzwerks

| Parameter | Operand      | Werte          |                |
|-----------|--------------|----------------|----------------|
| OUT0      | Tag_Output_0 | unverändert    | unverändert    |
| OUT1      | Tag_Output_1 | DW#16#FFFFFFFF | unverändert    |
| ELSE      | Tag_Output_2 | unverändert    | DW#16#003E4A7D |

Wenn der Eingang "Tag\_Input" den Signalzustand "1" liefert, wird die Anweisung "Demultiplexen" ausgeführt. Abhängig vom Wert des Operanden "Tag\_Number" wird der Wert am Eingang IN in den entsprechenden Ausgang kopiert.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## Schieben und Rotieren

### SHR: Rechts schieben

#### Beschreibung

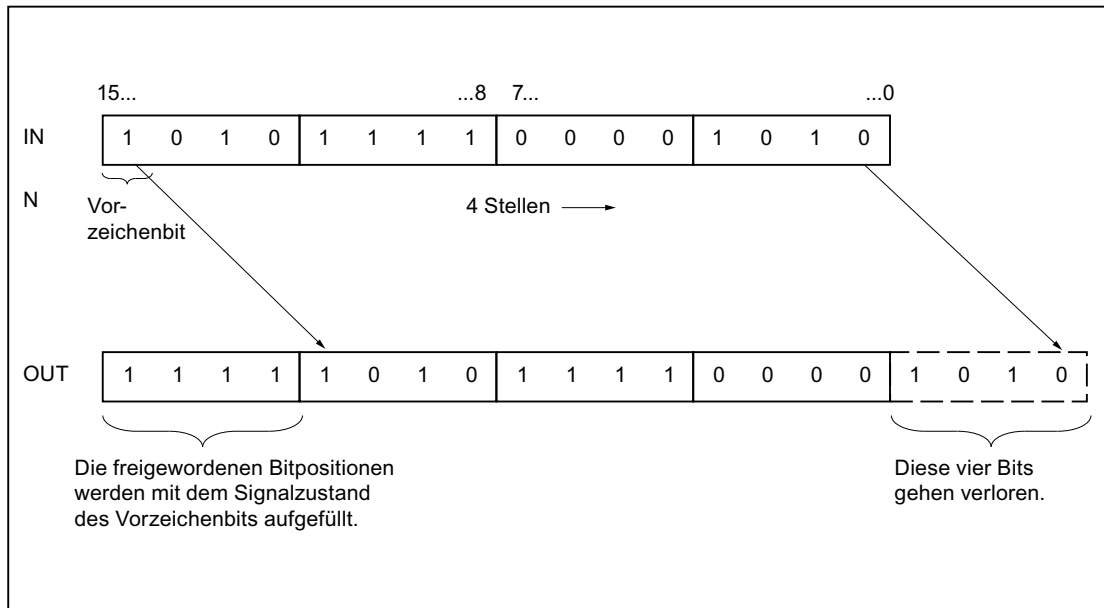
Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp Integer um vier Bitstellen nach rechts verschoben wird:



**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts schieben":

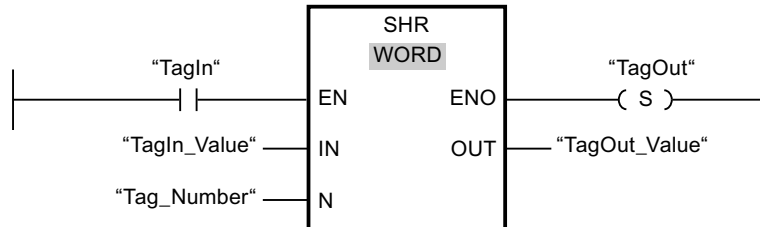
| Parameter | Deklaration | Datentyp              |                           | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------------|---------------------------|------------------------------|--|
|           |             | S7-1200               | S7-1500                   |                              |  |
| EN        | Input       | BOOL                  | BOOL                      | E, A, M, D, L                | Freigabeeingang  |
| ENO       | Output      | BOOL                  | BOOL                      | E, A, M, D, L                | Freigabeausgang  |
| IN        | Input       | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L oder Konstante | Wert, der verschoben wird                              |
| N         | Input       | USINT, UINT, UDINT    | USINT, UINT, UDINT, ULINT | E, A, M, D, L oder Konstante | Anzahl der Bitstellen, um die der Wert verschoben wird |
| OUT       | Output      | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L                | Ergebnis der Anweisung                                 |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0011 1111 1010 1111 |
| N         | Tag_Number   | 3                   |
| OUT       | TagOut_Value | 0000 0111 1111 0101 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Rechts schieben" ausgeführt. Der Inhalt des Operanden "TagIn\_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## SHL: Links schieben

### Beschreibung

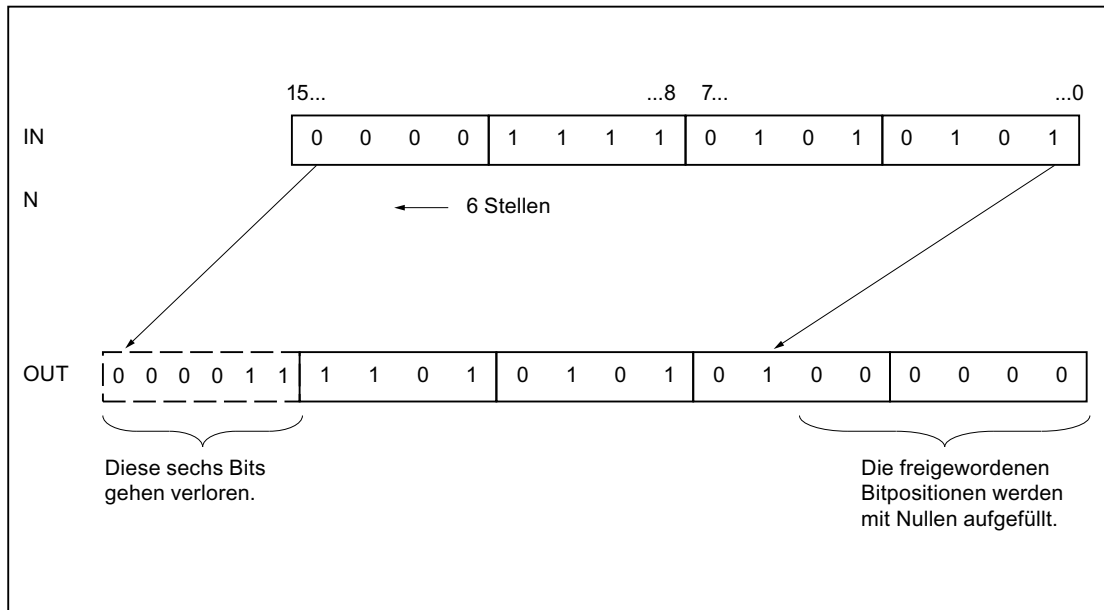
Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen im rechten Bereich des Operanden werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links schieben":

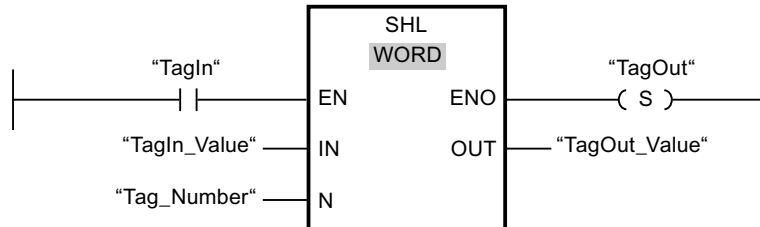
| Parameter | Deklaration | Datentyp              |                           | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------------|---------------------------|------------------------------|--|
|           |             | S7-1200               | S7-1500                   |                              |  |
| EN        | Input       | BOOL                  | BOOL                      | E, A, M, D, L                | Freigabeeingang  |
| ENO       | Output      | BOOL                  | BOOL                      | E, A, M, D, L                | Freigabeausgang  |
| IN        | Input       | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L oder Konstante | Wert, der verschoben wird                              |
| N         | Input       | USINT, UINT, UDINT    | USINT, UINT, UDINT, ULINT | E, A, M, D, L oder Konstante | Anzahl der Bitstellen, um die der Wert verschoben wird |
| OUT       | Output      | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L                | Ergebnis der Anweisung                                 |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0011 1111 1010 1111 |
| N         | Tag_Number   | 4                   |
| OUT       | TagOut_Value | 1111 1010 1111 0000 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Links schieben" ausgeführt. Der Inhalt des Operanden "TagIn\_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ROR: Rechts rotieren

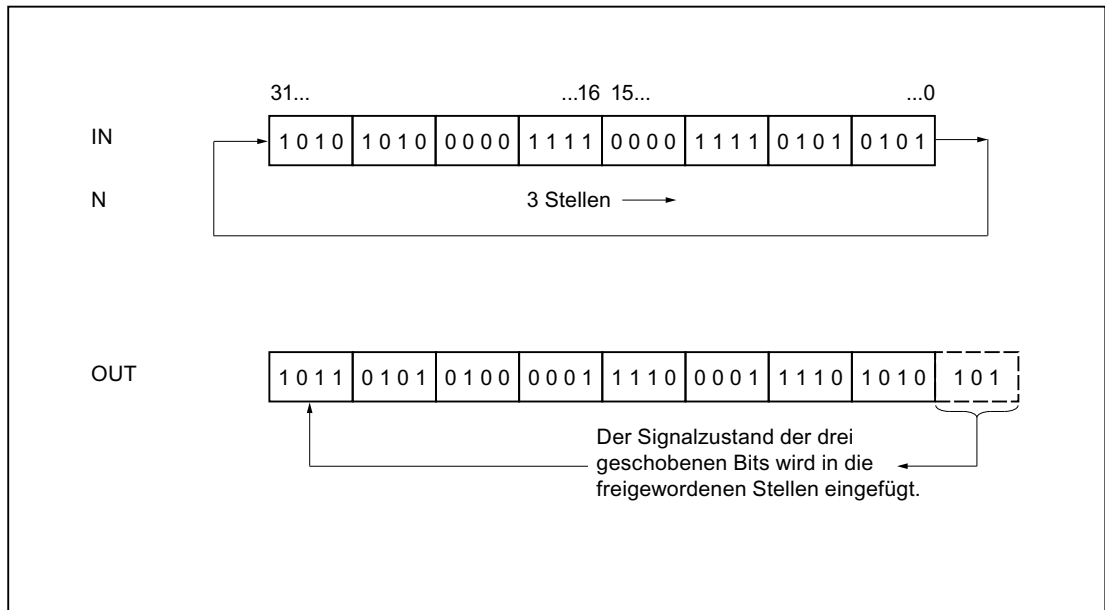
### Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts rotieren":

| Parameter | Deklaration | Datentyp                 |                              | Speicherbereich                 | Beschreibung  |
|-----------|-------------|--------------------------|------------------------------|---------------------------------|---|
|           |             | S7-1200                  | S7-1500                      |                                 |   |
| EN        | Input       | BOOL                     | BOOL                         | E, A, M, D, L                   | Freigabeeingang                                     |
| ENO       | Output      | BOOL                     | BOOL                         | E, A, M, D, L                   | Freigabeausgang                                     |
| IN        | Input       | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | E, A, M, D, L<br>oder Konstante | Wert, der rotiert wird                              |
| N         | Input       | USINT,<br>UINT,<br>UDINT | USINT, UINT,<br>UDINT, ULINT | E, A, M, D, L<br>oder Konstante | Anzahl der Bitstellen, um die der Wert rotiert wird |
| OUT       | Output      | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | E, A, M, D, L                   | Ergebnis der Anweisung                              |

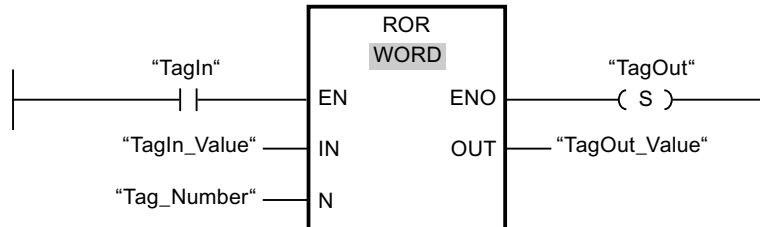
Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0000 1111 1001 0101 |
| N         | Tag_Number   | 5                   |
| OUT       | TagOut_Value | 1010 1000 0111 1100 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Rechts rotieren" ausgeführt. Der Inhalt des Operanden "TagIn\_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ROL: Links rotieren

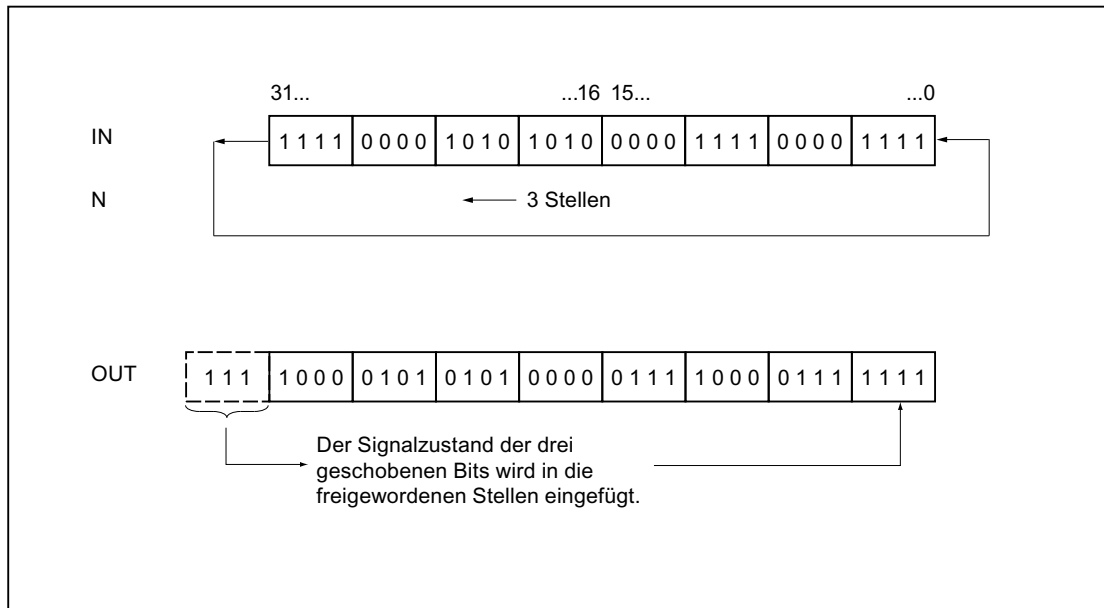
### Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links rotieren":

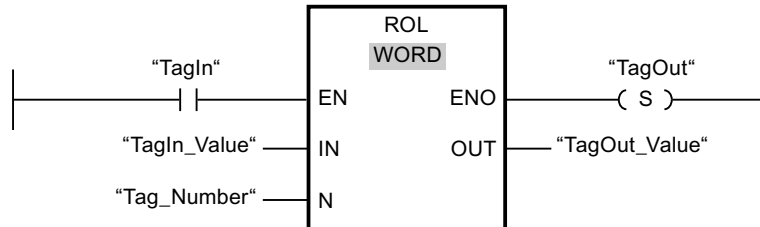
| Parameter | Deklaration | Datentyp              |                           | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------------------|---------------------------|------------------------------|---|
|           |             | S7-1200               | S7-1500                   |                              |   |
| EN        | Input       | BOOL                  | BOOL                      | E, A, M, D, L                | Freigabeeingang                                     |
| ENO       | Output      | BOOL                  | BOOL                      | E, A, M, D, L                | Freigabeausgang                                     |
| IN        | Input       | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L oder Konstante | Wert, der rotiert wird                              |
| N         | Input       | USINT, UINT, UDINT    | USINT, UINT, UDINT, ULINT | E, A, M, D, L oder Konstante | Anzahl der Bitstellen, um die der Wert rotiert wird |
| OUT       | Output      | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L                | Ergebnis der Anweisung                              |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 1010 1000 1111 0110 |
| N         | Tag_Number   | 5                   |
| OUT       | TagOut_Value | 0001 1110 1101 0101 |

Wenn der Eingang "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Links rotieren" ausgeführt. Der Inhalt des Operanden "TagIn\_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## Weitere Anweisungen

### DRUM: Schrittschaltwerk realisieren

## Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT\_WORD) mit den programmierten Werten des Parameters OUT\_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S\_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S\_PRESET) für jeden Schritt festgelegt. Zu Beginn

eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST\_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S\_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT\_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT\_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT\_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT\_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT\_WORD).

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstantz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L                | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang  |
| RESET     | Input       | BOOL     | E, A, M, D, L oder Konstante | Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung. |

| Parameter               | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-------------------------|-------------|----------|------------------------------|---|
| JOG                     | Input       | BOOL     | E, A, M, D, L oder Konstante | Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.                        |
| DRUM_EN                 | Input       | BOOL     | E, A, M, D, L oder Konstante | Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen. |
| LST_STEP                | Input       | BYTE     | E, A, M, D, L oder Konstante | Nummer des zuletzt programmierten Schritts.   |
| EVENT(i),<br>1 ≤ i ≤ 16 | Input       | BOOL     | E, A, M, D, L oder Konstante | Ereignisbit (i); Anfänglicher Signalzustand ist "1".  |
| OUT(j),<br>1 ≤ j ≤ 16   | Output      | BOOL     | E, A, M, D, L                | Ausgabebit (j)  |
| Q                       | Output      | BOOL     | E, A, M, D, L                | Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.                           |
| OUT_WORD                | Output      | WORD     | E, A, M, D, L, P             | Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.  |
| ERR_CODE                | Output      | WORD     | E, A, M, D, L, P             | Fehlerinformation   |
| JOG_HIS                 | Static      | BOOL     | E, A, M, D, L                | Verlaufsbit zum Parameter JOG   |
| EOD                     | Static      | BOOL     | E, A, M, D, L                | Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.                           |
| DSP                     | Static      | BYTE     | E, A, M, D, L, P             | Voreingestellter Schritt des Schrittschaltwerks   |
| DSC                     | Static      | BYTE     | E, A, M, D, L, P             | Aktueller Schritt des Schrittschaltwerks  |
| DCC                     | Static      | DWORD    | E, A, M, D, L, P             | Aktueller Zählwert des Schrittschaltwerks   |

| Parameter | Deklaration | Datentyp      | Speicherbereich              | Beschreibung   |
|-----------|-------------|---------------|------------------------------|--|
| DTBP      | Static      | WORD          | E, A, M, D, L, P             | Voreingestellte Zeitbasis des Schrittschaltwerks   |
| PREV_TIME | Static      | DWORD         | E, A, M, D, L oder Konstante | Vorherige Systemzeit   |
| S_PRESET  | Static      | ARRAY of WORD | E, A, M, D, L                | Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.                             |
| OUT_VAL   | Static      | ARRAY of BOOL | E, A, M, D, L                | Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].   |
| S_MASK    | Static      | ARRAY of BOOL | E, A, M, D, L                | Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1". |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

| ERR_CODE*  | Erläuterung   |
|--|---|
| W#16#0000  | Kein Fehler   |
| W#16#000B  | Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.                        |
| W#16#000C  | Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP. |
| W#16#000D  | Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.           |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## DCAT: Diskreter Steuerungszeitalarm

### Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O\_FB bzw. C\_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD\_HIS, ET (nur wenn  $ET < PT$  ist), OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD\_HIS werden auf "1" gesetzt.
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt.
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET (nur wenn  $ET < PT$  ist), OA, CA und CMD\_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD\_HIS den Signalzustand "1" anliegt und der Parameter O\_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD\_HIS und O\_FB auf "1" gesetzt sind und der Parameter C\_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O\_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD\_HIS und C\_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD\_HIS und O\_FB den Signalzustand "0" liefern und der Parameter C\_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C\_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O\_FB und C\_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Diskreter Steuerungszeitalarm":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L                | Freigabeeingang   |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang   |
| CMD       | Input       | BOOL     | E, A, M, D, L oder Konstante | Der Signalzustand "0" gibt den Befehl "Schließen" an.<br>Der Signalzustand "1" gibt den Befehl "Öffnen" an. |
| O_FB      | Input       | BOOL     | E, A, M, D, L oder Konstante | Feedback-Eingang beim Öffnen  |
| C_FB      | Input       | BOOL     | E, A, M, D, L oder Konstante | Feedback-Eingang beim Schließen   |
| Q         | Output      | BOOL     | E, A, M, D, L                | Zeigt den Status des Parameters CMD   |
| OA        | Output      | BOOL     | E, A, M, D, L                | Alarmausgang beim Öffnen  |
| CA        | Output      | BOOL     | E, A, M, D, L                | Alarmausgang beim Schließen   |
| ET        | Static      | DINT     | D, L                         | Aktuelle, abgelaufene Zeit;<br>ein Takt = 1 ms  |
| PT        | Static      | DINT     | D, L                         | Voreingestellter Zeitwert;<br>ein Takt = 1 ms   |
| PREV_TIME | Static      | DWORD    | D, L                         | Vorherige Systemzeit  |
| CMD_HIS   | Static      | BOOL     | D, L                         | Verlaufsbit zu CMD  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

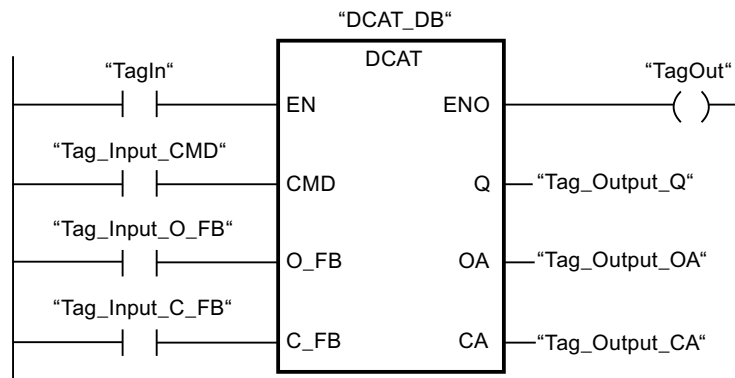
Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA



erhalten den Signalzustand "0". Der Parameter CMD\_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

**Hinweis**

Statische Parameter können Sie im Datenbaustein initialisieren.



Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

**Vor der Bearbeitung**

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand        | Wert  |
|-----------|----------------|-------|
| CMD       | Tag_Input_CMD  | TRUE  |
| O_FB      | Tag_Input_O_FB | FALSE |
| C_FB      | Tag_Input_C_FB | FALSE |
| Q         | Tag_Output_Q   | FALSE |
| OA        | Tag_Output_OA  | FALSE |
| CA        | Tag_Output_CA  | FALSE |

Im Instanz-Datenbaustein "DCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| ET        | DBD4    | L#12  |
| PT        | DBD8    | L#222 |
| CMD_HIS   | DBX16.0 | FALSE |

**Nach der Bearbeitung**

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand       | Wert  |
|-----------|---------------|-------|
| Q         | Tag_Output_Q  | TRUE  |
| OA        | Tag_Output_OA | FALSE |
| CA        | Tag_Output_CA | FALSE |

Im Instanz-Datenbaustein "DCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert |
|-----------|---------|------|
| ET        | DBD4    | L#0  |
| CMD_HIS   | DBX16.0 | TRUE |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**MCAT: Motorsteuerungszeitalarm**

**Beschreibung**

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

**Ausführung der Anweisung "Motorsteuerungszeitalarm"**

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

| Eingangsparameter |        |        |        |        |        |       |       | Ausgangsparameter |    |    |    |    |        |        |   |         |
|-------------------|--------|--------|--------|--------|--------|-------|-------|-------------------|----|----|----|----|--------|--------|---|---------|
| ET                | O_H IS | C_H IS | O_C MD | C_C MD | S_C MD | O_F B | C_F B | OO                | CO | OA | CA | ET | O_H IS | C_H IS | Q | Zustand |
| X                 | 1      | 1      | X      | X      | X      | X     | X     | 0                 | 0  | 1  | 1  | PT | 0      | 0      | 0 | Alarm   |
| X                 | X      | X      | X      | X      | X      | 1     | 1     | 0                 | 0  | 1  | 1  | PT | 0      | 0      | 0 | Alarm   |
| X                 | X      | X      | X      | X      | 1      | X     | X     | 0                 | 0  | 0  | 0  | X  | 0      | 0      | 1 | Stop    |
| X                 | X      | X      | 1      | 1      | X      | X     | X     | 0                 | 0  | 0  | 0  | X  | 0      | 0      | 1 | Stop    |

| Eingangsparameter   |   |  |   |   |   |   |   | Ausgangsparameter |   |   |   |     |   |   |   |                   |
|---|---|--|---|---|---|---|---|-------------------|---|---|---|-----|---|---|---|-------------------|
| X   | 0 | X  | 1 | 0 | 0 | X | X | 1                 | 0 | 0 | 0 | 0   | 1 | 0 | 1 | Öffnen starten    |
| <PT   | 1 | 0  | X | 0 | 0 | 0 | X | 1                 | 0 | 0 | 0 | INC | 1 | 0 | 1 | Öffnen            |
| X   | 1 | 0  | X | 0 | 0 | 1 | 0 | 0                 | 0 | 0 | 0 | PT  | 1 | 0 | 1 | Geöffnet          |
| >= PT   | 1 | 0  | X | 0 | 0 | 0 | X | 0                 | 0 | 1 | 0 | PT  | 1 | 0 | 0 | Alarm öffnen      |
| X   | X | 0  | 0 | 1 | 0 | X | X | 0                 | 1 | 0 | 0 | 0   | 0 | 1 | 1 | Schließen starten |
| < PT  | 0 | 1  | 0 | X | 0 | X | 0 | 0                 | 1 | 0 | 0 | INC | 0 | 1 | 1 | Schließen         |
| X   | 0 | 1  | 0 | X | 0 | 0 | 1 | 0                 | 0 | 0 | 0 | PT  | 0 | 1 | 1 | Geschlossen       |
| >= PT   | 0 | 1  | 0 | X | 0 | X | 0 | 0                 | 0 | 0 | 1 | PT  | 0 | 1 | 0 | Alarm schließen   |
| X   | 0 | 0  | 0 | 0 | 0 | X | X | 0                 | 0 | 0 | 0 | X   | 0 | 0 | 1 | Gestoppt          |
| Legende:  |   |  |   |   |   |   |   |                   |   |   |   |     |   |   |   |                   |
| INC   |   | Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren |   |   |   |   |   |                   |   |   |   |     |   |   |   |                   |
| PT  |   | PT wird auf den gleichen Wert wie ET gesetzt                                 |   |   |   |   |   |                   |   |   |   |     |   |   |   |                   |
| X   |   | Nicht anwendbar  |   |   |   |   |   |                   |   |   |   |     |   |   |   |                   |
| < PT  |   | ET < PT  |   |   |   |   |   |                   |   |   |   |     |   |   |   |                   |
| >= PT   |   | ET >= PT   |   |   |   |   |   |                   |   |   |   |     |   |   |   |                   |
| <p>Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt:</p> <p>OO = FALSE<br/>                     CO = FALSE<br/>                     OA = FALSE<br/>                     CA = FALSE<br/>                     ET = PT<br/>                     Q = TRUE</p> |   |  |   |   |   |   |   |                   |   |   |   |     |   |   |   |                   |

Die Anweisung "Motorsteuerungszeitalarm" liefert keine Fehlerinformationen.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung               |
|-----------|-------------|----------|------------------------------|----------------------------|
| EN        | Input       | BOOL     | E, A, M, D, L                | Freigabeeingang            |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang            |
| O_CMD     | Input       | BOOL     | E, A, M, D, L oder Konstante | Befehlseingabe "Öffnen"    |
| C_CMD     | Input       | BOOL     | E, A, M, D, L oder Konstante | Befehlseingabe "Schließen" |
| S_CMD     | Input       | BOOL     | E, A, M, D, L oder Konstante | Befehlseingabe "Stoppen"   |

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| O_FB      | Input       | BOOL     | E, A, M, D, L oder Konstante | Feedback-Eingang beim Öffnen                         |
| C_FB      | Input       | BOOL     | E, A, M, D, L oder Konstante | Feedback-Eingang beim Schließen                      |
| OO        | Output      | BOOL     | E, A, M, D, L                | Ausgang "Öffnen"                                     |
| CO        | Output      | BOOL     | E, A, M, D, L                | Ausgang "Schließen"                                  |
| OA        | Output      | BOOL     | E, A, M, D, L                | Alarmausgang beim Öffnen                             |
| CA        | Output      | BOOL     | E, A, M, D, L                | Alarmausgang beim Schließen                          |
| Q         | Output      | BOOL     | E, A, M, D, L                | Der Signalzustand "0" zeigt eine Fehlerbedingung an. |
| ET        | Static      | DINT     | D, L                         | Aktuelle, abgelaufene Zeit; ein Takt = 1 ms          |
| PT        | Static      | DINT     | D, L                         | Voreingestellter Zeitwert; ein Takt = 1 ms           |
| PREV_TIME | Static      | DWORD    | D, L                         | Vorherige Systemzeit                                 |
| O_HIS     | Static      | BOOL     | D, L                         | Verlaufsbit "Öffnen"                                 |
| C_HIS     | Static      | BOOL     | D, L                         | Verlaufsbit "Schließen"                              |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

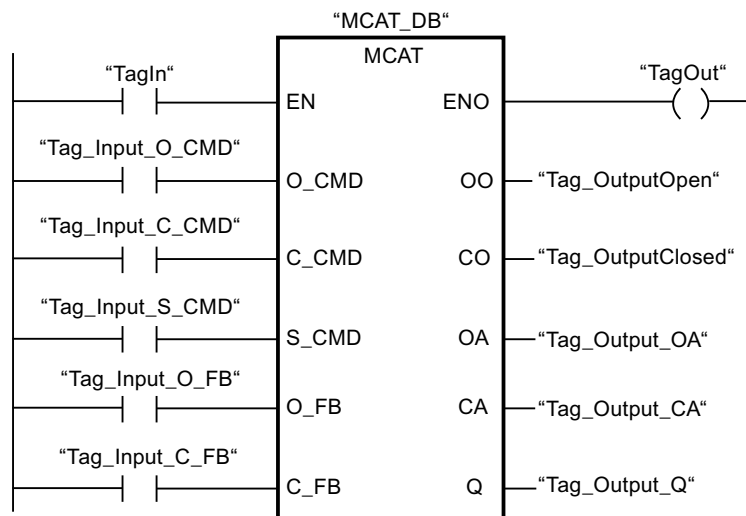
Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

---

#### Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

---



Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

### Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand          | Wert  |
|-----------|------------------|-------|
| O_CMD     | Tag_Input_O_CMD  | TRUE  |
| C_CMD     | Tag_Input_C_CMD  | FALSE |
| S_CMD     | Tag_Input_S_CMD  | FALSE |
| O_FB      | Tag_Input_O_FB   | FALSE |
| C_FB      | Tag_Input_C_FB   | FALSE |
| OO        | Tag_OutputOpen   | FALSE |
| CO        | Tag_OutputClosed | FALSE |
| OA        | Tag_Output_OA    | FALSE |
| CA        | Tag_Output_CA    | FALSE |
| Q         | Tag_Output_Q     | FALSE |

Im Instanz-Datenbaustein "MCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| ET        | DBD4    | L#2   |
| PT        | DBD8    | L#22  |
| O_HIS     | DBX16.0 | TRUE  |
| C_HIS     | DBX16.1 | FALSE |

### Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand          | Wert  |
|-----------|------------------|-------|
| OO        | Tag_OutputOpen   | TRUE  |
| CO        | Tag_OutputClosed | FALSE |
| OA        | Tag_Output_OA    | FALSE |
| CA        | Tag_Output_CA    | FALSE |
| Q         | Tag_Output_Q     | TRUE  |

Im Instanz-Datenbaustein "MCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| ET        | DBD4    | L#0   |
| O_HIS     | DBX16.0 | TRUE  |
| CMD_HIS   | DBX16.1 | FALSE |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**IMC: Eingabebits mit den Bits einer Maske vergleichen**

**Beschreibung**

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN\_BIT0 bis IN\_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN\_BIT0 wird mit dem Wert der Maske CMP\_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP\_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP\_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR\_CODE wird eine Fehlermeldung ausgegeben.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

| Parameter | Deklaration | Datentyp | Speicherbereich                 | Beschreibung  |
|-----------|-------------|----------|---------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L                   | Freigabeeingang                                     |
| ENO       | Output      | BOOL     | E, A, M, D, L                   | Freigabeausgang                                     |
| IN_BIT0   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 0 wird mit Bit 0 der Maske verglichen.   |
| IN_BIT1   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 1 wird mit Bit 1 der Maske verglichen.   |
| IN_BIT2   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 2 wird mit Bit 2 der Maske verglichen.   |
| IN_BIT3   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 3 wird mit Bit 3 der Maske verglichen.   |
| IN_BIT4   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 4 wird mit Bit 4 der Maske verglichen.   |
| IN_BIT5   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 5 wird mit Bit 5 der Maske verglichen.   |
| IN_BIT6   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 6 wird mit Bit 6 der Maske verglichen.   |
| IN_BIT7   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 7 wird mit Bit 7 der Maske verglichen.   |
| IN_BIT8   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 8 wird mit Bit 8 der Maske verglichen.   |
| IN_BIT9   | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 9 wird mit Bit 9 der Maske verglichen.   |
| IN_BIT10  | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 10 wird mit Bit 10 der Maske verglichen. |
| IN_BIT11  | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 11 wird mit Bit 11 der Maske verglichen. |
| IN_BIT12  | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 12 wird mit Bit 12 der Maske verglichen. |
| IN_BIT13  | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 13 wird mit Bit 13 der Maske verglichen. |
| IN_BIT14  | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 14 wird mit Bit 14 der Maske verglichen. |
| IN_BIT15  | Input       | BOOL     | E, A, M, D, L oder Konstante    | Eingabebit 15 wird mit Bit 15 der Maske verglichen. |
| CMP_STEP  | Input       | BYTE     | E, A, M, D, L, P oder Konstante | Schrittnummer der Maske, mit der verglichen wird.   |

| Parameter | Deklaration | Datentyp      | Speicherbereich  | Beschreibung   |
|-----------|-------------|---------------|------------------|--|
| OUT       | Output      | BOOL          | E, A, M, D, L    | Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde.<br>Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.            |
| ERR_CODE  | Output      | WORD          | E, A, M, D, L, P | Fehlerinformation  |
| CMP_VAL   | Static      | ARRAY OF WORD | E, A, M, D, L    | Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE ausgegeben werden:

| Fehlercode* (W#16#...) | Erläuterung                                       |
|------------------------|---|
| 0000                   | Kein Fehler                                       |
| 000A                   | Der Wert am Parameter CMP_STEP ist größer als 15. |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### SMC: Matrixscanner

#### Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN\_BIT0 bis IN\_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN\_BIT0 wird mit dem Wert der Maske CMP\_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der



Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT\_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT\_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT\_STEP um "1" größer als der Wert am Parameter LAST.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L                | Freigabeeingang                                     |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang                                     |
| IN_BIT0   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 0 wird mit Bit 0 der Maske verglichen.   |
| IN_BIT1   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 1 wird mit Bit 1 der Maske verglichen.   |
| IN_BIT2   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 2 wird mit Bit 2 der Maske verglichen.   |
| IN_BIT3   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 3 wird mit Bit 3 der Maske verglichen.   |
| IN_BIT4   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 4 wird mit Bit 4 der Maske verglichen.   |
| IN_BIT5   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 5 wird mit Bit 5 der Maske verglichen.   |
| IN_BIT6   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 6 wird mit Bit 6 der Maske verglichen.   |
| IN_BIT7   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 7 wird mit Bit 7 der Maske verglichen.   |
| IN_BIT8   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 8 wird mit Bit 8 der Maske verglichen.   |
| IN_BIT9   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 9 wird mit Bit 9 der Maske verglichen.   |
| IN_BIT10  | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 10 wird mit Bit 10 der Maske verglichen. |
| IN_BIT11  | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 11 wird mit Bit 11 der Maske verglichen. |

| Parameter | Deklaration | Datentyp      | Speicherbereich              | Beschreibung   |
|-----------|-------------|---------------|------------------------------|--|
| IN_BIT12  | Input       | BOOL          | E, A, M, D, L oder Konstante | Eingabebit 12 wird mit Bit 12 der Maske verglichen.  |
| IN_BIT13  | Input       | BOOL          | E, A, M, D, L oder Konstante | Eingabebit 13 wird mit Bit 13 der Maske verglichen.  |
| IN_BIT14  | Input       | BOOL          | E, A, M, D, L oder Konstante | Eingabebit 14 wird mit Bit 14 der Maske verglichen.  |
| IN_BIT15  | Input       | BOOL          | E, A, M, D, L oder Konstante | Eingabebit 15 wird mit Bit 15 der Maske verglichen.  |
| OUT       | Output      | BOOL          | E, A, M, D, L                | Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde.<br>Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.                              |
| ERR_CODE  | Output      | WORD          | E, A, M, D, L, P             | Fehlerinformation  |
| OUT_STEP  | Output      | BYTE          | E, A, M, D, L, P             | Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde. |
| LAST      | Static      | BYTE          | E, A, M, D, L, P             | Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.   |
| CMP_VAL   | Static      | ARRAY OF WORD | E, A, M, D, L                | Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.             |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

| Fehlercode* (W#16#...)   | Erläuterung                                   |
|--|---|
| 0000   | Kein Fehler                                   |
| 000E   | Der Wert am Parameter LAST ist größer als 15. |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**LEAD\_LAG: Lead- und Lag-Algorithmus****Beschreibung**

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$\text{OUT} = \left[ \frac{\text{LG\_TIME}}{\text{LG\_TIME} + \text{SAMPLE\_T}} \right] \text{PREV\_OUT} + \text{GAIN} \left[ \frac{\text{LD\_TIME} + \text{SAMPLE\_T}}{\text{LG\_TIME} + \text{SAMPLE\_T}} \right] \text{IN} - \text{GAIN} \left[ \frac{\text{LD\_TIME}}{\text{LG\_TIME} + \text{SAMPLE\_T}} \right] * \text{PREV\_IN}$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD\_TIME, LG\_TIME und SAMPLE\_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei  $\text{LG\_TIME} > 4 + \text{SAMPLE\_T}$  an folgende Funktion an:

$$\text{OUT} = \text{GAIN} * ((1 + \text{LD\_TIME} * s) / (1 + \text{LG\_TIME} * s)) * \text{IN}$$

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR\_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Lead- und Lag-Algorithmus":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung  |
|-----------|-------------|----------|------------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L                      | Freigabeeingang   |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang   |
| IN        | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.<br>Am Parameter IN können auch Konstanten angegeben werden. |
| SAMPLE_T  | Input       | INT      | E, A, M, D, L, P<br>oder Konstante | Abtastzeit<br>Am Parameter SAMPLE_T können auch Konstanten angegeben werden.  |
| OUT       | Output      | REAL     | E, A, M, D, L                      | Ergebnis der Anweisung  |
| ERR_CODE  | Output      | WORD     | E, A, M, D, L                      | Fehlerinformation   |
| LD_TIME   | Static      | REAL     | E, A, M, D, L, P<br>oder Konstante | Voreilzeit in der gleichen Einheit wie die Abtastzeit.  |
| LG_TIME   | Static      | REAL     | E, A, M, D, L, P<br>oder Konstante | Nacheilzeit in der gleichen Einheit wie die Abtastzeit.   |
| GAIN      | Static      | REAL     | E, A, M, D, L, P<br>oder Konstante | Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).                                |
| PREV_IN   | Static      | REAL     | E, A, M, D, L, P<br>oder Konstante | Vorheriger Eingang  |
| PREV_OUT  | Static      | REAL     | E, A, M, D, L, P<br>oder Konstante | Vorheriger Ausgang  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

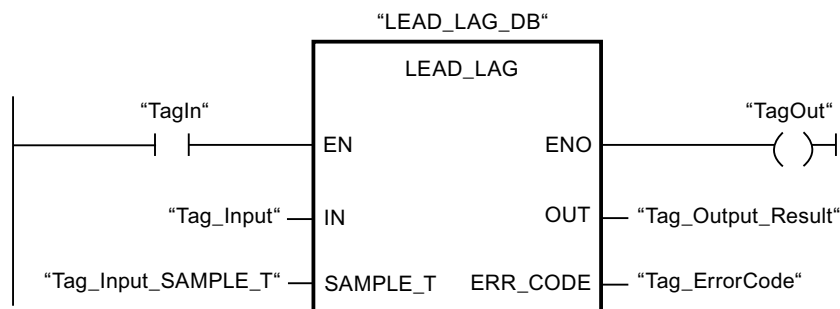
| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 0009   | Der Wert am Parameter GAIN ist kleiner oder gleich Null. |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

### Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.



Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

### Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand             | Wert |
|-----------|---------------------|------|
| IN        | Tag_Input           | 2.0  |
| SAMPLE_T  | Tag_InputSampleTime | 10   |

Im Instanz-Datenbaustein "LEAD\_LAG\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert |
|-----------|---------|------|
| LD_TIME   | DBD12   | 2.0  |
| LG_TIME   | DBD16   | 2.0  |
| GAIN      | DBD20   | 1.0  |
| PREV_IN   | DBD24   | 6.0  |
| PREV_OUT  | DBD28   | 6.0  |

**Nach der Bearbeitung**

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand           | Wert |
|-----------|-------------------|------|
| OUT       | Tag_Output_Result | 2.0  |

Im Instanz-Datenbaustein "LEAD\_LAD\_DB" der Anweisung werden die folgenden Werte gespeichert:

| Parameter | Operand | Wert |
|-----------|---------|------|
| PREV_IN   | DBD24   | 2.0  |
| PREV_OUT  | DBD28   | 2.0  |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**SEG: Bitmuster für 7-Segment-Anzeige erzeugen**

**Beschreibung**

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

| Eingangsziffer<br>r<br>(Binär) | Belegung der<br>Segmente<br>- g f e d c b a | Anzeige<br>(Hexadezimal) | Sieben-Segment-Anzeige |
|--------------------------------|---|--------------------------|------------------------|
| 0000                           | 00111111                                    | 0                        |                        |
| 0001                           | 00000110                                    | 1                        |                        |
| 0010                           | 01011011                                    | 2                        |                        |
| 0011                           | 01001111                                    | 3                        |                        |
| 0100                           | 01100110                                    | 4                        |                        |
| 0101                           | 01101101                                    | 5                        |                        |
| 0110                           | 01111101                                    | 6                        |                        |
| 0111                           | 00000111                                    | 7                        |                        |
| 1000                           | 01111111                                    | 8                        |                        |
| 1001                           | 01100111                                    | 9                        |                        |
| 1010                           | 01110111                                    | A                        |                        |

|      |          |   |
|------|----------|---|
| 1011 | 01111100 | B |
| 1100 | 00111001 | C |
| 1101 | 01011110 | D |
| 1110 | 01111001 | E |
| 1111 | 01110001 | F |

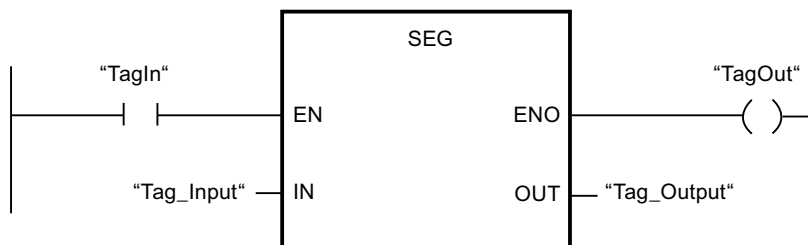
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

| Parameter | Deklaration | Datentyp | Speicherbereich                 | Beschreibung                          |
|-----------|-------------|----------|---------------------------------|---------------------------------------|
| EN        | Input       | BOOL     | E, A, M, D, L                   | Freigabeeingang                       |
| ENO       | Output      | BOOL     | E, A, M, D, L                   | Freigabeausgang                       |
| IN        | Input       | WORD     | E, A, M, D, L, P oder Konstante | Quellwort mit vier Hexadezimalziffern |
| OUT       | Output      | DWORD    | E, A, M, D, L, P                | Bitmuster für die 7-Segment-Anzeige   |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter   | Operand    | Wert          |   |
|-------------|------------|---------------|---|
| Hexadezimal | Binär      |               |   |
| IN          | Tag_Input  | W#16#1234     | 0001 0010 0011 0100   |
| OUT         | Tag_Output | DW#16065B4F66 | 000 00110 0101 1011 0100<br>1111 0110 0110<br>Anzeige: 1234 |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## BCDCPL: Zehnerkomplement erzeugen

### Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

$$10000000 \text{ (als BCD)} \\ - \text{7-stelliger BCD-Wert} \\ \hline \text{Zehnerkomplement (als BCD)}$$

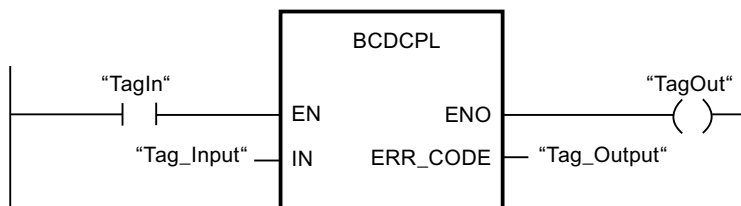
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung           |
|-----------|-------------|----------|------------------------------------|------------------------|
| EN        | Input       | BOOL     | E, A, M, D, L                      | Freigabeeingang        |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang        |
| IN        | Input       | DWORD    | E, A, M, D, L, P<br>oder Konstante | 7-stellige BCD-Zahl    |
| ERR_CODE  | Output      | DWORD    | E, A, M, D, L, P                   | Ergebnis der Anweisung |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

| Parameter | Operand    | Wert*          |
|-----------|------------|----------------|
| IN        | Tag_Input  | DW#16#01234567 |
| ERR_CODE  | Tag_Output | DW#16#08765433 |

\*Die Fehlercodes im Programmmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".



**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**BITSUM: Anzahl der gesetzten Bits zählen**

**Beschreibung**

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET\_VAL ausgegeben.

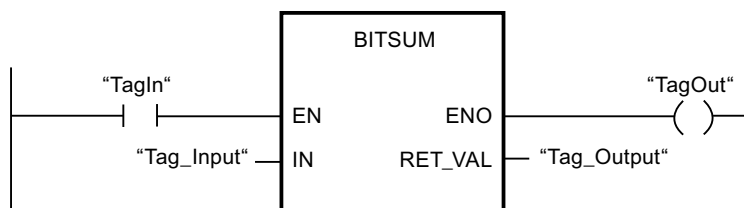
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung                                   |
|-----------|-------------|----------|------------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L                      | Freigabeeingang                                |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang                                |
| IN        | Input       | DWORD    | E, A, M, D, L, P<br>oder Konstante | Operand, dessen gesetzten Bits gezählt werden. |
| RET_VAL   | Output      | INT      | E, A, M, D, L, P                   | Anzahl der gesetzten Bits                      |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

| Parameter | Operand    | Wert*               |
|-----------|------------|---------------------|
| IN        | Tag_Input  | DW#16#12345678      |
| RET_VAL   | Tag_Output | W#16#000D (13 Bits) |

\*Die Fehlercodes im Programmmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**9.7.2.2 FUP**

**Bitverknüpfungen**

**&: UND-Verknüpfung**

**Beschreibung**

Mit der Anweisung "UND-Verknüpfung" fragen Sie die Signalzustände zweier oder mehrerer angegebener Operanden ab und werten sie entsprechend der UND-Wahrheitstabelle aus.

Beträgt der Signalzustand aller Operanden "1", so ist die Bedingung erfüllt und die Anweisung liefert das Ergebnis "1". Beträgt der Signalzustand eines Operanden "0", ist die Bedingung nicht erfüllt und die Anweisung erzeugt das Ergebnis "0".

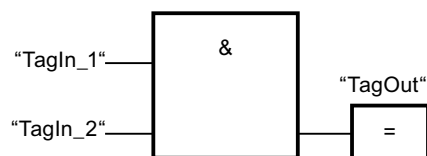
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "UND-Verknüpfung":

| Parameter | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung  |
|-----------|-------------|----------|-----------------|---------------------|---|
|           |             |          | S7-1200         | S7-1500             |   |
| <Operand> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird. |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand der Operanden "TagIn\_1" und "TagIn\_2" "1" ist und zurückgesetzt, wenn der Signalzustand der Operanden "TagIn\_1" und "TagIn\_2" "0" ist.

**Siehe auch**

- UND-Wahrheitstabelle (Seite 1885)
- Beispiel für das Erfassen der Richtung eines Förderbands (Seite 1598)
- Beispiel für das Kontrollieren der Raumtemperatur (Seite 1602)
- Übersicht über die gültigen Datentypen (Seite 1087)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)
- Eingang einfügen (Seite 1889)

**UND-Wahrheitstabelle****Verknüpfungsergebnisse**

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch UND entstehen:

| Signalzustand des ersten Operanden | Signalzustand des zweiten Operanden | Verknüpfungsergebnis |
|------------------------------------|-------------------------------------|----------------------|
| 1                                  | 1                                   | 1                    |
| 0                                  | 1                                   | 0                    |
| 1                                  | 0                                   | 0                    |
| 0                                  | 0                                   | 0                    |

**Siehe auch**

- &: UND-Verknüpfung (Seite 1884)
- Übersicht über die gültigen Datentypen (Seite 1087)

**>=1: ODER-Verknüpfung****Beschreibung**

Mit der Anweisung "ODER-Verknüpfung" fragen Sie die Signalzustände zweier oder mehrerer angegebener Operanden ab und werten sie entsprechend der ODER-Wahrheitstabelle aus.

Beträgt der Signalzustand eines der Operanden "1", so ist die Bedingung erfüllt und die Anweisung liefert das Ergebnis "1". Beträgt der Signalzustand aller Operanden "0", ist die Bedingung nicht erfüllt und die Anweisung erzeugt das Ergebnis "0".

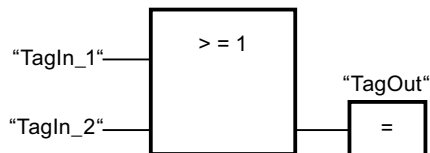
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER-Verknüpfung":

| Parameter | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung  |
|-----------|-------------|----------|-----------------|---------------------|---|
|           |             |          | S7-1200         | S7-1500             |   |
| <Operand> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird. |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand des Operanden "TagIn\_1" oder des Operanden "TagIn\_2" "1" ist.

**Siehe auch**

- ODER-Wahrheitstabelle (Seite 1886)
- Beispiel für das Steuern eines Förderbands (Seite 1597)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)
- Übersicht über die gültigen Datentypen (Seite 1087)
- Eingang einfügen (Seite 1889)

**ODER-Wahrheitstabelle**

**Verknüpfungsergebnisse**

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch ODER entstehen:

| Signalzustand des ersten Operanden | Signalzustand des zweiten Operanden | Verknüpfungsergebnis |
|------------------------------------|-------------------------------------|----------------------|
| 1                                  | 0                                   | 1                    |
| 0                                  | 1                                   | 1                    |
| 1                                  | 1                                   | 1                    |
| 0                                  | 0                                   | 0                    |

**Siehe auch**

>=1: ODER-Verknüpfung (Seite 1885)

Übersicht über die gültigen Datentypen (Seite 1087)

**X: EXKLUSIV ODER-Verknüpfung****Beschreibung**

Mit der Anweisung "EXKLUSIV ODER-Verknüpfung" fragen Sie das Ergebnis einer Signalzustandsabfrage entsprechend der EXKLUSIV ODER-Wahrheitstabelle ab.

Bei der Anweisung "EXKLUSIV ODER-Verknüpfung" ist der Signalzustand "1", wenn der Signalzustand eines der beiden angegebenen Operanden "1" ist. Bei der Abfrage von mehr als zwei Operanden ist das gemeinsame Verknüpfungsergebnis "1", wenn eine ungerade Anzahl der abgefragten Operanden das Abfrageergebnis "1" liefert.

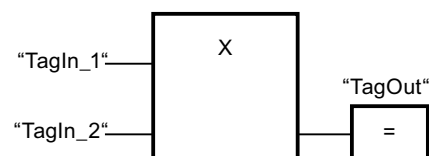
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER-Verknüpfung":

| Parameter | Deklaration | Datentyp | Speicherbereich |                        | Beschreibung  |
|-----------|-------------|----------|-----------------|------------------------|---|
|           |             |          | S7-1200         | S7-1500                |   |
| <Operand> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L,<br>T, Z | Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird. |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn der Signalzustand eines der beiden Operanden "TagIn\_1" und "TagIn\_2" "1" ist. Wenn beide Operanden den Signalzustand "1" oder "0" liefern, wird der Ausgang "TagOut" zurückgesetzt.

**Siehe auch**

EXKLUSIV ODER-Wahrheitstabelle (Seite 1888)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Übersicht über die gültigen Datentypen (Seite 1087)

Eingang einfügen (Seite 1889)

**EXKLUSIV ODER-Wahrheitstabelle**

**Verknüpfungsergebnisse**

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von zwei Operanden durch EXKLUSIV ODER entstehen:

| Signalzustand des ersten Operanden | Signalzustand des zweiten Operanden | Verknüpfungsergebnis |
|------------------------------------|-------------------------------------|----------------------|
| 1                                  | 0                                   | 1                    |
| 0                                  | 1                                   | 1                    |
| 1                                  | 1                                   | 0                    |
| 0                                  | 0                                   | 0                    |

Die folgende Tabelle zeigt die Ergebnisse, die bei der Verknüpfung von drei Operanden durch EXKLUSIV ODER entstehen:

| Signalzustand des ersten Operanden | Signalzustand des zweiten Operanden | Signalzustand des dritten Operanden | Verknüpfungsergebnis |
|------------------------------------|-------------------------------------|-------------------------------------|----------------------|
| 1                                  | 0                                   | 0                                   | 1                    |
| 0                                  | 1                                   | 1                                   | 0                    |
| 0                                  | 1                                   | 0                                   | 1                    |
| 1                                  | 0                                   | 1                                   | 0                    |
| 0                                  | 0                                   | 1                                   | 1                    |
| 1                                  | 1                                   | 0                                   | 0                    |
| 1                                  | 1                                   | 1                                   | 1                    |
| 0                                  | 0                                   | 0                                   | 0                    |

**Siehe auch**

X: EXKLUSIV ODER-Verknüpfung (Seite 1887)

Übersicht über die gültigen Datentypen (Seite 1087)

## Eingang einfügen

### Beschreibung

Mit der Anweisung "Eingang einfügen" erweitern Sie die Box einer der folgenden Anweisungen durch einen Eingang:

- "UND-Verknüpfung"
- "ODER-Verknüpfung"
- "EXKLUSIV ODER-Verknüpfung"

Durch die Erweiterung einer Anweisungsbox können Sie den Signalzustand mehrerer Operanden abfragen.

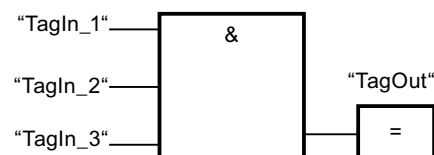
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingang einfügen":

| Parameter | Deklaration | Datentyp | Speicherbereich |                        | Beschreibung  |
|-----------|-------------|----------|-----------------|------------------------|---|
|           |             |          | S7-1200         | S7-1500                |   |
| <Operand> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L,<br>T, Z | Der Operand gibt das Bit an, dessen Signalzustand abgefragt wird. |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Box der Anweisung "UND-Verknüpfung" wurde durch einen zusätzlichen Eingang erweitert, an dem der Signalzustand des Operanden "TagIn\_3" abgefragt wird. Der Ausgang "TagOut" wird gesetzt, wenn die Operanden "TagIn\_1", "TagIn\_2" und "TagIn\_3" den Signalzustand "1" liefern.

### Siehe auch

- &: UND-Verknüpfung (Seite 1884)
- >=1: ODER-Verknüpfung (Seite 1885)
- X: EXKLUSIV ODER-Verknüpfung (Seite 1887)
- Übersicht über die gültigen Datentypen (Seite 1087)

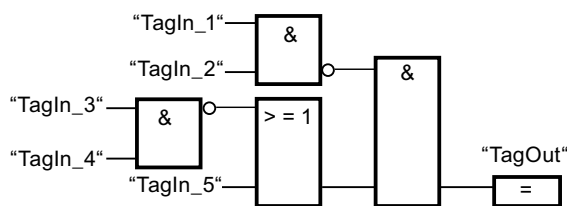
## VKE invertieren

### Beschreibung

Mit der Anweisung "VKE invertieren" invertieren Sie den Signalzustand des Verknüpfungsergebnisses (VKE).

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Eingang "TagIn\_1" und/oder "TagIn\_2" liefert den Signalzustand "0".
- Der Eingang "TagIn\_3" und/oder "TagIn\_4" liefert den Signalzustand "0" oder der Eingang "TagIn\_5" liefert den Signalzustand "1".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### =: Zuweisung

### Beschreibung

Mit der Anweisung "Zuweisung" setzen Sie das Bit eines angegebenen Operanden. Wenn das Verknüpfungsergebnis (VKE) am Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf den Signalzustand "1" gesetzt. Wenn der Signalzustand am Boxeingang "0" ist, wird das Bit des angegebenen Operanden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Boxeingang wird direkt dem über der Zuweisen-Box stehenden Operanden zugewiesen.

### Parameter

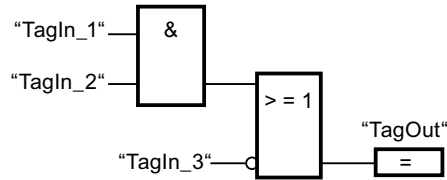
Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                          |
|-----------|-------------|----------|-----------------|---------------------------------------|
| <Operand> | Output      | BOOL     | E, A, M, D, L   | Operand, dem das VKE zugewiesen wird. |



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" am Ausgang der Anweisung "Zuweisung" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Eingänge "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Signalzustand am Eingang "TagIn\_3" ist "0".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 1599)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 1602)

## /=: Zuweisung negieren

## Beschreibung

Mit der Anweisung "Zuweisung negieren" kehren Sie das Verknüpfungsergebnis (VKE) um und weisen es dem über der Box stehenden Operanden zu. Wenn das VKE am Eingang der Box "1" ist, wird der Operand zurückgesetzt. Wenn das VKE am Eingang der Box "0" ist, wird der Operand auf den Signalzustand "1" gesetzt.

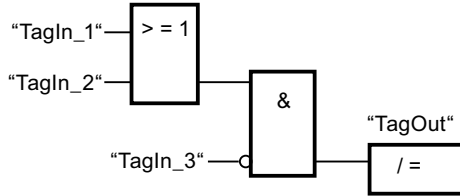
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zuweisung negieren":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                                   |
|-----------|-------------|----------|-----------------|--|
| <Operand> | Output      | BOOL     | E, A, M, D, L   | Operand, dem das negierte VKE zugewiesen wird. |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird zurückgesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" oder "TagIn\_2" hat den Signalzustand "1".
- Der Operand "TagIn\_3" hat den Signalzustand "0".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**R: Ausgang rücksetzen**

**Beschreibung**

Mit der Anweisung "Ausgang rücksetzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "0" zurück.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Boxeingang "1" ist. Wenn der Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf "0" zurückgesetzt. Bei einem VKE von "0" am Boxeingang, bleibt der Signalzustand des angegebenen Operanden unverändert.

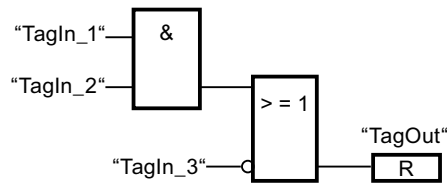
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang rücksetzen":

| Parameter | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung                                   |
|-----------|-------------|----------|-----------------|---------------------|--|
|           |             |          | S7-1200         | S7-1500             |  |
| <Operand> | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Operand, der bei VKE = "1" zurückgesetzt wird. |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Operand "TagIn\_3" liefert den Signalzustand "0".

## Siehe auch

Beispiel für das Steuern eines Förderbands (Seite 1597)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 1598)

Übersicht über die gültigen Datentypen (Seite 1087)

## S: Ausgang setzen

### Beschreibung

Mit der Anweisung "Ausgang setzen" setzen Sie den Signalzustand eines angegebenen Operanden auf "1".

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Boxeingang "1" ist. Wenn der Boxeingang den Signalzustand "1" liefert, wird der angegebene Operand auf "1" gesetzt. Bei einem VKE von "0" am Boxeingang, bleibt der Signalzustand des angegebenen Operanden unverändert.

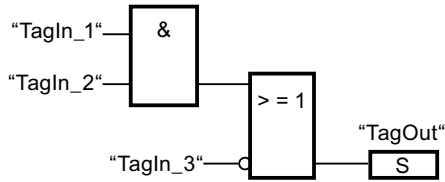
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ausgang setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                             |
|-----------|-------------|----------|-----------------|--|
| <Operand> | Output      | BOOL     | E, A, M, D, L   | Operand, der bei VKE = "1" gesetzt wird. |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird gesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Operand "TagIn\_3" liefert den Signalzustand "0".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**SET\_BF: Bitfeld setzen**

**Beschreibung**

Mit der Anweisung "Bitfeld setzen" setzen Sie mehrere Bit ab einer bestimmten Adresse.

Die Anzahl der zu setzenden Bit bestimmen Sie durch den Wert des Eingangs N. Die Adresse des ersten zu setzenden Bit wird durch (<Operand>) festgelegt. Wenn der Wert des Eingangs N größer als die Anzahl der Bit in einem selektierten Byte ist, werden die Bit des nächstfolgenden Byte gesetzt. Die Bit bleiben so lange gesetzt, bis sie explizit durch eine andere Anweisung zurückgesetzt werden.

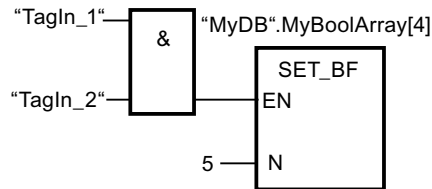
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung                                |
|-----------|-------------|----------|--|---|
| EN        | Input       | BOOL     | E, A, M, D, L  | Freigabeeingang                             |
| N         | Input       | UINT     | Konstante  | Anzahl der zu setzenden Bit                 |
| <Operand> | Output      | BOOL     | E, A, M<br>Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL | Zeiger auf das erste Bit, das gesetzt wird. |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, werden 5 Bits ab der Adresse des Operanden "MyDB".MyBoolArray[4] gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## RESET\_BF: Bitfeld rücksetzen

### Beschreibung

Mit der Anweisung "Bitfeld rücksetzen" setzen Sie mehrere Bit ab einer bestimmten Adresse zurück.

Die Anzahl der zurückzusetzenden Bit bestimmen Sie durch den Wert des Eingangs N. Die Adresse des ersten zurückzusetzenden Bit wird durch (<Operand>) festgelegt. Wenn der Wert des Eingangs N größer als die Anzahl der Bit in einem selektierten Byte ist, werden die Bit des nächstfolgenden Byte zurückgesetzt. Die Bit bleiben so lange zurückgesetzt, bis sie explizit durch eine andere Anweisung gesetzt werden.

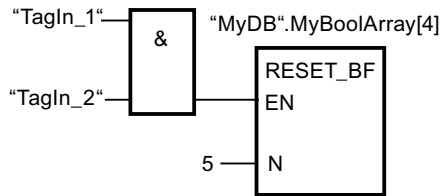
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitfeld rücksetzen":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung                                      |
|-----------|-------------|----------|--|---|
| EN        | Input       | BOOL     | E, A, M, D, L  | Freigabeeingang                                   |
| N         | Input       | UINT     | Konstante  | Anzahl der zurückzusetzenden Bit.                 |
| <Operand> | Output      | BOOL     | E, A, M<br>Bei einem DB oder einem IDB ein Element eines ARRAY[..] of BOOL | Zeiger auf das erste Bit, das zurückgesetzt wird. |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, werden 5 Bits ab der Adresse des Operanden "MyDB".MyBoolArray[4] zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**SR: Flipflop setzen/rücksetzen**

**Beschreibung**

Mit der Anweisung "Flipflop setzen/rücksetzen" setzen oder rücksetzen Sie das Bit eines angegebenen Operanden abhängig vom Signalzustand an den Eingängen S und R1. Wenn der Signalzustand am Eingang S "1" und am Eingang R1 "0" ist, wird der angegebene Operand auf "1" gesetzt. Wenn der Signalzustand am Eingang S "0" und am Eingang R1 "1" ist, wird der angegebene Operand auf "0" zurückgesetzt.

Der Eingang R1 dominiert den Eingang S. Bei einem Signalzustand "1" an beiden Eingängen S und R1 wird der Signalzustand des angegebenen Operanden auf "0" zurückgesetzt.

Bei einem Signalzustand "0" an beiden Eingängen S und R1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

**Parameter**

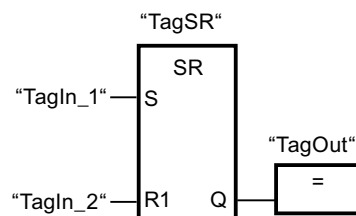
Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop setzen/rücksetzen":

| Parameter | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung         |
|-----------|-------------|----------|-----------------|---------------------|----------------------|
|           |             |          | S7-1200         | S7-1500             |                      |
| S         | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Setzen freigeben     |
| R1        | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Rücksetzen freigeben |

| Parameter | Deklaration | Datentyp | Speicherbereich |               | Beschreibung                                 |
|-----------|-------------|----------|-----------------|---------------|--|
|           |             |          | S7-1200         | S7-1500       |  |
| <Operand> | InOut       | BOOL     | E, A, M, D, L   | E, A, M, D, L | Operand, der gesetzt oder zurückgesetzt wird |
| Q         | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L | Signalzustand des Operanden                  |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Operanden "TagSR" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Der Operand "TagIn\_2" liefert den Signalzustand "0".

Die Operanden "TagSR" und "TagOut" werden zurückgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Operand "TagIn\_1" liefert den Signalzustand "0" und der Operand "TagIn\_2" liefert den Signalzustand "1".
- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## RS: Flipflop rücksetzen/setzen

### Beschreibung

Mit der Anweisung "Flipflop rücksetzen/setzen" rücksetzen oder setzen Sie das Bit eines angegebenen Operanden abhängig vom Signalzustand an den Eingängen R und S1. Wenn der Signalzustand am Eingang R "1" und am Eingang S1 "0" ist, wird der angegebene Operand auf "0" zurückgesetzt. Wenn der Signalzustand am Eingang R "0" und am Eingang S1 "1" ist, wird der angegebene Operand auf "1" gesetzt.

Der Eingang S1 dominiert den Eingang R. Bei einem Signalzustand "1" an beiden Eingängen R und S1 wird der Signalzustand des angegebenen Operanden auf "1" gesetzt.

Bei einem Signalzustand "0" an beiden Eingängen R und S1 wird die Anweisung nicht ausgeführt. Der Signalzustand des Operanden bleibt in diesem Fall unverändert.

Der aktuelle Signalzustand des Operanden wird auf den Ausgang Q übertragen und kann an diesem abgefragt werden.

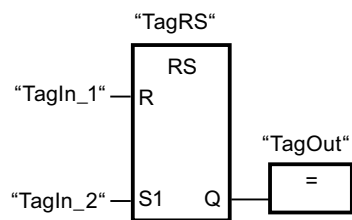
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Flipflop rücksetzen/setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung                                  |
|-----------|-------------|----------|-----------------|---------------------|---|
|           |             |          | S7-1200         | S7-1500             |   |
| R         | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Rücksetzen freigeben                          |
| S1        | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Setzen freigeben                              |
| <Operand> | InOut       | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Operand, der zurückgesetzt oder gesetzt wird. |
| Q         | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Signalzustand des Operanden                   |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Operanden "TagRS" und "TagOut" werden zurückgesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Der Operand "TagIn\_2" liefert den Signalzustand "0".

Die Operanden "TagRS" und "TagOut" werden gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "0" und der Operand "TagIn\_2" liefert den Signalzustand "1".
- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".



**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**P: Operand auf positive Signalflanke abfragen****Beschreibung**

Mit der Anweisung "Operand auf positive Signalflanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "0" auf "1" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Wenn eine positive Signalflanke erfasst wird, liefert der Ausgang der Anweisung den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

**Hinweis**

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

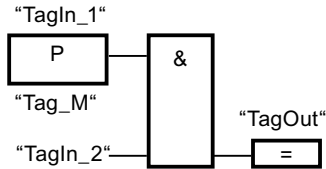
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf positive Signalflanke abfragen":

| Parameter  | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung   |
|------------|-------------|----------|-----------------|---------------------|--|
|            |             |          | S7-1200         | S7-1500             |  |
| <Operand1> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Abzufragendes Signal   |
| <Operand2> | InOut       | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird. |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Am Eingang "TagIn\_1" liegt eine positive Signalfanke vor.
- Der Signalzustand des Operanden "TagIn\_2" ist "1".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Beispiel für das Erfassen der Richtung eines Förderbands (Seite 1598)

### N: Operand auf negative Signalfanke abfragen

#### Beschreibung

Mit der Anweisung "Operand auf negative Signalfanke abfragen" erfassen Sie, ob im Signalzustand eines angegebenen Operanden (<Operand1>) eine Änderung von "1" auf "0" vorliegt. Die Anweisung vergleicht den aktuellen Signalzustand von <Operand1> mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" erkennt, liegt eine negative Signalfanke vor.

Wenn eine negative Signalfanke erfasst wird, liefert der Ausgang der Anweisung den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Den abzufragenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

---

#### Hinweis

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

---

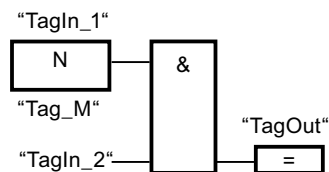
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand auf negative Signalflanke abfragen":

| Parameter  | Deklaration | Datentyp | Speicherbereich |                     | Beschreibung   |
|------------|-------------|----------|-----------------|---------------------|--|
|            |             |          | S7-1200         | S7-1500             |  |
| <Operand1> | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L, T, Z | Abzufragendes Signal   |
| <Operand2> | InOut       | BOOL     | E, A, M, D, L   | E, A, M, D, L       | Flankenmerker, in dem der Signalzustand der vorherigen Abfrage gespeichert wird. |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Am Eingang "TagIn\_1" liegt eine negative Signalflanke vor.
- Der Signalzustand des Operanden "TagIn\_2" ist "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## P=: Operand bei positiver Signalflanke setzen

### Beschreibung

Mit der Anweisung "Operand bei positiver Signalflanke setzen" setzen Sie einen angegebenen Operanden (<Operand2>), wenn eine Änderung im Verknüpfungsergebnis (VKE) von "0" auf "1" vorliegt. Die Anweisung vergleicht das aktuelle VKE mit dem VKE der vorherigen Abfrage, das in einem Flankenmerker (<Operand1>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalflanke vor.

Wenn eine positive Signalflanke erfasst wird, wird <Operand2> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den zu setzenden Operanden (<Operand2>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand1>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

**Hinweis**

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenbewertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

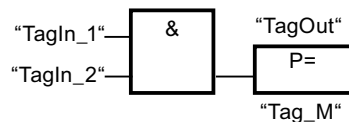
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand bei positiver Signalflanke setzen":

| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|------------|-------------|----------|-----------------|---|
| <Operand2> | Output      | BOOL     | E, A, M, D, L   | Operand, der bei einer positiven Signalflanke gesetzt wird. |
| <Operand1> | InOut       | BOOL     | E, A, M, D, L   | Flankenmerker   |

**Beispiel**

Das folgende Beispiel zeigt die Parameter der Anweisung:



Der Ausgang "TagOut" wird für einen Programmzyklus gesetzt, wenn der Signalzustand am Eingang der Anweisungsbox von "0" auf "1" wechselt (positive Signalflanke). In allen anderen Fällen führt der Ausgang "TagOut" den Signalzustand "0".

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**N=: Operand bei negativer Signalflanke setzen**

**Beschreibung**

Mit der Anweisung "Operand bei negativer Signalflanke setzen" setzen Sie einen angegebenen Operanden (<Operand1>), wenn eine Änderung im Verknüpfungsergebnis (VKE) von "1" auf "0" vorliegt. Die Anweisung vergleicht das aktuelle VKE mit dem VKE der

vorherigen Abfrage, das in einem Flankenmerker (<Operand2>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Wenn eine negative Signalflanke erfasst wird, wird <Operand1> für einen Programmzyklus auf den Signalzustand "1" gesetzt. In allen anderen Fällen liefert der Operand den Signalzustand "0".

Den zu setzenden Operanden (<Operand1>) geben Sie in den Operandenplatzhalter oberhalb der Anweisung an. Den Flankenmerker (<Operand2>) geben Sie in den Operandenplatzhalter unterhalb der Anweisung an.

### Hinweis

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

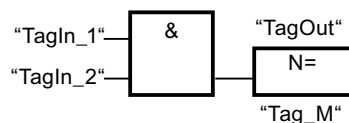
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Operand bei negativer Signalflanke setzen":

| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|------------|-------------|----------|-----------------|---|
| <Operand1> | Output      | BOOL     | E, A, M, D, L   | Operand, der bei einer negativen Signalflanke gesetzt wird. |
| <Operand2> | InOut       | BOOL     | E, A, M, D, L   | Flankenmerker   |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Operand "TagOut" wird für einen Programmzyklus gesetzt, wenn der Signalzustand am Eingang der Anweisungsbox von "1" auf "0" wechselt (negative Signalflanke). In allen anderen Fällen führt der Operand "TagOut" den Signalzustand "0".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## P\_TRIG: VKE auf positive Signalfanke abfragen

### Beschreibung

Mit der Anweisung "VKE auf positive Signalfanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "0" auf "1" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der in einem Flankenmerker (<Operand>) gespeichert wird. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalfanke vor.

Wenn eine positive Signalfanke erfasst wird, liefert der Ausgang der Anweisung den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

### Hinweis

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

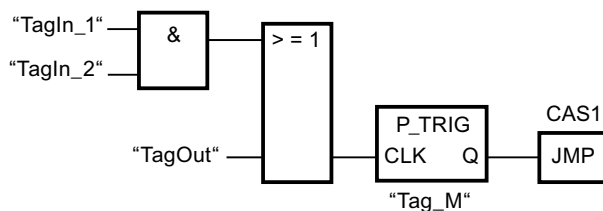
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf positive Signalfanke abfragen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| CLK       | Input       | BOOL     | E, A, M, D, L   | Aktuelles VKE  |
| <Operand> | InOut       | BOOL     | M, D            | Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird. |
| Q         | Output      | BOOL     | E, A, M, D, L   | Ergebnis der Flankenauswertung   |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Im Flankenmerker "Tag\_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## N\_TRIG: VKE auf negative Signalflanke abfragen

### Beschreibung

Mit der Anweisung "VKE auf negative Signalflanke abfragen" fragen Sie eine Änderung im Signalzustand des Verknüpfungsergebnisses (VKE) von "1" nach "0" ab. Die Anweisung vergleicht den aktuellen Signalzustand des VKE mit dem Signalzustand der vorherigen Abfrage, der im Flankenmerker (<Operand>) gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Wenn eine negative Signalflanke erfasst wird, liefert der Ausgang der Anweisung den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

---

### Hinweis

Die Adresse des Flankenmerkers darf nicht mehrfach im Programm verwendet werden, da in diesem Fall das Merkerbit überschrieben wird. Dadurch wird die Flankenauswertung irritiert, so dass das Ergebnis nicht mehr eindeutig ist. Der Speicherbereich des Flankenmerkers muss in einem DB (Static-Bereich bei FB) oder im Merkerbereich liegen.

---

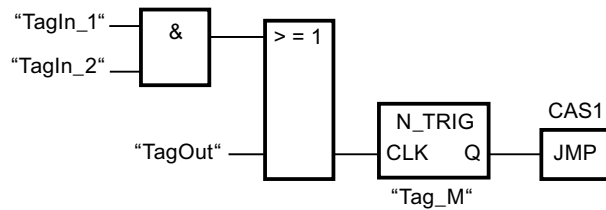
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VKE auf negative Signalflanke abfragen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| CLK       | Input       | BOOL     | E, A, M, D, L   | Aktuelles VKE  |
| <Operand> | InOut       | BOOL     | M, D            | Flankenmerker, in dem das VKE der vorherigen Abfrage gespeichert wird. |
| Q         | Output      | BOOL     | E, A, M, D, L   | Ergebnis der Flankenauswertung   |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Im Flankenmerker "Tag\_M" wird das VKE der vorherigen Abfrage gespeichert. Wenn eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, wird der Sprung zur Sprungmarke CAS1 ausgeführt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### R\_TRIG: Variable bei positiver Signalfanke setzen

#### Beschreibung

Mit der Anweisung "Variable bei positiver Signalfanke setzen" setzen Sie eine angegebene Variable im Instanz-DB, wenn eine Änderung im Verknüpfungsergebnis (VKE) von "0" auf "1" vorliegt. Die Anweisung vergleicht das aktuelle VKE am Eingang CLK mit dem VKE der vorherigen Abfrage, die im angegebenen Instanz-DB gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalfanke vor.

Wenn eine positive Signalfanke erfasst wird, wird die Variable im Instanz-DB auf den Signalzustand "1" gesetzt und der Ausgang Q liefert den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der Flankenmerker in einem eigenen Datenbaustein (Einzelinstantz) oder als lokale Variable (Multiinstantz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".



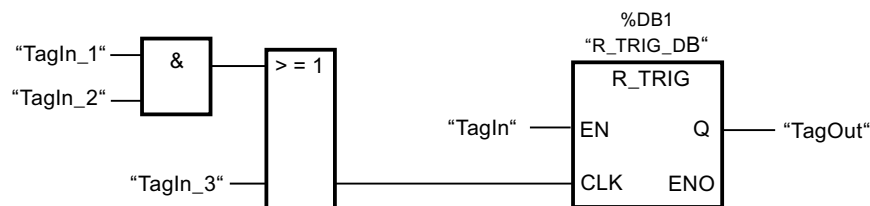
## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Variable bei positiver Signalfanke setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich              |                                    | Beschreibung                                      |
|-----------|-------------|----------|------------------------------|------------------------------------|---|
|           |             |          | S7-1200                      | S7-1500                            |   |
| EN        | Input       | BOOL     | E, A, M, D, L                | E, A, M, D, L, T, Z                | Freigabeeingang                                   |
| ENO       | Output      | BOOL     | E, A, M, D, L                | E, A, M, D, L                      | Freigabeausgang                                   |
| CLK       | Input       | BOOL     | E, A, M, D, L oder Konstante | E, A, M, D, L, T, Z oder Konstante | Eingehendes Signal, dessen Flanke abgefragt wird. |
| Q         | Output      | BOOL     | E, A, M, D, L                | E, A, M, D, L                      | Ergebnis der Flankenauwertung                     |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



In der Variablen im Instanz-DB "R\_TRIG\_DB" wird das VKE der vorherigen Abfrage gespeichert. Wenn in den Operanden "TagIn\_1" und "TagIn\_2" oder im Operanden "TagIn\_3" eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, liefert der Ausgang "TagOut" den Signalzustand "1".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## F\_TRIG: Variable bei negativer Signalfanke setzen

### Beschreibung

Mit der Anweisung "Variable bei negativer Signalfanke setzen" setzen Sie eine angegebene Variable im Instanz-DB, wenn eine Änderung im Verknüpfungsergebnis (VKE) von "1" auf "0" vorliegt. Die Anweisung vergleicht das aktuelle VKE am Eingang CLK mit dem VKE der vorherigen Abfrage, die im angegebenen Instanz-DB gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalfanke vor.

Wenn eine negative Signalflanke erfasst wird, wird die Variable im Instanz-DB auf den Signalzustand "1" gesetzt und der Ausgang Q liefert den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der Flankenmerker in einem eigenen Datenbaustein (Einzelnstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

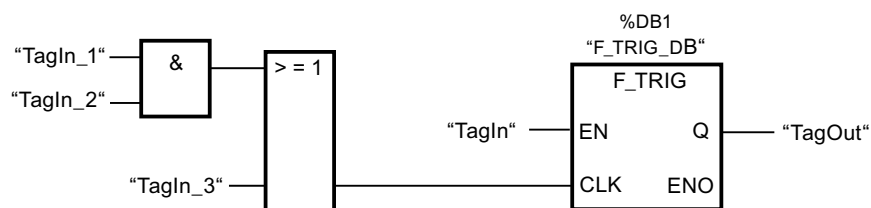
### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Variable bei negativer Signalflanke setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich              |                                    | Beschreibung                                      |
|-----------|-------------|----------|------------------------------|------------------------------------|---|
|           |             |          | S7-1200                      | S7-1500                            |   |
| EN        | Input       | BOOL     | E, A, M, D, L                | E, A, M, D, L, T, Z                | Freigabeeingang                                   |
| ENO       | Output      | BOOL     | E, A, M, D, L                | E, A, M, D, L                      | Freigabeausgang                                   |
| CLK       | Input       | BOOL     | E, A, M, D, L oder Konstante | E, A, M, D, L, T, Z oder Konstante | Eingehendes Signal, dessen Flanke abgefragt wird. |
| Q         | Output      | BOOL     | E, A, M, D, L                | E, A, M, D, L                      | Ergebnis der Flankenauswertung                    |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



In der Variablen im Instanz-DB "F\_TRIG\_DB" wird das VKE der vorherigen Abfrage gespeichert. Wenn in den Operanden "TagIn\_1" und "TagIn\_2" oder im Operanden "TagIn\_3" eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, liefert der Ausgang "TagOut" den Signalzustand "1".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Zeiten

### IEC-Zeiten

#### TP: Impuls erzeugen

#### Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Ausgang Q für die parametrisierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die parametrisierte Zeitdauer PT ab. Der Ausgang Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf (positive Signalflanke) des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Ausgang Q nicht, solange die Zeitdauer PT läuft.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die parametrisierte Zeitdauer PT erreicht ist und der Signalzustand am Eingang IN "0" ist, wird der Ausgang ET zurückgesetzt.

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

---

#### Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

#### Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TP\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

#### Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TP\_TIME oder TP\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME, TP\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelninstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Impuls erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

### Parameter

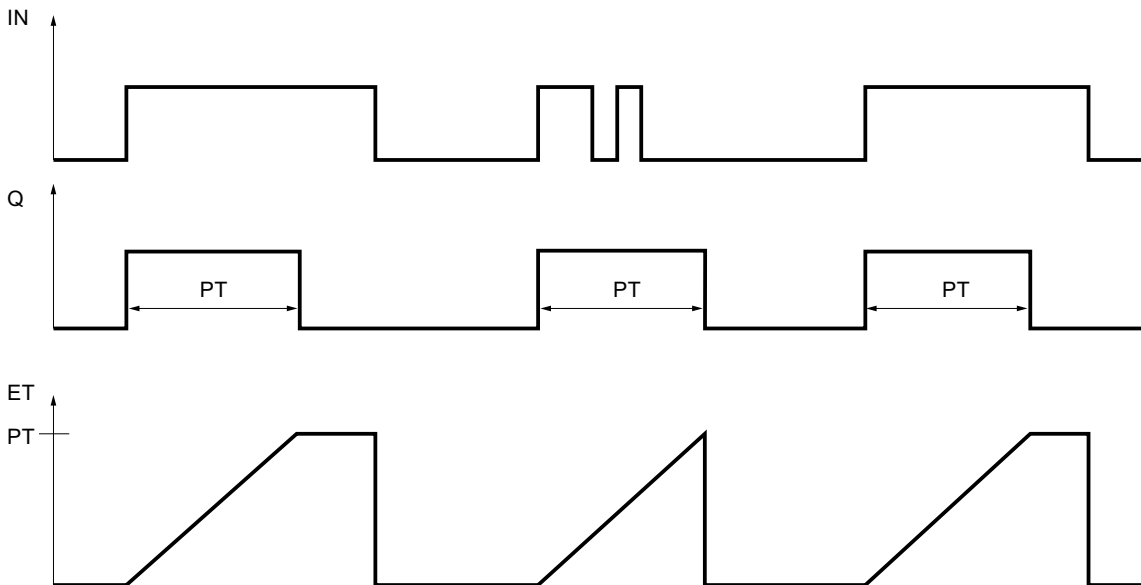
Die folgende Tabelle zeigt die Parameter der Anweisung "Impuls erzeugen":

| Parameter | Deklaration | Datentyp |             | Speicherbereich              |                                 | Beschreibung   |
|-----------|-------------|----------|-------------|------------------------------|---------------------------------|--|
|           |             | S7-1200  | S7-1500     | S7-1200                      | S7-1500                         |  |
| IN        | Input       | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, T, Z, P          | Starteingang   |
| PT        | Input       | TIME     | TIME, LTIME | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Zeitdauer des Impulses.<br>Der Wert am Parameter PT muss positiv sein. |
| Q         | Output      | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Impulsausgang  |
| ET        | Output      | TIME     | TIME, LTIME | E, A, M, D, L                | E, A, M, D, L, P                | Aktueller Zeitwert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":



## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Beispiel für das Kontrollieren der Raumtemperatur (Seite 1602)

## TON: Einschaltverzögerung erzeugen

### Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Ausgangs Q um die parametrisierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalfanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Ausgang Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalfanke am Starteingang erfasst wird.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Der Ausgang ET wird zurückgesetzt, sobald der Signalzustand am Eingang IN auf "0" wechselt.

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

---

#### **Hinweis**

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

### **Für CPU S7-1200**

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TON\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### **Für CPU S7-1500**

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TON\_TIME oder TON\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME, TON\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Einschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

## Parameter

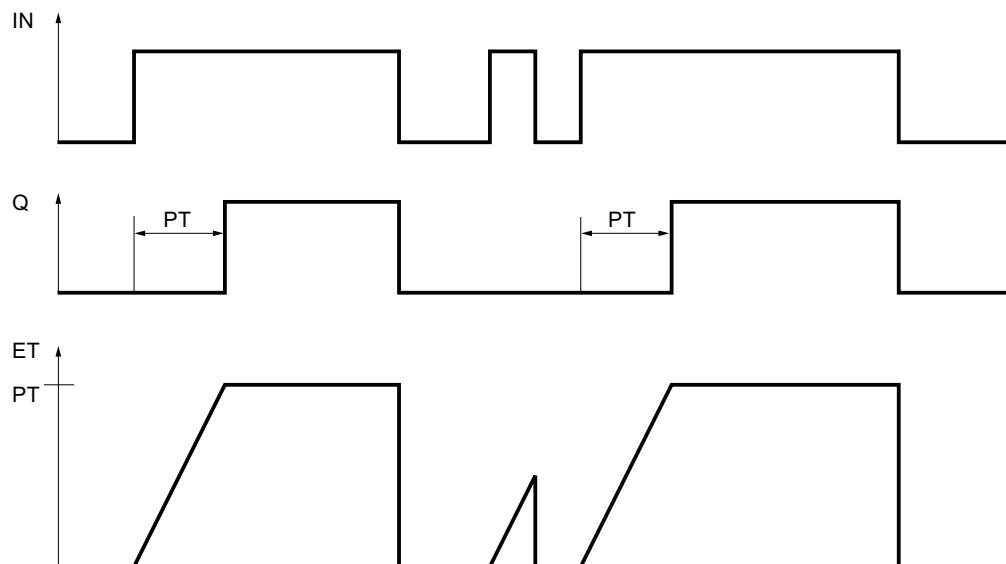
Die folgende Tabelle zeigt die Parameter der Anweisung "Einschaltverzögerung erzeugen":

| Parameter | Deklaration | Datentyp |             | Speicherbereich              |                                 | Beschreibung   |
|-----------|-------------|----------|-------------|------------------------------|---------------------------------|--|
|           |             | S7-1200  | S7-1500     | S7-1200                      | S7-1500                         |  |
| IN        | Input       | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, T, Z, P          | Starteingang   |
| PT        | Input       | TIME     | TIME, LTIME | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Zeitdauer der Einschaltverzögerung.<br>Der Wert am Parameter PT muss positiv sein. |
| Q         | Output      | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird.                             |
| ET        | Output      | TIME     | TIME, LTIME | E, A, M, D, L                | E, A, M, D, L, P                | Aktueller Zeitwert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":



## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## TOF: Ausschaltverzögerung erzeugen

### Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Ausgangs Q um die parametrisierte Zeitdauer PT. Der Ausgang Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalflanke). Wenn der Signalzustand am Eingang IN wieder auf "0" wechselt (positive Signalflanke), läuft die parametrisierte Zeitdauer PT ab. Der Ausgang Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeitdauer PT wird der Ausgang Q zurückgesetzt. Falls der Signalzustand am Eingang IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Ausgang Q bleibt weiterhin auf "1" gesetzt.

Am Ausgang ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Ausgang ET solange auf dem aktuellen Wert stehen, bis der Eingang IN wieder auf "1" wechselt. Wenn der Eingang IN vor dem Ablauf der Zeitdauer PT auf "1" wechselt, wird der Ausgang ET auf den Wert T#0s zurückgesetzt.

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

---

### Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

## Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TOF\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

## Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TOF\_TIME oder TOF\_LTIMER, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME, TOF\_LTIMER, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)



Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstantz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Ausschaltverzögerung erzeugen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

## Parameter

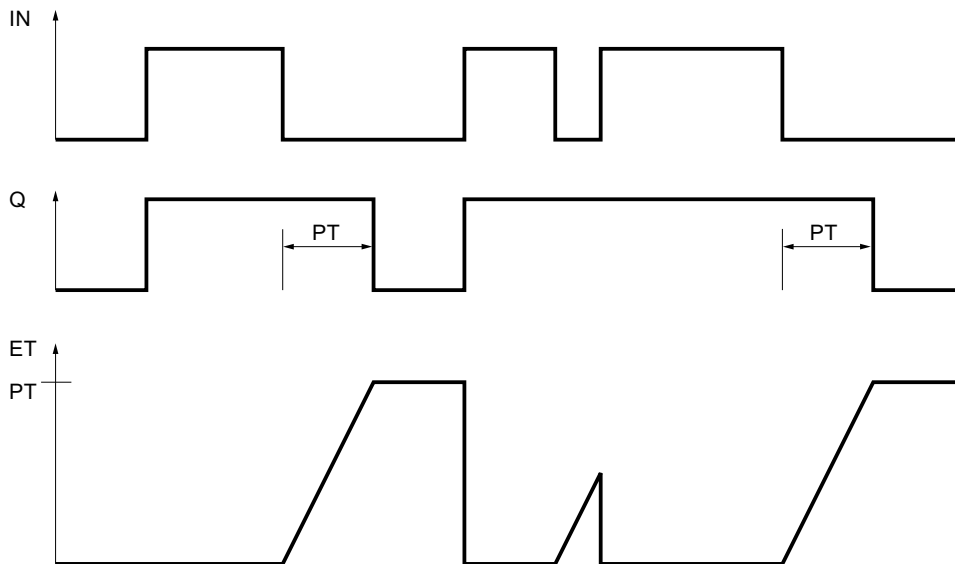
Die folgende Tabelle zeigt die Parameter der Anweisung "Ausschaltverzögerung erzeugen":

| Parameter | Deklaration | Datentyp |             | Speicherbereich              |                                 | Beschreibung  |
|-----------|-------------|----------|-------------|------------------------------|---------------------------------|---|
|           |             | S7-1200  | S7-1500     | S7-1200                      | S7-1500                         |   |
| IN        | Input       | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, T, Z, P          | Starteingang  |
| PT        | Input       | TIME     | TIME, LTIME | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Zeitdauer der Ausschaltverzögerung<br>Der Wert am Parameter PT muss positiv sein. |
| Q         | Output      | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, P                | Ausgang, der nach dem Ablauf der Zeit PT zurückgesetzt wird.                      |
| ET        | Output      | TIME     | TIME, LTIME | E, A, M, D, L                | E, A, M, D, L, P                | Aktueller Zeitwert  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":



### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### TONR: Zeit akkumulieren

#### Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines durch den Parameter PT gesetzten Zeitraums. Wenn das Verknüpfungsergebnis (VKE) am Eingang IN von "0" auf "1" wechselt (positive Signalfanke), wird die Anweisung ausgeführt und die parametrisierte Zeitdauer PT gestartet. Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Eingang IN erfasst werden. Die akkumulierte Zeit wird am Ausgang ET ausgegeben und kann an diesem abgefragt werden. Wenn der aktuelle Zeitwert PT erreicht ist, liefert der Ausgang Q den Signalzustand "1". Der Ausgang Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Eingang IN auf "0" wechselt.

Der Eingang R setzt die Ausgänge ET und Q unabhängig vom Signalzustand am Starteingang zurück.

Jedem Aufruf der Anweisung "Zeit akkumulieren" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

## Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TONR\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

## Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TONR\_TIME oder TONR\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME, TONR\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstantz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerkes angeordnet werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

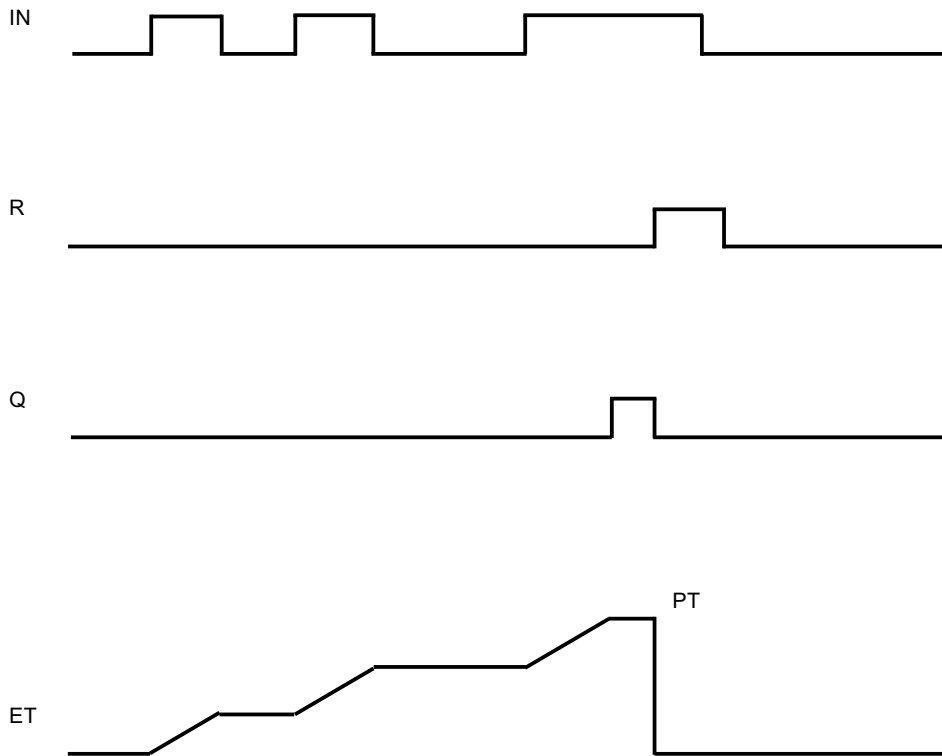
| Parameter | Deklaration | Datentyp |             | Speicherbereich              |                                 | Beschreibung   |
|-----------|-------------|----------|-------------|------------------------------|---------------------------------|--|
|           |             | S7-1200  | S7-1500     | S7-1200                      | S7-1500                         |  |
| IN        | Input       | BOOL     | BOOL        | E, A, M, D, L                | E, A, M, D, L, T, Z, P          | Starteingang   |
| R         | Input       | BOOL     | BOOL        | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Rücksetzeingang  |
| PT        | Input       | TIME     | TIME, LTIME | E, A, M, D, L oder Konstante | E, A, M, D, L, P oder Konstante | Maximale Dauer der Zeiterfassung.<br>Der Wert am Parameter PT muss positiv sein. |

| Parameter | Deklaration | Datentyp |             | Speicherbereich |                  | Beschreibung   |
|-----------|-------------|----------|-------------|-----------------|------------------|--|
|           |             | S7-1200  | S7-1500     | S7-1200         | S7-1500          |  |
| Q         | Output      | BOOL     | BOOL        | E, A, M, D, L   | E, A, M, D, L, P | Ausgang, der nach dem Ablauf der Zeit PT gesetzt wird. |
| ET        | Output      | TIME     | TIME, LTIME | E, A, M, D, L   | E, A, M, D, L, P | Aktueller Zeitwert                                     |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit akkumulieren":



### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## TP: Zeit als Impuls starten

### Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Impuls. Die IEC-Zeit wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) wechselt. Die IEC-Zeit läuft mit der angegebenen Zeitdauer ab, unabhängig vom weiteren Verlauf des VKE. Auch die Erfassung einer neuen positiven Signalfanke beeinflusst den Ablauf der IEC-Zeit nicht. Solange die IEC-Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn die IEC-Zeit abgelaufen ist, liefert der Zeitstatus den Signalzustand "0".

---

#### Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablafebene befinden, da jede Abfrage der Ausgänge Q oder ET die IEC\_TIMER-Struktur aktualisiert.

---

### Für CPU S7-1200

Die Anweisung "Zeit als Impuls starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER oder TP\_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### Für CPU S7-1500

Die Anweisung "Zeit als Impuls starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER, IEC\_LTIMER, TP\_TIME oder TP\_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME, TP\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente Q der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe einer binären Verknüpfung abfragen. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

Die Ausführung der Anweisung "Zeit als Impuls starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

### Parameter

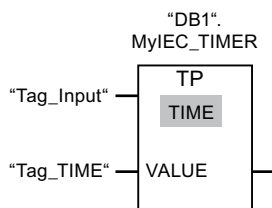
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

| Parameter  | Deklaration | Datentyp           |  | Speicherbereich              | Beschreibung                           |
|------------|-------------|--------------------|--|------------------------------|--|
|            |             | S7-1200            | S7-1500                                  |                              |  |
| VALUE      | Input       | TIME               | TIME, LTIME                              | E, A, M, D, L oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft. |
| <IEC-Zeit> | InOut       | IEC_TIMER, TP_TIME | IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME | D, L                         | IEC-Zeit, die gestartet wird.          |

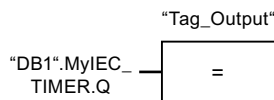
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Anweisung "Zeit als Impuls starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag\_Input" von "0" auf "1" wechselt. Die Zeit "DB1".MyIEC\_TIMER wird für die Zeitdauer gestartet, die im Operanden "TagTime" gespeichert ist.



Solange die Zeit "DB1".MyIEC\_TIMER läuft, liefert der Zeitstatus ("DB1".MyIEC\_TIMER.Q) den Signalzustand "1" und der Operand "Tag\_Output" wird gesetzt. Wenn die IEC-Zeit abgelaufen ist, wechselt der Signalzustand des Zeitstatus auf "0" und der Operand "Tag\_Output" wird zurückgesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### TON: Zeit als Einschaltverzögerung starten

### Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Einschaltverzögerung. Die IEC-Zeit wird gestartet, wenn das

Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) wechselt. Die IEC-Zeit läuft mit der angegebenen Zeitdauer ab. Der Ausgang liefert den Signalzustand "1", wenn das VKE am Eingang der Anweisung den Signalzustand "1" führt. Wenn das VKE vor dem Ablauf der Zeit auf "0" wechselt, wird die laufende IEC-Zeit zurückgesetzt. Die Abfrage des Zeitstatus auf "1" liefert in diesem Fall den Signalzustand "0". Mit der Erfassung der nächsten positiven Signalflanke am Eingang der Anweisung startet die IEC-Zeit wieder.

---

**Hinweis**

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablafebene befinden, da jede Abfrage der Ausgänge Q oder ET die IEC\_TIMER-Struktur aktualisiert.

---

**Für CPU S7-1200**

Die Anweisung "Zeit als Einschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER oder TON\_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

**Für CPU S7-1500**

Die Anweisung "Zeit als Einschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER, IEC\_LTIMER, TON\_TIME oder TON\_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME, TON\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe einer binären Verknüpfung abfragen. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

Die Ausführung der Anweisung "Zeit als Einschaltverzögerung starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

**Parameter**

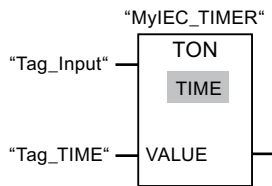
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

| Parameter  | Deklaration | Datentyp            |  | Speicherbereich                 | Beschreibung                           |
|------------|-------------|---------------------|--|---------------------------------|--|
|            |             | S7-1200             | S7-1500                                    |                                 |  |
| VALUE      | Input       | TIME                | TIME, LTIME                                | E, A, M, D, L<br>oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft. |
| <IEC-Zeit> | InOut       | IEC_TIMER, TON_TIME | IEC_TIMER, IEC_LTIMER, TON_TIME, TON_LTIME | D, L                            | IEC-Zeit, die gestartet wird.          |

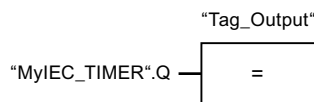
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Anweisung "Zeit als Einschaltverzögerung starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag\_Input" von "0" auf "1" wechselt. Die Zeit "MyIEC\_TIMER" wird für die Zeitdauer gestartet, die im Operanden "Tag\_TIME" gespeichert ist.



Wenn die Zeit "MyIEC\_TIMER" abgelaufen ist und der Operand "Tag\_Input" den Signalzustand "1" führt, liefert die Abfrage des Zeitstatus ("MyIEC\_TIMER".Q) den Signalzustand "1" und der Operand "Tag\_Output" wird gesetzt. Wenn der Signalzustand des Operanden "Tag\_Input" auf "0" wechselt, liefert die Abfrage des Zeitstatus den Signalzustand "0" und der Operand "Tag\_Output" wird zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)



## TOF: Zeit als Ausschaltverzögerung starten

### Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine IEC-Zeit mit einer bestimmten Zeitdauer als Ausschaltverzögerung. Die Abfrage des Zeitstatus auf "1" liefert den Signalzustand "1", wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung den Signalzustand "0" führt. Wenn das VKE von "1" auf "0" (negative Signalflanke) wechselt, wird die IEC-Zeit mit der angegebenen Zeitdauer gestartet. Solange die IEC-Zeit läuft, bleibt der Zeitstatus auf den Signalzustand "1". Wenn die Zeit abgelaufen ist und das VKE am Eingang der Anweisung den Signalzustand "0" führt, wird der Zeitstatus auf den Signalzustand "0" gesetzt. Wechselt das VKE vor dem Ablauf der Zeit auf "1", wird die laufende IEC-Zeit zurückgesetzt und der Zeitstatus bleibt auf den Signalzustand "1".

---

#### Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablafebenebenen befinden, da jede Abfrage der Ausgänge Q oder ET die IEC\_TIMER-Struktur aktualisiert.

---

### Für CPU S7-1200

Die Anweisung "Zeit als Ausschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER oder TOF\_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### Für CPU S7-1500

Die Anweisung "Zeit als Ausschaltverzögerung starten" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER, IEC\_LTIMER, TOF\_TIME oder TOF\_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME, TOF\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe einer binären Verknüpfung abfragen. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

Die Ausführung der Anweisung "Zeit als Ausschaltverzögerung starten" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

### Parameter

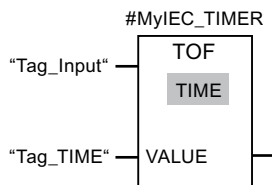
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

| Parameter  | Deklaration | Datentyp            |  | Speicherbereich              | Beschreibung                           |
|------------|-------------|---------------------|--|------------------------------|--|
|            |             | S7-1200             | S7-1500                                    |                              |  |
| VALUE      | Input       | TIME                | TIME, LTIME                                | E, A, M, D, L oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft. |
| <IEC-Zeit> | InOut       | IEC_TIMER, TOF_TIME | IEC_TIMER, IEC_LTIMER, TOF_TIME, TOF_LTIME | D, L                         | IEC-Zeit, die gestartet wird.          |

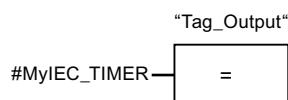
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Anweisung "Zeit als Ausschaltverzögerung starten" wird ausgeführt, wenn der Signalzustand des Operanden "Tag\_Input" von "1" auf "0" wechselt. Die Zeit #MyIEC\_TIMER wird für die Zeitdauer gestartet, die im Operanden "Tag\_TIME" gespeichert ist.



Solange die Zeit #MyIEC\_TIMER läuft, liefert die Abfrage des Zeitstatus (#MyIEC\_TIMER.Q) den Signalzustand "1" und der Operand "Tag\_Output" wird gesetzt. Wenn die Zeit abgelaufen ist und der Operand "Tag\_Input" den Signalzustand "0" führt, liefert die Abfrage des Zeitstatus den Signalzustand "0". Wechselt der Signalzustand des Operanden "Tag\_Input" vor dem Ablauf der Zeit #MyIEC\_TIMER auf "1", wird die Zeit zurückgesetzt. Beim Signalzustand "1" des Operanden "Tag\_Input" liefert die Abfrage des Zeitstatus den Signalzustand "1".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## TONR: Zeit akkumulieren

### Beschreibung

Mit der Anweisung "Zeit akkumulieren" erfassen Sie, wie lange das Signal am Eingang der Anweisung "1" ist. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) wechselt. Solange das VKE "1" ist, wird die Zeit erfasst. Wechselt das VKE auf "0", wird die Anweisung angehalten. Wechselt das VKE erneut auf "1", wird die Zeiterfassung fortgesetzt. Die Abfrage des Zeitstatus auf "1" liefert den Signalzustand "1", wenn die erfasste Zeit den Wert der angegebenen Zeitdauer überschreitet und das VKE am Eingang der Spule "1" führt.

Der Zeitstatus und die aktuell abgelaufene Zeit können durch die Anweisung "Zeit rücksetzen" auf "0" zurückgesetzt werden.

---

### Hinweis

Der Start und die Abfrage der IEC-Zeit können sich in verschiedenen Ablauebene befinden, da jede Abfrage der Ausgänge Q oder ET die IEC\_TIMER-Struktur aktualisiert.

---

### Für CPU S7-1200

Die Anweisung "Zeit akkumulieren" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER oder TONR\_TIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME oder IEC\_TIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

### Für CPU S7-1500

Die Anweisung "Zeit akkumulieren" legt ihre Daten in einer Struktur des Datentyps IEC\_TIMER, IEC\_LTIMER, TONR\_TIME oder TONR\_LTIME ab. Sie können die Struktur wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME, TONR\_LTIME, IEC\_TIMER oder IEC\_LTIMER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_TIMER)

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die zugeordnete Zeit.

Der aktuelle Zeitstatus wird in der Strukturkomponente ET der IEC-Zeit abgelegt. Den Zeitstatus können Sie mithilfe einer binären Verknüpfung abfragen. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

Die Ausführung der Anweisung "Zeit akkumulieren" setzt eine Vorverknüpfung voraus. Sie kann nur am Ende des Netzwerks angeordnet werden.

**Parameter**

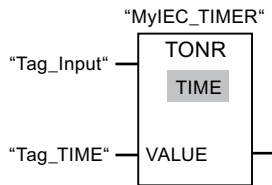
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit akkumulieren":

| Parameter  | Deklaration | Datentyp             |  | Speicherbereich              | Beschreibung                           |
|------------|-------------|----------------------|--|------------------------------|--|
|            |             | S7-1200              | S7-1500                                      |                              |  |
| VALUE      | Input       | TIME                 | TIME, LTIME                                  | E, A, M, D, L oder Konstante | Zeitdauer, mit der die IEC-Zeit läuft. |
| <IEC-Zeit> | InOut       | IEC_TIMER, TONR_TIME | IEC_TIMER, IEC_LTIMER, TONR_TIME, TONR_LTIME | D, L                         | IEC-Zeit, die gestartet wird.          |

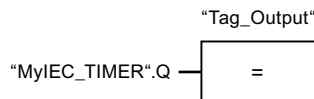
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die Anweisung "Zeit akkumulieren" wird bei einer positiven Signalfanke im VKE ausgeführt. Solange der Operand "Tag\_Input" den Signalzustand "1" liefert, wird die Zeit erfasst.



Wenn die erfasste Zeit den Wert des Operanden "Tag\_TIME" überschreitet, liefert die Abfrage des Zeitstatus ("MyIEC\_TIMER".Q) den Signalzustand "1" und der Operand "Tag\_Output" wird gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

RT: Zeit rücksetzen (Seite 1927)

## RT: Zeit rücksetzen

### Beschreibung

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine IEC-Zeit auf "0" zurück. Die IEC-Zeit, die zurückgesetzt werden soll, geben Sie an, indem Sie in den Platzhalter über der Anweisung den Namen des Datenbausteins eintragen, der die Struktur der IEC-Zeit enthält.

Die Anweisung wird nur ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Boxeingang "1" ist. Wenn die Anweisung ausgeführt wird, werden die Strukturkomponenten der IEC-Zeit im angegebenen Datenbaustein auf "0" zurückgesetzt. Wenn das VKE am Boxeingang "0" ist, wird die Anweisung nicht ausgeführt.

Die Anweisung beeinflusst das VKE nicht. Das VKE am Boxeingang wird direkt auf den Boxausgang übertragen.

Der Anweisung "Zeit rücksetzen" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung und nicht bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage der Daten ist nur gleich vom Aufruf der Anweisung bis zum nächsten Aufruf der Anweisung.

### Parameter

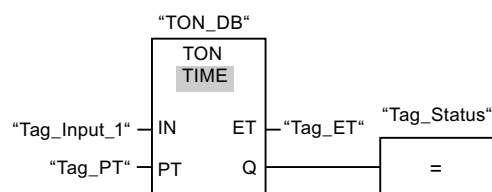
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit rücksetzen":

| Parameter  | Deklaration | Datentyp  |  | Speicherbereich | Beschreibung                      |
|------------|-------------|---|--|-----------------|-----------------------------------|
|            |             | S7-1200   | S7-1500  |                 |                                   |
| <IEC-Zeit> | InOut       | IEC_TIMER,<br>TP_TIME,<br>TON_TIME,<br>TOF_TIME,<br>TONR_TIME | IEC_TIMER,<br>IEC_LTIMER,<br>TP_TIME,<br>TP_LTIME,<br>TON_TIME,<br>TON_LTIME,<br>TOF_TIME,<br>TOF_LTIME,<br>TONR_TIME,<br>TONR_LTIME | D, L            | IEC-Zeit, die zurückgesetzt wird. |

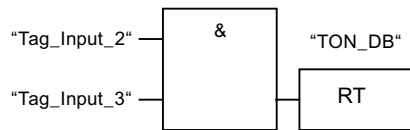
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "Tag\_Input\_1" von "0" auf "1" wechselt, wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON\_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag\_PT" vorgegeben ist.



Wenn die Operanden "Tag\_Input\_2" und "Tag\_Input\_3" den Signalzustand "1" liefern, wird die Anweisung "Zeit rücksetzen" ausgeführt und die im Datenbaustein "TON\_DB" abgelegte IEC-Zeit zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**PT: Zeitdauer laden**

**Beschreibung**

Mit der Anweisung "Zeitdauer laden" parametrisiert die Zeitdauer einer IEC-Zeit. Die Anweisung wird in jedem Zyklus ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisungsbox den Signalzustand "1" führt. Die Anweisung schreibt die angegebene Zeitdauer in die Struktur der angegebenen IEC-Zeit.

---

**Hinweis**

Wenn die angegebene IEC-Zeit während der Ausführung der Anweisung läuft, überschreibt die Anweisung die aktuelle Zeitdauer der angegebenen IEC-Zeit. Dadurch kann sich der Zeitstatus der IEC-Zeit ändern.

---

Der Anweisung "Zeitdauer laden" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht bei einem Aufruf der Anweisung und bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

## Parameter

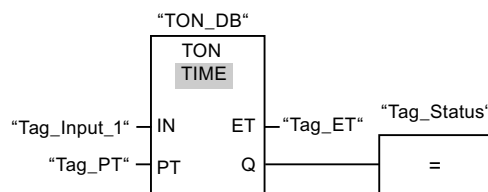
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitdauer laden":

| Parameter  | Deklaration | Datentyp  |  | Speicherbereich                 | Beschreibung                                     |
|------------|-------------|---|--|---------------------------------|--|
|            |             | S7-1200   | S7-1500  |                                 |  |
| PT         | Input       | TIME  | TIME, LTIME  | E, A, M, D, L<br>oder Konstante | Zeitdauer  |
| <IEC-Zeit> | InOut       | IEC_TIMER,<br>TP_TIME,<br>TON_TIME,<br>TOF_TIME,<br>TONR_TIME | IEC_TIMER,<br>IEC_LTIMER,<br>TP_TIME,<br>TP_LTIME,<br>TON_TIME,<br>TON_LTIME,<br>TOF_TIME,<br>TOF_LTIME,<br>TONR_TIME,<br>TONR_LTIME | D, L                            | IEC-Zeit,<br>deren Dauer<br>eingestellt<br>wird. |

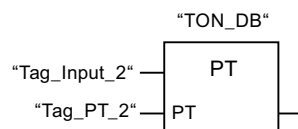
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "Tag\_Input\_1" von "0" auf "1" wechselt, wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON\_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag\_PT" vorgegeben ist.



Wenn der Operand "Tag\_Input\_2" den Signalzustand "1" führt, wird die Anweisung "Zeitdauer laden" ausgeführt. Die Anweisung schreibt die Zeitdauer "Tag\_PT\_2" in den Instanz-Datenbaustein "TON\_DB" und überschreibt dabei den Wert des Operanden "Tag\_PT" innerhalb des Datenbausteins. Dadurch kann sich der Signalzustand des Zeitstatus bei der nächsten Abfrage bzw. beim Zugriff auf "MyTimer".Q oder "MyTimer".ET ändern.

### Hinweis

Der "Tag\_Input\_2" wird als Impulsmerker ausgeführt, damit das Laden der Zeitdauer nur über einen Programmzyklus erfolgt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## SIMATIC-Zeiten

### S\_IMPULS: Zeit als Impuls parametrieren und starten

#### Beschreibung

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn der Signalzustand am Eingang S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten. In diesem Fall ist der Signalzustand am Ausgang Q "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als Impuls parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

---

#### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".



## Parameter

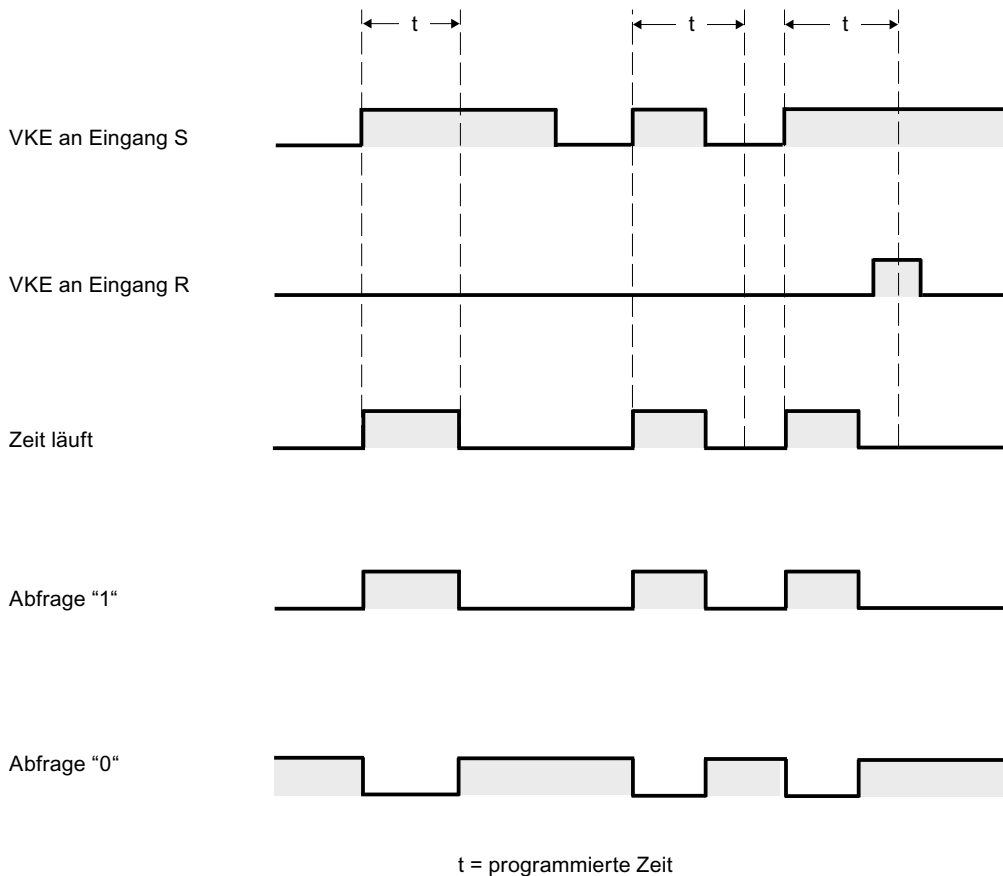
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, T, Z, D, L, P                | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Zeitdauer   |
| R         | Input       | BOOL         | E, A, M, T, Z, D, L, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L, P                      | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

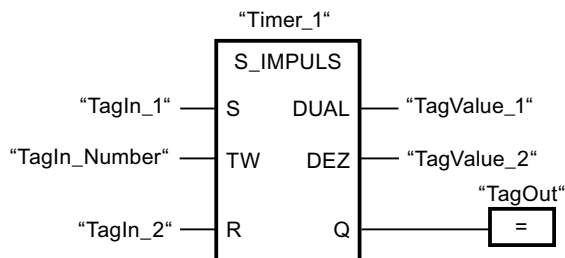
### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":



### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, solange der Operand "TagIn\_1" den Signalzustand "1" liefert. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit "Timer\_1" angehalten. Der Operand "TagOut" wird in diesem Fall auf "0" zurückgesetzt.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft und der Operand "TagIn\_1" den Signalzustand "1" hat. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## S\_VIMP: Zeit als verlängerten Impuls parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Solange die Zeit läuft, liefert der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Ausgang Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der am Eingang TW programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Die Anweisung "Zeit als verlängerten Impuls parametrieren und starten" benötigt für die Flankenbewertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusbeginn andere Werte als am Zyklusende liefert.

---

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

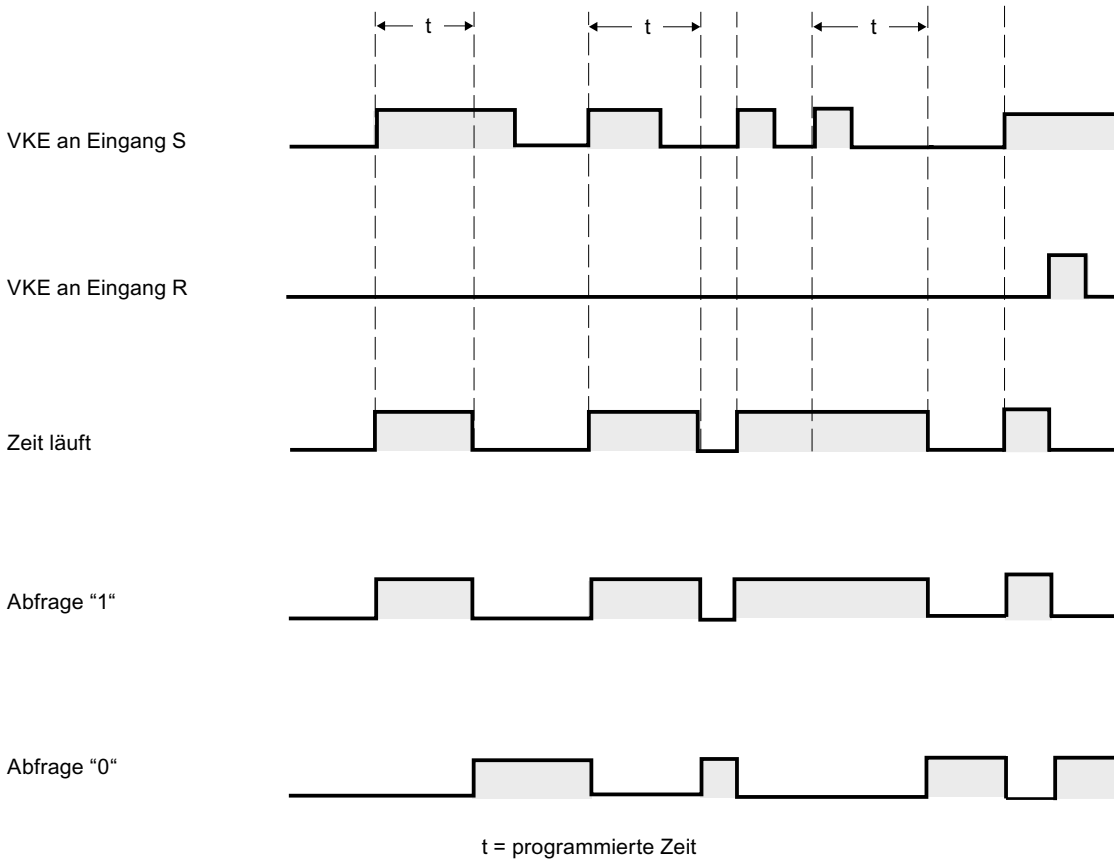
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, T, Z, D, L, P                | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Zeitdauer   |
| R         | Input       | BOOL         | E, A, M, T, Z, D, L, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L, P                      | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

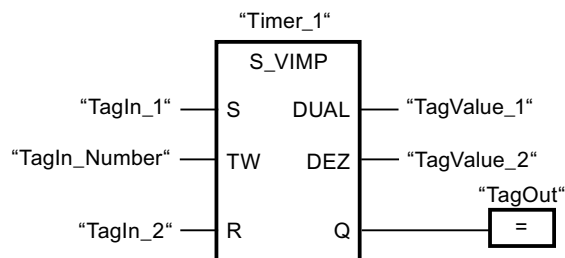
### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":



### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, ohne von einer fallenden Flanke am Eingang S beeinträchtigt zu werden. Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

Der Operand "TagOut" liefert den Signalzustand "1", solange die Zeit läuft. Wenn die Zeit abgelaufen ist oder zurückgesetzt wurde, wird der Operand "TagOut" auf "0" zurückgesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## S\_EVERZ: Zeit als Einschaltverzögerung parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, solange der Signalzustand am Eingang S "1" ist. Wenn die Zeit ordnungsgemäß abgelaufen ist und am Eingang S der Signalzustand "1" noch anliegt, liefert der Ausgang Q den Signalzustand "1". Wechselt der Signalzustand am Eingang S von "1" auf "0", während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Ausgang Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Eingang R zurückgesetzt, auch wenn die Zeit nicht läuft und das VKE am Eingang S "1" ist.

Die Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

---

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

## Parameter

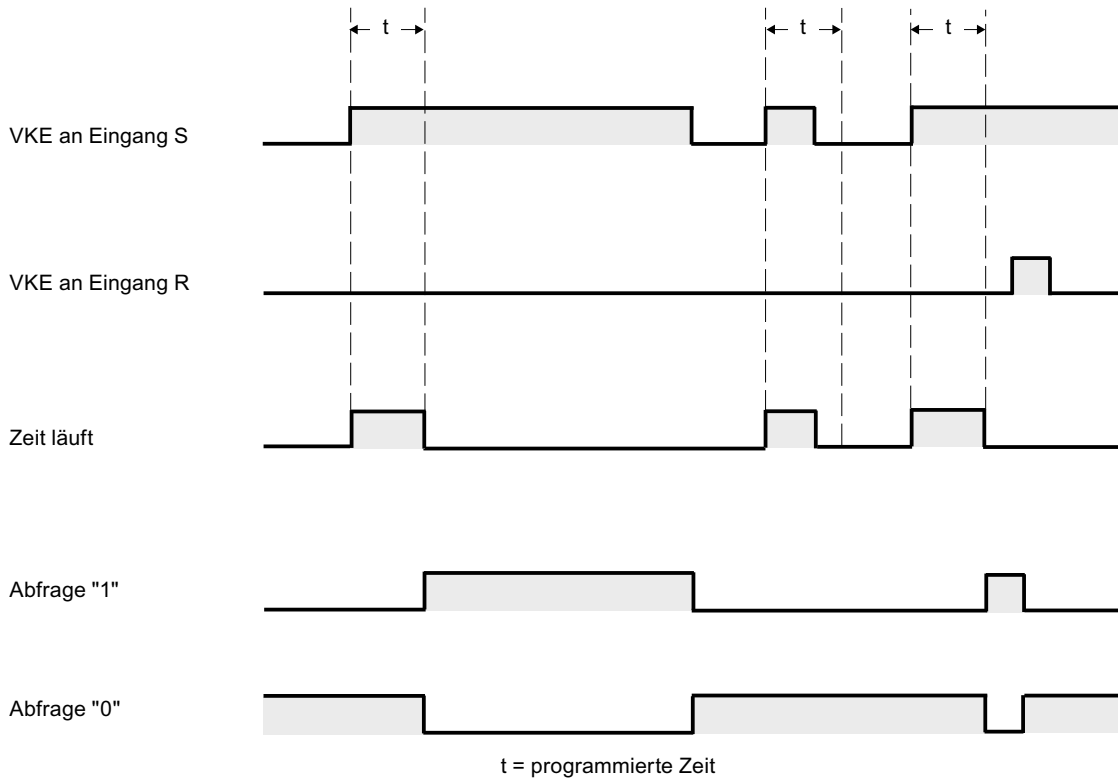
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, T, Z, D, L, P                | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Zeitdauer   |
| R         | Input       | BOOL         | E, A, M, T, Z, D, L, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L, P                      | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

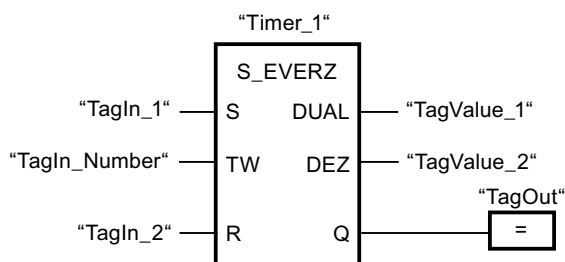
### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":



### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "1" ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Der Operand "TagOut" liefert in diesem Fall den Signalzustand "0".



**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**S\_SEVERZ: Zeit als speichernde Einschaltverzögerung parametrieren und starten****Beschreibung**

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab, auch wenn der Signalzustand am Eingang S auf "0" wechselt. Wenn die Zeit abgelaufen ist, liefert der Ausgang Q den Signalzustand "1" unabhängig vom Signalzustand am Eingang S. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer (TW) neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Starteingang S. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

---

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

### Parameter

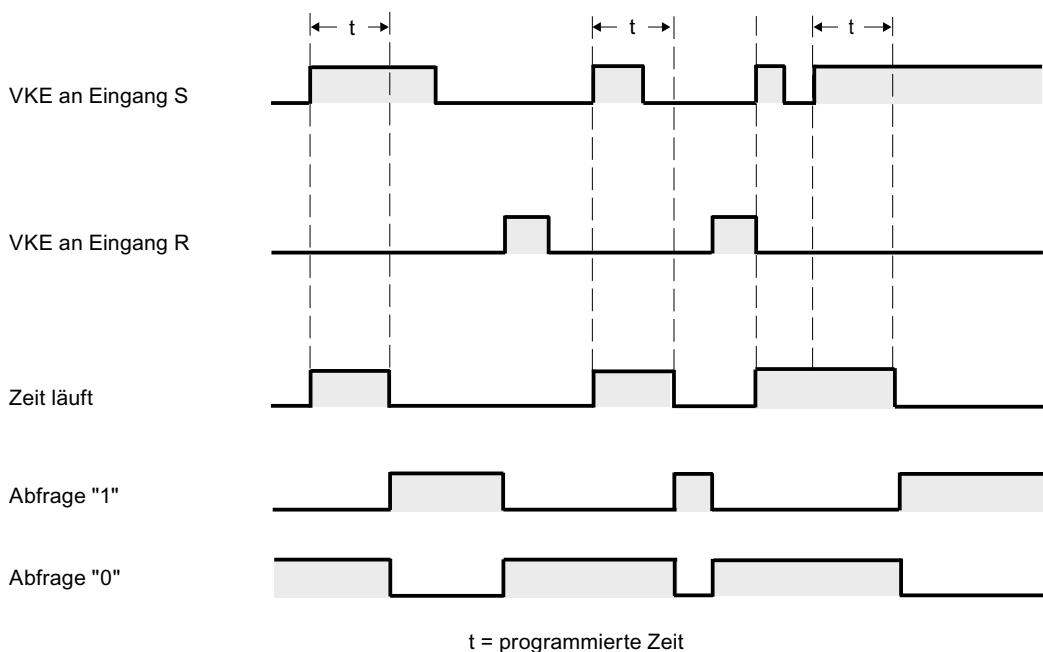
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, T, Z, D, L, P                | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Zeitdauer   |
| R         | Input       | BOOL         | E, A, M, T, Z, D, L, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L, P                      | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

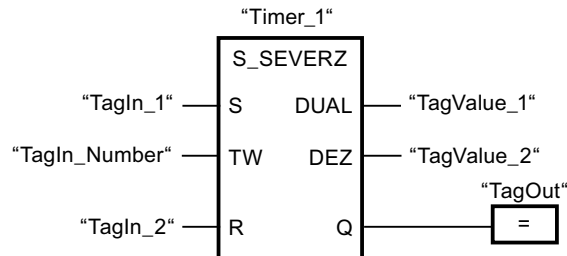
### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, auch wenn der Signalzustand des Operanden "TagIn\_1" auf "0" wechselt. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## S\_AVERZ: Zeit als Ausschaltverzögerung parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Eingang S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer (TW) ab. Solange die Zeit läuft oder der Eingang S den Signalzustand "1" liefert, führt der Ausgang Q den Signalzustand "1". Wenn die Zeit abgelaufen ist und der Signalzustand "0" ist, wird der Ausgang Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1", während die Zeit läuft, wird die Zeit angehalten. Erst wenn eine negative Signalflanke am Eingang S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird dualcodiert am Ausgang DUAL und BCD-codiert am Ausgang DEZ ausgegeben.

Der Signalzustand "1" am Eingang R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Ausgang Q ist in diesem Fall "0".

Die Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" benötigt für die Flankenbewertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

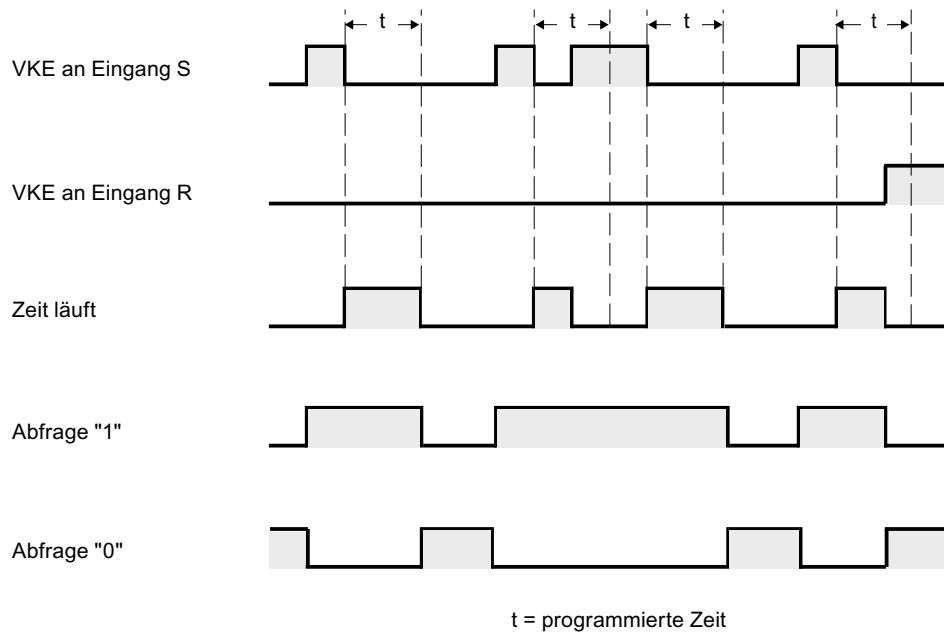
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich                       | Beschreibung  |
|-----------|-------------|--------------|---------------------------------------|---|
| <Zeit>    | InOut/Input | TIMER        | T                                     | Zeit der Anweisung<br>Die Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL         | E, A, M, T, Z, D, L, P                | Starteingang  |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante          | Zeitdauer   |
| R         | Input       | BOOL         | E, A, M, T, Z, D, L, P oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (Dualcodiert)                                      |
| DEZ       | Output      | WORD         | E, A, M, D, L, P                      | Aktueller Zeitwert (BCD-Format)                                       |
| Q         | Output      | BOOL         | E, A, M, D, L, P                      | Status der Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

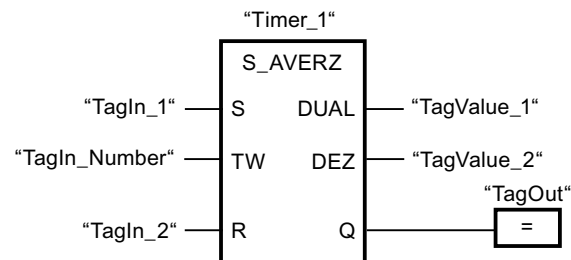
## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "1" auf "0" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Der Operand "TagOut" wird auf "1" gesetzt, wenn die Zeit läuft oder der Operand "TagIn\_1" den Signalzustand "0" liefert. Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", während die Zeit läuft, wird die Zeit zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## SI: Zeit als Impuls starten

### Beschreibung

Mit der Anweisung "Zeit als Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Starteingang erfasst wird. Die Zeit läuft solange mit der angegebenen Zeitdauer, wie das VKE den Signalzustand "1" liefert. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "1" auf "0", bevor der Zeitwert abgelaufen ist, wird die Zeit angehalten. In diesem Fall ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

---

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

### Parameter

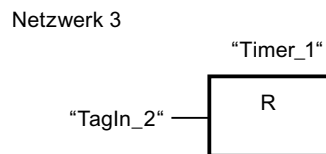
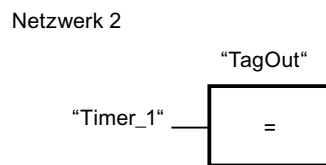
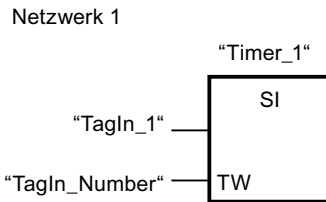
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Impuls starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-----------|-------------|--------------|------------------------------|--|
| <Operand> | Input       | BOOL         | E, A, M, T, Z, D, L, P       | Starteingang   |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer  |
| <Zeit>    | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, solange der Signalzustand des Operanden "TagIn\_1" "1" ist. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", dann wird die Zeit zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## SV: Zeit als verlängerten Impuls starten

### Beschreibung

Mit der Anweisung "Zeit als verlängerten Impuls starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wechselt das VKE von "0" auf "1", während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer neu gestartet. Die Abfrage des Zeitstatus auf "1" ergibt den Signalzustand "0", wenn die Zeit abgelaufen ist.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte

Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als verlängerten Impuls starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als verlängerten Impuls starten":

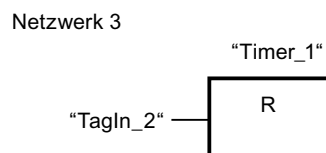
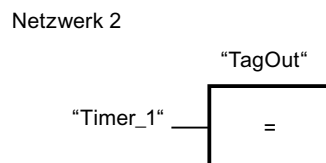
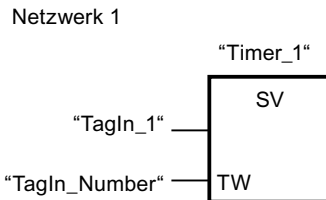
| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-----------|-------------|--------------|------------------------------|--|
| <Operand> | Input       | BOOL         | E, A, M, T, Z, D, L, P       | Starteingang   |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer  |
| <Zeit>    | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab, ohne von einer negativen Flanke im VKE beeinflusst zu werden. Solange die Zeit läuft, liefert der Operand "TagOut" den Signalzustand "1". Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", bevor die Zeit abgelaufen ist, wird die Zeit neu gestartet.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## SE: Zeit als Einschaltverzögerung starten

### Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalflanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, solange das VKE "1" ist. Wenn die Zeit abgelaufen ist und das VKE noch den Signalzustand "1" hat, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn das VKE von "1" auf "0" wechselt, während die Zeit läuft, wird die Zeit angehalten. Die Abfrage des Zeitstatus auf "1" ergibt in diesem Fall den Signalzustand "0".

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Einschaltverzögerung starten" benötigt für die Flankenbewertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

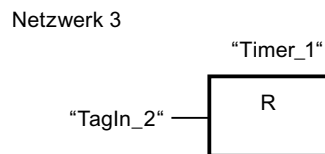
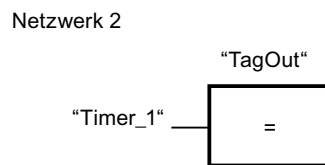
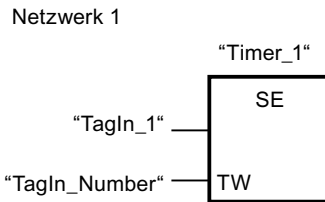
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Einschaltverzögerung starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-----------|-------------|--------------|------------------------------|--|
| <Operand> | Input       | BOOL         | E, A, M, T, Z, D, L, P       | Starteingang   |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer  |
| <Zeit>    | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Wenn die Zeit abgelaufen ist und das VKE den Signalzustand "1" hat, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", bevor die Zeit abgelaufen ist, wird die Zeit angehalten. Wenn der Signalzustand des Operanden "TagIn\_2" "1" ist, dann wird die Zeit "Timer\_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## SS: Zeit als speichernde Einschaltverzögerung starten

### Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab, auch wenn das VKE auf den Signalzustand "0" wechselt. Wenn die Zeit abgelaufen ist, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Nach dem Ablauf der Zeit ist ein Neustart der Zeit erst dann möglich, wenn diese explizit zurückgesetzt wurde.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte

Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als speichernde Einschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

---

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

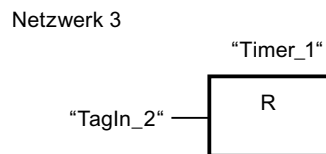
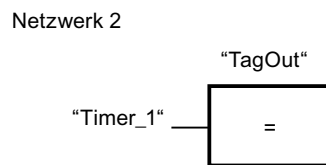
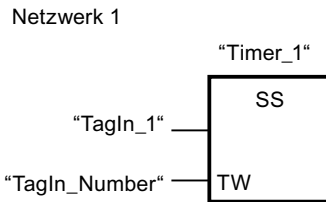
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als speichernde Einschaltverzögerung starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-----------|-------------|--------------|------------------------------|--|
| <Operand> | Input       | BOOL         | E, A, M, T, Z, D, L, P       | Starteingang   |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer  |
| <Zeit>    | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Wenn die Zeit abgelaufen ist, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn\_2" "1" ist, dann wird die Zeit "Timer\_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## SA: Zeit als Ausschaltverzögerung starten

### Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung starten" starten Sie eine programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "1" auf "0" (negative Signalflanke) am Starteingang erfasst wird. Die Zeit läuft mit der angegebenen Zeitdauer ab. Solange die Zeit läuft, ergibt die Abfrage des Zeitstatus auf "1" den Signalzustand "1". Wenn das VKE von "0" auf "1" wechselt, während die Zeit läuft, wird die Zeit zurückgesetzt. Die Zeit wird immer dann neu gestartet, wenn das VKE von "1" auf "0" wechselt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TW programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an.

Die Anweisung "Zeit als Ausschaltverzögerung starten" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Parameter**

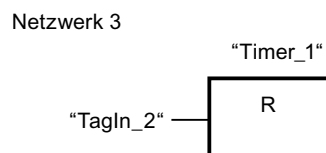
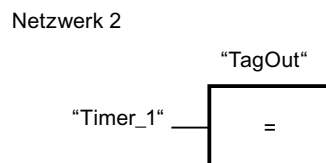
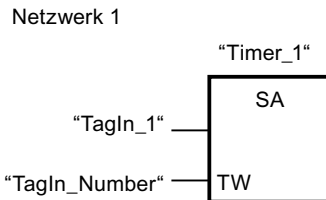
Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit als Ausschaltverzögerung starten":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-----------|-------------|--------------|------------------------------|--|
| <Operand> | Input       | BOOL         | E, A, M, T, Z, D, L, P       | Starteingang   |
| TW        | Input       | S5TIME, WORD | E, A, M, D, L oder Konstante | Zeitdauer  |
| <Zeit>    | InOut/Input | TIMER        | T                            | Zeit, die gestartet wird.<br>Die Anzahl der Zeiten ist von der CPU abhängig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "1" auf "0" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "TagIn\_Number" ab. Solange die Zeit läuft, wird der Operand "TagOut" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "TagIn\_1" von "1" auf "0", während die Zeit läuft, wird die Zeit neu gestartet. Wenn der Signalzustand des Operanden "TagIn\_2" "1" ist, dann wird die Zeit "Timer\_1" zurückgesetzt, d. h. sie wird angehalten und der aktuelle Zeitwert wird auf "0" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Zähler

### IEC-Zähler

### CTU: Vorwärts zählen

## Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Ausgang CV hoch. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins erhöht. Bei der ersten Ausführung der Anweisung ist der aktuelle Zählwert am Ausgang CV auf Null gestellt. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke erhöht, bis er den oberen

Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Eingang CU keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Der Signalzustand am Ausgang Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert am Ausgang CV wird auf "0" zurückgesetzt und in einem Flankenmerker gespeichert, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CU keine Wirkung auf die Anweisung.

**Hinweis**

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

**Für CPU S7-1200**

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTU_SINT / CTU_USINT</li> <li>• CTU_INT / CTU_UINT</li> <li>• CTU_DINT / CTU_UDINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> |

**Für CPU S7-1500**

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTU_SINT / CTU_USINT</li> <li>• CTU_INT / CTU_UINT</li> <li>• CTU_DINT / CTU_UDINT</li> <li>• CTU_LINT / CTU_ULINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:



- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTU oder IEC\_COUNTER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Ausführung der Anweisung "Vorwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

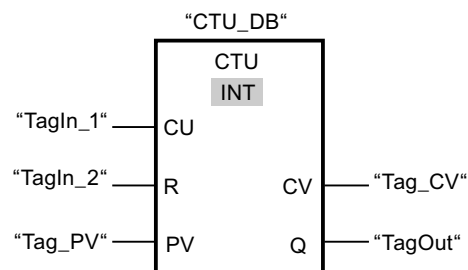
| Parameter | Deklaration | Datentyp               | Speicherbereich                       | Beschreibung                              |
|-----------|-------------|------------------------|---------------------------------------|---|
| CU        | Input       | BOOL                   | E, A, M, D, L oder Konstante          | Zähleingang                               |
| R         | Input       | BOOL                   | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang                           |
| PV        | Input       | Ganzzahlen             | E, A, M, D, L, P oder Konstante       | Wert, bei dem der Ausgang Q gesetzt wird. |
| Q         | Output      | BOOL                   | E, A, M, D, L                         | Zählerstatus                              |
| CV        | Output      | Ganzzahlen, CHAR, DATE | E, A, M, D, L, P                      | Aktueller Zählwert                        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag\_CV"

um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des angegebenen Datentyps (INT = 32767) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "TagOut" übernommen. Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag\_PV" ist. In allen anderen Fällen liefert der Ausgang "TagOut" den Signalzustand "0".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

### CTD: Rückwärts zählen

#### Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Ausgang CV runter. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Ausgang CV um eins verringert. Bei der ersten Ausführung der Anweisung wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Eingang CD keine Wirkung mehr auf die Anweisung.

Am Ausgang Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich "0" ist, wird der Ausgang Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert am Ausgang CV wird auf den Wert des Parameters PV gesetzt und in einem Flankenmerker gespeichert, wenn der Signalzustand am Eingang LD von "0" auf "1" wechselt. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand am Eingang CD keine Wirkung auf die Anweisung.

---

#### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

## Für CPU S7-1200

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTD_SINT / CTD_USINT</li> <li>• CTD_INT / CTD_UINT</li> <li>• CTD_DINT / CTD_UDINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> |

## Für CPU S7-1500

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTD_SINT / CTD_USINT</li> <li>• CTD_INT / CTD_UINT</li> <li>• CTD_DINT / CTD_UDINT</li> <li>• CTD_LINT / CTD_ULINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTD oder IEC\_COUNTER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelnstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Ausführung der Anweisung "Rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

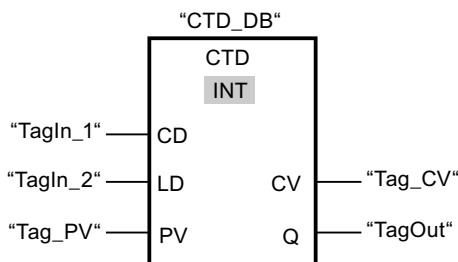
| Parameter | Deklaration | Datentyp               | Speicherbereich                       | Beschreibung                              |
|-----------|-------------|------------------------|---------------------------------------|---|
| CD        | Input       | BOOL                   | E, A, M, D, L oder Konstante          | Zähleingang                               |
| LD        | Input       | BOOL                   | E, A, M, D, L, T, Z, P oder Konstante | Ladeeingang                               |
| PV        | Input       | Ganzzahlen             | E, A, M, D, L, P oder Konstante       | Wert, bei dem der Ausgang Q gesetzt wird. |
| Q         | Output      | BOOL                   | E, A, M, D, L                         | Zählerstatus                              |
| CV        | Output      | Ganzzahlen, CHAR, DATE | E, A, M, D, L, P                      | Aktueller Zählwert                        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" wechselt, wird die Anweisung "Rückwärts zählen" ausgeführt und der Wert am Ausgang "Tag\_CV" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des angegebenen Datentyps (INT = -32768) erreicht ist.

Der Wert am Parameter PV wird als Grenze für die Bestimmung des Ausgangs "TagOut" übernommen. Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich "0" ist. In allen anderen Fällen liefert der Ausgang "TagOut" den Signalzustand "0".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## CTUD: Vorwärts und rückwärts zählen

### Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Ausgang CV hoch und runter. Wenn der Signalzustand am Eingang CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht und am Ausgang CV abgelegt. Wenn der Signalzustand am Eingang CD von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Ausgang CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Ausgang CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Ausgang CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang LD auf "1" wechselt, wird der Zählwert am Ausgang CV auf den Wert des Parameters PV gesetzt und in einem Flankenmerker gespeichert. Solange am Eingang LD der Signalzustand "1" ansteht, hat der Signalzustand an den Eingängen CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf "0" gesetzt und in einem Flankenmerker gespeichert, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Eingängen CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Ausgang QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Ausgang QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QU "0". Am Parameter PV können Sie auch eine Konstante angeben.

Am Ausgang QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Ausgang QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Ausgang QD "0".

---

### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPU S7-1200

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTUD_SINT / CTUD_USINT</li> <li>• CTUD_INT / CTUD_UINT</li> <li>• CTUD_DINT / CTUD_UDINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> |

Für CPU S7-1500

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTUD_SINT / CTUD_USINT</li> <li>• CTUD_INT / CTUD_UINT</li> <li>• CTUD_DINT / CTUD_UDINT</li> <li>• CTUD_LINT / CTUD_ULINT</li> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTUD oder IEC\_COUNTER im Abschnitt "Static" eines Bausteins (z. B. #MyIEC\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Ausführung der Anweisung "Vorwärts und rückwärts zählen" setzt eine Vorverknüpfung voraus. Sie kann innerhalb oder am Ende des Netzwerks angeordnet werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts und rückwärts zählen":

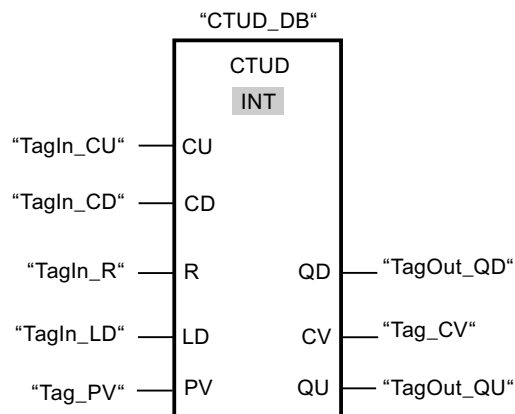
| Parameter | Deklaration | Datentyp               | Speicherbereich                       | Beschreibung                               |
|-----------|-------------|------------------------|---------------------------------------|--|
| CU        | Input       | BOOL                   | E, A, M, D, L oder Konstante          | Vorwärtszähleingang                        |
| CD        | Input       | BOOL                   | E, A, M, D, L oder Konstante          | Rückwärtszähleingang                       |
| R         | Input       | BOOL                   | E, A, M, D, L, T, Z, P oder Konstante | Rücksetzeingang                            |
| LD        | Input       | BOOL                   | E, A, M, D, L, T, Z, P oder Konstante | Ladeeingang                                |
| PV        | Input       | Ganzzahlen             | E, A, M, D, L, P oder Konstante       | Wert, bei dem der Ausgang QU gesetzt wird. |
| QU        | Output      | BOOL                   | E, A, M, D, L                         | Status Vorwärtszähler                      |
| QD        | Output      | BOOL                   | E, A, M, D, L                         | Status Rückwärtszähler                     |
| CV        | Output      | Ganzzahlen, CHAR, DATE | E, A, M, D, L, P                      | Aktueller Zählwert                         |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand am Eingang "TagIn\_CU" oder am Eingang "TagIn\_CD" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Vorwärts und rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn\_CU" vorliegt, wird der aktuelle Zählwert um eins erhöht und am Ausgang "Tag\_CV" abgelegt. Wenn eine positive Signalflanke am Eingang "TagIn\_CD" vorliegt, wird der Zählwert um eins verringert und am Ausgang "Tag\_CV" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Eingang CU so

lange erhöht, bis er den oberen Grenzwert (INT = 32767) erreicht hat. Bei einer positiven Signalflanke am Eingang CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert (INT = -32768) erreicht hat.

Der Ausgang "TagOut\_GU" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert am Eingang "Tag\_PV" ist. In allen anderen Fällen liefert der Ausgang "TagOut\_QU" den Signalzustand "0".

Der Ausgang "TagOut\_QD" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich "0" ist. In allen anderen Fällen führt der Ausgang "TagOut\_QD" den Signalzustand "0".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 1599)

### SIMATIC-Zähler

#### ZAEHLER: Parametrieren und vorwärts/rückwärts zählen

#### Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Wenn in einem Programmzyklus an den Eingängen ZV und ZR eine positive Signalflanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE an den Eingängen ZV und ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV, ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

---

#### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---



Die Anweisung "Parametrieren und vorwärts/rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

## Parameter

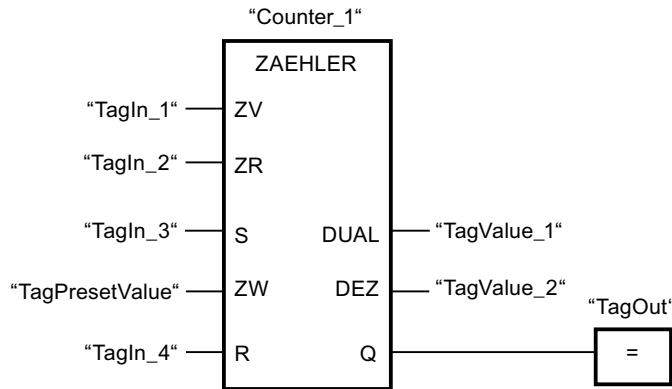
Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts/rückwärts zählen":

| Parameter | Deklaration | Datentyp           | Speicherbereich                    | Beschreibung  |
|-----------|-------------|--------------------|------------------------------------|---|
| <Zähler>  | InOut/Input | COUNTER            | Z                                  | Zähler der Anweisung<br>Die Anzahl der Zähler ist von der CPU abhängig. |
| ZV        | Input       | BOOL               | E, A, M, D, L, T, Z                | Vorwärtszähleingang   |
| ZR        | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Rückwärtszähleingang  |
| S         | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Eingang zum Voreinstellen des Zählers                                   |
| ZW        | Input       | WORD               | E, A, M, D, L, Z oder Konstante    | Voreingestellter Zählwert (C#0 bis C#999)                               |
| R         | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD, S5TIME, DATE | E, A, M, D, L                      | Aktueller Zählwert (Hexadezimal)  |
| DEZ       | Output      | WORD, S5TIME, DATE | E, A, M, D, L                      | Aktueller Zählwert (BCD-Format)   |
| Q         | Output      | BOOL               | E, A, M, D, L                      | Status des Zählers  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand am Eingang "TagIn\_1" oder am Eingang "TagIn\_2" von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung "Parametrieren und vorwärts/rückwärts zählen" ausgeführt. Wenn eine positive Signalflanke am Eingang "TagIn\_1" vorliegt und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalflanke am Eingang "TagIn\_2" vorliegt und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand am Eingang "TagIn\_3" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn\_4" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue\_1" und BCD-codiert im Operanden "TagValue\_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**Z\_VORW: Parametrieren und vorwärts zählen**

**Beschreibung**

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Eingang ZV von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZV "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZV und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

---

#### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

Die Anweisung "Parametrieren und vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

## Parameter

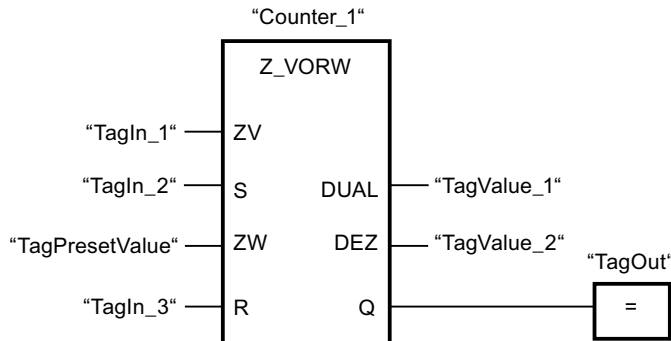
Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und vorwärts zählen":

| Parameter | Deklaration | Datentyp           | Speicherbereich                    | Beschreibung  |
|-----------|-------------|--------------------|------------------------------------|---|
| <Zähler>  | InOut/Input | COUNTER            | Z                                  | Zähler der Anweisung<br>Die Anzahl der Zähler ist von der CPU abhängig. |
| ZV        | Input       | BOOL               | E, A, M, D, L, T, Z                | Vorwärtszähleingang   |
| S         | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Eingang zum Voreinstellen des Zählers                                   |
| ZW        | Input       | WORD               | E, A, M, D, L, Z oder Konstante    | Voreingestellter Zählwert (C#0 bis C#999)                               |
| R         | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD, S5TIME, DATE | E, A, M, D, L                      | Aktueller Zählwert (Hexadezimal)  |
| DEZ       | Output      | WORD, S5TIME, DATE | E, A, M, D, L                      | Aktueller Zählwert (BCD-Format)   |
| Q         | Output      | BOOL               | E, A, M, D, L                      | Status des Zählers  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand am Eingang "TagIn\_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "TagIn\_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn\_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue\_1" und BCD-codiert im Operanden "TagValue\_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**Z\_RUECK: Parametrieren und rückwärts zählen**

**Beschreibung**

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Eingang ZR von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Ausgang DUAL hexadezimal und am Ausgang DEZ BCD-codiert ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Eingang S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters ZW gesetzt. Wird der Zähler gesetzt und das VKE am Eingang ZR "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Eingang R auf "1" wechselt. Solange am Eingang R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Eingängen ZR und S keine Wirkung auf den Zählwert.

Der Signalzustand am Ausgang Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Ausgang Q den Signalzustand "0".

#### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

Die Anweisung "Parametrieren und rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann innerhalb oder am Ende des Netzwerks angeordnet werden.

## Parameter

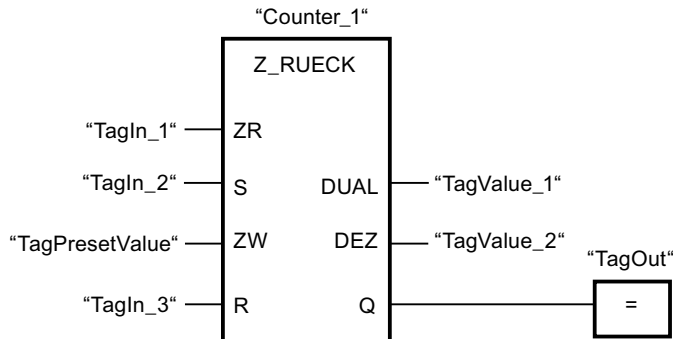
Die folgende Tabelle zeigt die Parameter der Anweisung "Parametrieren und rückwärts zählen":

| Parameter | Deklaration | Datentyp           | Speicherbereich                    | Beschreibung  |
|-----------|-------------|--------------------|------------------------------------|---|
| <Zähler>  | InOut/Input | COUNTER            | Z                                  | Zähler der Anweisung<br>Die Anzahl der Zähler ist von der CPU abhängig. |
| ZR        | Input       | BOOL               | E, A, M, D, L oder Konstante       | Rückwärtszähleingang  |
| S         | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Eingang zum Voreinstellen des Zählers                                   |
| ZW        | Input       | WORD               | E, A, M, D, L, Z oder Konstante    | Voreingestellter Zählwert (C#0 bis C#999)                               |
| R         | Input       | BOOL               | E, A, M, D, L, T, Z oder Konstante | Rücksetzeingang   |
| DUAL      | Output      | WORD, S5TIME, DATE | E, A, M, D, L                      | Aktueller Zählwert (Hexadezimal)  |
| DEZ       | Output      | WORD, S5TIME, DATE | E, A, M, D, L                      | Aktueller Zählwert (BCD-Format)   |
| Q         | Output      | BOOL               | E, A, M, D, L                      | Status des Zählers  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand am Eingang "TagIn\_1" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand am Eingang "TagIn\_2" von "0" auf "1", wird der Zählwert auf den Wert des Operanden "TagPresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "TagIn\_3" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "TagValue\_1" und BCD-codiert im Operanden "TagValue\_2" gespeichert.

Der Ausgang "TagOut" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**SZ: Zähleranfangswert setzen**

**Beschreibung**

Mit der Anweisung "Zähleranfangswert setzen" stellen Sie den Wert eines Zählers ein. Die Anweisung wird ausgeführt, wenn das Verknüpfungsergebnis (VKE) am Starteingang der Anweisung von "0" auf "1" wechselt. Wenn die Anweisung ausgeführt wird, wird der Zähler auf den angegebenen Zählwert gesetzt.

Die Anweisung "Zähleranfangswert setzen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

## Parameter

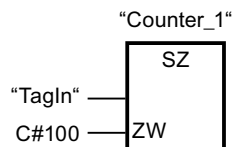
Die folgende Tabelle zeigt die Parameter der Anweisung "Zähleranfangswert setzen":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| <Operand> | Input       | BOOL     | E, A, M, T, Z, D, L          | Starteingang  |
| ZW        | Input       | WORD     | E, A, M, D, L oder Konstante | Wert, mit dem der Zähler im BCD-Format voreingestellt wird. (C#0 bis C#999) |
| <Zähler>  | InOut/Input | COUNTER  | Z                            | Zähler, der voreingestellt wird.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Signalzustand des Operanden "TagIn" von "0" auf "1" wechselt, wird der Zähler "Counter\_1" mit dem Wert "100" voreingestellt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## ZV: Vorwärts zählen

### Beschreibung

Mit der Anweisung "Vorwärts zählen" erhöhen Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke am Starteingang vorliegt. Der Zählwert kann so lange erhöht werden, bis der Grenzwert "999" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr inkrementiert.

Die Anweisung "Vorwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Vorwärts zählen":

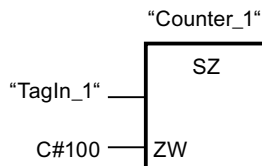
| Parameter | Deklaration | Datentyp | Speicherbereich     | Beschreibung                     |
|-----------|-------------|----------|---------------------|----------------------------------|
| <Operand> | Input       | BOOL     | E, A, M, T, Z, D, L | Starteingang                     |
| <Zähler>  | InOut/Input | COUNTER  | Z                   | Zähler, dessen Wert erhöht wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

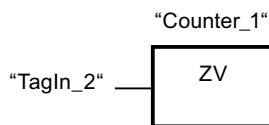
**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

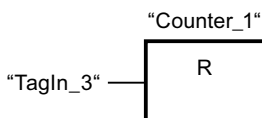
Netzwerk 1



Netzwerk 2



Netzwerk 3



Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter\_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter\_1" wird um eins erhöht, wenn der Signalzustand des Operanden "TagIn\_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn\_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter\_1" auf "0" zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)



## ZR: Rückwärts zählen

### Beschreibung

Mit der Anweisung "Rückwärts zählen" verringern Sie den Wert des angegebenen Zählers um eins, wenn eine positive Signalflanke am Starteingang vorliegt. Der Zählwert kann so lange verringert werden, bis der Grenzwert "0" erreicht ist. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verändert.

Die Anweisung "Rückwärts zählen" benötigt für die Flankenauswertung eine Vorverknüpfung und kann nur am rechten Rand des Netzwerks angeordnet werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rückwärts zählen":

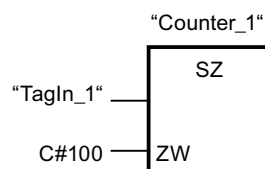
| Parameter | Deklaration | Datentyp | Speicherbereich     | Beschreibung                         |
|-----------|-------------|----------|---------------------|--------------------------------------|
| <Operand> | Input       | BOOL     | E, A, M, T, Z, D, L | Starteingang                         |
| <Zähler>  | InOut/Input | COUNTER  | Z                   | Zähler, dessen Wert verringert wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

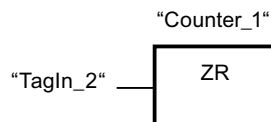
### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

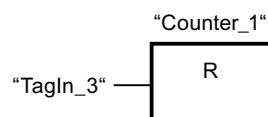
Netzwerk 1



Netzwerk 2



Netzwerk 3



Wechselt der Signalzustand des Operanden "TagIn\_1" von "0" auf "1" (positive Signalflanke), wird der Zähler "Counter\_1" mit dem Wert "100" voreingestellt.

Der Wert des Zählers "Counter\_1" wird um eins verringert, wenn der Signalzustand des Operanden "TagIn\_2" von "0" auf "1" wechselt.

Wenn der Operand "TagIn\_3" den Signalzustand "1" liefert, wird der Wert des Zählers "Counter\_1" auf "0" zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**Vergleicher**

**CMP ==: Gleich**

**Beschreibung**

Mit der Anweisung "Gleich" fragen Sie ab, ob der Wert am Eingang IN1 gleich dem Wert am Eingang IN2 ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| IN1          | IN2         | VKE der Anweisung |
|--------------|-------------|-------------------|
| 'AA'         | 'AA'        | 1                 |
| 'Hallo Welt' | 'HalloWelt' | 0                 |
| 'AA'         | 'aa'        | 0                 |

Die Anweisung "Gleich" vergleicht auch einzelne Zeichen einer Zeichenkette (STRING). Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Bei aktivierter IEC-Prüfung müssen die zu vergleichenden Operanden vom gleichen Datentyp sein. Wenn die IEC-Prüfung nicht aktiviert ist, muss die Breite (Länge) der Operanden gleich sein. Beim Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

---

**Hinweis**

**Vergleich von Gleitpunktzahlen**

Wenn Sie die Datentypen REAL oder LREAL vergleichen wollen, dann verwenden Sie statt der Anweisung "CMP ==: Gleich" die Anweisung "IN\_RANGE: Wert innerhalb Bereich".

---

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gleich":

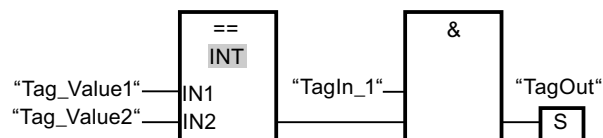
| Parameter | Deklaration | Datentyp  |   | Speicherbereich                    | Beschreibung              |
|-----------|-------------|---|---|------------------------------------|---------------------------|
|           |             | S7-1200   | S7-1500   |                                    |                           |
| IN1       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, DATE,<br>TOD, DTL | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, LTIME,<br>DATE, TOD,<br>LTOD, DTL,<br>DT, LDT | E, A, M, D, L, P<br>oder Konstante | Erster<br>Vergleichswert  |
| IN2       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, DATE,<br>TOD, DTL | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, LTIME,<br>DATE, TOD,<br>LTOD, DTL,<br>DT, LDT | E, A, M, D, L, P<br>oder Konstante | Zweiter<br>Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" = "Tag\_Value2" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## CMP <>: Ungleich

### Beschreibung

Mit der Anweisung "Ungleich" fragen Sie ab, ob der Wert am Eingang IN1 ungleich dem Wert am Eingang IN2 ist.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| IN1          | IN2         | VKE der Anweisung |
|--------------|-------------|-------------------|
| 'AA'         | 'aa'        | 1                 |
| 'Hallo Welt' | 'HalloWelt' | 1                 |
| 'AA'         | 'AA'        | 0                 |

Die Anweisung "Ungleich" vergleicht auch einzelne Zeichen einer Zeichenkette (STRING). Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Bei aktivierter IEC-Prüfung müssen die zu vergleichenden Operanden vom gleichen Datentyp sein. Wenn die IEC-Prüfung nicht aktiviert ist, muss die Breite (Länge) der Operanden gleich sein. Beim Vergleichen von Gleitpunktzahlen müssen die zu vergleichenden Operanden unabhängig von der Einstellung der IEC-Prüfung vom gleichen Datentyp sein.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ungleich":

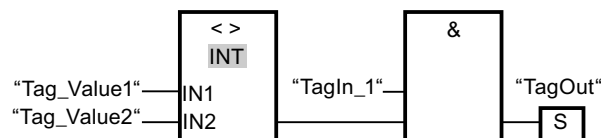
| Parameter | Deklaration | Datentyp  |   | Speicherbereich                    | Beschreibung              |
|-----------|-------------|---|---|------------------------------------|---------------------------|
|           |             | S7-1200   | S7-1500   |                                    |                           |
| IN1       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, DATE,<br>TOD, DTL | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, LTIME,<br>DATE, TOD,<br>LTOD, DTL,<br>DT, LDT | E, A, M, D, L, P<br>oder Konstante | Erster<br>Vergleichswert  |
| IN2       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, DATE,<br>TOD, DTL | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, LTIME,<br>DATE, TOD,<br>LTOD, DTL,<br>DT, LDT | E, A, M, D, L, P<br>oder Konstante | Zweiter<br>Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" <> "Tag\_Value2" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

**CMP >=: Größer gleich**

**Beschreibung**

Mit der Anweisung "Größer gleich" fragen Sie ab, ob der Wert am Eingang IN1 größer oder gleich dem Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| IN1          | IN2          | VKE der Anweisung |
|--------------|--------------|-------------------|
| 'BB'         | 'AA'         | 1                 |
| 'AAA'        | 'AA'         | 1                 |
| 'Hallo Welt' | 'Hallo Welt' | 1                 |
| 'Hallo Welt' | 'HalloWelt'  | 0                 |
| 'AA'         | 'aa'         | 0                 |
| 'AAA'        | 'a'          | 0                 |

Die Anweisung "Größer gleich" vergleicht auch einzelne Zeichen einer Zeichenkette (STRING). Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am Eingang IN1 größer (jünger) oder gleich dem Zeitpunkt am Eingang IN2 ist.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer gleich":

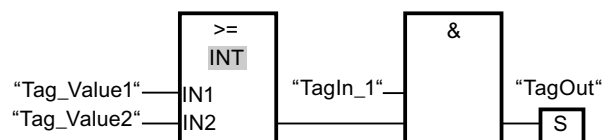
| Parameter | Deklaration | Datentyp  |   | Speicherbereich                 | Beschreibung           |
|-----------|-------------|---|---|---------------------------------|------------------------|
|           |             | S7-1200   | S7-1500   |                                 |                        |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" >= "Tag\_Value2" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 1599)

## CMP <=: Kleiner gleich

### Beschreibung

Mit der Anweisung "Kleiner gleich" fragen Sie ab, ob der Wert am Eingang IN1 kleiner oder gleich dem Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die kürzere Zeichenkette als kleiner.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| IN1          | IN2          | VKE der Anweisung |
|--------------|--------------|-------------------|
| 'AA'         | 'aa'         | 1                 |
| 'AAA'        | 'a'          | 1                 |
| 'Hallo Welt' | 'Hallo Welt' | 1                 |
| 'HalloWelt'  | 'Hallo Welt' | 0                 |
| 'BB'         | 'AA'         | 0                 |
| 'AAA'        | 'AA'         | 0                 |

Die Anweisung "Kleiner gleich" vergleicht auch einzelne Zeichen einer Zeichenkette (STRING). Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am Eingang IN1 kleiner (älter) oder gleich dem Zeitpunkt am Eingang IN2 ist.



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner gleich":

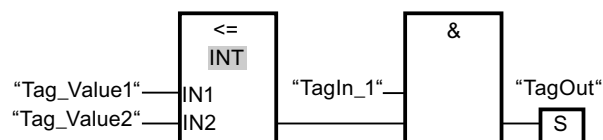
| Parameter | Deklaration | Datentyp  |   | Speicherbereich                    | Beschreibung              |
|-----------|-------------|---|---|------------------------------------|---------------------------|
|           |             | S7-1200   | S7-1500   |                                    |                           |
| IN1       | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, DATE,<br>TOD, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, LTIME,<br>DATE, TOD,<br>LTOD, DTL,<br>DT, LDT | E, A, M, D, L, P<br>oder Konstante | Erster<br>Vergleichswert  |
| IN2       | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, DATE,<br>TOD, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, LTIME,<br>DATE, TOD,<br>LTOD, DTL,<br>DT, LDT | E, A, M, D, L, P<br>oder Konstante | Zweiter<br>Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" <= "Tag\_Value2" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## CMP >: Größer

### Beschreibung

Mit der Anweisung "Größer" fragen Sie ab, ob der Wert am Eingang IN1 größer als der Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die längere Zeichenkette als größer.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| IN1   | IN2  | VKE der Anweisung |
|-------|------|-------------------|
| 'BB'  | 'AA' | 1                 |
| 'AAA' | 'AA' | 1                 |
| 'AA'  | 'aa' | 0                 |
| 'AAA' | 'a'  | 0                 |

Die Anweisung "Größer" vergleicht auch einzelne Zeichen einer Zeichenkette (STRING). Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am Eingang IN1 größer (jünger) als der Zeitpunkt am Eingang IN2 ist.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Größer":

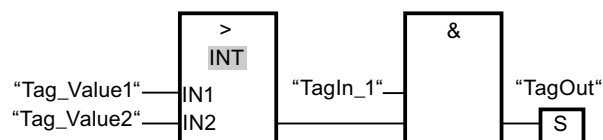
| Parameter | Deklaration | Datentyp  |   | Speicherbereich                 | Beschreibung           |
|-----------|-------------|---|---|---------------------------------|------------------------|
|           |             | S7-1200   | S7-1500   |                                 |                        |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" > "Tag\_Value2" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## CMP <: Kleiner

### Beschreibung

Mit der Anweisung "Kleiner" fragen Sie ab, ob der Wert am Eingang IN1 kleiner als der Wert am Eingang IN2 ist. Beide Vergleichswerte müssen vom gleichen Datentyp sein.

Wenn die Bedingung des Vergleichs erfüllt ist, liefert die Anweisung das Verknüpfungsergebnis (VKE) "1". Bei nicht erfüllter Bedingung des Vergleichs liefert die Anweisung das VKE "0".

Beim Vergleichen von Zeichenketten werden die einzelnen Zeichen über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Der Vergleich wird von links nach rechts durchgeführt. Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Wenn der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette ist, gilt die kürzere Zeichenkette als kleiner.

Die folgende Tabelle zeigt Beispiele für Vergleiche von Zeichenketten:

| <Operand1> | <Operand2> | VKE der Anweisung |
|------------|------------|-------------------|
| 'AA'       | 'aa'       | 1                 |
| 'AAA'      | 'a'        | 1                 |
| 'BB'       | 'AA'       | 0                 |
| 'AAA'      | 'AA'       | 0                 |

Die Anweisung "Kleiner" vergleicht auch einzelne Zeichen einer Zeichenkette (STRING). Die Nummer des zu vergleichenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" verglichen.

Beim Vergleichen von Zeitwerten ist das VKE der Anweisung "1", wenn der Zeitpunkt am Eingang IN1 kleiner (älter) als der Zeitpunkt am Eingang IN2 ist.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Kleiner":

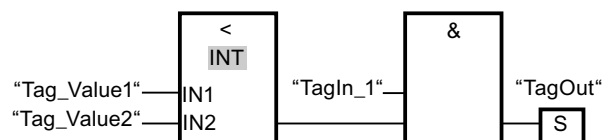
| Parameter | Deklaration | Datentyp  |   | Speicherbereich                 | Beschreibung           |
|-----------|-------------|---|---|---------------------------------|------------------------|
|           |             | S7-1200   | S7-1500   |                                 |                        |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Erster Vergleichswert  |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, DATE, TOD, DTL | Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT | E, A, M, D, L, P oder Konstante | Zweiter Vergleichswert |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Der Operand "TagIn\_1" liefert den Signalzustand "1".
- Die Bedingung der Vergleichsanweisung ist erfüllt, wenn "Tag\_Value1" < "Tag\_Value2" ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Beispiel für das Erfassen des Füllstands eines Lagerbereichs (Seite 1599)

## IN\_RANGE: Wert innerhalb Bereich

### Beschreibung

Mit der Anweisung "Wert innerhalb Bereich" fragen Sie ab, ob der Wert am Eingang VAL innerhalb eines bestimmten Wertebereichs liegt.

Die Grenzen des Wertebereichs legen Sie durch die Eingänge MIN und MAX fest. Die Anweisung "Wert innerhalb Bereich" vergleicht den Wert am Eingang VAL mit den Werten der Eingänge MIN und MAX und führt das Ergebnis auf den Boxausgang. Wenn der Wert am Eingang VAL den Vergleich  $MIN \leq VAL$  oder  $VAL \leq MAX$  erfüllt, liefert der Boxausgang den Signalzustand "1". Bei nicht erfülltem Vergleich steht am Boxausgang der Signalzustand "0".

Für die Bearbeitung der Vergleichsfunktion wird vorausgesetzt, dass die zu vergleichenden Werte vom gleichen Datentyp sind.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert innerhalb Bereich":

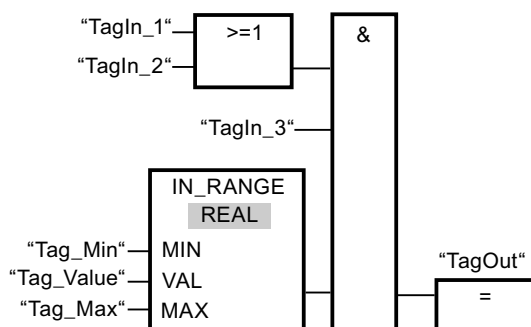
| Parameter  | Deklaration | Datentyp                        | Speicherbereich                 | Beschreibung                       |
|------------|-------------|---------------------------------|---------------------------------|------------------------------------|
| MIN        | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Untere Grenze<br>des Wertebereichs |
| VAL        | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Vergleichswert                     |
| MAX        | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Obere Grenze des<br>Wertebereichs  |
| Boxausgang | Output      | BOOL                            | E, A, M, D, L                   | Ergebnis des<br>Vergleichs         |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" oder "TagIn\_2" liefern den Signalzustand "1".
- Der Operand "TagIn\_3" liefert den Signalzustand "1".
- Der Wert des Operanden "Tag\_Value" liegt innerhalb des Wertebereichs, der durch die aktuellen Werte der Operanden "Tag\_Min" und "Tag\_Max" vorgegeben ist ( $MIN \leq VAL$  oder  $VAL \leq MAX$ ).

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## OUT\_RANGE: Wert außerhalb Bereich

### Beschreibung

Mit der Anweisung "Wert außerhalb Bereich" fragen Sie ab, ob der Wert am Eingang VAL außerhalb eines bestimmten Wertebereichs liegt.

Die Grenzen des Wertebereichs legen Sie durch die Eingänge MIN und MAX fest. Die Anweisung "Wert außerhalb Bereich" vergleicht den Wert am Eingang VAL mit den Werten der Eingänge MIN und MAX und führt das Ergebnis auf den Boxausgang. Wenn der Wert am Eingang VAL den Vergleich  $MIN > VAL$  oder  $VAL > MAX$  erfüllt, liefert der Boxausgang den Signalzustand "1". Der Boxausgang liefert auch den Signalzustand "1", wenn ein angegebener Operand vom Datentyp REAL einen ungültigen Wert aufweist.

Der Boxausgang liefert den Signalzustand "0", wenn der Wert am Eingang VAL die Bedingung  $MIN > VAL$  oder  $VAL > MAX$  nicht erfüllt.

Für die Bearbeitung der Vergleichsfunktion wird vorausgesetzt, dass die zu vergleichenden Werte vom gleichen Datentyp sind.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert außerhalb Bereich":

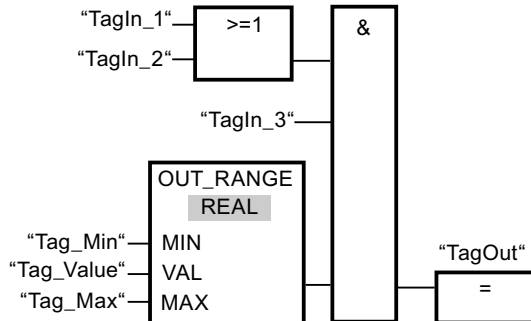
| Parameter  | Deklaration | Datentyp                        | Speicherbereich                 | Beschreibung                       |
|------------|-------------|---------------------------------|---------------------------------|------------------------------------|
| MIN        | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Untere Grenze<br>des Wertebereichs |
| VAL        | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Vergleichswert                     |
| MAX        | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Obere Grenze des<br>Wertebereichs  |
| Boxausgang | Output      | BOOL                            | E, A, M, D, L                   | Ergebnis des<br>Vergleichs         |

Aus der Klappliste "<???"> der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Der Ausgang "TagOut" wird gesetzt, wenn die folgenden Bedingungen erfüllt sind:

- Die Operanden "TagIn\_1" und "TagIn\_2" liefern den Signalzustand "1".
- Der Operand "TagIn\_3" liefert den Signalzustand "1".
- Der Wert des Operanden "Tag\_Value" liegt außerhalb des Wertebereichs, der durch die Werte der Operanden "Tag\_Min" und "Tag\_Max" vorgegeben ist (MIN > VAL oder VAL > MAX).

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**OK: Gültigkeit prüfen**

**Beschreibung**

Mit der Anweisung "Gültigkeit prüfen" fragen Sie ab, ob der Wert eines Operanden (<Operand>) eine gültige Gleitpunktzahl ist. Die Prüfung wird in jedem Programmzyklus durchgeführt. Wenn der Operandenwert zum Zeitpunkt der Abfrage eine gültige Gleitpunktzahl ist, liefert der Boxausgang den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "Gültigkeit prüfen" "0".

Die Anweisung "Gültigkeit prüfen" können Sie in Verbindung mit dem EN-Mechanismus nutzen. Wenn Sie die Anweisungsbox an einen Freigabeeingang EN anschließen, wird der Freigabeeingang nur bei einer positiven Abfrage der Wertgültigkeit gesetzt. Durch diese Funktionalität können Sie sicherstellen, dass eine Anweisung nur dann aktiviert wird, wenn der Wert des angegebenen Operanden eine gültige Gleitpunktzahl ist.



## Parameter

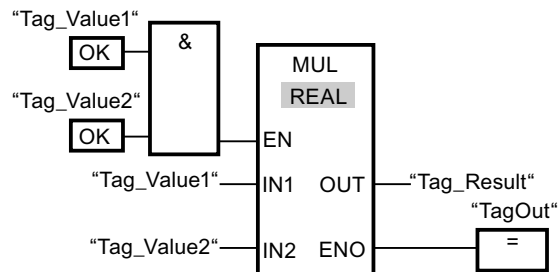
Das folgende Beispiel zeigt die Funktionsweise der Anweisung "Gültigkeit prüfen":

| Parameter | Deklaration | Datentyp         | Speicherbereich | Beschreibung            |
|-----------|-------------|------------------|-----------------|-------------------------|
| <Operand> | Input       | Gleitpunktzahlen | E, A, M, D, L   | Wert, der geprüft wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Werte der Operanden "Tag\_Value1" und "Tag\_Value2" gültige Gleitpunktzahlen aufweisen, wird die Anweisung "Multiplizieren" (MUL) aktiviert und der Freigabeausgang ENO gesetzt. Bei der Bearbeitung der Anweisung "Multiplizieren" (MUL) wird der Wert des Operanden "Tag\_Value1" mit dem Wert des Operanden "Tag\_Value2" multipliziert. Das Multiplikationsergebnis wird dann in den Operanden "Tag\_Result" gespeichert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden die Ausgänge ENO und "TagOut" auf den Signalzustand "1" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## NOT\_OK: Ungültigkeit prüfen

### Beschreibung

Mit der Anweisung "Ungültigkeit prüfen" fragen Sie ab, ob der Wert eines Operanden (<Operand>) eine ungültige Gleitpunktzahl ist. Die Prüfung wird in jedem Programmzyklus durchgeführt. Wenn der Operandenwert zum Zeitpunkt der Abfrage eine ungültige Gleitpunktzahl ist, liefert der Boxausgang den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Boxausgang "0".

### Parameter

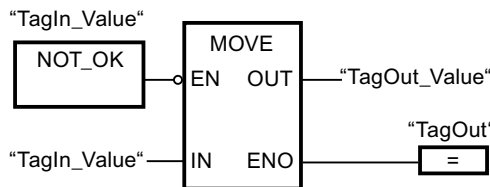
Die folgende Tabelle zeigt die Parameter der Anweisung "Ungültigkeit prüfen":

| Parameter | Deklaration | Datentyp         | Speicherbereich | Beschreibung            |
|-----------|-------------|------------------|-----------------|-------------------------|
| <Operand> | Input       | Gleitpunktzahlen | E, A, M, D, L   | Wert, der geprüft wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Wert des Operanden "TagIn\_Value" eine ungültige Gleitpunktzahl ist, wird die Anweisung "Wert kopieren" (MOVE) nicht ausgeführt. Der Ausgang "TagOut" wird auf den Signalzustand "0" zurückgesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## Mathematische Funktionen

### CALCULATE: Berechnen

#### Beschreibung

Mit der Anweisung "Berechnen" definieren und führen Sie, abhängig vom gewählten Datentyp, einen Ausdruck (Formel) zum Berechnen mathematischer Operationen oder komplexer Verknüpfungen aus.

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung auswählen. Abhängig vom ausgewählten Datentyp können Sie die Funktionalität bestimmter Anweisungen kombinieren, um eine komplexe Berechnung auszuführen. Die Angabe des zu berechnenden Ausdrucks erfolgt über einen Dialog, den Sie über das Symbol "Taschenrechner" am rechten oberen Rand der Anweisungsbox öffnen können. Im Ausdruck können Namen der Eingangsparameter und die Syntax der Anweisungen enthalten sein. Die Angabe von Operandennamen oder -adressen ist nicht zulässig.

Die folgende Tabelle zeigt die Anweisungen, die, abhängig vom gewählten Datentyp, im Ausdruck der Anweisung "Berechnen" kombiniert werden, ausgeführt werden können:

| Datentyp   | Anweisung  | Syntax   | Beispiel   |
|--|--|----------|--|
| Bitfolgen  | AND: UND verknüpfen  | AND      | IN1 AND IN2 OR IN3   |
|  | OR: ODER verknüpfen  | OR       |  |
|  | XOR: EXCLUSIV ODER verknüpfen                              | XOR      |  |
|  | INV: Einerkomplement erzeugen                              | NOT      |  |
|  | SWAP: Anordnung ändern <sup>1)</sup>                       | SWAP     |  |
| Ganzzahlen   | ADD: Addieren  | +        | (IN1 + IN2) * IN3;<br>(ABS(IN2))*(ABS(IN1))  |
|  | SUB: Subtrahieren  | -        |  |
|  | MUL: Multiplizieren  | *        |  |
|  | DIV: Dividieren  | /        |  |
|  | MOD: Divisionsrest gewinnen                                | MOD      |  |
|  | INV: Einerkomplement erzeugen                              | NOT      |  |
|  | NEG: Zweierkomplement erzeugen                             | -(in1)   |  |
|  | ABS: Absolutwert bilden                                    | ABS( )   |  |
| Gleitpunktzahlen                                   | ADD: Addieren  | +        | ((SIN(IN2)*SIN(IN2)+<br>SIN(IN3)*SIN(IN3))/<br>IN3;<br>(SQR(SIN(IN2))+<br>SQR(COS(IN3)))/IN2 |
|  | SUB: Subtrahieren  | -        |  |
|  | MUL: Multiplizieren  | *        |  |
|  | DIV: Dividieren  | /        |  |
|  | EXPT: Potenzieren  | **       |  |
|  | ABS: Absolutwert bilden                                    | ABS( )   |  |
|  | SQR: Quadrat bilden  | SQR( )   |  |
|  | SQRT: Quadratwurzel bilden                                 | SQRT( )  |  |
|  | LN: Natürlichen Logarithmus bilden                         | LN( )    |  |
|  | EXP: Exponentialwert bilden                                | EXP( )   |  |
|  | FRAC: Nachkommastellen ermitteln                           | FRAC( )  |  |
|  | SIN: Sinuswert bilden                                      | SIN( )   |  |
|  | COS: Cosinuswert bilden                                    | COS( )   |  |
|  | TAN: Tangenswert bilden                                    | TAN( )   |  |
|  | ASIN: Arcussinuswert bilden                                | ASIN( )  |  |
|  | ACOS: Arcuscosinuswert bilden                              | ACOS( )  |  |
|  | ATAN: Arcustangenswert bilden                              | ATAN( )  |  |
|  | NEG: Zweierkomplement erzeugen                             | -(in1)   |  |
|  | TRUNC: Ganzzahl erzeugen                                   | TRUNC( ) |  |
|  | ROUND: Zahl runden   | ROUND( ) |  |
|  | CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen   | CEIL( )  |  |
|  | FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen | FLOOR( ) |  |
| <sup>1)</sup> Nicht für den Datentyp BYTE möglich. |  |          |  |

Die Anweisungsbox enthält im Grundzustand mindestens 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert.

Die Werte der Eingänge werden zur Ausführung des angegebenen Ausdrucks verwendet. Im Ausdruck müssen dabei nicht alle definierten Eingänge verwendet werden. Das Ergebnis der Anweisung wird an den Boxausgang OUT übertragen.

Wenn Sie im Ausdruck Eingänge verwenden, die in der Box nicht verfügbar sind, werden diese automatisch eingefügt. Voraussetzung dafür ist, dass die Nummerierung der neu zu definierenden Eingänge im Ausdruck lückenlos ist. Sie können zum Beispiel den Eingang IN4 im Ausdruck nicht verwenden, wenn der Eingang IN3 nicht definiert ist.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis oder ein Zwischenergebnis der Anweisung "Berechnen" liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.
- Bei der Ausführung einer der im Ausdruck angegebenen Anweisungen ist ein Fehler aufgetreten.

## Parameter

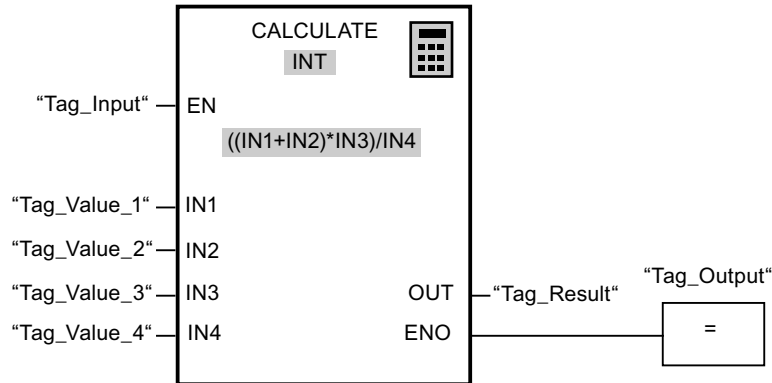
Die folgende Tabelle zeigt die Parameter der Anweisung "Berechnen":

| Parameter | Deklaration | Datentyp                                | Speicherbereich                 |                                 | Beschreibung                                     |
|-----------|-------------|---|---------------------------------|---------------------------------|--|
|           |             |   | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL                                    | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                                  |
| ENO       | Output      | BOOL                                    | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                                  |
| IN1       | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Erster verfügbarer Eingang                       |
| IN2       | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter verfügbarer Eingang                      |
| INn       | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zusätzlich eingefügte Eingänge                   |
| OUT       | Output      | Bitfolgen, Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Ausgang, an den das Endergebnis übertragen wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand     | Wert |
|-----------|-------------|------|
| IN1       | Tag_Value_1 | 4    |
| IN2       | Tag_Value_2 | 4    |
| IN3       | Tag_Value_3 | 3    |
| IN4       | Tag_Value_4 | 2    |
| OUT       | Tag_Result  | 12   |

Wenn der Eingang "Tag\_Input" den Signalzustand "1" liefert, wird die Anweisung "Berechnen" ausgeführt. Der Wert des Operanden "Tag\_Value\_1" wird mit dem Wert des Operanden "Tag\_Value\_2" addiert. Die Summe wird mit dem Wert des Operanden "Tag\_Value\_3" multipliziert. Das Produkt wird durch den Wert des Operanden "Tag\_Value\_4" dividiert. Der Quotient wird als Endergebnis in den Operanden "Tag\_Result" am Ausgang OUT der Anweisung übertragen. Wenn keine Fehler während der Bearbeitung der einzelnen Anweisungen auftreten, werden der Freigabeausgang ENO und der Operand "Tag\_Output" auf "1" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ADD: Addieren

### Beschreibung

Mit der Anweisung "Addieren" addieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragen die Summe am Ausgang OUT ab (OUT := IN1+IN2).

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter addiert. Die Summe wird am Ausgang OUT gespeichert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Addieren":

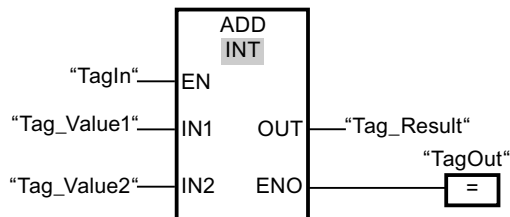
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 |                                 | Beschreibung                                 |
|-----------|-------------|------------------------------|---------------------------------|---------------------------------|--|
|           |             |                              | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL                         | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                              |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                              |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Erster Summand                               |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter Summand                              |
| INn       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Optionale Eingangswerte, die addiert werden. |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Summe  |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weiter Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch":

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Addieren" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" addiert. Das Ergebnis der Addition wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Auswahl eines Datentyps (Seite 1336)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

## SUB: Subtrahieren

### Beschreibung

Mit der Anweisung "Subtrahieren" subtrahieren Sie den Wert am Eingang IN2 vom Wert am Eingang IN1 und fragen die Differenz am Ausgang OUT ab ( $OUT := IN1 - IN2$ ).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Subtrahieren":

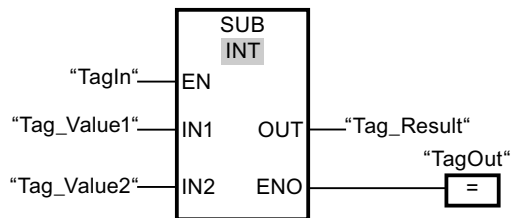
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 |                                 | Beschreibung    |
|-----------|-------------|------------------------------|---------------------------------|---------------------------------|-----------------|
|           |             |                              | S7-1200                         | S7-1500                         |                 |
| EN        | Input       | BOOL                         | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Minuend         |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Subtrahend      |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Differenz       |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Subtrahieren" ausgeführt. Der Wert des Operanden "Tag\_Value2" wird vom Wert des Operanden "Tag\_Value1" subtrahiert. Das Ergebnis der Subtraktion wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Auswahl eines Datentyps (Seite 1336)



## MUL: Multiplizieren

### Beschreibung

Mit der Anweisung "Multiplizieren" multiplizieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 und fragen das Produkt am Ausgang OUT ab ( $OUT := IN1 * IN2$ ).

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter multipliziert. Das Produkt wird am Ausgang OUT gespeichert.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplizieren":

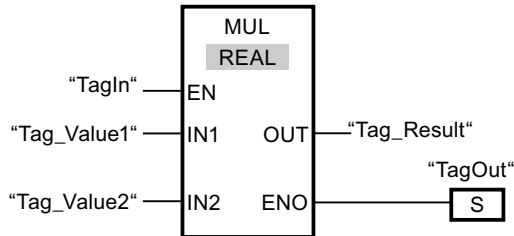
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 |                                 | Beschreibung                                       |
|-----------|-------------|------------------------------|---------------------------------|---------------------------------|--|
|           |             |                              | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL                         | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                                    |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                                    |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Erster Wert der Multiplikation                     |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter Wert der Multiplikation                    |
| INn       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Optionale Eingangswerte, die multipliziert werden. |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Produkt  |

Aus der Klappliste "<???">" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Multiplizieren" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" multipliziert. Das Ergebnis der Multiplikation wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Auswahl eines Datentyps (Seite 1336)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

### DIV: Dividieren

#### Beschreibung

Mit der Anweisung "Dividieren" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Quotienten am Ausgang OUT ab ( $OUT := IN1/IN2$ ).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dividieren":

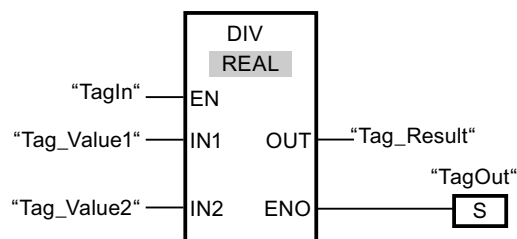
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 |                                 | Beschreibung        |
|-----------|-------------|------------------------------|---------------------------------|---------------------------------|---------------------|
|           |             |                              | S7-1200                         | S7-1500                         |                     |
| EN        | Input       | BOOL                         | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang     |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang     |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Dividend            |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Divisor             |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Wert des Quotienten |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Dividieren" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird durch den Wert des Operanden "Tag\_Value2" dividiert. Das Ergebnis der Division wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Auswahl eines Datentyps (Seite 1336)

## MOD: Divisionsrest gewinnen

### Beschreibung

Mit der Anweisung "Divisionsrest gewinnen" dividieren Sie den Wert am Eingang IN1 durch den Wert am Eingang IN2 und fragen den Divisionsrest am Ausgang OUT ab.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Divisionsrest gewinnen":

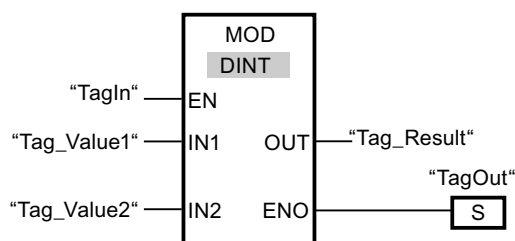
| Parameter | Deklaration | Datentyp   | Speicherbereich                 |                                 | Beschreibung    |
|-----------|-------------|------------|---------------------------------|---------------------------------|-----------------|
|           |             |            | S7-1200                         | S7-1500                         |                 |
| EN        | Input       | BOOL       | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang |
| ENO       | Output      | BOOL       | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang |
| IN1       | Input       | Ganzzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Dividend        |
| IN2       | Input       | Ganzzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Divisor         |
| OUT       | Output      | Ganzzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Divisionsrest   |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Divisionsrest gewinnen" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird durch den Wert des Operanden "Tag\_Value2" dividiert. Der Divisionsrest wird im Operanden "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Auswahl eines Datentyps (Seite 1336)

**NEG: Zweierkomplement erzeugen****Beschreibung**

Mit der Anweisung "Zweierkomplement erzeugen" wechseln Sie das Vorzeichen des Werts am Eingang IN und fragen das Ergebnis am Ausgang OUT ab. Wenn z. B. am Eingang IN ein positiver Wert ansteht, wird am Ausgang OUT das negative Äquivalent dieses Wertes ausgegeben.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Das Ergebnis der Anweisung liegt außerhalb des Bereichs, der für den am Ausgang OUT angegebenen Datentyp zulässig ist.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zweierkomplement erzeugen":

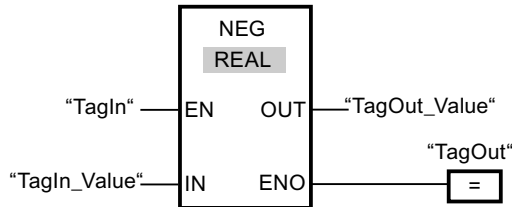
| Parameter | Deklaration | Datentyp                          |   | Speicherbereich                 |                                 | Beschreibung                       |
|-----------|-------------|-----------------------------------|---|---------------------------------|---------------------------------|------------------------------------|
|           |             | S7-1200                           | S7-1500                                 | S7-1200                         | S7-1500                         |                                    |
| EN        | Input       | BOOL                              | BOOL                                    | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                    |
| ENO       | Output      | BOOL                              | BOOL                                    | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                    |
| IN        | Input       | SINT, INT, DINT, Gleitpunktzahlen | SINT, INT, DINT, LINT, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert                       |
| OUT       | Output      | SINT, INT, DINT, Gleitpunktzahlen | SINT, INT, DINT, LINT, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Zweierkomplement des Eingangswerts |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Zweierkomplement erzeugen" ausgeführt. Das Vorzeichen des Werts am Eingang "TagIn\_Value" wird gewechselt und das Ergebnis wird am Ausgang "TagOut\_Value" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

### INC: Inkrementieren

#### Beschreibung

Mit der Anweisung "Inkrementieren" ändern Sie den Wert des Operanden am Parameter IN/ OUT auf den nächst höheren Wert und fragen das Ergebnis ab.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Inkrementieren":

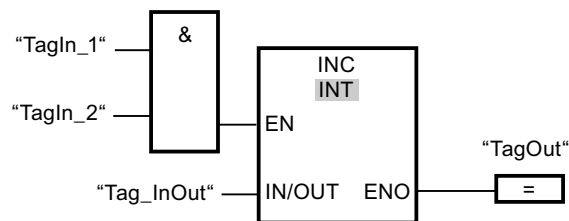
| Parameter | Deklaration | Datentyp   | Speicherbereich |                     | Beschreibung                  |
|-----------|-------------|------------|-----------------|---------------------|-------------------------------|
|           |             |            | S7-1200         | S7-1500             |                               |
| EN        | Input       | BOOL       | E, A, M, D, L   | E, A, M, D, L, T, Z | Freigabeeingang               |
| ENO       | Output      | BOOL       | E, A, M, D, L   | E, A, M, D, L       | Freigabeausgang               |
| IN/OUT    | InOut       | Ganzzahlen | E, A, M, D, L   | E, A, M, D, L       | Wert, der inkrementiert wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird der Wert des Operanden "Tag\_InOut" um eins erhöht und der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

## DEC: Dekrementieren

### Beschreibung

Mit der Anweisung "Dekrementieren" ändern Sie den Wert des Operanden am Parameter IN/OUT auf den nächst niedrigeren Wert und fragen das Ergebnis ab.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Dekrementieren":

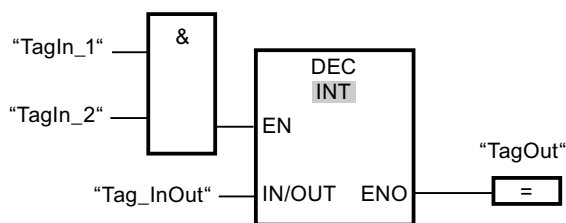
| Parameter | Deklaration | Datentyp   | Speicherbereich |                     | Beschreibung                  |
|-----------|-------------|------------|-----------------|---------------------|-------------------------------|
|           |             |            | S7-1200         | S7-1500             |                               |
| EN        | Input       | BOOL       | E, A, M, D, L   | E, A, M, D, L, T, Z | Freigabeeingang               |
| ENO       | Output      | BOOL       | E, A, M, D, L   | E, A, M, D, L       | Freigabeausgang               |
| IN/OUT    | InOut       | Ganzzahlen | E, A, M, D, L   | E, A, M, D, L       | Wert, der dekrementiert wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird der Wert des Operanden "Tag\_InOut" um eins verringert und der Ausgang "TagOut" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)



## ABS: Absolutwert bilden

### Beschreibung

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag des Wertes, der am Eingang IN angegeben ist. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Absolutwert bilden":

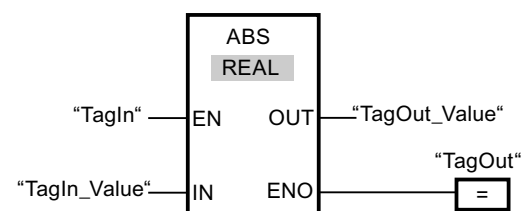
| Parameter | Deklaration | Datentyp                          |   | Speicherbereich                 |                                 | Beschreibung                  |
|-----------|-------------|-----------------------------------|---|---------------------------------|---------------------------------|-------------------------------|
|           |             | S7-1200                           | S7-1500                                 | S7-1200                         | S7-1500                         |                               |
| EN        | Input       | BOOL                              | BOOL                                    | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang               |
| ENO       | Output      | BOOL                              | BOOL                                    | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang               |
| IN        | Input       | SINT, INT, DINT, Gleitpunktzahlen | SINT, INT, DINT, LINT, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert                  |
| OUT       | Output      | SINT, INT, DINT, Gleitpunktzahlen | SINT, INT, DINT, LINT, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Absolutwert des Eingangswerts |

Aus der Klappliste "<???"> der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert   |
|-----------|--------------|--------|
| IN        | TagIn_Value  | -6.234 |
| OUT       | TagOut_Value | 6.234  |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Absolutwert bilden" ausgeführt. Die Anweisung berechnet den Absolutbetrag des Wertes am Eingang "TagIn\_Value" und gibt das Ergebnis am Ausgang "TagOut\_Value" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

**MIN: Minimum ermitteln**

**Beschreibung**

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und schreiben den kleinsten Wert in den Ausgang OUT. Die Anzahl der Eingänge ist an der Anweisungsbox durch zusätzliche Eingänge erweiterbar. Die Eingänge werden an der Box aufsteigend nummeriert.

Im Grundzustand enthält die Anweisung min. zwei Eingänge (IN1 und IN2) bis max. 100 Eingänge.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

| Parameter | Deklaration | Datentyp |         | Speicherbereich |                     | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|---------------------|-----------------|
|           |             | S7-1200  | S7-1500 | S7-1200         | S7-1500             |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L, T, Z | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L       | Freigabeausgang |

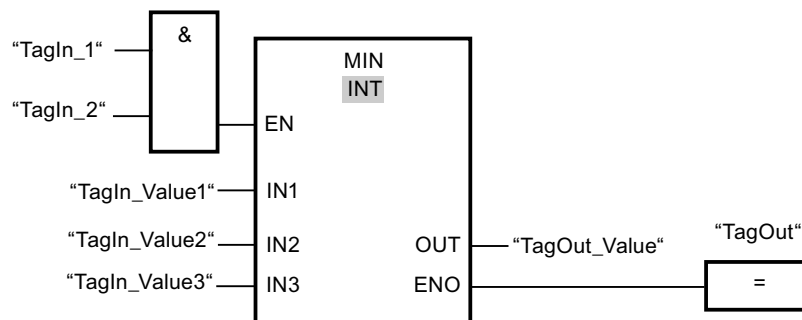
| Parameter | Deklaration | Datentyp                                      |   | Speicherbereich                 |                                 | Beschreibung  |
|-----------|-------------|---|---|---------------------------------|---------------------------------|---|
|           |             | S7-1200                                       | S7-1500   | S7-1200                         | S7-1500                         |   |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Erster Eingangswert   |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter Eingangswert  |
| INn       | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zusätzlich eingefügte Eingänge, deren Werte verglichen werden |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis  |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert  |
|-----------|--------------|-------|
| IN1       | TagIn_Value1 | 12222 |
| IN2       | TagIn_Value2 | 14444 |
| IN3       | TagIn_Value3 | 13333 |
| OUT       | TagOut_Value | 12222 |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" haben, wird die Anweisung "Minimum ermitteln" ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den kleinsten Wert ("TagIn\_Value1") in den Ausgang "TagOut\_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

### MAX: Maximum ermitteln

#### Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und schreiben den größten Wert in den Ausgang OUT. Die Anzahl der Eingänge ist an der Anweisungsbox durch zusätzliche Eingänge erweiterbar. Die Eingänge werden an der Box aufsteigend nummeriert.

Im Grundzustand enthält die Anweisung min. zwei Eingänge (IN1 und IN2) bis max. 100 Eingänge.

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

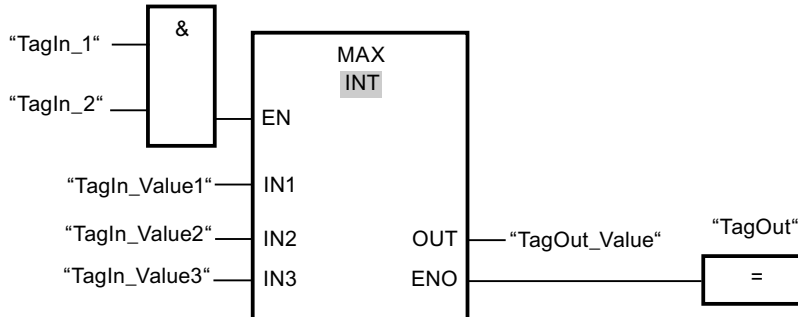
| Parameter | Deklaration | Datentyp                                      |   | Speicherbereich                 |                                 | Beschreibung   |
|-----------|-------------|---|---|---------------------------------|---------------------------------|--|
|           |             | S7-1200                                       | S7-1500   | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL  | BOOL  | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeingang   |
| ENO       | Output      | BOOL  | BOOL  | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang  |
| IN1       | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Erster Eingangswert  |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter Eingangswert   |
| INn       | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zusätzlich eingefügte Eingänge, deren Werte verglichen werden. |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis   |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert  |
|-----------|--------------|-------|
| IN1       | TagIn_Value1 | 12222 |
| IN2       | TagIn_Value2 | 14444 |
| IN3       | TagIn_Value3 | 13333 |
| OUT       | TagOut_Value | 14444 |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" haben, wird die Anweisung "Maximum ermitteln" ausgeführt. Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("TagIn\_Value2") in den Ausgang "TagOut\_Value". Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

**LIMIT: Limitieren**

**Beschreibung**

Mit der Anweisung "Limitieren" begrenzen Sie den Wert am Eingang IN auf die Werte an den Eingängen MN und MX. Wenn der Wert am Eingang IN die Bedingung  $MN \leq IN \leq MX$  erfüllt, wird er in den Ausgang OUT kopiert. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Ausgang OUT auf den Wert des Eingangs MN gesetzt. Bei einer Überschreitung der Obergrenze MX wird der Ausgang OUT auf den Wert des Eingangs MX gesetzt.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis undefiniert und der Freigabeausgang ENO ist "0".

Der Freigabeausgang ENO hat den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN hat den Signalzustand "0".
- Die angegebenen Variablen sind nicht vom gleichen Datentyp.
- Ein Operand weist einen ungültigen Wert auf.
- Der Wert am Eingang MN ist größer als der Wert am Eingang MX.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Limitieren":

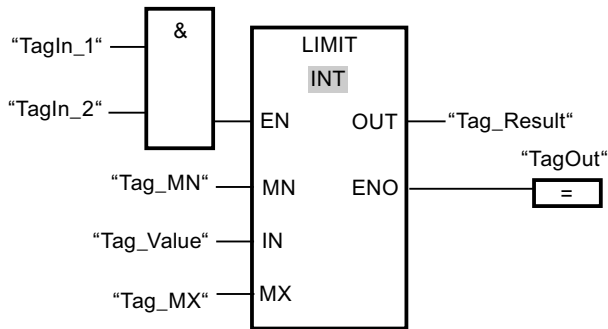
| Parameter | Deklaration | Datentyp                                      |   | Speicherbereich                 |                                 | Beschreibung    |
|-----------|-------------|---|---|---------------------------------|---------------------------------|-----------------|
|           |             | S7-1200                                       | S7-1500   | S7-1200                         | S7-1500                         |                 |
| EN        | Input       | BOOL  | BOOL  | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang |
| ENO       | Output      | BOOL  | BOOL  | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang |
| MN        | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Untergrenze     |
| IN        | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert    |
| MX        | Input       | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Obergrenze      |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen, TIME, TOD, DATE | Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|-------|
| MN        | Tag_MN     | 12000 |
| IN        | Tag_Value  | 8000  |
| MX        | Tag_MX     | 16000 |
| OUT       | Tag_Result | 12000 |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" haben, wird die Anweisung "Limitieren" ausgeführt. Der Wert des Operanden "Tag\_Value" wird mit den Werten der Operanden "Tag\_MN" und "Tag\_MX" verglichen. Da der Wert des Operanden "Tag\_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag\_MN" in den Ausgang "Tag\_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

**SQR: Quadrat bilden**

**Beschreibung**

Mit der Anweisung "Quadrat bilden" quadrieren Sie den am Eingang IN stehenden Wert und fragen das Ergebnis am Ausgang OUT ab.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft.



- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadrat bilden":

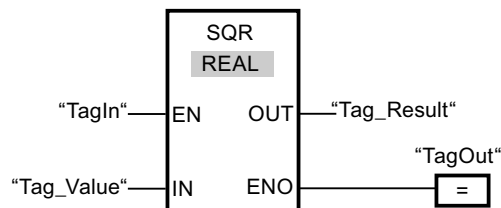
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung               |
|-----------|-------------|------------------|---------------------------------|---------------------------------|----------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                            |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang            |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang            |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert               |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Quadrat des Eingangswertes |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert |
|-----------|------------|------|
| IN        | Tag_Value  | 5.0  |
| OUT       | Tag_Result | 25.0 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Quadrat bilden" ausgeführt. Die Anweisung quadriert den Wert des Operanden "Tag\_Value" und gibt das Ergebnis am Ausgang "Tag\_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**SQRT: Quadratwurzel bilden**

**Beschreibung**

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem am Eingang IN stehenden Wert die Quadratwurzel und fragen das Ergebnis am Ausgang OUT ab. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück. Wenn der Wert am Eingang IN "0" ist, ist das Ergebnis auch "0".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Quadratwurzel bilden":

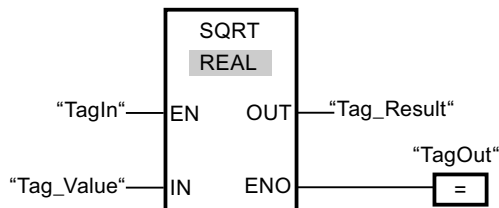
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                    |
|-----------|-------------|------------------|---------------------------------|---------------------------------|---------------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                                 |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                 |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                 |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert                    |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L                   | E, A, M, D, L                   | Quadratwurzel des Eingangswerts |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert |
|-----------|------------|------|
| IN        | Tag_Value  | 25.0 |
| OUT       | Tag_Result | 5.0  |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Quadratwurzel bilden" ausgeführt. Die Anweisung zieht aus dem Wert des Operanden "Tag\_Value" die Quadratwurzel und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## LN: Natürlichen Logarithmus bilden

### Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem am Eingang IN stehenden Wert den natürlichen Logarithmus zur Basis  $e$  ( $e = 2.718282$ ). Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert der Ausgang OUT eine ungültige Gleitpunktzahl zurück.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN ist negativ.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Natürlichen Logarithmus bilden":

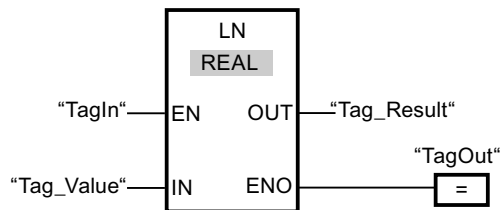
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                              |
|-----------|-------------|------------------|---------------------------------|---------------------------------|---|
|           |             |                  | S7-1200                         | S7-1500                         |   |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                           |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                           |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert                              |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Natürlicher Logarithmus des Eingangswerts |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Natürlichen Logarithmus bilden" ausgeführt. Die Anweisung bildet den natürlichen Logarithmus des Wertes am Eingang "Tag\_Value" und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## EXP: Exponentialwert bilden

### Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e ( $e = 2.718282$ ) und dem am Eingang IN angegebenen Wert. Das Ergebnis wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden ( $OUT = e^{IN}$ ).

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Exponentialwert bilden":

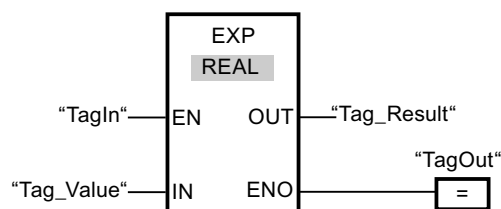
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                         |
|-----------|-------------|------------------|---------------------------------|---------------------------------|--------------------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                                      |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                      |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                      |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert                         |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Exponentialwert des Eingangswerts IN |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Exponentialwert bilden" ausgeführt. Die Anweisung bildet die Potenz aus der Basis e und dem Wert des Operanden "Tag\_Value" und gibt das Ergebnis am Ausgang "Tag\_Result" aus. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**SIN: Sinuswert bilden**

**Beschreibung**

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Sinuswert bilden":

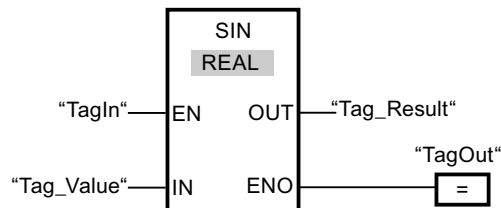
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                  |
|-----------|-------------|------------------|---------------------------------|---------------------------------|-------------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                               |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang               |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang               |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Größe des Winkels im Bogenmaß |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Sinus des angegebenen Winkels |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | +1.570796 ( $\pi/2$ ) |
| OUT       | Tag_Result | 1.0                   |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Sinuswert bilden" ausgeführt. Die Anweisung berechnet den Sinus des am Eingang "Tag\_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## COS: Cosinuswert bilden

### Beschreibung

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Cosinuswert bilden":

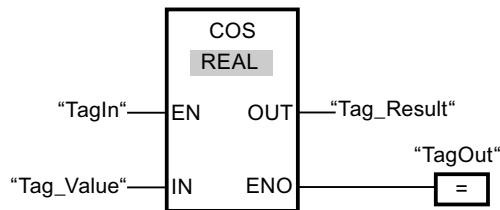
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                    |
|-----------|-------------|------------------|---------------------------------|---------------------------------|---------------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                                 |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                 |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                 |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Größe des Winkels im Bogenmaß   |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Cosinus des angegebenen Winkels |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | +1.570796 ( $\pi/2$ ) |
| OUT       | Tag_Result | 0                     |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Cosinuswert bilden" ausgeführt. Die Anweisung berechnet den Cosinus des am Eingang "Tag\_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.



**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**TAN: Tangenswert bilden****Beschreibung**

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens eines Winkels. Die Größe des Winkels wird am Eingang IN im Bogenmaß angegeben. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Tangenswert bilden":

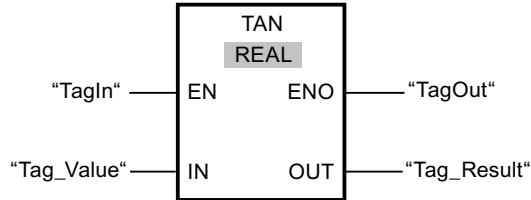
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                    |
|-----------|-------------|------------------|---------------------------------|---------------------------------|---------------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                                 |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                 |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                 |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Größe des Winkels im Bogenmaß   |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Tangens des angegebenen Winkels |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                |
|-----------|------------|---------------------|
| IN        | Tag_Value  | +3.141593 ( $\pi$ ) |
| OUT       | Tag_Result | 0                   |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Tangenswert bilden" ausgeführt. Die Anweisung berechnet den Tangens des am Eingang "Tag\_Value" angegebenen Winkels und legt das Ergebnis am Ausgang "Tag\_Result" ab. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**ASIN: Arcussinuswert bilden**

**Beschreibung**

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Sinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von  $-\pi/2$  bis  $+\pi/2$  liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcussinuswert bilden":

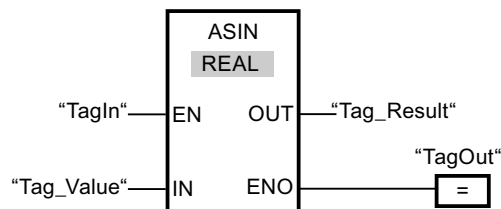
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                  |
|-----------|-------------|------------------|---------------------------------|---------------------------------|-------------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                               |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang               |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang               |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Sinuswert                     |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Größe des Winkels im Bogenmaß |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | 1.0                   |
| OUT       | Tag_Result | +1.570796 ( $\pi/2$ ) |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcussinuswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Sinuswert am Eingang "Tag\_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**ACOS: Arcuscosinuswert bilden**

**Beschreibung**

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von 0 bis  $+\pi$  liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.
- Der Wert am Eingang IN liegt außerhalb des erlaubten Wertebereichs (-1 bis +1).

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcuscosinuswert bilden":

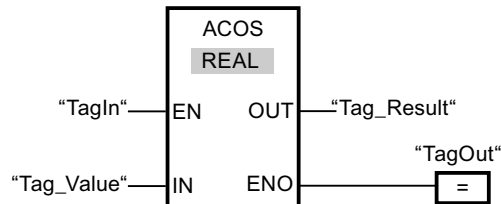
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                  |
|-----------|-------------|------------------|---------------------------------|---------------------------------|-------------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                               |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang               |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang               |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Cosinuswert                   |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Größe des Winkels im Bogenmaß |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | 0                     |
| OUT       | Tag_Result | +1.570796 ( $\pi/2$ ) |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcuscosinuswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Cosinuswert am Eingang "Tag\_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ATAN: Arcustangenswert bilden

### Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus dem am Eingang IN angegebenen Tangenswert die Größe des Winkels, der diesem Wert entspricht. Am Eingang IN dürfen nur gültige Gleitpunktzahlen angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß am Ausgang OUT ausgegeben und kann in einem Wertebereich von  $-\pi/2$  bis  $+\pi/2$  liegen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang IN ist keine gültige Gleitpunktzahl.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Arcustangenswert bilden":

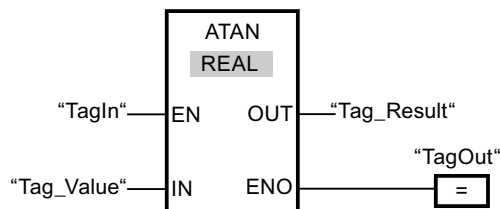
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung                  |
|-----------|-------------|------------------|---------------------------------|---------------------------------|-------------------------------|
|           |             |                  | S7-1200                         | S7-1500                         |                               |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang               |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang               |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Tangenswert                   |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Größe des Winkels im Bogenmaß |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                  |
|-----------|------------|-----------------------|
| IN        | Tag_Value  | 1.0                   |
| OUT       | Tag_Result | +0.785398 ( $\pi/4$ ) |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Arcustangenswert bilden" ausgeführt. Die Anweisung berechnet die Größe des Winkels, der dem Tangenswert am Eingang "Tag\_Value" entspricht. Das Ergebnis der Anweisung wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**FRAC: Nachkommastellen ermitteln****Beschreibung**

Mit der Anweisung "Nachkommastellen ermitteln" ermitteln Sie die Nachkommastellen des Wertes am Eingang IN. Das Ergebnis der Abfrage wird am Ausgang OUT abgelegt und kann an diesem abgefragt werden. Wenn am Eingang IN z. B. der Wert 123,4567 ansteht, liefert der Ausgang OUT den Wert 0,4567.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung der Anweisung auf, z. B. am Eingang liegt keine gültige Gleitpunktzahl an.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Nachkommastellen ermitteln":

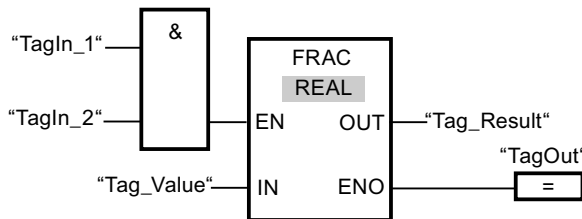
| Parameter | Deklaration | Datentyp         | Speicherbereich                 |                                 | Beschreibung  |
|-----------|-------------|------------------|---------------------------------|---------------------------------|---|
|           |             |                  | S7-1200                         | S7-1500                         |   |
| EN        | Input       | BOOL             | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang   |
| ENO       | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang   |
| IN        | Input       | Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert, dessen Nachkommastellen ermittelt werden. |
| OUT       | Output      | Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Nachkommastellen des Eingangswerts am Eingang IN        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|-------|
| IN        | Tag_Value  | 2.555 |
| OUT       | Tag_Result | 0.555 |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Nachkommastellen ermitteln" gestartet. Die Nachkommastellen aus dem Wert des Operanden "Tag\_Value" werden in den Operanden "Tag\_Result" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

**EXPT: Potenzieren**

**Beschreibung**

Mit der Anweisung "Potenzieren" potenzieren Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2. Das Ergebnis der Anweisung wird am Ausgang OUT abgelegt und kann an diesem abgefragt werden (OUT = IN1<sup>IN2</sup>).

Der Eingang IN1 kann nur mit gültigen Gleitpunktzahlen belegt werden. Zur Belegung des Eingangs IN2 sind auch Ganzzahlen zulässig.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung der Anweisung auf, z. B. es tritt ein Überlauf auf.



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Potenzieren":

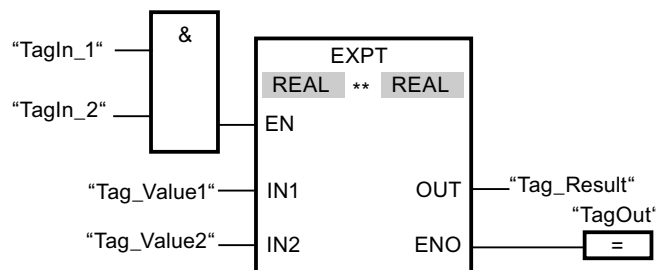
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 |                                 | Beschreibung                                |
|-----------|-------------|------------------------------|---------------------------------|---------------------------------|---|
|           |             |                              | S7-1200                         | S7-1500                         |   |
| EN        | Input       | BOOL                         | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                             |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                             |
| IN1       | Input       | Gleitpunktzahlen             | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Basiswert                                   |
| IN2       | Input       | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Wert, mit dem der Basiswert potenziert wird |
| OUT       | Output      | Gleitpunktzahlen             | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis                                    |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Potenzieren" gestartet. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" potenziert. Das Ergebnis wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

**Verschieben**

**MOVE: Wert kopieren**

**Beschreibung**

Mit der Anweisung "Wert kopieren" übertragen Sie den Inhalt des Operanden am Eingang IN zum Operanden am Ausgang OUT1. Die Übertragung erfolgt dabei immer in Richtung aufsteigender Adresse.

Die folgende Tabelle zeigt die möglichen Übertragungen für die CPU-Familie S7-1200:

| Quelle (IN)                              | Ziel (OUT1)                |   |
|--|----------------------------|---|
|  | Mit IEC-Prüfung            | Ohne IEC-Prüfung  |
| BYTE                                     | BYTE, WORD, DWORD          | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD, CHAR       |
| WORD                                     | WORD, DWORD                | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD, CHAR       |
| DWORD                                    | DWORD                      | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, REAL, TIME, DATE, TOD, CHAR |
| SINT                                     | SINT                       | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| USINT                                    | USINT, UINT, UDINT         | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| INT                                      | INT                        | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| UINT                                     | UINT, UDINT                | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| DINT                                     | DINT                       | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| UDINT                                    | UDINT                      | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD             |
| REAL                                     | REAL                       | DWORD, REAL   |
| LREAL                                    | LREAL                      | LREAL   |
| TIME                                     | TIME                       | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME                        |
| DATE                                     | DATE                       | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, DATE                        |
| TOD                                      | TOD                        | BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TOD                         |
| DTL                                      | DTL                        | DTL   |
| CHAR                                     | CHAR                       | BYTE, WORD, DWORD, CHAR, Zeichen einer Zeichenkette <sup>1)</sup>                   |
| Zeichen einer Zeichenkette <sup>1)</sup> | Zeichen einer Zeichenkette | CHAR, Zeichen einer Zeichenkette  |
| ARRAY <sup>2)</sup>                      | ARRAY                      | ARRAY   |
| STRUCT                                   | STRUCT                     | STRUCT  |

| Quelle (IN)        | Ziel (OUT1)        |                    |
|--------------------|--------------------|--------------------|
|                    | Mit IEC-Prüfung    | Ohne IEC-Prüfung   |
| PLC-Datentyp (UDT) | PLC-Datentyp (UDT) | PLC-Datentyp (UDT) |
| IEC_TIMER          | IEC_TIMER          | IEC_TIMER          |
| IEC_SCOUNTER       | IEC_SCOUNTER       | IEC_SCOUNTER       |
| IEC_USCOUNTER      | IEC_USCOUNTER      | IEC_USCOUNTER      |
| IEC_COUNTER        | IEC_COUNTER        | IEC_COUNTER        |
| IEC_UCOUNTER       | IEC_UCOUNTER       | IEC_UCOUNTER       |
| IEC_DCOUNTER       | IEC_DCOUNTER       | IEC_DCOUNTER       |
| IEC_UDCOUNTER      | IEC_UDCOUNTER      | IEC_UDCOUNTER      |

Die folgende Tabelle zeigt die möglichen Übertragungen für die CPU-Familie S7-1500:

| Quelle (IN) | Ziel (OUT1)               |   |
|-------------|---------------------------|---|
|             | Mit IEC-Prüfung           | Ohne IEC-Prüfung  |
| BYTE        | BYTE, WORD, DWORD, LWORD  | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR         |
| WORD        | WORD, DWORD, LWORD        | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, S5TIME, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR |
| DWORD       | DWORD, LWORD              | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR   |
| LWORD       | LWORD                     | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LREAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR  |
| SINT        | SINT                      | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |
| USINT       | USINT, UINT, UDINT, ULINT | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |
| INT         | INT                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |
| UINT        | UINT, UDINT, ULINT        | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |
| DINT        | DINT                      | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD               |

| Quelle (IN)                              | Ziel (OUT1)                |   |
|--|----------------------------|---|
|  | Mit IEC-Prüfung            | Ohne IEC-Prüfung  |
| UDINT                                    | UDINT, ULINT               | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD |
| LINT                                     | LINT                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD |
| ULINT                                    | ULINT                      | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD |
| REAL                                     | REAL                       | DWORD, REAL   |
| LREAL                                    | LREAL                      | LWORD, LREAL  |
| S5TIME                                   | S5TIME                     | WORD, S5TIME  |
| TIME                                     | TIME                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME                              |
| LTIME                                    | LTIME                      | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTIME                             |
| DATE                                     | DATE                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, DATE                              |
| DT                                       | DT                         | DT  |
| LDT                                      | LDT                        | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LDT                               |
| TOD                                      | TOD                        | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TOD                               |
| LTOD                                     | LTOD                       | BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTOD                              |
| DTL                                      | DTL                        | DTL   |
| CHAR                                     | CHAR                       | BYTE, WORD, DWORD, LWORD, CHAR, Zeichen einer Zeichenkette <sup>1)</sup>                                      |
| Zeichen einer Zeichenkette <sup>1)</sup> | Zeichen einer Zeichenkette | CHAR, Zeichen einer Zeichenkette  |
| ARRAY <sup>2)</sup>                      | ARRAY                      | ARRAY   |
| STRUCT                                   | STRUCT                     | STRUCT  |
| COUNTER                                  | COUNTER, WORD, INT         | WORD, DWORD, INT, UINT, DINT, UDINT   |
| TIMER                                    | TIMER, WORD, INT           | WORD, DWORD, INT, UINT, DINT, UDINT   |
| PLC-Datentyp (UDT)                       | PLC-Datentyp (UDT)         | PLC-Datentyp (UDT)  |
| IEC_TIMER                                | IEC_TIMER                  | IEC_TIMER   |
| IEC_LTIMER                               | IEC_LTIMER                 | IEC_LTIMER  |
| IEC_SCOUNTER                             | IEC_SCOUNTER               | IEC_SCOUNTER  |
| IEC_USCOUNTER                            | IEC_USCOUNTER              | IEC_USCOUNTER   |
| IEC_COUNTER                              | IEC_COUNTER                | IEC_COUNTER   |
| IEC_UCOUNTER                             | IEC_UCOUNTER               | IEC_UCOUNTER  |

| Quelle (IN)       | Ziel (OUT1)     |                  |
|-------------------|-----------------|------------------|
|                   | Mit IEC-Prüfung | Ohne IEC-Prüfung |
| IEC_DCOUNT<br>R   | IEC_DCOUNTER    | IEC_DCOUNTER     |
| IEC_UDCOUN<br>TER | IEC_UDCOUNTER   | IEC_UDCOUNTER    |
| IEC_LCOUNT<br>R   | IEC_LCOUNTER    | IEC_LCOUNTER     |
| IEC_ULCOUN<br>TER | IEC_ULCOUNTER   | IEC_ULCOUNTER    |

<sup>1)</sup> Mit der Anweisung "Wert kopieren" können Sie auch einzelne Zeichen einer Zeichenkette (STRING) zu Operanden vom Datentyp CHAR übertragen. Die Nummer des zu übertragenden Zeichens wird in eckigen Klammern neben dem Operandennamen angegeben. Bei der Angabe "MyString[2]" wird zum Beispiel das zweite Zeichen der Zeichenkette "MyString" übertragen. Die Übertragung vom Operanden des Datentyps CHAR zu einzelnen Zeichen einer Zeichenkette ist ebenfalls möglich. Sie können auch ein bestimmtes Zeichen einer Zeichenkette durch das Zeichen einer anderen Zeichenkette ersetzen.

<sup>2)</sup> Das Übertragen von ganzen Feldern (ARRAY) ist nur dann möglich, wenn die Feldkomponenten der Operanden am Eingang IN und am Ausgang OUT1 vom gleichen Datentyp sind.

Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 überschreitet, gehen die höherwertigen Bits des Quellwerts verloren. Wenn die Bitlänge des Datentyps am Eingang IN die Bitlänge des Datentyps am Ausgang OUT1 unterschreitet, werden die höherwertigen Bits des Zielwerts mit Nullen überschrieben.

Die Anweisungsbox enthält im Grundzustand 1 Ausgang (OUT1). Die Anzahl der Ausgänge ist erweiterbar. Die eingefügten Ausgänge werden an der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung wird der Inhalt des Operanden am Eingang IN zu allen verfügbaren Ausgängen übertragen. Die Anweisungsbox kann nicht erweitert werden, wenn strukturierte Datentypen (DTL, STRUCT, ARRAY) oder Zeichen einer Zeichenkette (STRING) übertragen werden.

Zum Kopieren von Operanden des Datentyps ARRAY können auch die Anweisungen "Bereich kopieren" (MOVE\_BLK) und "Bereich ununterbrechbar kopieren" (UMOVE\_BLK) verwendet werden. Operanden des Datentyps STRING können mit der Anweisung "Zeichenkette verschieben" (S\_MOVE) kopiert werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert kopieren":

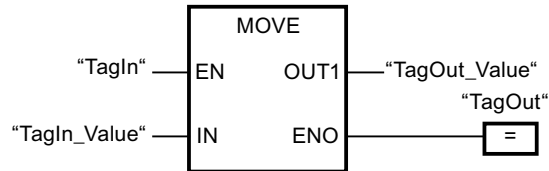
| Parameter | Deklaration | Datentyp |         | Speicherbereich |                     | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|---------------------|-----------------|
|           |             | S7-1200  | S7-1500 | S7-1200         | S7-1500             |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L, T, Z | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L       | Freigabeausgang |

| Parameter | Deklaration | Datentyp   |  | Speicherbereich              |                              | Beschreibung   |
|-----------|-------------|--|--|------------------------------|------------------------------|--|
|           |             | S7-1200  | S7-1500  | S7-1200                      | S7-1500                      |  |
| IN        | Input       | Bitfolgen,<br>Ganzzahlen,<br><br>Gleitpunktzahlen, DATE, TIME, TOD, DTL, CHAR, STRUCT, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT) | Bitfolgen,<br>Ganzzahlen,<br><br>Gleitpunktzahlen, DATE, DT, LDT, S5TIME, TIME, LTIME, TOD, LTOD, DTL, CHAR, STRUCT, ARRAY, TIMER, COUNTER, IEC-Datentypen, PLC-Datentyp (UDT) | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Element, mit dem die Zieladresse überschrieben wird. |
| OUT1      | Output      | Bitfolgen,<br>Ganzzahlen,<br><br>Gleitpunktzahlen, DATE, TIME, TOD, DTL, CHAR, STRUCT, ARRAY, IEC-Datentypen, PLC-Datentyp (UDT) | Bitfolgen,<br>Ganzzahlen,<br><br>Gleitpunktzahlen, DATE, DT, LDT, S5TIME, TIME, LTIME, TOD, LTOD, DTL, CHAR, STRUCT, ARRAY, TIMER, COUNTER, IEC-Datentypen, PLC-Datentyp (UDT) | E, A, M, D, L                | E, A, M, D, L                | Zieladresse  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0011 1111 1010 1111 |
| OUT1      | TagOut_Value | 0011 1111 1010 1111 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Wert kopieren" ausgeführt. Die Anweisung kopiert die Inhalte des Operanden "TagIn\_Value" zum Operanden "TagOut\_Value" und setzt den Ausgang "TagOut" auf den Signalzustand "1".

## Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)
- MOVE\_BLK: Bereich kopieren (Seite 2037)
- UMOVE\_BLK: Bereich ununterbrechbar kopieren (Seite 2043)
- S\_MOVE: Zeichenkette verschieben (Seite 2364)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

## FieldRead: Feld lesen

### Beschreibung

Mit der Anweisung "Feld lesen" lesen Sie eine bestimmte Komponente aus dem am Parameter MEMBER angegebenen Feld aus und übertragen deren Inhalt in die Variable am Parameter VALUE. Den Index der zu lesenden Feldkomponente legen Sie am Parameter INDEX fest. Am Parameter MEMBER geben Sie die erste Komponente des Felds an, aus dem gelesen wird.

Die Datentypen der Feldkomponente am Parameter MEMBER und der Variablen am Parameter VALUE müssen mit dem Datentyp der Anweisung "Feld lesen" übereinstimmen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die am Parameter INDEX angegebene Feldkomponente ist in dem am Parameter MEMBER angegebenen Feld nicht definiert.
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Feld lesen":

| Parameter | Deklaration | Datentyp   |  | Speicherbereich                 |                                 | Beschreibung  |
|-----------|-------------|--|--|---------------------------------|---------------------------------|---|
|           |             | S7-1200  | S7-1500  | S7-1200                         | S7-1500                         |   |
| EN        | Input       | BOOL   | BOOL   | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang   |
| ENO       | Output      | BOOL   | BOOL   | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang   |
| INDEX     | Input       | DINT   | DINT   | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Index der Feldkomponente, deren Inhalt ausgelesen wird        |
| MEMBER    | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME, DATE, TOD und CHAR als Komponente einer ARRAY-Variablen | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD und CHAR als Komponente einer ARRAY-Variablen | D, L                            | D, L                            | Erste Komponente des Felds, aus dem gelesen wird              |
| VALUE     | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME, DATE, TOD und CHAR                                      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD und CHAR                                      | E, A, M, D, L, P                | E, A, M, D, L, P                | Operand, in den der Inhalt der Feldkomponente übertragen wird |

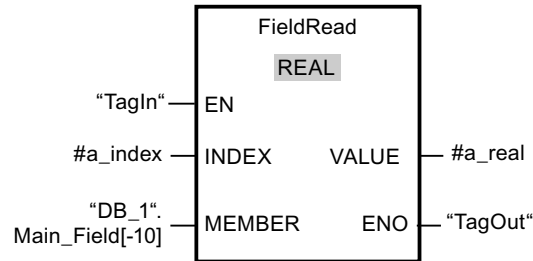
Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Variable               | Wert   |
|-----------|------------------------|--|
| INDEX     | a_index                | 4  |
| MEMBER    | "DB_1".Main_Field[-10] | Erste Komponente des Felds "Main_Field[-10..10] of REAL" im Datenbaustein "DB_1" |
| VALUE     | a_real                 | Komponente mit Index 4 des Felds "Main_Field[-10..10] of REAL"                   |

Die Feldkomponente mit Index 4 wird aus dem Feld "Main\_Field[-10...10] of REAL" ausgelesen und in die Variable "a\_real" geschrieben. Die zu lesende Feldkomponente wird durch den Wert am Parameter INDEX festgelegt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## FieldWrite: Feld schreiben

### Beschreibung

Mit der Anweisung "Feld schreiben" übertragen Sie den Inhalt der Variablen am Eingang VALUE in eine bestimmte Komponente des Felds am Ausgang MEMBER. Den Index der Feldkomponente, die beschrieben wird, legen Sie durch den Wert am Eingang INDEX fest. Am Ausgang MEMBER geben Sie die erste Komponente des Felds an, in das geschrieben wird.

Die Datentypen der am Ausgang MEMBER angegebenen Feldkomponente und der Variablen am Eingang VALUE müssen mit dem Datentyp der Anweisung "Feld schreiben" übereinstimmen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Die am Eingang INDEX angegebene Feldkomponente ist in dem am Ausgang MEMBER angegebenen Feld nicht definiert.
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Feld schreiben":

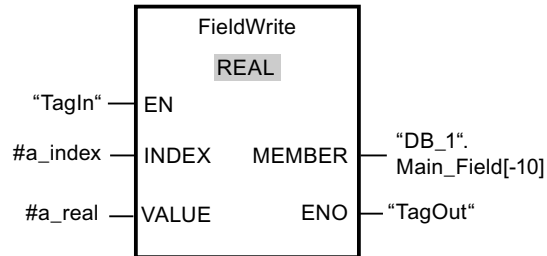
| Parameter | Deklaration | Datentyp   |  | Speicherbereich                 |                                 | Beschreibung  |
|-----------|-------------|--|--|---------------------------------|---------------------------------|---|
|           |             | S7-1200  | S7-1500  | S7-1200                         | S7-1500                         |   |
| EN        | Input       | BOOL   | BOOL   | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang   |
| ENO       | Output      | BOOL   | BOOL   | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang   |
| INDEX     | Input       | DINT   | DINT   | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Index der Feldkomponente, die mit dem Inhalt von VALUE beschrieben wird       |
| VALUE     | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME, DATE, TOD und CHAR                                      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD und CHAR                                      | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Operand, dessen Inhalt kopiert wird   |
| MEMBER    | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME, DATE, TOD und CHAR als Komponente einer ARRAY-Variablen | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten, DATE, TOD, LTOD und CHAR als Komponente einer ARRAY-Variablen | D, L                            | D, L                            | Erste Komponente des Felds, in welches der Inhalt von VALUE geschrieben wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand                | Wert   |
|-----------|------------------------|--|
| INDEX     | a_index                | 4  |
| VALUE     | a_real                 | 10.54  |
| MEMBER    | "DB_1".Main_Field[-10] | Erste Komponente des Felds "Main_Field[-10..10] of REAL" im Datenbaustein "DB_1" |

Der Wert "10.54" der Variablen "a\_real" wird in die Feldkomponente mit Index 4 des Felds "Main\_Field[-10 ... 10] of REAL" geschrieben. Der Index der Feldkomponente, in die der Inhalt der Variablen "a\_real" übertragen wird, wird durch den Wert am Eingang INDEX festgelegt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## MOVE\_BLK: Bereich kopieren

### Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein Array of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der Array-Struktur überschritten wird. Wird die Bytegrenze der Array-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

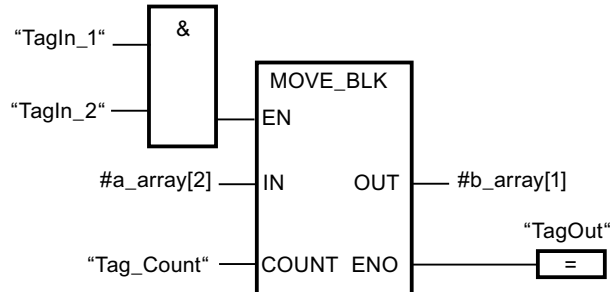
| Parameter         | Deklaration | Datentyp   |   | Speicherbereich                    |                                    | Beschreibung  |
|-------------------|-------------|--|---|------------------------------------|------------------------------------|---|
|                   |             | S7-1200  | S7-1500   | S7-1200                            | S7-1500                            |   |
| EN                | Input       | BOOL   | BOOL  | E, A, M, D, L                      | E, A, M, D, L, T, Z                | Freigabeausgang   |
| ENO               | Output      | BOOL   | BOOL  | E, A, M, D, L                      | E, A, M, D, L                      | Freigabeausgang   |
| IN <sup>1)</sup>  | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME,<br>DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten,<br>DATE,<br>CHAR,<br>TOD, LTOD | D, L                               | D, L                               | Das erste Element des Quellbereichs, das kopiert wird                                   |
| COUNT             | Input       | USINT,<br>UINT,<br>UDINT   | USINT,<br>UINT,<br>UDINT,<br>ULINT  | E, A, M, D, L, P oder<br>Konstante | E, A, M, D, L, P oder<br>Konstante | Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden         |
| OUT <sup>1)</sup> | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME,<br>DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten,<br>DATE,<br>CHAR,<br>TOD, LTOD | D, L                               | D, L                               | Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden |

<sup>1)</sup> Die angegebenen Datentypen können nur als Elemente einer Array-Struktur verwendet werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Bereich kopieren" ausgeführt. Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a\_array und kopiert deren Inhalte in die Ausgangsvariable #b\_array, beginnend bei dem zweiten Element. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## MOVE\_BLK\_VARIANT: Bereich kopieren

### Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Sie können Elemente eines Arrays in ein anderes Array des selben Datentyps kopieren. Die Größen (Anzahl der Elemente) von Quell- und Ziel-Array dürfen unterschiedlich sein. Sie können mehrere Elemente innerhalb eines Arrays umkopieren oder einzelne Elemente kopieren.

Zum Zeitpunkt der Bausteinerstellung, wenn Sie die Anweisung verwenden, muss der Array noch nicht bekannt sein, da die Quelle und das Ziel per VARIANT übergeben werden.

Die Zählung an den Parametern SRC\_INDEX und DEST\_INDEX beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des Arrays.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als zur Verfügung gestellt werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

| Parameter | Deklaration | Datentyp                                  | Speicherbereich              | Beschreibung   |
|-----------|-------------|---|------------------------------|--|
| EN        | Input       | BOOL                                      | E, A, M, D, L                | Freigabeeingang  |
| ENO       | Output      | BOOL                                      | E, A, M, D, L                | Freigabeausgang  |
| SRC       | Input       | VARIANT<br>(Array oder einzelnes Element) | E, A, D, L                   | Das erste Element des Quellbereichs, das kopiert wird  |
| COUNT     | Input       | UDINT                                     | E, A, M, D, L oder Konstante | Gibt die Anzahl der Elemente, die kopiert werden, an.<br>Geben Sie dem Parameter COUNT den Wert „1“, wenn am Parameter SRC oder am Parameter DEST kein Array angegeben ist.  |
| SRC_INDEX | Input       | UDINT                                     | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• Wenn am Parameter SRC ein Array angegeben ist, dann gibt der Parameter SRC_INDEX das erste Element am Parameter SRC an, das kopiert werden soll.</li> <li>• Wenn am Parameter SRC kein Array angegeben ist, dann geben Sie dem Parameter SRC_INDEX den Wert „0“.</li> </ul> |

| Parameter  | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|------------|-------------|----------|------------------------------|---|
| DEST_INDEX | Input       | UDINT    | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• Wenn am Parameter DEST ein Array angegeben ist, dann gibt der Parameter DEST_INDEX das erste Element am Parameter DEST an, das kopiert werden soll.</li> <li>• Wenn am Parameter DEST kein Array angegeben ist, dann geben Sie dem Parameter DEST_INDEX den Wert „0“.</li> </ul> |
| DEST       | InOut       | VARIANT  | E, A, D, L                   | Das erste Element des Quellbereichs, das kopiert wird   |
| RET_VAL    | Output      | INT      | E, A, M, D, L                | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

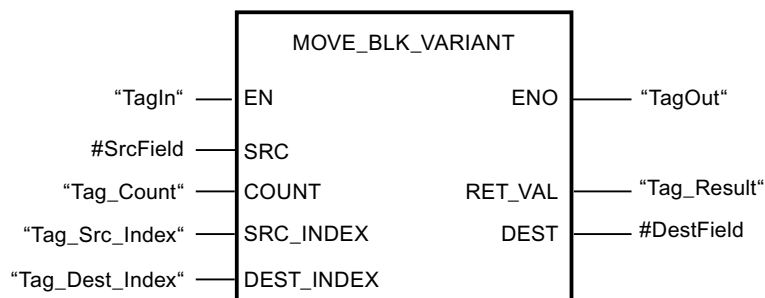
Die folgende Tabelle zeigt die Bedeutung der Werte des Parameter RET\_VAL:

| Fehlercode*<br>(W#16#...) | Erläuterung                           |
|---------------------------|---------------------------------------|
| 0000                      | Kein Fehler                           |
| 8082                      | Datentypen stimmen nicht überein      |
| 80B4                      | Datentypen stimmen nicht überein      |
| 8151                      | Codeerzeugungsfehler am Parameter SRC |
| 8152                      | Codeerzeugungsfehler am Parameter SRC |
| 8153                      | Codeerzeugungsfehler am Parameter SRC |
| 8251                      | Ungültiger Wert am Parameter COUNT    |
| 8254                      | Ungültiger Wert am Parameter COUNT    |
| 8281                      | Ungültiger Wert am Parameter COUNT    |

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 8282   | Ungültiger Wert am Parameter COUNT   |
| 8382   | Der Wert am Parameter SRC_INDEX befindet sich außerhalb der Grenzwerte des VARIANT         |
| 8383   | Der Wert am Parameter SRC_INDEX befindet sich außerhalb des oberen Grenzwerts des Arrays.  |
| 8482   | Der Wert am Parameter DEST_INDEX befindet sich außerhalb der Grenzwerte des VARIANT        |
| 8483   | Der Wert am Parameter DEST_INDEX befindet sich außerhalb des oberen Grenzwerts des Arrays. |
| 8534   | Der Parameter DEST ist schreibgeschützt  |
| 8551   | Codeerzeugungsfehler am Parameter DEST   |
| 8552   | Codeerzeugungsfehler am Parameter DEST   |
| 8553   | Codeerzeugungsfehler am Parameter DEST   |
| 85A2   | Der Parameter DEST ist schreibgeschützt  |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand       | Wert  |
|-----------|---------------|---|
| SRC       | #SrcField     | Der lokale Operand #SrcField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekanntes, UDT. |
| COUNT     | Tag_Count     | 3   |
| SRC_INDEX | Tag_Src_Index | 4   |



| Parameter  | Operand        | Wert   |
|------------|----------------|--|
| DEST_INDEX | Tag_Dest_Index | 2  |
| DEST       | #DestField     | Der lokale Operand #DestField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekanntes, UDT. |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Bereich kopieren" ausgeführt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## UMOVE\_BLK: Bereich ununterbrechbar kopieren

### Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich) ununterbrechbar. Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des Elements am Eingang IN definiert.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

---

### Hinweis

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

---

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein Array of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der Array-Struktur überschritten wird. Wird die Bytegrenze der Array-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

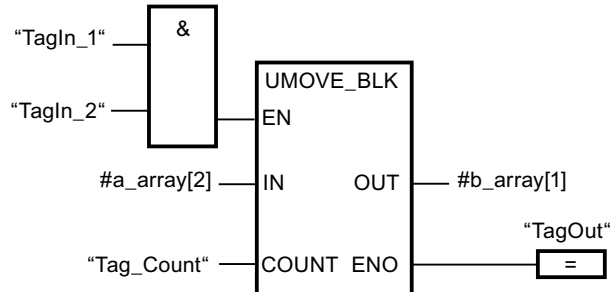
| Parameter         | Deklaration | Datentyp   |   | Speicherbereich                 |                                 | Beschreibung  |
|-------------------|-------------|--|---|---------------------------------|---------------------------------|---|
|                   |             | S7-1200  | S7-1500   | S7-1200                         | S7-1500                         |   |
| EN                | Input       | BOOL   | BOOL  | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang   |
| ENO               | Output      | BOOL   | BOOL  | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang   |
| IN <sup>1)</sup>  | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME,<br>DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten,<br>DATE,<br>CHAR,<br>TOD, LTOD | D, L                            | D, L                            | Das erste Element des Quellbereichs, das kopiert wird                                   |
| COUNT             | Input       | USINT,<br>UINT,<br>UDINT   | USINT,<br>UINT,<br>UDINT,<br>ULINT  | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Anzahl der Elemente, die aus dem Quellbereich in den Zielbereich kopiert werden         |
| OUT <sup>1)</sup> | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME,<br>DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten,<br>DATE,<br>CHAR,<br>TOD, LTOD | D, L                            | D, L                            | Das erste Element des Zielbereichs, in den die Inhalte des Quellbereichs kopiert werden |

<sup>1)</sup> Die angegebenen Datentypen können nur als Elemente einer Array-Struktur verwendet werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Bereich ununterbrechbar kopieren" ausgeführt. Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a\_array und kopiert deren Inhalte in die Ausgangsvariable #b\_array, beginnend bei dem zweiten Element. Der Kopiervorgang kann durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## FILL\_BLK: Bereich befüllen

### Beschreibung

Mit der Anweisung "Bereich befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN selektiert und so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein Array of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der Array-Struktur überschritten wird. Wird die Bytegrenze der Array-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

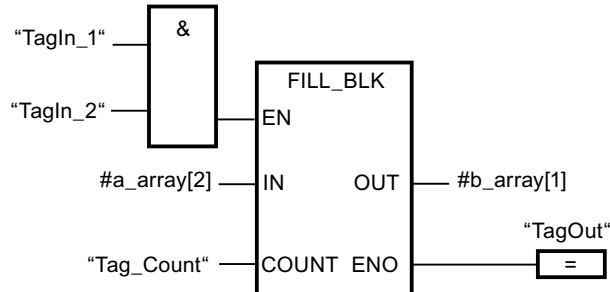
| Parameter         | Deklaration | Datentyp   |  | Speicherbereich                 |                                 | Beschreibung                                  |
|-------------------|-------------|--|--|---------------------------------|---------------------------------|---|
|                   |             | S7-1200  | S7-1500  | S7-1200                         | S7-1500                         |   |
| EN                | Input       | BOOL   | BOOL   | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                               |
| ENO               | Output      | BOOL   | BOOL   | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                               |
| IN <sup>1)</sup>  | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME, DATE, TOD, CHAR | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Element, mit dem der Zielbereich befüllt wird |
| COUNT             | Input       | USINT, UINT, UDINT   | USINT, UINT, UDINT, ULINT  | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Anzahl der Kopierwiederholungen               |
| OUT <sup>2)</sup> | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME, DATE, TOD, CHAR | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | D, L                            | D, L                            | Adresse im Zielbereich, ab der befüllt wird   |

<sup>1)</sup> Die angegebenen Datentypen können auch als Elemente einer Array-Struktur verwendet werden.  
<sup>2)</sup> Die angegebenen Datentypen können nur als Elemente einer Array-Struktur verwendet werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Bereich befüllen" ausgeführt. Die Anweisung kopiert drei Mal das dritte Element (#a\_array[2]) der Variablen #a\_array in die Ausgangsvariable #b\_array. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden die Ausgänge ENO und "TagOut" auf den Signalzustand "1" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## UFILL\_BLK: Bereich ununterbrechbar befüllen

### Beschreibung

Mit der Anweisung "Bereich ununterbrechbar befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Wert des Eingangs IN ununterbrechbar. Der Zielbereich wird ab der Adresse befüllt, die am Ausgang OUT angegeben ist. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN selektiert und so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

**Hinweis**

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar befüllen" erhöhen.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Wenn ein Array of BOOL kopiert wird, bleibt der Freigabeausgang ENO bei einem Überlauf solange auf "1" gesetzt, bis die Bytegrenze der Array-Struktur überschritten wird. Wird die Bytegrenze der Array-Struktur durch den Wert am Eingang COUNT überschritten, wird der Freigabeausgang ENO auf "0" zurückgesetzt.

Mit der Anweisung "Bereich ununterbrechbar befüllen" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar befüllen":

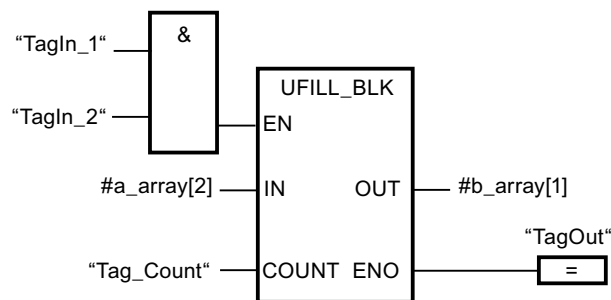
| Parameter        | Deklaration | Datentyp   |  | Speicherbereich                 |                                 | Beschreibung                                  |
|------------------|-------------|--|--|---------------------------------|---------------------------------|---|
|                  |             | S7-1200  | S7-1500  | S7-1200                         | S7-1500                         |   |
| EN               | Input       | BOOL   | BOOL   | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                               |
| ENO              | Output      | BOOL   | BOOL   | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                               |
| IN <sup>1)</sup> | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME, DATE, CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten, DATE, CHAR, TOD, LTOD | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Element, mit dem der Zielbereich befüllt wird |
| COUNT            | Input       | USINT, UINT, UDINT   | USINT, UINT, UDINT, ULINT  | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Anzahl der Kopierwiederholungen               |

| Parameter   | Deklaration | Datentyp   |   | Speicherbereich |         | Beschreibung                                |
|---|-------------|--|---|-----------------|---------|---|
|   |             | S7-1200  | S7-1500   | S7-1200         | S7-1500 |   |
| OUT <sup>2)</sup>   | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, TIME,<br>DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen, Zeiten,<br>DATE,<br>CHAR,<br>TOD, LTOD | D, L            | D, L    | Adresse im Zielbereich, ab der befüllt wird |
| <sup>1)</sup> Die angegebenen Datentypen können auch als Elemente einer Array-Struktur verwendet werden.<br><sup>2)</sup> Die angegebenen Datentypen können nur als Elemente einer Array-Struktur verwendet werden. |             |  |   |                 |         |   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Wenn die Operanden "TagIn\_1" und "TagIn\_2" den Signalzustand "1" liefern, wird die Anweisung "Bereich ununterbrechbar befüllen" ausgeführt. Die Anweisung kopiert drei Mal das dritte Element (#a\_array[2]) der Variablen #a\_array in die Ausgangsvariable #b\_array. Der Kopiervorgang kann dabei durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden die Ausgänge ENO und "TagOut" auf den Signalzustand "1" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

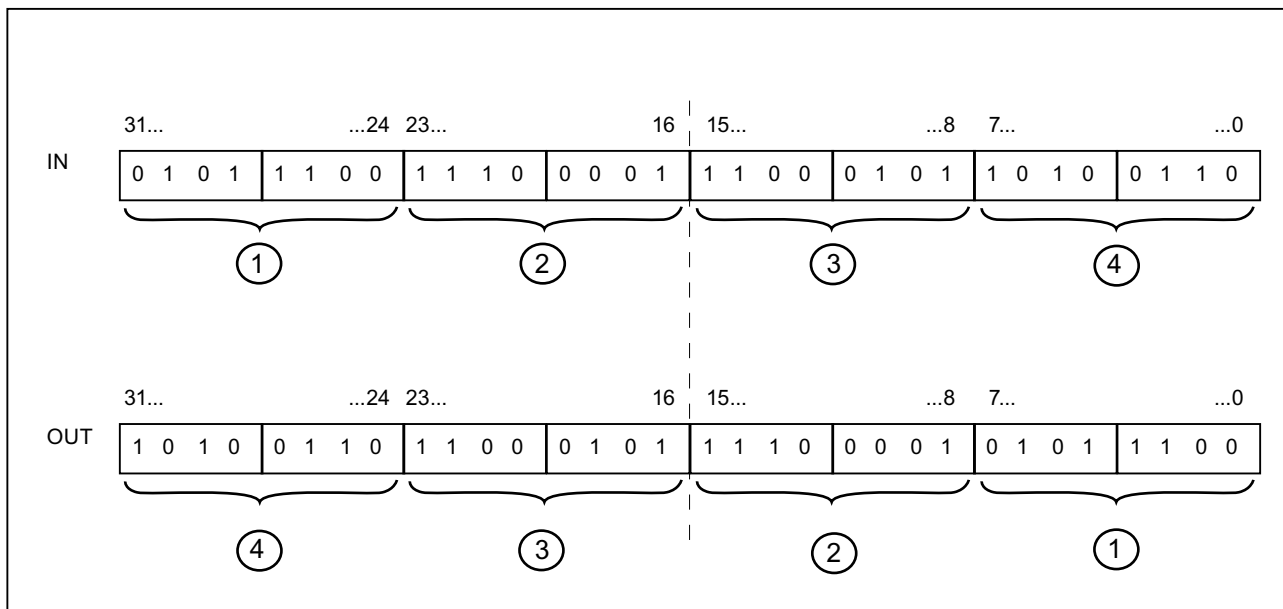
Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**SWAP: Anordnung ändern**

**Beschreibung**

Mit der Anweisung "Anordnung ändern" ändern Sie die Anordnung der Bytes am Eingang IN und fragt das Ergebnis am Ausgang OUT ab.

Das folgende Bild zeigt, wie die Bytes eines Operanden vom Datentyp DWORD mit der Anweisung "Anordnung ändern" geändert werden:



**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Anordnung ändern":

| Parameter | Deklaration | Datentyp |         | Speicherbereich |                     | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|---------------------|-----------------|
|           |             | S7-1200  | S7-1500 | S7-1200         | S7-1500             |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L, T, Z | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L       | Freigabeausgang |



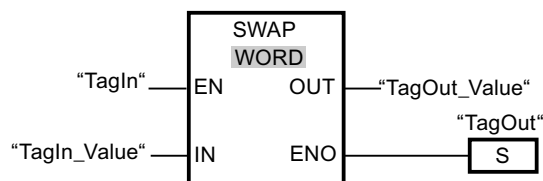
| Parameter | Deklaration | Datentyp    |                    | Speicherbereich                 |                                 | Beschreibung                           |
|-----------|-------------|-------------|--------------------|---------------------------------|---------------------------------|--|
|           |             | S7-1200     | S7-1500            | S7-1200                         | S7-1500                         |  |
| IN        | Input       | WORD, DWORD | WORD, DWORD, LWORD | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Operand, dessen Bytes geändert werden. |
| OUT       | Output      | WORD, DWORD | WORD, DWORD, LWORD | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis                               |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0000 1111 0101 0101 |
| OUT       | TagOut_Value | 0101 0101 0000 1111 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Anordnung ändern" ausgeführt. Die Anordnung der Bytes wird geändert und im Operanden "TagOut\_Value" abgelegt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## Array-Datenbausteine

### ReadFromArrayDB: Aus Array-Datenbaustein lesen

#### Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein lesen" lesen Sie Daten aus einem Array-DB.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des Arrays.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein lesen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, aus dem gelesen wird  |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element, das gelesen wird  |
| VALUE     | InOut       | VARIANT  | E, A, M, D, L   | Wert, der gelesen wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 80B4   | Datentypen stimmen nicht überein   |
| 80B5   | Kopiervorgang wurde abgebrochen  |
| 8131   | Der Datenbaustein existiert nicht  |
| 8132   | Der Datenbaustein ist zu kurz  |
| 8133   | Der Datenbaustein liegt im Ladespeicher  |
| 8134   | Der Datenbaustein ist schreibgeschützt   |
| 8135   | Der Datenbaustein ist kein Array-DB  |
| 8282   | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350   | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352   | Codeerzeugungsfehler   |
| 8353   | Codeerzeugungsfehler   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## WriteToArrayDB: In Array-Datenbaustein schreiben

### Beschreibung

Mit der Anweisung "In Array-Datenbaustein schreiben" schreiben Sie Daten in einen Array-DB.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein schreiben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, in den geschrieben wird   |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element im DB, in das geschrieben wird   |
| VALUE     | Input       | VARIANT  | E, A, M, D, L   | Wert, der geschrieben wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...) | Erläuterung  |
|---------------------------|--|
| 0000                      | Kein Fehler  |
| 80B4                      | Datentypen stimmen nicht überein   |
| 80B5                      | Kopiervorgang wurde abgebrochen  |
| 8131                      | Der Datenbaustein existiert nicht  |
| 8132                      | Der Datenbaustein ist zu kurz  |
| 8133                      | Der Datenbaustein liegt im Ladespeicher  |
| 8134                      | Der Datenbaustein ist schreibgeschützt   |
| 8135                      | Der Datenbaustein ist kein Array-DB  |
| 8282                      | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350                      | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352                      | Codeerzeugungsfehler   |

| Fehlercode*<br>(W#16#...)  | Erläuterung          |
|--|----------------------|
| 8353   | Codeerzeugungsfehler |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                      |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen

### Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen" lesen Sie Daten aus einem Array-DB im Ladespeicher.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Wenn der Array-DB mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalfanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalfanke erfasst wird, ist die Anweisung beendet. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1" und innerhalb dieses Zyklus wird der ausgelesene Wert am Parameter VALUE ausgegeben. Bei allen anderen Programmzyklen wird der Wert am Parameter VALUE nicht verändert.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| REQ       | Input       | BOOL     | E, A, M, D, L   | REQ = "1": Mit dem Auslesen des Array-DBs beginnen   |
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, aus dem gelesen wird  |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element, das gelesen wird  |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | BUSY = "1": Das Auslesen des Array-DBs ist noch nicht abgeschlossen  |
| DONE      | Output      | BOOL     | E, A, M, D, L   | DONE = "1": Die Anweisung wurde erfolgreich ausgeführt   |
| VALUE     | InOut       | VARIANT  | E, A, M, D, L   | Wert, der gelesen wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Parameter**

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode* (W#16#...) | Erläuterung                      |
|------------------------|----------------------------------|
| 0000                   | Kein Fehler                      |
| 80B4                   | Datentypen stimmen nicht überein |
| 80B5                   | Kopiervorgang wurde abgebrochen  |

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 8131   | Der Datenbaustein existiert nicht  |
| 8132   | Der Datenbaustein ist zu kurz  |
| 8133   | Der Datenbaustein liegt nicht im Ladespeicher                                  |
| 8134   | Der Datenbaustein ist schreibgeschützt   |
| 8135   | Der Datenbaustein ist kein Array-DB  |
| 8282   | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350   | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352   | Codeerzeugungsfehler   |
| 8353   | Codeerzeugungsfehler   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben

### Beschreibung

Mit der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben" schreiben Sie Daten in einen Array-DB im Ladespeicher.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Wenn der Array-DB mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalfanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalfanke erfasst wird, ist die Anweisung beendet und der Wert des Parameters VALUE wird in den Datenbaustein geschrieben. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| REQ       | Input       | BOOL     | E, A, M, D, L   | REQ = "1": Mit dem Schreiben in den Array-DB beginnen  |
| DB        | Input       | DB_ANY   | D               | Datenbaustein, in den geschrieben wird   |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element im DB, in das geschrieben wird   |
| VALUE     | Input       | VARIANT  | E, A, M, D, L   | Wert, der geschrieben wird   |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | BUSY = "1": Das Schreiben in den Array-DB ist noch nicht abgeschlossen   |
| DONE      | Output      | BOOL     | E, A, M, D, L   | DONE = "1": Die Anweisung wurde erfolgreich ausgeführt   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".



## Parameter

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 80B4   | Datentypen stimmen nicht überein   |
| 80B5   | Kopiervorgang wurde abgebrochen  |
| 8131   | Der Datenbaustein existiert nicht  |
| 8132   | Der Datenbaustein ist zu kurz  |
| 8133   | Der Datenbaustein liegt nicht im Ladespeicher                                  |
| 8134   | Der Datenbaustein ist schreibgeschützt   |
| 8135   | Der Datenbaustein ist kein Array-DB  |
| 8282   | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350   | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352   | Codeerzeugungsfehler   |
| 8353   | Codeerzeugungsfehler   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## Weitere

### BLKMOV: Bereich kopieren

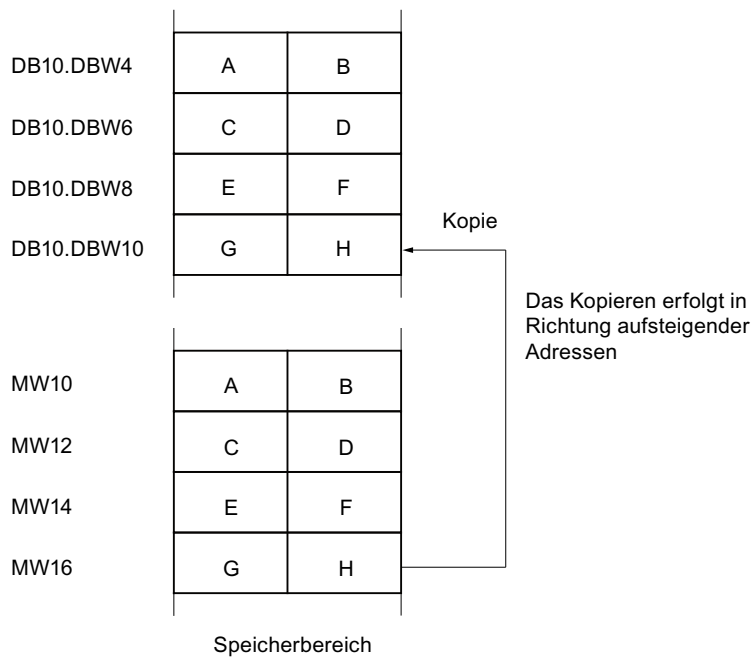
#### Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

#### Hinweis

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:



### Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben. Andernfalls ist die Konsistenz der Zieldaten nicht gewährleistet.

### Unterbrechbarkeit

Es gibt keine Begrenzung der Schachtelungstiefe.

### Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

### Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

## Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

| Parameter | Deklaration | Datentyp | Speicherbereich  |                     | Beschreibung   |
|-----------|-------------|----------|------------------|---------------------|--|
|           |             |          | S7-1200          | S7-1500             |  |
| EN        | Input       | BOOL     | E, A, M, D, L    | E, A, M, D, L, T, Z | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L    | E, A, M, D, L       | Freigabeausgang  |
| SRCBLK    | Input       | VARIANT  | E, A, M, D, L    | E, A, M, D, L       | Angabe des Speicherbereichs, der kopiert wird (Quellbereich).  |
| RET_VAL   | Output      | INT      | E, A, M, D, L, P | E, A, M, D, L, P    | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |
| DSTBLK    | Output      | VARIANT  | E, A, M, D, L    | E, A, M, D, L       | Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).  |

### Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 8091   | Die zulässige Schachtelungstiefe wurde überschritten   |
| 8092   | Die Anweisung kann nicht ausgeführt werden, weil ein angegebener Datenbaustein schreibgeschützt, nicht ablauffähig oder nicht geladen ist. |
| allgemeine Fehlerinformation   | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

### UBLKMOV: Bereich ununterbrechbar kopieren

#### Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

---

#### Hinweis

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

---

## Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

## Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

## Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichlänge "1" angegeben werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

| Parameter | Deklaration | Datentyp | Speicherbereich |                        | Beschreibung    |
|-----------|-------------|----------|-----------------|------------------------|-----------------|
|           |             |          | S7-1200         | S7-1500                |                 |
| EN        | Input       | BOOL     | E, A, M, D, L   | E, A, M, D, L,<br>T, Z | Freigabeeingang |
| ENO       | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L          | Freigabeausgang |

| Parameter | Deklaration | Datentyp | Speicherbereich  |                  | Beschreibung   |
|-----------|-------------|----------|------------------|------------------|--|
|           |             |          | S7-1200          | S7-1500          |  |
| SRCBLK    | Input       | VARIANT  | E, A, M, D, L    | E, A, M, D, L    | Angabe des Speicherbereichs, der kopiert wird (Quellbereich).  |
| RET_VAL   | Output      | INT      | E, A, M, D, L, P | E, A, M, D, L, P | Fehlerinformation:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |
| DSTBLK    | Output      | VARIANT  | E, A, M, D, L    | E, A, M, D, L    | Angabe des Speicherbereichs, in den kopiert wird (Zielbereich).  |

### Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode (W#16#...)  | Erläuterung   |
|--|---|
| 0000   | Kein Fehler   |
| 8091   | Der Quellbereich liegt in einem nicht ablaufrelevanten Datenbaustein. |
| allgemeine Fehlerinformation   | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"                    |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

### FILL: Bereich befüllen

#### Beschreibung

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen"

kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

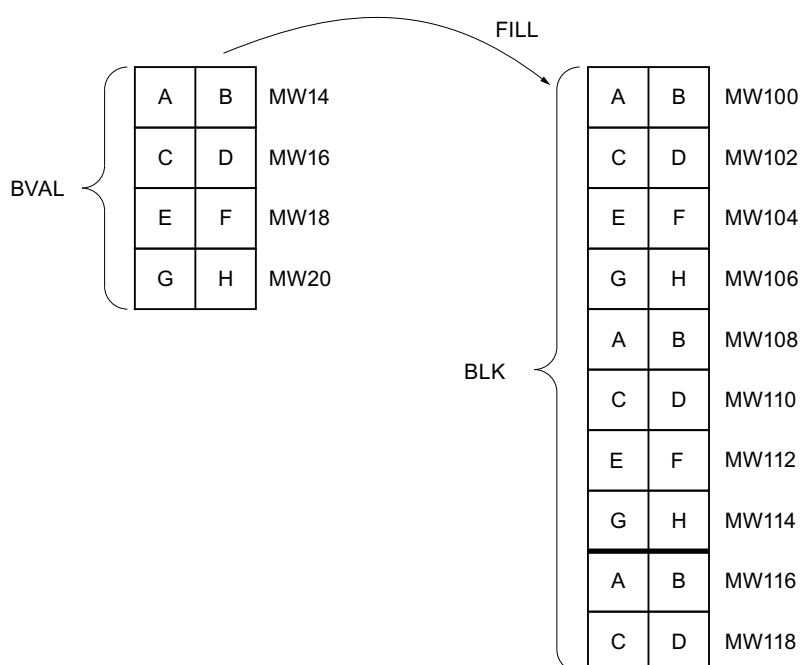
Den Quell- und Zielbereich definieren Sie durch VARIANT.

### Hinweis

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

Für Bausteine mit dem Attribut "optimierter Bausteinzugriff" können Sie die Anweisung "FILL\_BLK: Bereich befüllen" verwenden.

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:



### Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

### Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker

- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

### Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrisierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss er absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| BVAL      | Input       | VARIANT  | E, A, M, D, L   | Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird.                                 |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |
| BLK       | Output      | VARIANT  | E, A, M, D, L   | Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.  |



## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| allgemeine Fehlerinformation   | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen" |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)  
 Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)  
 GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

### Umwandler

#### CONVERT: Wert konvertieren

#### Beschreibung

Mit der Anweisung "Wert konvertieren" lesen Sie den Inhalt des Parameters IN und konvertieren ihn entsprechend auf den in der Anweisungsbox parametrisierten Datentypen. Der konvertierte Wert wird am Ausgang OUT ausgegeben.

Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.
- Für den Eingang IN ist in der Anweisungsbox ein Operand vom Datentyp BYTE, WORD, DWORD oder LWORD parametrisiert, dessen hochwertigstes Bit gesetzt ist. Am Ausgang OUT ist in der Anweisungsbox eine vorzeichenbehaftete Ganzzahl (SINT, INT, DINT, LINT) angegeben, die die gleiche Bitlänge wie der Operand am Eingang IN hat.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Wert konvertieren":

| Parameter | Deklaration | Datentyp  | Speicherbereich                 |                                 | Beschreibung                |
|-----------|-------------|---|---------------------------------|---------------------------------|-----------------------------|
|           |             |   | S7-1200                         | S7-1500                         |                             |
| EN        | Input       | BOOL  | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang             |
| ENO       | Output      | BOOL  | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang             |
| IN        | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, CHAR, BCD16, BCD32 | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Wert, der konvertiert wird. |
| OUT       | Output      | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, CHAR, BCD16, BCD32 | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis der Konvertierung  |

Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Bitfolgen (BYTE, WORD, DWORD, LWORD) sind in der Anweisungsbox nicht auswählbar. Wenn Sie einen Operanden vom Datentyp BYTE, WORD, DWORD oder LWORD an einem Parameter der Anweisung angeben, wird der Wert des Operanden als eine vorzeichenlose Ganzzahl mit der gleichen Bitlänge interpretiert. Der Datentyp BYTE wird in diesem Fall als USINT, WORD als UINT, DWORD als UDINT und LWORD als LINT interpretiert.

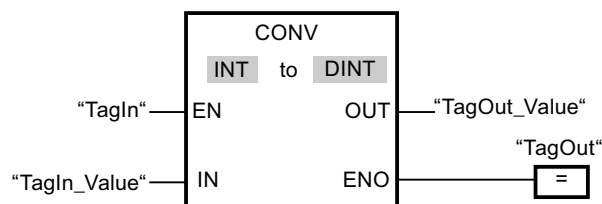
**Hinweis**

Für CPU S7-1500 gilt: Es können die Datentypen DWORD und LWORD ausgewählt werden, wenn als IN-Datentyp REAL oder LREAL ausgewählt wurde.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird der Inhalt des Operanden "TagIn\_Value" gelesen und in eine Ganzzahl (16 Bit) umgewandelt. Das Ergebnis wird im

Operanden "TagOut\_Value" abgelegt. Der Ausgang "TagOut" wird auf "1" gesetzt, wenn die Anweisung fehlerfrei ausgeführt wurde.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ROUND: Zahl runden

### Beschreibung

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die am nächsten liegende Ganzzahl um. Wenn der Eingangswert zwischen zwei Zahlen liegt, wird die gerade Zahl gewandelt. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zahl runden":

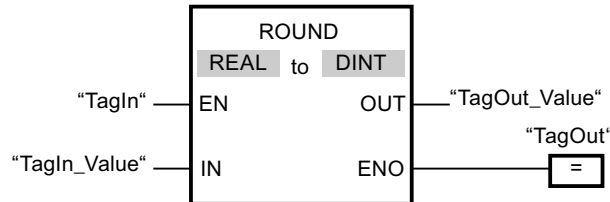
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 |                                 | Beschreibung                     |
|-----------|-------------|------------------------------|---------------------------------|---------------------------------|----------------------------------|
|           |             |                              | S7-1200                         | S7-1500                         |                                  |
| EN        | Input       | BOOL                         | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                  |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                  |
| IN        | Input       | Gleitpunktzahlen             | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert, der gerundet wird. |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis der Rundung             |

Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert       |             |
|-----------|--------------|------------|-------------|
| IN        | TagIn_Value  | 1.50000000 | -1.50000000 |
| OUT       | TagOut_Value | 2          | -2          |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Zahl runden" ausgeführt. Die Gleitpunktzahl am Eingang "TagIn\_Value" wird zur nächstliegenden geraden Ganzzahl gerundet und am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen**

**Beschreibung**

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst höhere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und kann an diesem abgefragt werden. Der Ausgangswert kann dabei größer oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen":

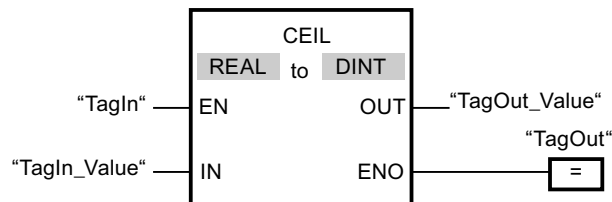
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 |                                 | Beschreibung                         |
|-----------|-------------|------------------------------|---------------------------------|---------------------------------|--------------------------------------|
|           |             |                              | S7-1200                         | S7-1500                         |                                      |
| EN        | Input       | BOOL                         | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                      |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                      |
| IN        | Input       | Gleitpunktzahlen             | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert als Gleitpunktzahl      |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis mit nächst höherer Ganzzahl |

Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert       |             |
|-----------|--------------|------------|-------------|
| IN        | TagIn_Value  | 0.50000000 | -0.50000000 |
| OUT       | TagOut_Value | 1          | 0           |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" ausgeführt. Die Gleitpunktzahl am Eingang "TagIn\_Value" wird zur nächst höheren Ganzzahl gerundet und am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**FLOOR: Aus Gleitpunktzahl nächst niedrigere Ganzzahl erzeugen**

**Beschreibung**

Mit der Anweisung "Aus Gleitpunktzahl nächst niedrigere Ganzzahl erzeugen" runden Sie den Wert am Eingang IN zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt diese in die nächst niedrigere Ganzzahl um. Das Ergebnis der Anweisung wird am Ausgang OUT ausgegeben und an diesem abgefragt werden. Der Ausgangswert kann dabei kleiner oder gleich dem Eingangswert sein.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Gleitpunktzahl nächst niedrigere Ganzzahl erzeugen":

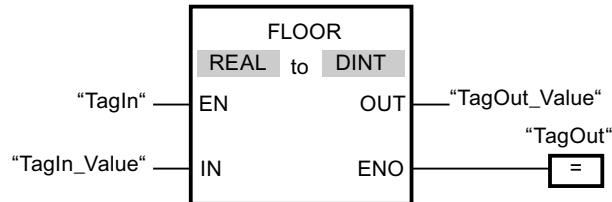
| Parameter | Deklaration | Datentyp                     | Speicherbereich                 |                                 | Beschreibung                             |
|-----------|-------------|------------------------------|---------------------------------|---------------------------------|--|
|           |             |                              | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL                         | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                          |
| ENO       | Output      | BOOL                         | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                          |
| IN        | Input       | Gleitpunktzahlen             | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert als Gleitpunktzahl          |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis mit nächst niedrigerer Ganzzahl |

Aus den Klapplisten "<???">" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert       |             |
|-----------|--------------|------------|-------------|
| IN        | TagIn_Value  | 0.50000000 | -0.50000000 |
| OUT       | TagOut_Value | 0          | -1          |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" ausgeführt. Die Gleitpunktzahl am Eingang "TagIn\_Value" wird zur nächst niedrigeren Ganzzahl gerundet und am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## TRUNC: Ganzzahl erzeugen

### Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Wert am Eingang IN eine Ganzzahl. Der Wert am Eingang IN wird als Gleitpunktzahl interpretiert. Die Anweisung selektiert nur den ganzzahligen Anteil der Gleitpunktzahl und gibt diesen ohne die Nachkommastellen am Ausgang OUT aus.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Es treten Fehler während der Bearbeitung auf, z. B. es tritt ein Überlauf auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Ganzzahl erzeugen":

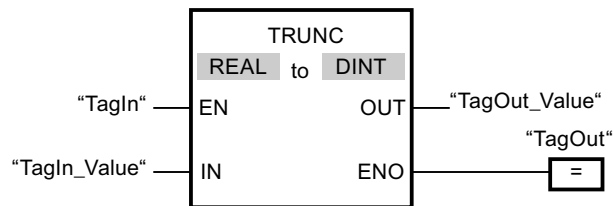
| Parameter | Deklaration | Datentyp                     | Speicherbereich              | Beschreibung  |
|-----------|-------------|------------------------------|------------------------------|---|
| EN        | Input       | BOOL                         | E, A, M, D, L                | Freigabeeingang                                     |
| ENO       | Output      | BOOL                         | E, A, M, D, L                | Freigabeausgang                                     |
| IN        | Input       | Gleitpunktzahlen             | E, A, M, D, L oder Konstante | Eingangswert als Gleitpunktzahl                     |
| OUT       | Output      | Ganzzahlen, Gleitpunktzahlen | E, A, M, D, L                | Ergebnis mit ganzzahligem Anteil der Gleitpunktzahl |

Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert       |             |
|-----------|--------------|------------|-------------|
| IN        | TagIn_Value  | 1.50000000 | -1.50000000 |
| OUT       | TagOut_Value | 1          | -1          |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Ganzzahl erzeugen" ausgeführt. Der ganzzahlige Anteil der Gleitpunktzahl am Eingang "TagIn\_Value" wird in eine Ganzzahl umgewandelt und am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, wird der Ausgang "TagOut" gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

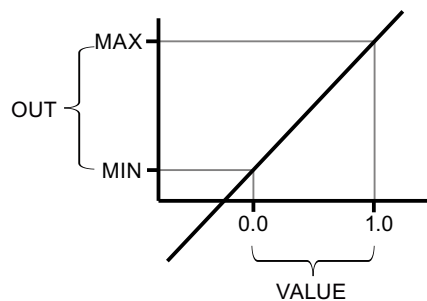


## SCALE\_X: Skalieren

### Beschreibung

Mit der Anweisung "Skalieren" skalieren Sie den Wert am Eingang VALUE, indem Sie diesen auf einem bestimmten Wertebereich abbilden. Bei der Ausführung der Anweisung "Skalieren" wird der Gleitpunktwert am Eingang VALUE auf den Wertebereich skaliert, der durch die Parameter MIN und MAX definiert wurde. Das Ergebnis der Skalierung ist eine Ganzzahl, die am Ausgang OUT abgelegt wird.

Das folgende Bild zeigt beispielhaft, wie Werte skaliert werden können:



Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

$$\text{OUT} = [\text{VALUE} * (\text{MAX} - \text{MIN})] + \text{MIN}$$

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang MIN ist größer oder gleich dem Wert am Eingang MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Es tritt ein Überlauf auf.
- Der Wert am Eingang VALUE ist NaN (Not a number = Ergebnis einer ungültigen Rechenoperation).

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

| Parameter | Deklaration | Datentyp                        | Speicherbereich                 | Beschreibung   |
|-----------|-------------|---------------------------------|---------------------------------|--|
| EN        | Input       | BOOL                            | E, A, M, D, L                   | Freigabeeingang  |
| ENO       | Output      | BOOL                            | E, A, M, D, L                   | Freigabeausgang  |
| MIN       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Untere Grenze<br>des Wertebereichs   |
| VALUE     | Input       | Gleitpunktzahlen                | E, A, M, D, L oder<br>Konstante | Wert, der skaliert<br>wird.<br>Bei Angabe einer<br>Konstante<br>müssen Sie diese<br>deklarieren. |
| MAX       | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Obere Grenze des<br>Wertebereichs  |
| OUT       | Output      | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L                   | Ergebnis der<br>Skalierung   |

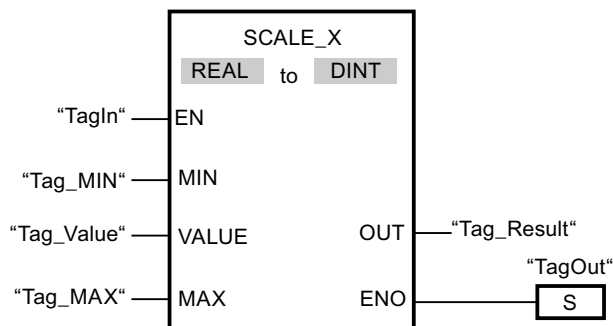
Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert |
|-----------|------------|------|
| MIN       | Tag_MIN    | 10   |
| VALUE     | Tag_Value  | 0.5  |
| MAX       | Tag_MAX    | 30   |
| OUT       | Tag_Result | 20   |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Skalieren" ausgeführt. Der Wert am Eingang "Tag\_Value" wird auf den Wertebereich skaliert, der durch die Werte an den Eingängen "Tag\_MIN" und "Tag\_MAX" definiert wurde. Das Ergebnis wird am Ausgang "Tag\_Result" abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

NORM\_X: Normieren (Seite 2077)

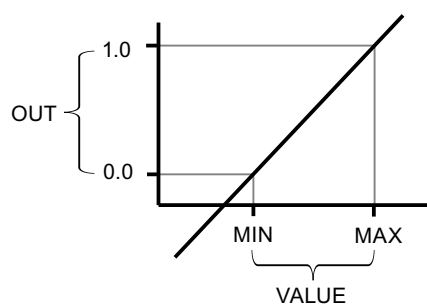
Konstanten deklarieren (Seite 1202)

## NORM\_X: Normieren

### Beschreibung

Mit der Anweisung "Normieren" normieren Sie den Wert der Variablen am Eingang VALUE, indem Sie diesen auf einer linearen Skala abbilden. Mit den Parametern MIN und MAX definieren Sie die Grenzen des Wertebereichs, der auf die Skala gespiegelt wird. Abhängig von der Lage des zu normierenden Werts in diesem Wertebereich wird das Ergebnis berechnet und am Ausgang OUT als Gleitpunktzahl abgelegt. Wenn der zu normierende Wert gleich dem Wert am Eingang MIN ist, liefert der Ausgang OUT den Wert "0.0". Wenn der zu normierende Wert gleich dem Wert am Eingang MAX ist, liefert Ausgang OUT den Wert "1.0".

Das folgende Bild zeigt beispielhaft, wie Werte normiert werden können:



Die Anweisung "Normieren" arbeitet mit der folgenden Gleichung:

$$OUT = (VALUE - MIN) / (MAX - MIN)$$

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert am Eingang MIN ist größer oder gleich dem Wert am Eingang MAX.
- Der Wert einer angegebenen Gleitpunktzahl liegt außerhalb des Bereichs der normalisierten Zahlen nach IEEE-754.
- Der Wert am Eingang VALUE ist NaN (Ergebnis einer ungültigen Rechenoperation).

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Normieren":

| Parameter   | Deklaration | Datentyp                        | Speicherbereich                 | Beschreibung                       |
|---|-------------|---------------------------------|---------------------------------|------------------------------------|
| EN  | Input       | BOOL                            | E, A, M, D, L                   | Freigabeeingang                    |
| ENO   | Output      | BOOL                            | E, A, M, D, L                   | Freigabeausgang                    |
| MIN <sup>1)</sup>   | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Untere Grenze<br>des Wertebereichs |
| VALUE <sup>1)</sup>   | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Wert, der normiert<br>wird.        |
| MAX <sup>1)</sup>   | Input       | Ganzzahlen,<br>Gleitpunktzahlen | E, A, M, D, L oder<br>Konstante | Obere Grenze des<br>Wertebereichs  |
| OUT   | Output      | Gleitpunktzahlen                | E, A, M, D, L                   | Ergebnis der<br>Normierung         |
| <sup>1)</sup> Bei Verwendung von Konstanten an diesen drei Parametern genügt es, wenn Sie eine davon deklarieren. |             |                                 |                                 |                                    |

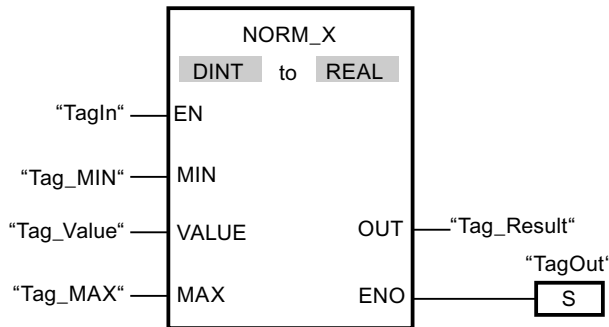
Aus den Klapplisten "<???" der Anweisungsbox können Sie die Datentypen der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert |
|-----------|------------|------|
| MIN       | Tag_MIN    | 10   |
| VALUE     | Tag_Value  | 20   |
| MAX       | Tag_MAX    | 30   |
| OUT       | Tag_Result | 0.5  |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Normieren" ausgeführt. Der Wert am Eingang "Tag\_Value" wird dem Wertebereich zugeordnet, der durch die Werte an den Eingängen "Tag\_MIN" und "Tag\_MAX" definiert wurde. Entsprechend dem definierten Wertebereich wird der Variablenwert am Eingang "Tag\_Value" normiert. Das Ergebnis wird am Ausgang "Tag\_Result" als Gleitpunktzahl abgelegt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- SCALE\_X: Skalieren (Seite 2075)
- Konstanten deklarieren (Seite 1202)

Weitere

**SCALE: Skalieren**

**Beschreibung**

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf den der Eingabewert skaliert wird, legen Sie durch die Parameter LO\_LIM und HI\_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

$$OUT = [((FLOAT (IN) - K1)/(K2-K1)) * (HI\_LIM-LO\_LIM)] + LO\_LIM$$

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI\_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO\_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO\_LIM > HI\_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Skalieren":

| Parameter | Deklaration | Datentyp | Speicherbereich                 |                                 | Beschreibung                    |
|-----------|-------------|----------|---------------------------------|---------------------------------|---------------------------------|
|           |             |          | S7-1200                         | S7-1500                         |                                 |
| EN        | Input       | BOOL     | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                 |
| ENO       | Output      | BOOL     | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                 |
| IN        | Input       | INT      | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingabewert, der skaliert wird. |

| Parameter | Deklaration | Datentyp | Speicherbereich                    |                                    | Beschreibung  |
|-----------|-------------|----------|------------------------------------|------------------------------------|---|
|           |             |          | S7-1200                            | S7-1500                            |   |
| HI_LIM    | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | E, A, M, D, L, P<br>oder Konstante | Oberer<br>Grenzwert   |
| LO_LIM    | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | E, A, M, D, L, P<br>oder Konstante | Unterer<br>Grenzwert  |
| BIPOLAR   | Input       | BOOL     | E, A, M, D, L<br>oder Konstante    | E, A, M, D, L<br>oder Konstante    | Gibt an, ob der<br>Wert am<br>Parameter IN<br>als bipolar<br>oder unipolar<br>interpretiert<br>wird. Der<br>Parameter<br>kann die<br>folgenden<br>Werte<br>annehmen:<br>1: Bipolar<br>0: Unipolar |
| OUT       | Output      | REAL     | E, A, M, D, L, P                   | E, A, M, D, L, P                   | Ergebnis der<br>Anweisung   |
| RET_VAL   | Output      | WORD     | E, A, M, D, L, P                   | E, A, M, D, L, P                   | Fehlerinformati<br>onen   |

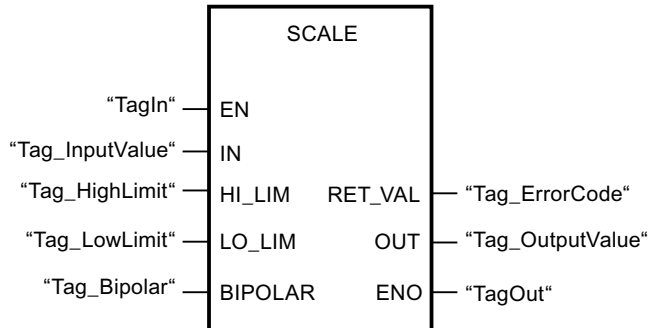
## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung   |
|--|---|
| 0000   | Kein Fehler   |
| 0008   | Der Wert des Parameters IN ist größer als der Wert der Konstante "K2" oder kleiner als der Wert der Konstante "K1". |
| allgemeine<br>Fehlerinformati<br>on  | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"  |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand         | Wert        |
|-----------|-----------------|-------------|
| IN        | Tag_InputValue  | 22          |
| HI_LIM    | Tag_HighLimit   | 100.0       |
| LO_LIM    | Tag_LowLimit    | 0.0         |
| BIPOLAR   | Tag_Bipolar     | 1           |
| OUT       | Tag_OutputValue | 50.03978588 |
| RET_VAL   | Tag_ErrorCode   | W#16#0000   |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

**UNSCALE: Deskalieren**

**Beschreibung**

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandelt sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf den der Eingabewert deskaliert wird, legen Sie durch die Parameter LO\_LIM und HI\_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

$$OUT = [ ((IN-LO\_LIM)/(HI\_LIM-LO\_LIM)) * (K2-K1) ] + K1$$

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:



- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert des oberen Grenzwert (HI\_LIM) ist, wird das Ergebnis der Anweisung auf den Wert der Konstante "K2" gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante des unteren Grenzwert (LO\_LIM) ist, wird das Ergebnis der Anweisung auf den Wert der Konstante "K1" gesetzt und ein Fehler ausgegeben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Deskalisieren":

| Parameter | Deklaration | Datentyp | Speicherbereich                 |                                 | Beschreibung   |
|-----------|-------------|----------|---------------------------------|---------------------------------|--|
|           |             |          | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL     | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang  |
| ENO       | Input       | BOOL     | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang  |
| IN        | Input       | REAL     | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingabewert, der in einen ganzzahligen Wert deskaliert wird.   |
| HI_LIM    | Input       | REAL     | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Oberer Grenzwert   |
| LO_LIM    | Input       | REAL     | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Unterer Grenzwert  |
| BIPOLAR   | Input       | BOOL     | E, A, M, D, L oder Konstante    | E, A, M, D, L oder Konstante    | Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen:<br>1: Bipolar<br>0: Unipolar |

| Parameter | Deklaration | Datentyp | Speicherbereich  |                  | Beschreibung           |
|-----------|-------------|----------|------------------|------------------|------------------------|
|           |             |          | S7-1200          | S7-1500          |                        |
| OUT       | Output      | INT      | E, A, M, D, L, P | E, A, M, D, L, P | Ergebnis der Anweisung |
| RET_VAL   | Output      | WORD     | E, A, M, D, L, P | E, A, M, D, L, P | Fehlerinformationen    |

### Parameter RET\_VAL

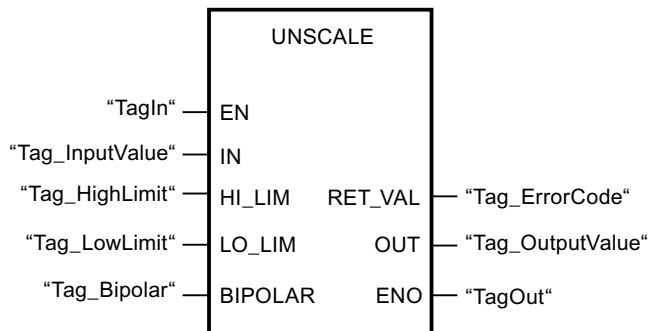
Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode* (W#16#...)       | Erläuterung  |
|------------------------------|--|
| 0000                         | Kein Fehler  |
| 0008                         | Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM). |
| allgemeine Fehlerinformation | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"   |

\*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand         | Wert        |
|-----------|-----------------|-------------|
| IN        | Tag_InputValue  | 50.03978588 |
| HI_LIM    | Tag_HighLimit   | 100.0       |
| LO_LIM    | Tag_LowLimit    | 0.0         |
| BIPOLAR   | Tag_Bipolar     | 1           |
| OUT       | Tag_OutputValue | 22          |
| RET_VAL   | Tag_ErrorCode   | W#16#0000   |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)  
Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)  
GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)

## Programmsteuerung

### JMP: Springen bei VKE = 1

#### Beschreibung

Mit der Anweisung "Springen bei VKE = 1" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisungsbox angegeben.

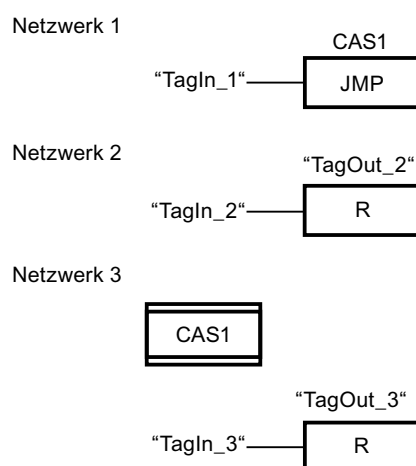
Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn die Bedingung am Eingang der Anweisung nicht erfüllt ist (VKE = 0), wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

#### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn\_1" den Signalzustand "1" liefert, wird die Anweisung "Springen bei VKE = 1" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und

im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn\_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut\_3" zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**JMPN: Springen bei VKE = 0**

**Beschreibung**

Mit der Anweisung "Springen bei VKE = 0" unterbrechen Sie die lineare Bearbeitung des Programms und setzen sie in einem anderen Netzwerk fort, wenn das Verknüpfungsergebnis am Eingang der Anweisung "0" ist. Das Zielnetzwerk muss durch eine Sprungmarke (LABEL) gekennzeichnet werden. Die Bezeichnung der Sprungmarke wird in den Platzhalter oberhalb der Anweisungsbox angegeben.

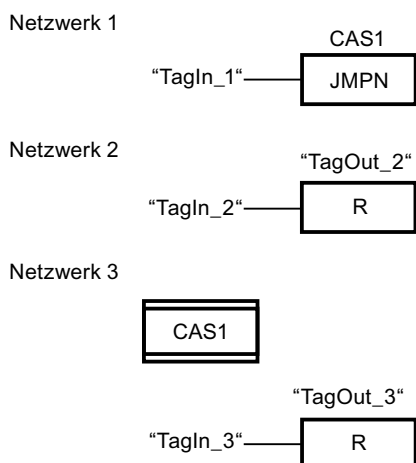
Die angegebene Sprungmarke muss im gleichen Baustein liegen, in dem die Anweisung ausgeführt wird. Ihre Bezeichnung darf nur einmal im Baustein vergeben sein. Innerhalb eines Netzwerks darf es nur eine springende Spule geben.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "0" ist, wird der Sprung in das Netzwerk ausgeführt, das durch die angegebene Sprungmarke gekennzeichnet ist. Der Sprung kann in Richtung höherer oder niedrigerer Netzwerknummern erfolgen.

Wenn das VKE am Eingang der Anweisung "1" ist, wird die Programmbearbeitung im nächsten Netzwerk fortgesetzt.

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn\_1" den Signalzustand "0" liefert, wird die Anweisung "Springen bei VKE = 0" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn\_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut\_3" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

## LABEL: Sprungmarke

### Beschreibung

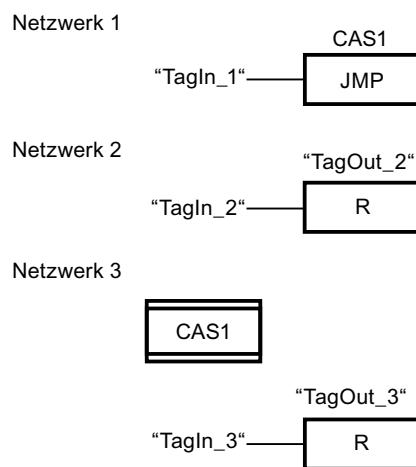
Die Sprungmarke kennzeichnet ein Zielnetzwerk, in dem nach einer ausgeführten Sprunganweisung die Programmbearbeitung fortgesetzt werden soll.

Die Sprungmarke und die Anweisung, in der die Sprungmarke als Sprungziel angegeben ist, müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb des Bausteins nur einmal vergeben werden. Sie können maximal 32 Sprungmarken bei Verwendung einer CPU S7-1200 und maximal 256 Sprungmarken bei Verwendung einer CPU S7-1500 deklarieren.

In einem Netzwerk kann nur eine Sprungmarke platziert werden. Jede Sprungmarke kann von mehreren Stellen angesprungen werden.

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn der Operand "TagIn\_1" den Signalzustand "1" liefert, wird die Anweisung "Springen bei VKE = 1" ausgeführt. Die lineare Bearbeitung des Programms wird dadurch unterbrochen und im Netzwerk 3 fortgesetzt, das durch die Sprungmarke CAS1 gekennzeichnet ist. Wenn der Eingang "TagIn\_3" den Signalzustand "1" liefert, wird der Ausgang "TagOut\_3" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

## JMP\_LIST: Sprungliste definieren

### Beschreibung

Mit der Anweisung "Sprungliste definieren" definieren Sie mehrere bedingte Sprünge und setzen die Programmbearbeitung abhängig vom Wert des Parameters K in einem definierten Zielnetzwerk fort.

Die Sprünge definieren Sie durch Sprungmarken (LABEL), die Sie an den Ausgängen der Anweisungsbox angeben. Die Anweisungsbox enthält im Grundzustand min. 2 Ausgänge (DEST0 und DEST1). Die Anzahl der Ausgänge ist erweiterbar. Sie können maximal 32 Ausgänge bei Verwendung einer CPU S7-1200 und maximal 256 Ausgänge bei Verwendung einer CPU S7-1500 deklarieren.

Die Nummerierung der Ausgänge beginnt mit dem Wert "0" und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. An den Ausgängen der Anweisung können nur Sprungmarken angegeben werden. Die Angabe von Anweisungen oder Operanden ist nicht zulässig.

Durch den Wert des Parameters K geben Sie die Nummer des Ausgangs und damit die Sprungmarke an, bei der die Programmbearbeitung fortgesetzt werden soll. Wenn der Wert am Parameter K größer als die Anzahl der verfügbaren Ausgänge ist, wird die lineare Bearbeitung des Programms nicht unterbrochen, sondern im nächsten Netzwerk fortgesetzt.

### Parameter

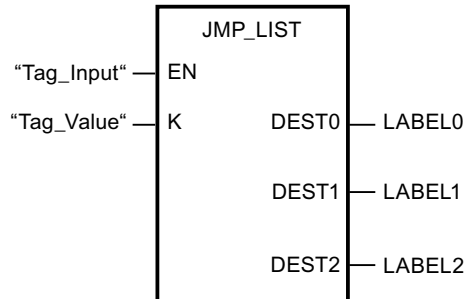
Die folgende Tabelle zeigt die Parameter der Anweisung "Sprungliste definieren":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, L, D                | Freigabeeingang  |
| K         | Input       | UINT     | E, A, M, L, D oder Konstante | Gibt die Nummer des Ausgangs und damit den Sprung an, der ausgeführt wird. |
| DEST0     | -           | -        | -                            | Erste Sprungmarke  |
| DEST1     | -           | -        | -                            | Zweite Sprungmarke   |
| DESTn     | -           | -        | -                            | Optionale Sprungmarken   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand/Sprungmarke | Wert   |
|-----------|---------------------|--|
| K         | Tag_Value           | 1  |
| DEST0     | LABEL0              | Sprung in das Netzwerk, das mit der Sprungmarke "LABEL0" gekennzeichnet ist. |
| DEST1     | LABEL1              | Sprung in das Netzwerk, das mit der Sprungmarke "LABEL1" gekennzeichnet ist. |
| DEST2     | LABEL2              | Sprung in das Netzwerk, das mit der Sprungmarke "LABEL2" gekennzeichnet ist. |

Wenn der Operand "Tag\_Input" den Signalzustand "1" liefert, wird die Anweisung "Sprungliste definieren" ausgeführt. Die Programmbearbeitung wird entsprechend dem Wert des Operanden "Tag\_Value" in dem Netzwerk fortgesetzt, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

## SWITCH: Sprungverteiler

### Beschreibung

Mit der Anweisung "Sprungverteiler" definieren Sie mehrere Programmsprünge, die abhängig vom Ergebnis einer oder mehrer Vergleichsanweisungen ausgeführt werden.

Den zu vergleichenden Wert geben Sie am Parameter K an. Dieser Wert wird mit den Werten verglichen, die die einzelnen Eingänge liefern. Die Art des Vergleichs wählen Sie für jeden

Eingang einzeln aus. Die Verfügbarkeit der verschiedenen Vergleichsanweisungen hängt dabei vom Datentyp der Anweisung ab.

Die folgende Tabelle zeigt die Vergleichsanweisungen, die abhängig vom gewählten Datentyp verfügbar sind:

| Datentyp  |  | Anweisung      | Syntax |
|---|--|----------------|--------|
| S7-1200   | S7-1500  |                |        |
| Bitfolgen   | Bitfolgen  | Gleich         | ==     |
|   |  | Ungleich       | <>     |
| Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE, TOD | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME, DATE,<br>TOD, LTOD, LDT | Gleich         | ==     |
|   |  | Ungleich       | <>     |
|   |  | Größer gleich  | >=     |
|   |  | Kleiner gleich | <=     |
|   |  | Größer         | >      |
|   |  | Kleiner        | <      |

Den Datentyp der Anweisung wählen Sie aus der Klappliste "<???" der Anweisungsbox. Wenn Sie eine Vergleichsanweisung auswählen und der Datentyp der Anweisung noch nicht festgelegt ist, werden in der Klappliste "<???" nur die Datentypen aufgelistet, die für die ausgewählte Vergleichsanweisung zulässig sind.

Die Bearbeitung der Anweisung beginnt mit dem ersten Vergleich und wird so lange ausgeführt, bis eine Vergleichsbedingung erfüllt ist. Ist eine Vergleichsbedingung erfüllt, werden die nachfolgenden Vergleichsbedingungen nicht berücksichtigt. Wenn keine der angegebenen Vergleichsbedingungen erfüllt ist, wird der Sprung am Ausgang ELSE ausgeführt. Wenn am Ausgang ELSE keine Sprungmarke definiert ist, wird die lineare Bearbeitung des Programms nicht unterbrochen, sondern im nächsten Netzwerk fortgesetzt.

Die Anweisungsbox enthält im Grundzustand min. 2 Ausgänge (DEST0 und DEST1). Die Anzahl der Ausgänge ist erweiterbar. Die Nummerierung der Ausgänge beginnt mit dem Wert "0" und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. An den Ausgängen der Anweisung geben Sie Sprungmarken (LABEL) an. Die Angabe von Anweisungen oder Operanden ist an den Ausgängen der Anweisung nicht zulässig.

Zu jedem zusätzlichen Ausgang wird automatisch auch ein Eingang eingefügt. Der an einem Ausgang programmierte Sprung wird ausgeführt, wenn die Vergleichsbedingung des entsprechenden Eingangs erfüllt ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Sprungverteiler":

| Parameter | Deklaration | Datentyp |         | Speicherbereich              | Beschreibung                        |
|-----------|-------------|----------|---------|------------------------------|-------------------------------------|
|           |             | S7-1200  | S7-1500 |                              |                                     |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L                | Freigabeeingang                     |
| K         | Input       | UINT     | UINT    | E, A, M, D, L oder Konstante | Gibt den zu vergleichenden Wert an. |

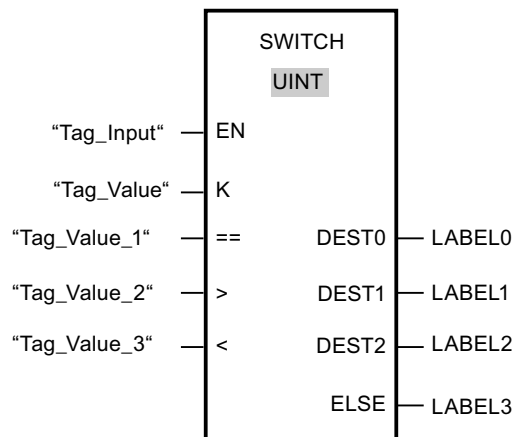


| Parameter         | Deklaration | Datentyp   |  | Speicherbereich              | Beschreibung  |
|-------------------|-------------|--|--|------------------------------|---|
|                   |             | S7-1200  | S7-1500  |                              |   |
| <Vergleichswerte> | Input       | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, DATE, TOD | Bitfolgen, Ganzzahlen, Gleitpunktzahlen, TIME, LTIME, DATE, TOD, LTOD, LDT | E, A, M, D, L oder Konstante | Eingangswerte, mit denen der Wert des Parameters K verglichen wird.                     |
| DEST0             | -           | -  | -  | -                            | Erste Sprungmarke   |
| DEST1             | -           | -  | -  | -                            | Zweite Sprungmarke  |
| DEST(n)           | -           | -  | -  | -                            | Optionale Sprungmarken (n = 2 bis 99)   |
| ELSE              | -           | -  | -  | -                            | Programmsprung, der ausgeführt wird, wenn keine der Vergleichsbedingungen erfüllt wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand/Sprungmarke | Wert |
|-----------|---------------------|------|
| K         | Tag_Value           | 23   |
| ==        | Tag_Value_1         | 20   |

| Parameter | Operand/Sprungmarke | Wert  |
|-----------|---------------------|---|
| >         | Tag_Value_2         | 21  |
| <         | Tag_Value_3         | 19  |
| DEST0     | LABEL0              | Sprung zur Sprungmarke "LABEL0", wenn der Wert des Parameters K gleich 20 ist.      |
| DEST1     | LABEL1              | Sprung zur Sprungmarke "LABEL1", wenn der Wert des Parameters K größer 21 ist.      |
| DEST2     | LABEL2              | Sprung zur Sprungmarke "LABEL2", wenn der Wert des Parameters K kleiner als 19 ist. |
| ELSE      | LABEL 3             | Sprung zur Sprungmarke "LABEL3", wenn keine der Vergleichsbedingungen erfüllt ist.  |

Wenn der Operand "Tag\_Input" auf den Signalzustand "1" wechselt, wird die Anweisung "Sprungverteiler" ausgeführt. Die Programmbearbeitung wird in dem Netzwerk fortgesetzt, das mit der Sprungmarke "LABEL1" gekennzeichnet ist.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

**RET: Zurück springen**

**Beschreibung**

Mit der Anweisung "Zurück springen" beenden Sie die Bearbeitung eines Bausteins. Hierdurch ergeben sich drei Arten, wie die Bearbeitung eines Bausteins beendet werden kann:

- Ohne Aufruf der Anweisung "Zurück springen"  
Nach der Ausführung des letzten Netzwerks wird der Baustein verlassen. Das ENO der Aufruffunktion wird auf den Signalzustand "1" gesetzt.
- Aufruf der Anweisung "Zurück springen" mit Vorverknüpfung (siehe Beispiel)  
Hat der linke Anschluss den Signalzustand "1", wird der Baustein verlassen. Das ENO der Aufruffunktion entspricht dem Operanden.
- Aufruf der Anweisung "Zurück springen" ohne Vorverknüpfung  
Der Baustein wird verlassen. Das ENO der Aufruffunktion entspricht dem Operanden.

**Hinweis**

In einem Netzwerk darf nur eine springende Spule verwendet werden ("Zurück springen", "Springen bei VKE = 1", "Springen bei VKE = 0").

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "Zurück springen" "1" ist, wird die Programmbearbeitung im aktuell aufgerufenen Baustein beendet und im aufrufenden Baustein (z. B. im aufrufenden OB) nach der Aufruffunktion fortgesetzt. Der Status (ENO) der Aufruffunktion wird durch den Parameter der Anweisung bestimmt. Dieser kann die folgenden Werte annehmen:

- RLO
- TRUE/FALSE
- <Operand>

Um die Werte des Parameters einzustellen, doppelklicken Sie auf die Anweisung und wählen Sie in der Klappliste den entsprechenden Wert aus.

Die folgende Tabelle zeigt den Status der Aufruffunktion, wenn die Anweisung "Zurück springen" in einem Netzwerk innerhalb des aufgerufenen Bausteins programmiert ist:

| VKE | Parameterwert | ENO der Aufruffunktion  |
|-----|---------------|---|
| 1   | RLO           | 1   |
|     | TRUE          | 1   |
|     | FALSE         | 0   |
|     | <Operand>     | <Operand>   |
| 0   | RLO           | Die Programmbearbeitung wird im nächsten Netzwerk des aufgerufenen Bausteins fortgesetzt. |
|     | TRUE          |   |
|     | FALSE         |   |
|     | <Operand>     |   |

Wenn ein OB beendet wird, dann wird durch das Ablaufebensystem ein anderer Baustein ausgewählt und gestartet bzw. weiter ausgeführt:

- Wenn der Programmzyklus-OB beendet wurde, wird dieser wieder gestartet.
- Wenn ein OB beendet wird, der einen anderen Baustein unterbrochen hat (z. B. ein Alarm-OB), so wird der unterbrochene Baustein (z. B. Programmzyklus-OB) ausgeführt.

## Parameter

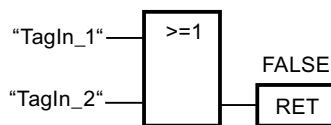
Die folgende Tabelle zeigt die Parameter der Anweisung "Zurück springen":

| Parameter                                    | Deklaration | Datentyp | Speicherbereich | Beschreibung                                |
|--|-------------|----------|-----------------|---|
| Status der aufrufenden Funktion bei VKE = 1: |             |          |                 |   |
| RLO  | -           | -        | -               | Wird auf den Signalzustand des VKE gesetzt. |
| TRUE   | -           | -        | -               | 1   |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                            |
|-----------|-------------|----------|-----------------|---|
| FALSE     | -           | -        | -               | 0                                       |
| <Operand> | Input       | BOOL     | E, A, M, D, L   | Signalzustand des angegebenen Operanden |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn einer der Operanden "TagIn\_1" oder "TagIn\_2" den Signalzustand "1" liefert, wird die Anweisung "Zurück springen" ausgeführt. Die Programmbearbeitung im aufgerufenen Baustein wird beendet und im aufrufenden Baustein fortgesetzt. Der Freigabeausgang ENO der Aufruffunktion wird auf den Signalzustand "0" zurückgesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**Laufzeitsteuerung**

**ENDIS\_PW: Passwort-Legitimierung einschränken und freigeben**

**Beschreibung**

Mit der Anweisung "Passwort-Legitimierung einschränken und freigeben" legen Sie fest, ob für die CPU eine Legitimation erlaubt ist oder nicht. Sie können legitimierte Verbindungen verhindern, auch wenn das korrekte Passwort bekannt ist. Wenn Sie die Anweisung aufrufen und der Parameter REQ den Signalzustand "0" hat, wird lediglich der momentan eingestellte Zustand an den Ausgangsparametern angezeigt, aber keine Einstellung verändert. Wenn der Parameter REQ den Signalzustand "1" hat, wird der Signalzustand von den Eingangsparametern (F\_PWD, FULL\_PWD, R\_PWD, HMI\_PWD) übernommen. FALSE bedeutet, dass die Legitimation per Passwort nicht erlaubt ist, TRUE bedeutet, das Passwort ist verwendbar. Das Sperren bzw. Freigeben der Passwörter kann einzeln erlaubt oder untersagt werden. Es können z. B. alle Passwörter nicht erlaubt sein, außer dem Failsafe-Passwort. So können Sie die Zugriffsmöglichkeiten auf eine kleine Anwendergruppe begrenzen. Die Ausgangsparameter (F\_PWD\_ON, FULL\_PWD\_ON, R\_PWD\_ON, HMI\_PWD\_ON) zeigen immer den aktuellen Status der Passwortverwendung, unabhängig vom Parameter REQ.

Die gleiche Einstellung kann im Frontpanel der CPU vorgenommen werden und die CPU speichert die jeweils letzte Einstellung.

Um ein ungewolltes Aussperren zu verhindern, kann bei einer S7-1500-CPU der Schutz durch ein Umlegen des Betriebsartenschalters nach STOP außer Kraft gesetzt werden. Der Schutz

wird nach dem Umlegen des Betriebsartenschalters in RUN automatisch wieder eingestellt, ohne dass die Anweisung "Passwort-Legitimierung einschränken und freigeben" erneut aufgerufen werden muss oder im Frontpanel weitere Aktionen notwendig sind.

Die Anweisung muss immer komplett durchführbar sein, d. h. es muss z. B. der Parameter F\_PWD immer den Signalzustand "0" haben, damit die Einstellungen abgespeichert werden können.

Unter den folgenden Bedingungen können gesperrte Passwörter wieder zugelassen werden:

- Die CPU wurde auf die Werkseinstellung zurückgesetzt.
- Das Frontpanel der S7-1500-CPU unterstützt einen Dialog, mit dessen Hilfe Sie zum passenden Menü navigieren und die Passwörter wieder zulassen können.
- Beim Aufruf der Anweisung "Passwort-Legitimierung einschränken und freigeben" hat der Eingangsparameter des gewünschten Passworts den Signalzustand "1".
- Den Betriebsartenschalter auf STOP setzen. Die Einschränkung der Passwort-Legitimierung wird wieder eingerichtet, sobald der Schalter wieder nach RUN bewegt wird.
- Das Stecken eines Transfermoduls in eine S7-1200-CPU.

---

#### Hinweis

Die Anweisung "Passwort-Legitimierung einschränken und freigeben" sperrt den Zugriff auf HMI Panels, wenn das HMI Passwort nicht freigegeben ist.

---

#### Hinweis

Bereits bestehende und legitimierte Verbindungen behalten ihre Zugriffs-Rechte und können über die Anweisung "Passwort-Legitimierung einschränken und freigeben" nicht eingeschränkt werden.

---

### Vorgehensweise bei verlorenem Passwort bei einer S7-1200-CPU

Wenn Sie das Passwort für eine passwortgeschützte CPU verloren haben, löschen Sie das passwortgeschützte Programm mit einer leeren Übertragungskarte. Die leere Übertragungskarte löscht den internen Ladespeicher der CPU. Dann können Sie ein neues Anwenderprogramm aus STEP 7 Basic in die CPU laden.

|  |
|--|
|  <b>WARNUNG</b> |
|--|

|                                  |
|----------------------------------|
| <b>Übertragungskarte stecken</b> |
|----------------------------------|

|  |
|--|
| Wenn Sie eine Übertragungskarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann. |
|--|

Sie müssen die Übertragungskarte ziehen, bevor Sie die CPU in RUN versetzen.

### Auswirkungen der Passwortverwendung auf die Betriebsarten

Die folgende Tabelle zeigt, welche Auswirkungen die Passwortverwendung durch die Anweisung "Passwort-Legitimierung einschränken und freigeben" auf die Betriebsarten und die entsprechenden Anwenderaktionen hat.

| Aktion  | Passwortschutz durch die Anweisung  |
|---|---|
| Grundzustand nach <ul style="list-style-type: none"> <li>Betriebsartenschalter auf STOP</li> <li>Speicher manuell zurücksetzen (PG, Schalter, Änderung des MC (Motion Control))</li> <li>Zurücksetzen auf Werkseinstellung</li> </ul> | Nicht aktiviert<br>(Keine Einschränkungen)  |
| Grundzustand nach NETZ-EIN  | Aktiviert (wenn vor NETZ-AUS eine Sperre aktiviert war)<br>Die Möglichkeit, Passwörter nicht zu erlauben, ist remanent. |
| Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch das Beenden der Anweisung, einen Fehler oder Kommunikation) oder STOP -> ANLAUF/ RUN/HALT   | Aktiviert<br>Die Passwörter dürfen weiterhin nicht verwendet werden.  |

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Passwort-Legitimierung einschränken und freigeben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L   | Freigabeausgang  |
| REQ       | Input       | BOOL     | E, A, M, D, L   | Wenn der Parameter REQ den Signalzustand "0" führt, wird der aktuell eingestellte Signalzustand der Passwörter abgefragt.  |
| F_PWD     | Input       | BOOL     | E, A, M, D, L   | Lese-/Schreibzugriff inklusive Failsafe <ul style="list-style-type: none"> <li>F_PWD = "0": Passwort nicht erlauben</li> <li>F_PWD = "1": Passwort erlauben</li> </ul> |
| FULL_PWD  | Input       | BOOL     | E, A, M, D, L   | Lese-/Schreibzugriff <ul style="list-style-type: none"> <li>FULL_PWD = "0": Passwort nicht erlauben</li> <li>FULL_PWD = "1": Passwort erlauben</li> </ul>              |

| Parameter   | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-------------|-------------|----------|-----------------|---|
| R_PWD       | Input       | BOOL     | E, A, M, D, L   | Lesezugriff <ul style="list-style-type: none"> <li>• R_PWD = "0": Passwort nicht erlauben</li> <li>• R_PWD = "1": Passwort erlauben</li> </ul>  |
| HMI_PWD     | Input       | BOOL     | E, A, M, D, L   | HMI-Zugriff <ul style="list-style-type: none"> <li>• HMI_PWD = "0": Passwort nicht erlauben</li> <li>• HMI_PWD = "1": Passwort erlauben</li> </ul>                                    |
| F_PWD_ON    | Output      | BOOL     | E, A, M, D, L   | Status Lese-/Schreibzugriff inklusive Failsafe <ul style="list-style-type: none"> <li>• F_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• F_PWD_ON = "1": Passwort erlaubt</li> </ul> |
| FULL_PWD_ON | Output      | BOOL     | E, A, M, D, L   | Status Lese-/Schreibzugriff <ul style="list-style-type: none"> <li>• FULL_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• FULL_PWD_ON = "1": Passwort erlaubt</li> </ul>              |
| R_PWD_ON    | Output      | BOOL     | E, A, M, D, L   | Status Lesezugriff <ul style="list-style-type: none"> <li>• R_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• R_PWD_ON = "1": Passwort erlaubt</li> </ul>                             |
| HMI_PWD_ON  | Output      | BOOL     | E, A, M, D, L   | Status HMI-Zugriff <ul style="list-style-type: none"> <li>• HMI_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• HMI_PWD_ON = "1": Passwort erlaubt</li> </ul>                         |
| RET_VAL     | Output      | WORD     | E, A, M, D, L   | Fehlerinformation   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 8090   | Die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird nicht unterstützt |
| 80D0   | Das Passwort für Failsafe ist nicht konfiguriert   |
| 80D1   | Der Lese-/Schreibzugriff ist nicht konfiguriert  |
| 80D2   | Der Lesezugriff ist nicht konfiguriert   |
| 80D3   | Der HMI-Zugriff ist nicht konfiguriert   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zur Fehlerbehandlung (Seite 1403)
- Grundlagen der lokalen Fehlerbehandlung (Seite 1405)
- Prioritäten bei der Fehlerausgabe (Seite 1406)
- Lokale Fehlerbehandlung für einen Baustein aktivieren (Seite 1407)

## RE\_TRIGR: Zyklusüberwachungszeit neu starten

### Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zykluszeitüberwachung der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann unabhängig von der Priorität in allen Bausteinen aufgerufen werden.

Wenn die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm, einem Diagnosealarm oder einem Weckalarm, aufgerufen wird, wird die Anweisung nicht ausgeführt und der Freigabeausgang ENO auf den Signalzustand "0" gesetzt.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann max. 10-mal in einem Programmzyklus aufgerufen werden.

### Parameter

Die Anweisung "Zyklusüberwachungszeit neu starten" hat keine Parameter.



**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zur Fehlerbehandlung (Seite 1403)
- Grundlagen der lokalen Fehlerbehandlung (Seite 1405)
- Prioritäten bei der Fehlerausgabe (Seite 1406)
- Lokale Fehlerbehandlung für einen Baustein aktivieren (Seite 1407)

**STP: Programm beenden****Beschreibung**

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOPP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang vom RUN in STOPP hängen von der CPU-Konfiguration ab.

Wenn das Verknüpfungsergebnis (VKE) am Eingang der Anweisung "1" ist, wechselt die CPU in den Betriebszustand STOPP und die Programmbearbeitung wird beendet. Der Signalzustand am Ausgang der Anweisung wird nicht ausgewertet.

Wenn das VKE am Eingang der Anweisung "0" ist, wird die Anweisung nicht ausgeführt.

**Parameter**

Die Anweisung "Programm beenden" hat keine Parameter.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)

**GET\_ERROR: Fehler lokal abfragen****Beschreibung**

Mit der Anweisung "Fehler lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Wenn das System während der Bausteinbearbeitung Fehler meldet, wird zum ersten aufgetretenen Fehler eine ausführliche Information im Operanden am Ausgang ERROR gespeichert.

Am Ausgang ERROR können nur Operanden vom Systemdatentyp "ErrorStruct" angegeben werden. Der Systemdatentyp "ErrorStruct" gibt die genaue Struktur vor, in der die Informationen zum aufgetretenen Fehler gespeichert werden. Mithilfe weiterer Anweisungen können Sie diese Struktur auswerten und eine entsprechende Reaktion programmieren. Wenn der erste aufgetretene Fehler behoben ist, gibt die Anweisung die Fehlerinformation zum nächsten aufgetretenen Fehler aus.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler lokal abfragen":

| Parameter | Deklaration | Datentyp    | Speicherbereich | Beschreibung        |
|-----------|-------------|-------------|-----------------|---------------------|
| ERROR     | Output      | ErrorStruct | D, L            | Fehlerinformationen |

**Datentyp "ErrorStruct"**

Die folgende Tabelle zeigt die Struktur des Datentyps "ErrorStruct":

| Strukturkomponente      | Datentyp    | Beschreibung   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
|-------------------------|-------------|--|------|--------|--------|-----|-----|-----|---|--|--|--|--|--|---|--|--|--|--|--------|---|--|--|------|--|--|---|----------|-------|--|--------|--|---|--|--|------|--|--------|---|--|--|------|--------|--------|---|-------------|--|------|--------|--------|---|-------------|-----------------|------|--------|--------|
| ERROR_ID                | WORD        | Fehler-ID  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| FLAGS                   | BYTE        | Zeigt, ob der Fehler während eines Bausteinaufrufs aufgetreten ist.<br>16#01: Fehler während eines Bausteinaufrufs<br>16#00: Kein Fehler während eines Bausteinaufrufs   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| REACTION                | BYTE        | Voreingestellte Reaktion:<br>0: Ignorieren (Schreibfehler),<br>1: Mit dem Ersatzwert "0" fortsetzen (Lesefehler),<br>2: Anweisung überspringen (Systemfehler)  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| CODE_ADDRESS            | CREF        | Informationen zur Adresse und Art des Bausteins  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| BLOCK_TYPE              | BYTE        | Bausteintyp, in dem der Fehler aufgetreten ist:<br>1: OB<br>2: FC<br>3: FB   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| CB_NUMBER               | UINT        | Nummer des Codebausteins   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| OFFSET                  | UDINT       | Verweis auf den internen Speicher  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| MODE                    | BYTE        | Zugriffsart: Abhängig von der Zugriffsart können die folgenden Informationen ausgegeben werden:<br><table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Mode</th> <th>(A)</th> <th>(B)</th> <th>(C)</th> <th>(D)</th> <th>(E)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>1</td> <td></td> <td></td> <td></td> <td></td> <td>Offset</td> </tr> <tr> <td>2</td> <td></td> <td></td> <td>Area</td> <td></td> <td></td> </tr> <tr> <td>3</td> <td>Location</td> <td>Scope</td> <td></td> <td>Number</td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td>Area</td> <td></td> <td>Offset</td> </tr> <tr> <td>5</td> <td></td> <td></td> <td>Area</td> <td>DB no.</td> <td>Offset</td> </tr> <tr> <td>6</td> <td>PtrNo./ Acc</td> <td></td> <td>Area</td> <td>DB no.</td> <td>Offset</td> </tr> <tr> <td>7</td> <td>PtrNo./ Acc</td> <td>Slot No./ Scope</td> <td>Area</td> <td>DB no.</td> <td>Offset</td> </tr> </tbody> </table> | Mode | (A)    | (B)    | (C) | (D) | (E) | 0 |  |  |  |  |  | 1 |  |  |  |  | Offset | 2 |  |  | Area |  |  | 3 | Location | Scope |  | Number |  | 4 |  |  | Area |  | Offset | 5 |  |  | Area | DB no. | Offset | 6 | PtrNo./ Acc |  | Area | DB no. | Offset | 7 | PtrNo./ Acc | Slot No./ Scope | Area | DB no. | Offset |
| Mode                    | (A)         | (B)  | (C)  | (D)    | (E)    |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| 0                       |             |  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| 1                       |             |  |      |        | Offset |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| 2                       |             |  | Area |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| 3                       | Location    | Scope  |      | Number |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| 4                       |             |  | Area |        | Offset |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| 5                       |             |  | Area | DB no. | Offset |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| 6                       | PtrNo./ Acc |  | Area | DB no. | Offset |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| 7                       | PtrNo./ Acc | Slot No./ Scope  | Area | DB no. | Offset |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| OPERAND_NUMBER          | UINT        | Operandennummer des Maschinenbefehls   |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |
| POINTER_NUMBER_LOCATION | UINT        | (A) Interner Zeiger  |      |        |        |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |             |  |      |        |        |   |             |                 |      |        |        |

| Strukturkomponente |           | Datentyp | Beschreibung   |
|--------------------|-----------|----------|--|
| SLOT_NUMBER_SCOPE  |           | UINT     | (B) Ablagebereich im internen Speicher   |
| DATA_ADDRESS       |           | NREF     | Informationen zur Adresse eines Operanden  |
|                    | AREA      | BYTE     | (C) Speicherbereich:<br>L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE<br>E: 16#81<br>A: 16#82<br>M: 16#83<br>DB: 16#84, 85, 8A, 8B<br>Bereichsverletzungen bei einer direkt editierbaren Variablen vom Datentyp DINT: 16#04 |
|                    | DB_NUMBER | UINT     | (D) Nummer des Datenbausteins  |
|                    | OFFSET    | UDINT    | (E) Relative Adresse des Operanden   |

### Strukturkomponente "ERROR\_ID"

Die folgende Tabelle zeigt die Werte, die an der Strukturkomponente "ERROR\_ID" ausgegeben werden können:

| ID*<br>(Hexadezimal) | ID*<br>(Dezimal) | Beschreibung  |
|----------------------|------------------|---|
| 0                    | 0                | Kein Fehler   |
| 2503                 | 9475             | Ungültiger Zeiger   |
| 2520                 | 9504             | Ungültiger STRING   |
| 2522                 | 9506             | Lesefehler: Operand außerhalb des gültigen Bereichs           |
| 2523                 | 9507             | Schreibfehler: Operand außerhalb des gültigen Bereichs        |
| 2524                 | 9508             | Lesefehler: Ungültiger Operand                                |
| 2525                 | 9509             | Schreibfehler: Ungültiger Operand                             |
| 2528                 | 9512             | Lesefehler: Datenausrichtung                                  |
| 2529                 | 9513             | Schreibfehler: Datenausrichtung                               |
| 252C                 | 9516             | Ungültiger Zeiger   |
| 2530                 | 9520             | Schreibfehler: Datenbaustein                                  |
| 2533                 | 9523             | Unzulässig verwendeter Zeiger                                 |
| 2538                 | 9528             | Zugriffsfehler: DB nicht vorhanden                            |
| 2539                 | 9529             | Zugriffsfehler: Falscher DB verwendet                         |
| 253A                 | 9530             | Globaler Datenbaustein existiert nicht                        |
| 253C                 | 9532             | Fehlerhafte Angabe oder die Funktion existiert nicht          |
| 253D                 | 9533             | Systemfunktion existiert nicht                                |
| 253E                 | 9534             | Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht |
| 253F                 | 9535             | Systembaustein existiert nicht                                |
| 2550                 | 9552             | Zugriffsfehler: DB nicht vorhanden                            |
| 2551                 | 9553             | Zugriffsfehler: Falscher DB verwendet                         |
| 2575                 | 9589             | Fehler in der Programmschachtelungstiefe                      |
| 2576                 | 9590             | Fehler in der Lokaldatenverteilung                            |

| ID* (Hexadezimal) | ID* (Dezimal) | Beschreibung   |
|-------------------|---------------|--|
| 2577              | 9591          | Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert. |
| 25A0              | 9632          | Interner Fehler im TP  |
| 25A1              | 9633          | Variable ist schreibgeschützt  |
| 25A2              | 9634          | Ungültiger Zahlenwert der Variable   |
| 2942              | 10562         | Lesefehler: Eingang  |
| 2943              | 10563         | Schreibfehler: Ausgang   |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

Der Freigabeausgang ENO der Anweisung "Fehler lokal abfragen" wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und eine Fehlerinformation vorliegt. Wenn eine dieser Bedingungen nicht zutrifft, wird die weitere Programmbearbeitung durch die Anweisung "Fehler lokal abfragen" nicht beeinflusst.

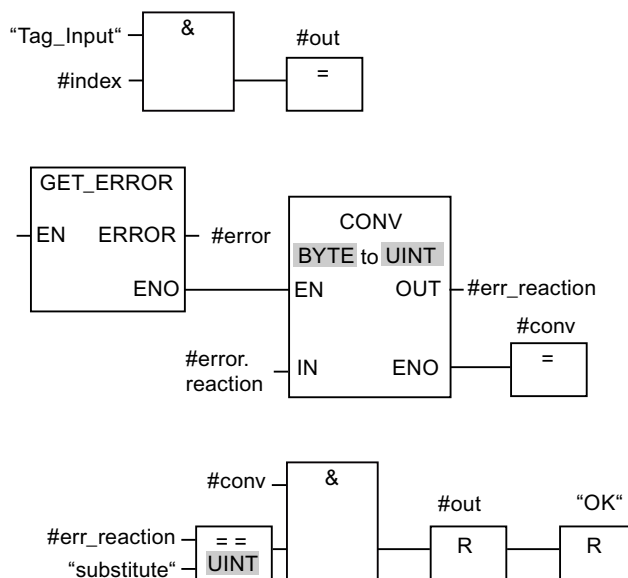
Die Anweisung "Fehler lokal abfragen" kann auch eingesetzt werden, um an den aufrufenden Baustein eine Meldung über den Fehlerzustand weiterzuleiten. Dafür muss die Anweisung im letzten Netzwerk des aufgerufenen Bausteins positioniert werden.

**Hinweis**

Die Anweisung "Fehler lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn "Fehler lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn ein Fehler auftritt, liefert die Anweisung "Fehler lokal abfragen" die Fehlerinformation an die lokal angelegte Struktur "#error" am Ausgang ERROR. Die Fehlerinformation wird konvertiert und mithilfe der Vergleichsanweisung "Gleich" ausgewertet. Als erster Vergleichswert wird der Anweisung die Information zur Fehlerart zugewiesen. Als zweiter Vergleichswert wird im Operanden "substitute" der Wert "1" angegeben. Wenn es sich um einen Lesefehler handelt, ist die Bedingung der Vergleichsanweisung erfüllt. Die Ausgänge "#out" und "OK" werden in diesem Fall zurückgesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zur Fehlerbehandlung (Seite 1403)

Grundlagen der lokalen Fehlerbehandlung (Seite 1405)

Prioritäten bei der Fehlerausgabe (Seite 1406)

Lokale Fehlerbehandlung für einen Baustein aktivieren (Seite 1407)

## GET\_ERR\_ID: Fehler-ID lokal abfragen

### Beschreibung

Mit der Anweisung "Fehler-ID lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Wenn das System während der Bausteinbearbeitung Fehler meldet, wird die Fehler-ID des ersten aufgetretenen Fehlers in der Variablen am Ausgang ID gespeichert. Am Ausgang ID können nur Variablen vom Datentyp WORD angegeben werden. Wenn der erste aufgetretene Fehler behoben ist, gibt die Anweisung die Fehler-ID des nächsten aufgetretenen Fehlers aus.

Der Ausgang der Anweisung "Fehler-ID lokal abfragen" wird nur gesetzt, wenn der Eingang der Anweisung den Signalzustand "1" liefert und eine Fehlerinformation vorliegt. Wenn eine dieser Bedingungen nicht zutrifft, wird die weitere Programmbearbeitung durch "Fehler-ID lokal abfragen" nicht beeinflusst.

Die Anweisung "Fehler-ID lokal abfragen" kann auch eingesetzt werden, um an den aufrufenden Baustein eine Meldung über den Fehlerzustand weiterzuleiten. Dafür muss die Anweisung im letzten Netzwerk des aufgerufenen Bausteins positioniert werden.

---

### Hinweis

Die Anweisung "Fehler-ID lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn die Anweisung "Fehler-ID lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

---

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler-ID lokal abfragen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung |
|-----------|-------------|----------|-----------------|--------------|
| ID        | Output      | WORD     | E, A, M, D, L   | Fehler-ID    |

### Parameter ID

Die folgende Tabelle zeigt die Werte, die am Parameter ID ausgegeben werden können:

| ID* (Hexadezimal) | ID* (Dezimal) | Beschreibung   |
|-------------------|---------------|--|
| 0                 | 0             | Kein Fehler  |
| 2503              | 9475          | Ungültiger Zeiger  |
| 2520              | 9504          | Ungültiger STRING  |
| 2522              | 9506          | Lesefehler: Operand außerhalb des gültigen Bereichs                              |
| 2523              | 9507          | Schreibfehler: Operand außerhalb des gültigen Bereichs                           |
| 2524              | 9508          | Lesefehler: Ungültiger Operand   |
| 2525              | 9509          | Schreibfehler: Ungültiger Operand  |
| 2528              | 9512          | Lesefehler: Datenausrichtung   |
| 2529              | 9513          | Schreibfehler: Datenausrichtung  |
| 252C              | 9516          | Ungültiger Zeiger  |
| 2530              | 9520          | Schreibfehler: Datenbaustein   |
| 2533              | 9523          | Unzulässig verwendeter Zeiger  |
| 2538              | 9528          | Zugriffsfehler: DB nicht vorhanden   |
| 2539              | 9529          | Zugriffsfehler: Falscher DB verwendet  |
| 253A              | 9530          | Globaler Datenbaustein existiert nicht   |
| 253C              | 9532          | Fehlerhafte Angabe oder die Funktion existiert nicht                             |
| 253D              | 9533          | Systemfunktion existiert nicht   |
| 253E              | 9534          | Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht                    |
| 253F              | 9535          | Systembaustein existiert nicht   |
| 2550              | 9552          | Zugriffsfehler: DB nicht vorhanden   |
| 2551              | 9553          | Zugriffsfehler: Falscher DB verwendet  |
| 2575              | 9589          | Fehler in der Programmschachtelungstiefe   |
| 2576              | 9590          | Fehler in der Lokaldatenverteilung   |
| 2577              | 9591          | Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert. |
| 25A0              | 9632          | Interner Fehler im TP  |
| 25A1              | 9633          | Variable ist schreibgeschützt  |
| 25A2              | 9634          | Ungültiger Zahlenwert der Variable   |
| 2942              | 10562         | Lesefehler: Eingang  |
| 2943              | 10563         | Schreibfehler: Ausgang   |

\*Die Fehlercodes im Programmmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zur Fehlerbehandlung (Seite 1403)
- Grundlagen der lokalen Fehlerbehandlung (Seite 1405)
- Prioritäten bei der Fehlerausgabe (Seite 1406)
- Lokale Fehlerbehandlung für einen Baustein aktivieren (Seite 1407)

**INIT\_RD: Alle remanenten Daten zurücksetzen****Beschreibung**

Mit der Anweisung "Alle remanenten Daten zurücksetzen" setzen Sie gleichzeitig alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler zurück. Die Anweisung kann nur innerhalb eines Anlauf-OBs ausgeführt werden, da die Ausführung die Programmzyklusdauer überschreiten würde.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Alle remanenten Daten zurücksetzen":

| Parameter | Deklaration | Datentyp | Speicherbereich                       | Beschreibung   |
|-----------|-------------|----------|---------------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z                   | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L                         | Freigabeausgang  |
| REQ       | Input       | BOOL     | E, A, M, D, L, T, Z<br>oder Konstante | Wenn der Eingang REQ den Signalzustand "1" führt, werden alle remanenten Daten zurückgesetzt.  |
| RET_VAL   | Output      | INT      | E, A, M, D, L                         | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter RET\_VAL

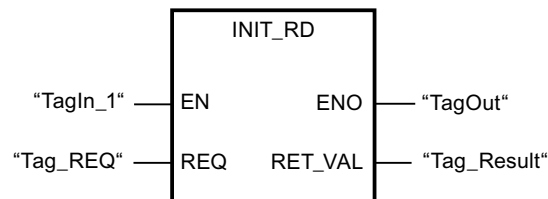
Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)    | Erläuterung   |
|------------------------------|---|
| 0000                         | Kein Fehler   |
| 80B5                         | Die Anweisung kann nicht ausgeführt werden, da sie nicht innerhalb eines Anlauf-OBs programmiert wurde. |
| allgemeine Fehlerinformation | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"  |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Wenn die Operanden "TagIn\_1" und "Tag\_REQ" den Signalzustand "1" liefern, wird die Anweisung ausgeführt. Alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler werden zurückgesetzt. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1".

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zur Fehlerbehandlung (Seite 1403)
- Grundlagen der lokalen Fehlerbehandlung (Seite 1405)
- Prioritäten bei der Fehlerausgabe (Seite 1406)
- Lokale Fehlerbehandlung für einen Baustein aktivieren (Seite 1407)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2103)



## WAIT: Zeitverzögerung programmieren

### Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten bis zu 32767 Mikrosekunden ( $\mu\text{s}$ ) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung "Zeitverzögerung programmieren".

Die Ausführung der Anweisung kann durch höherpriorie Ereignisse unterbrochen werden.

Die Anweisung "Zeitverzögerung programmieren" liefert keine Fehlerinformationen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitverzögerung programmieren":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung  |
|-----------|-------------|----------|------------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z                | Freigabeeingang   |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang   |
| WT        | Input       | INT      | E, A, M, D, L, P<br>oder Konstante | Verzögerungszeit<br>in Mikrosekunden<br>( $\mu\text{s}$ ) |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### Wortverknüpfungen

#### AND: UND verknüpfen

### Beschreibung

Mit der Anweisung "UND verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch UND und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch UND verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits der angegebenen Werte ausgeführt.

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend

nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch UND verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", nur wenn beide zu verknüpfende Bits auch den Signalzustand "1" liefern. Wenn eines der beiden zu verknüpfenden Bits den Signalzustand "0" führt, wird das entsprechende Ergebnisbit zurückgesetzt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UND verknüpfen":

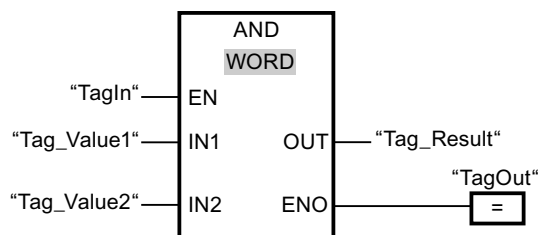
| Parameter | Deklaration | Datentyp  | Speicherbereich                 |                                 | Beschreibung                 |
|-----------|-------------|-----------|---------------------------------|---------------------------------|------------------------------|
|           |             |           | S7-1200                         | S7-1500                         |                              |
| EN        | Input       | BOOL      | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang              |
| ENO       | Output      | BOOL      | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang              |
| IN1       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Erster Wert der Verknüpfung  |
| IN2       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter Wert der Verknüpfung |
| INn       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Optionale Eingangswerte      |
| OUT       | Output      | Bitfolgen | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis der Anweisung       |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                |
|-----------|------------|---------------------|
| IN1       | Tag_Value1 | 0101 0101 0101 0101 |
| IN2       | Tag_Value2 | 0000 0000 0000 1111 |
| OUT       | Tag_Result | 0000 0000 0000 0101 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "UND verknüpfen" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" durch UND verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag\_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)

Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## OR: ODER verknüpfen

### Beschreibung

Mit der Anweisung "ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausganges OUT abgelegt. Die gleiche Verknüpfung wird für alle Bits der angegebenen Variablen ausgeführt.

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch ODER verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", wenn mindestens eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ODER verknüpfen":

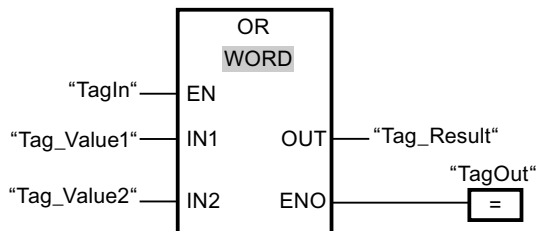
| Parameter | Deklaration | Datentyp  | Speicherbereich                 |                                 | Beschreibung                 |
|-----------|-------------|-----------|---------------------------------|---------------------------------|------------------------------|
|           |             |           | S7-1200                         | S7-1500                         |                              |
| EN        | Input       | BOOL      | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang              |
| ENO       | Output      | BOOL      | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang              |
| IN1       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Erster Wert der Verknüpfung  |
| IN2       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter Wert der Verknüpfung |
| INn       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Optionale Eingangswerte      |
| OUT       | Output      | Bitfolgen | E, A, M, D, L, P                | E, A, M, D, L, P                | Ergebnis der Anweisung       |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                |
|-----------|------------|---------------------|
| IN1       | Tag_Value1 | 0101 0101 0101 0101 |
| IN2       | Tag_Value2 | 0000 0000 0000 1111 |
| OUT       | Tag_Result | 0101 0101 0101 1111 |

Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung "ODER verknüpfen" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" durch ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag\_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)
- Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**XOR: EXKLUSIV ODER verknüpfen****Beschreibung**

Mit der Anweisung "EXKLUSIV ODER verknüpfen" verknüpfen Sie den Wert am Eingang IN1 mit dem Wert am Eingang IN2 bitweise durch EXKLUSIV ODER und fragen das Ergebnis am Ausgang OUT ab.

Bei der Bearbeitung der Anweisung wird das Bit 0 des Wertes am Eingang IN1 mit dem Bit 0 des Wertes am Eingang IN2 durch EXKLUSIV ODER verknüpft. Das Ergebnis wird im Bit 0 des Ausgangs OUT abgelegt. Die gleiche Verknüpfung wird für alle weiteren Bits des angegebenen Werts ausgeführt.

Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN1 und IN2). Die Anzahl der Eingänge ist in der Anweisungsbox erweiterbar. Die eingefügten Eingänge werden in der Box aufsteigend nummeriert. Bei der Ausführung der Anweisung werden die Werte aller verfügbaren Eingangsparameter durch EXKLUSIV ODER verknüpft. Das Ergebnis wird am Ausgang OUT abgelegt.

Ein Ergebnisbit hat den Signalzustand "1", wenn eines der beiden zu verknüpfenden Bits den Signalzustand "1" liefert. Wenn beide zu verknüpfende Bits den Signalzustand "1" oder "0" führen, wird das entsprechende Ergebnisbit zurückgesetzt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "EXKLUSIV ODER verknüpfen":

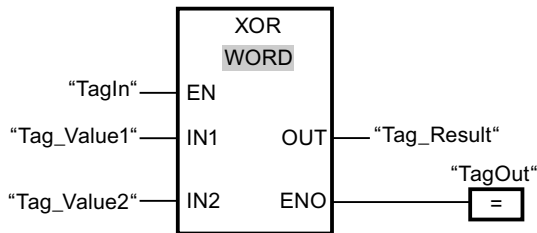
| Parameter | Deklaration | Datentyp  | Speicherbereich                |                                | Beschreibung                 |
|-----------|-------------|-----------|--------------------------------|--------------------------------|------------------------------|
|           |             |           | S7-1200                        | S7-1500                        |                              |
| EN        | Input       | BOOL      | E, A, M, D, L                  | E, A, M, D, L, T, Z            | Freigabeeingang              |
| ENO       | Output      | BOOL      | E, A, M, D, L                  | E, A, M, D, L                  | Freigabeausgang              |
| IN1       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstant | E, A, M, D, L, P oder Konstant | Erster Wert der Verknüpfung  |
| IN2       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstant | E, A, M, D, L, P oder Konstant | Zweiter Wert der Verknüpfung |
| INn       | Input       | Bitfolgen | E, A, M, D, L, P oder Konstant | E, A, M, D, L, P oder Konstant | Optionale Eingangswerte      |
| OUT       | Output      | Bitfolgen | E, A, M, D, L, P               | E, A, M, D, L, P               | Ergebnis der Anweisung       |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert                |
|-----------|------------|---------------------|
| IN1       | Tag_Value1 | 0101 0101 0101 0101 |
| IN2       | Tag_Value2 | 0000 0000 0000 1111 |
| OUT       | Tag_Result | 0101 0101 0101 1010 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "EXKLUSIV ODER verknüpfen" ausgeführt. Der Wert des Operanden "Tag\_Value1" wird mit dem Wert des Operanden "Tag\_Value2" durch EXKLUSIV ODER verknüpft. Das Ergebnis wird Bit für Bit gebildet und im Operanden "Tag\_Result" ausgegeben. Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Zusätzliche Eingänge und Ausgänge in FUP-Elemente einfügen (Seite 1351)
- Eingänge und Ausgänge einer Anweisung entfernen (Seite 1353)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**INV: Einerkomplement erzeugen**

**Beschreibung**

Mit der Anweisung "Einerkomplement erzeugen" invertieren Sie den Signalzustand der Bits am Eingang IN. Bei der Bearbeitung der Anweisung wird der Wert am Eingang IN mit einer Hexadezimalen Schablone (W#16#FFFF für 16 Bit-Zahlen oder DW#16#FFFF FFFF für 32 Bit-Zahlen) durch EXKLUSIV ODER verknüpft. Dadurch wird der Signalzustand der einzelnen Bits umgekehrt und am Ausgang OUT ausgegeben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Einerkomplement erzeugen":

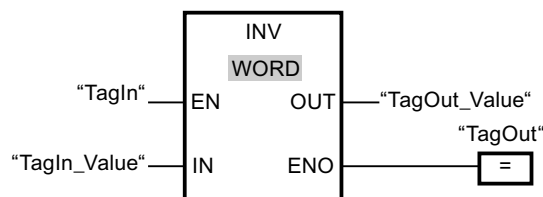
| Parameter | Deklaration | Datentyp              | Speicherbereich                 |                                 | Beschreibung                             |
|-----------|-------------|-----------------------|---------------------------------|---------------------------------|--|
|           |             |                       | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL                  | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                          |
| ENO       | Output      | BOOL                  | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                          |
| IN        | Input       | Bitfolgen, Ganzzahlen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert                             |
| OUT       | Output      | Bitfolgen, Ganzzahlen | E, A, M, D, L, P                | E, A, M, D, L, P                | Einerkomplement des Wertes am Eingang IN |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert      |         |
|-----------|--------------|-----------|---------|
| IN        | TagIn_Value  | W#16#000F | W#16#7E |
| OUT       | TagOut_Value | W#16#FFF0 | W#16#81 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Einerkomplement erzeugen" ausgeführt. Die Anweisung kehrt den Signalzustand der einzelnen Bits am Eingang "TagIn\_Value" um und schreibt das Ergebnis in den Ausgang "TagOut\_Value". Der Freigabeausgang ENO und der Ausgang "TagOut" werden auf den Signalzustand "1" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## DECO: Decodieren

### Beschreibung

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert am Eingang IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert am Eingang IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Decodieren":

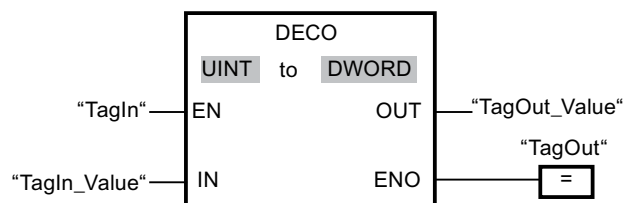
| Parameter | Deklaration | Datentyp  | Speicherbereich                 |                                 | Beschreibung   |
|-----------|-------------|-----------|---------------------------------|---------------------------------|--|
|           |             |           | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL      | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang                                      |
| ENO       | Output      | BOOL      | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang                                      |
| IN        | Input       | UINT      | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Position des Bits im Ausgangswert, das gesetzt wird. |
| OUT       | Output      | Bitfolgen | E, A, M, D, L, P                | E, A, M, D, L, P                | Ausgangswert   |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

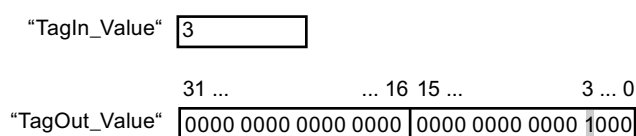
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:





Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung "Decodieren" ausgeführt. Die Anweisung liest die Bitnummer "3" aus dem Wert des Operanden "TagIn\_Value" am Eingang ab und setzt das dritte Bit im Wert des Operanden "TagOut\_Value" am Ausgang.

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Ausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ENCO: Encodieren

### Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und gibt ihn am Ausgang OUT aus.

Die Anweisung "Encodieren" selektiert das niederwertigste Bit des Werts am Eingang IN und schreibt dessen Bitnummer in die Variable am Ausgang OUT.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Encodieren":

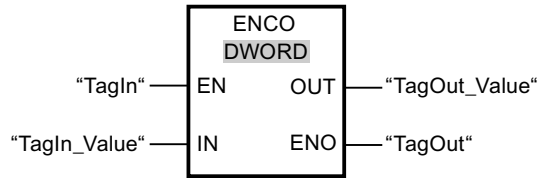
| Parameter | Deklaration | Datentyp  | Speicherbereich                 |                                 | Beschreibung    |
|-----------|-------------|-----------|---------------------------------|---------------------------------|-----------------|
|           |             |           | S7-1200                         | S7-1500                         |                 |
| EN        | Input       | BOOL      | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang |
| ENO       | Output      | BOOL      | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang |
| IN        | Input       | Bitfolgen | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert    |
| OUT       | Output      | INT       | E, A, M, D, L, P                | E, A, M, D, L, P                | Ausgangswert    |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

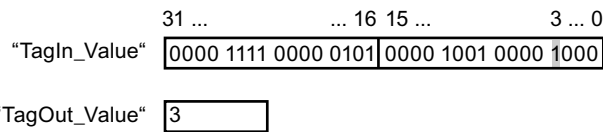
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:



Wenn der Operand "TagIn" den Signalzustand "1" führt, wird die Anweisung "Encodieren" ausgeführt. Die Anweisung selektiert Bitposition "3" als niederwertigstes gesetztes Bit am Eingang "TagIn\_Value" und schreibt den Wert "3" in die Variable am Ausgang "TagOut\_Value".

Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, führt der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**SEL: Selektieren**

**Beschreibung**

Mit der Anweisung "Selektieren" wählen Sie abhängig vom Signalzustand am Schalter (Eingang G) einen der Eingänge IN0 oder IN1 aus und kopieren dessen Inhalt in den Ausgang OUT. Wenn der Eingang G den Signalzustand "0" hat, wird der Wert am Eingang IN0 kopiert. Wenn der Eingang G den Signalzustand "1" hat, wird der Wert am Eingang IN1 in den Ausgang OUT kopiert.

Die Ausführung der Anweisung setzt den Signalzustand "1" am Freigabeeingang EN voraus und dass die Variablen an allen Parametern vom gleichen Datentyp sind.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Selektieren":

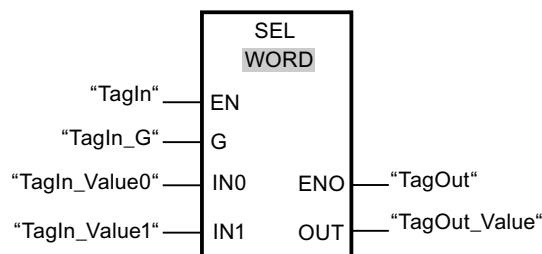
| Parameter | Deklaration | Datentyp   |  | Speicherbereich                       |  | Beschreibung         |
|-----------|-------------|--|--|---------------------------------------|--|----------------------|
|           |             | S7-1200  | S7-1500  | S7-1200                               | S7-1500                                  |                      |
| EN        | Input       | BOOL   | BOOL   | E, A, M, D, L                         | E, A, M, D, L                            | Freigabeeingang      |
| ENO       | Output      | BOOL   | BOOL   | E, A, M, D, L                         | E, A, M, D, L                            | Freigabeausgang      |
| G         | Input       | BOOL   | BOOL   | E, A, M, D, L,<br>oder<br>Konstante   | E, A, M, D, L, T,<br>Z oder<br>Konstante | Schalter             |
| IN0       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzah<br>len, TIME,<br>TOD, DATE,<br>CHAR | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahl<br>en, Zeiten,<br>TOD, LTOD,<br>DATE, LDT,<br>CHAR | E, A, M, D, L,<br>P oder<br>Konstante | E, A, M, D, L, P<br>oder Konstante       | Erster Eingangswert  |
| IN1       | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzah<br>len, TIME,<br>TOD, DATE,<br>CHAR | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahl<br>en, Zeiten,<br>TOD, LTOD,<br>DATE, LDT,<br>CHAR | E, A, M, D, L,<br>P oder<br>Konstante | E, A, M, D, L, P<br>oder Konstante       | Zweiter Eingangswert |
| OUT       | Output      | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzah<br>len, TIME,<br>TOD, DATE,<br>CHAR | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahl<br>en, Zeiten,<br>TOD, LTOD,<br>DATE, LDT,<br>CHAR | E, A, M, D, L,<br>P                   | E, A, M, D, L, P                         | Ergebnis             |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert      |         |
|-----------|--------------|-----------|---------|
| G         | TagIn_G      | 0         | 1       |
| IN0       | TagIn_Value0 | W#16#0000 | W#16#4C |
| IN1       | TagIn_Value1 | W#16#FFFF | W#16#5E |
| OUT       | TagOut_Value | W#16#0000 | W#16#5E |

Wenn der Operand "TagIn" den Signalzustand "1" hat, wird die Anweisung "Selektieren" ausgeführt. Abhängig vom Signalzustand am Eingang "TagIn\_G" wird der Wert am Eingang "TagIn\_Value0" oder "TagIn\_Value1" selektiert und in den Ausgang "TagOut\_Value" kopiert. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, hat der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**MUX: Multiplexen**

**Beschreibung**

Mit der Anweisung "Multiplexen" kopieren Sie den Inhalt eines ausgewählten Eingangs in den Ausgang OUT. Die Anweisungsbox enthält im Grundzustand min. 2 Eingänge (IN0 und IN1). Die Anzahl der auswählbaren Eingänge der Anweisungsbox ist erweiterbar. Sie können maximal 32 Eingänge deklarieren.

Die Eingänge werden in der Box automatisch nummeriert. Die Nummerierung fängt mit IN0 an und wird mit jedem neuen Eingang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Eingang, dessen Inhalt in den Ausgang OUT kopiert wird. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Eingänge ist, wird der Inhalt des Parameters ELSE in den Ausgang OUT kopiert und dem Freigabeausgang ENO den Signalzustand "0" zugewiesen.

Die Anweisung "Multiplexen" kann nur ausgeführt werden, wenn die Variablen an allen Eingängen sowie am Ausgang OUT vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, für den nur Ganzzahlen angegeben werden können.

Der Freigabeausgang ENO wird zurückgesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Eingänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplexen":

| Parameter | Deklarieren | Datentyp   |   | Speicherbereich                 |                                 | Beschreibung  |
|-----------|-------------|--|---|---------------------------------|---------------------------------|---|
|           |             | S7-1200  | S7-1500   | S7-1200                         | S7-1500                         |   |
| EN        | Input       | BOOL   | BOOL  | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang   |
| ENO       | Output      | BOOL   | BOOL  | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang   |
| K         | Input       | Ganzzahlen   | Ganzzahlen  | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Gibt den Eingang an, dessen Inhalt kopiert wird. <ul style="list-style-type: none"> <li>• Wenn K = 0 =&gt; Parameter IN0</li> <li>• Wenn K = 1 =&gt; Parameter IN1, etc.</li> </ul> |
| IN0       | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, TIME, TOD, DATE | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Erster Eingangswert   |
| IN1       | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, TIME, TOD, DATE | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Zweiter Eingangswert  |

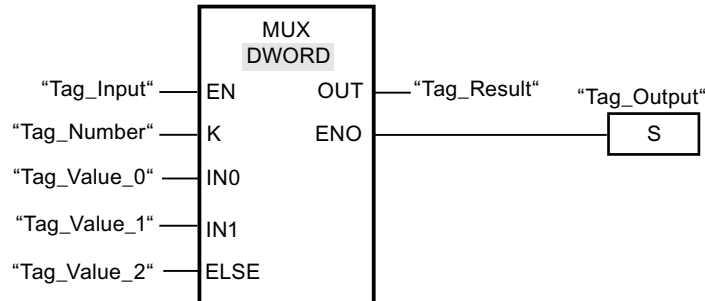
| Parameter | Deklarieren | Datentyp  |  | Speicherbereich                       |                                       | Beschreibung                                 |
|-----------|-------------|---|--|---------------------------------------|---------------------------------------|--|
|           |             | S7-1200   | S7-1500  | S7-1200                               | S7-1500                               |  |
| INn       | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, TIME,<br>TOD, DATE | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, Zeiten,<br>TOD,<br>LTOD,<br>DATE, LDT | E, A, M, D,<br>L, P oder<br>Konstante | E, A, M, D,<br>L, P oder<br>Konstante | Optionale Eingangswerte                      |
| ELSE      | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, TIME,<br>TOD, DATE | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, Zeiten,<br>TOD,<br>LTOD,<br>DATE, LDT | E, A, M, D,<br>L, P oder<br>Konstante | E, A, M, D,<br>L, P oder<br>Konstante | Gibt den Wert an, der bei K > n kopiert wird |
| OUT       | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, TIME,<br>TOD, DATE | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, Zeiten,<br>TOD,<br>LTOD,<br>DATE, LDT | E, A, M, D,<br>L, P                   | E, A, M, D,<br>L, P                   | Ausgang, in den der Wert kopiert wird        |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand     | Wert           |
|-----------|-------------|----------------|
| K         | Tag_Number  | 1              |
| IN0       | Tag_Value_0 | DW#16#00000000 |
| IN1       | Tag_Value_1 | DW#16#003E4A7D |
| ELSE      | Tag_Value_2 | DW#16#FFFF0000 |
| OUT       | Tag_Result  | DW#16#003E4A7D |

Wenn der Operand "Tag\_Input" den Signalzustand "1" liefert, wird die Anweisung "Multiplexen" ausgeführt. Entsprechend dem Wert des Operanden "Tag\_Number" wird der Wert am Eingang "Tag\_Value\_1" kopiert und dem Operanden am Ausgang "Tag\_Result" zugewiesen. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, werden die Ausgänge ENO und "Tag\_Output" gesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## DEMUX: Demultiplexen

### Beschreibung

Mit der Anweisung "Demultiplexen" kopieren Sie den Inhalt des Eingangs IN in einen ausgewählten Ausgang. Die Anweisungsbox enthält im Grundzustand min. 2 Ausgänge (OUT0 und OUT1). Die Anzahl der auswählbaren Ausgänge ist in der Anweisungsbox erweiterbar. Die Ausgänge werden in der Box automatisch nummeriert. Die Nummerierung fängt bei OUT0 an und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. Mit dem Parameter K bestimmen Sie den Ausgang, in welchen der Inhalt des Eingangs IN kopiert wird. Die anderen Ausgänge werden nicht verändert. Wenn der Wert des Parameters K größer als die Anzahl der verfügbaren Ausgänge ist, wird der Inhalt des Eingangs IN in den Parameter ELSE und dem Freigabeausgang ENO der Signalzustand "0" zugewiesen.

Die Anweisung "Demultiplexen" kann nur ausgeführt werden, wenn die Variablen am Eingang IN sowie an allen Ausgängen vom gleichen Datentyp sind. Eine Ausnahme ist der Parameter K, an welchem nur Ganzzahlen angegeben werden können.

Der Freigabeausgang ENO wird zurückgesetzt, wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Ausgänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Demultiplexen":

| Parameter | Deklaration | Datentyp  |  | Speicherbereich                 |                                 | Beschreibung   |
|-----------|-------------|---|--|---------------------------------|---------------------------------|--|
|           |             | S7-1200   | S7-1500  | S7-1200                         | S7-1500                         |  |
| EN        | Input       | BOOL  | BOOL   | E, A, M, D, L                   | E, A, M, D, L, T, Z             | Freigabeeingang  |
| ENO       | Output      | BOOL  | BOOL   | E, A, M, D, L                   | E, A, M, D, L                   | Freigabeausgang  |
| K         | Input       | Ganzzahlen  | Ganzzahlen   | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Gibt den Ausgang an, in welchen der Eingangswert (IN) kopiert wird. <ul style="list-style-type: none"> <li>• Wenn K = 0 =&gt; Parameter OUT0</li> <li>• Wenn K = 1 =&gt; Parameter OUT1, etc.</li> </ul> |
| IN        | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P oder Konstante | E, A, M, D, L, P oder Konstante | Eingangswert   |
| OUT0      | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, LDT | E, A, M, D, L, P                | E, A, M, D, L, P                | Erster Ausgang   |



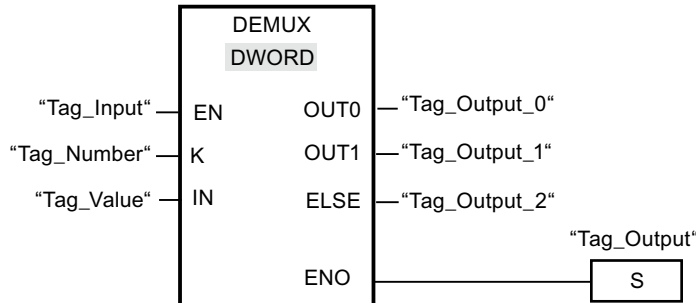
| Parameter | Deklaration | Datentyp  |  | Speicherbereich     |                     | Beschreibung  |
|-----------|-------------|---|--|---------------------|---------------------|---|
|           |             | S7-1200   | S7-1500  | S7-1200             | S7-1500             |   |
| OUT1      | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, TIME,<br>TOD, DATE | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, Zeiten,<br>TOD,<br>LTOD,<br>DATE, LDT | E, A, M, D,<br>L, P | E, A, M, D,<br>L, P | Zweiter Ausgang   |
| OUTn      | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, TIME,<br>TOD, DATE | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, Zeiten,<br>TOD,<br>LTOD,<br>DATE, LDT | E, A, M, D,<br>L, P | E, A, M, D,<br>L, P | Optionale Ausgänge  |
| ELSE      | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, TIME,<br>TOD, DATE | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen, Zeiten,<br>TOD,<br>LTOD,<br>DATE, LDT | E, A, M, D,<br>L, P | E, A, M, D,<br>L, P | Ausgang, in welchen der Eingangswert (IN) bei $K > n$ kopiert wird. |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den verfügbaren Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Tabelle 9-26 Eingangswerte der Anweisung "Demultiplexen" vor dem Ausführen des Netzwerks

| Parameter | Operand    | Werte          |                |
|-----------|------------|----------------|----------------|
| K         | Tag_Number | 1              | 4              |
| IN        | Tag_Value  | DW#16#FFFFFFFF | DW#16#003E4A7D |

Tabelle 9-27 Ausgangswerte der Anweisung "Demultiplexen" nach dem Ausführen des Netzwerks

| Parameter | Operand      | Werte          |                |
|-----------|--------------|----------------|----------------|
| OUT0      | Tag_Output_0 | unverändert    | unverändert    |
| OUT1      | Tag_Output_1 | DW#16#FFFFFFFF | unverändert    |
| ELSE      | Tag_Output_2 | unverändert    | DW#16#003E4A7D |

Wenn der Eingang "Tag\_Input" den Signalzustand "1" liefert, wird die Anweisung "Demultiplexen" ausgeführt. Abhängig vom Wert des Operanden "Tag\_Number" wird der Wert am Eingang IN in den entsprechenden Ausgang kopiert.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**Schieben und Rotieren**

**SHR: Rechts schieben**

**Beschreibung**

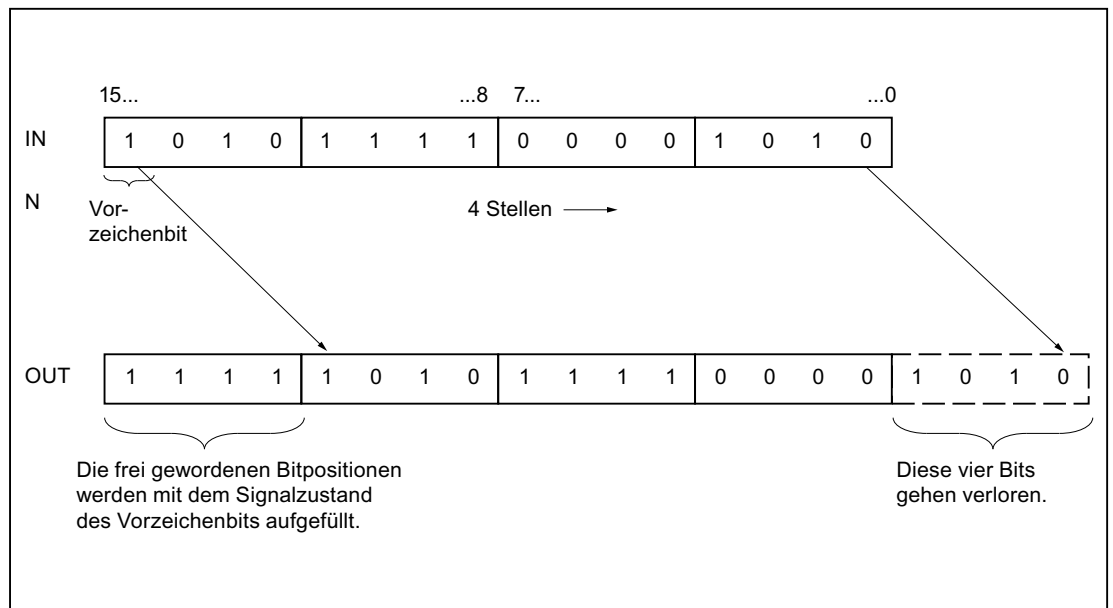
Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Eingang N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Eingang N "0" ist, wird der Wert am Eingang IN unverändert in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Eingang N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach rechts verschoben.

Bei Werten ohne Vorzeichen werden die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden mit Nullen aufgefüllt. Wenn der angegebene Wert ein Vorzeichen aufweist, werden die freien Bitstellen mit dem Signalzustand des Vorzeichenbits belegt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp Integer um vier Bitstellen nach rechts verschoben wird:



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts schieben":

| Parameter | Deklaration | Datentyp              |                       | Speicherbereich              |                              | Beschreibung              |
|-----------|-------------|-----------------------|-----------------------|------------------------------|------------------------------|---------------------------|
|           |             | S7-1200               | S7-1500               | S7-1200                      | S7-1500                      |                           |
| EN        | Input       | BOOL                  | BOOL                  | E, A, M, D, L                | E, A, M, D, L, T, Z          | Freigabeeingang           |
| ENO       | Output      | BOOL                  | BOOL                  | E, A, M, D, L                | E, A, M, D, L                | Freigabeausgang           |
| IN        | Input       | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Wert, der verschoben wird |

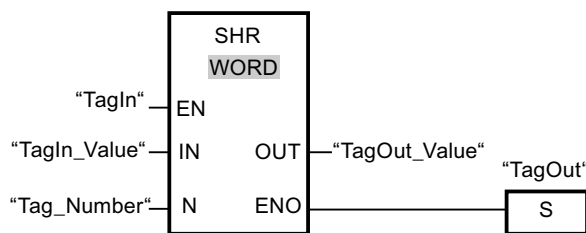
| Parameter | Deklaration | Datentyp              |                           | Speicherbereich              |                              | Beschreibung   |
|-----------|-------------|-----------------------|---------------------------|------------------------------|------------------------------|--|
|           |             | S7-1200               | S7-1500                   | S7-1200                      | S7-1500                      |  |
| N         | Input       | USINT, UINT, UDINT    | USINT, UINT, UDINT, ULINT | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Anzahl der Bitstellen, um die der Wert verschoben wird |
| OUT       | Output      | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L                | E, A, M, D, L                | Ergebnis der Anweisung                                 |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0011 1111 1010 1111 |
| N         | Tag_Number   | 3                   |
| OUT       | TagOut_Value | 0000 0111 1111 0101 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Rechts schieben" ausgeführt. Der Inhalt des Operanden "TagIn\_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## SHL: Links schieben

### Beschreibung

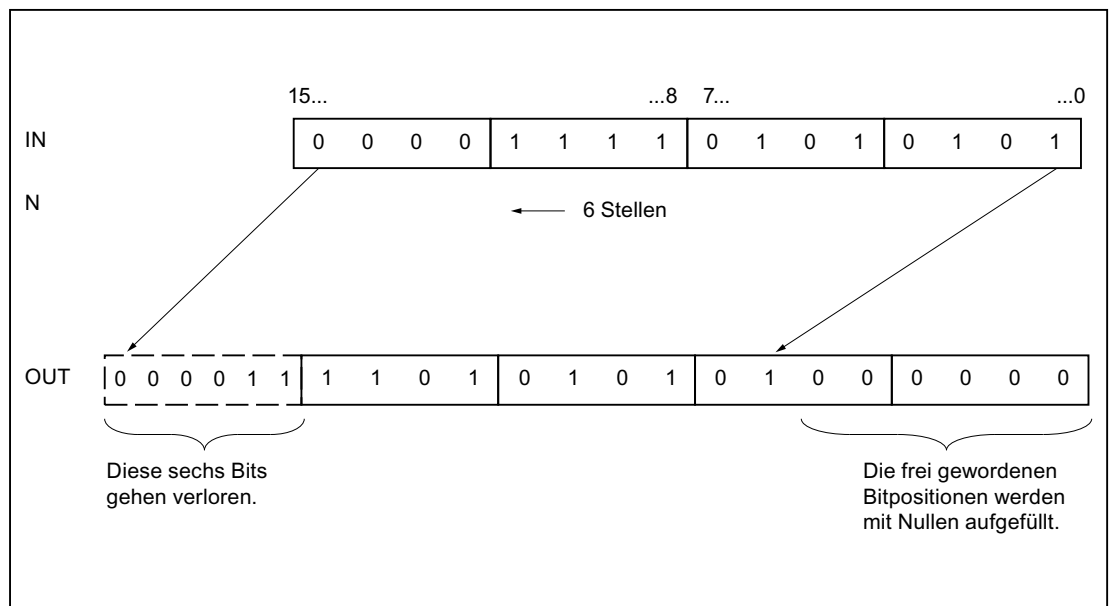
Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Eingang N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Eingang N "0" ist, wird der Wert am Eingang IN unverändert in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Eingang N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen im rechten Bereich des Operanden werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links schieben":

| Parameter | Deklaration | Datentyp |         | Speicherbereich |                     | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|---------------------|-----------------|
|           |             | S7-1200  | S7-1500 | S7-1200         | S7-1500             |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L, T, Z | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L       | Freigabeausgang |

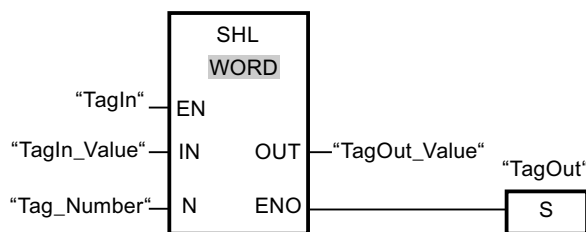
| Parameter | Deklaration | Datentyp              |                           | Speicherbereich              |                              | Beschreibung  |
|-----------|-------------|-----------------------|---------------------------|------------------------------|------------------------------|---|
|           |             | S7-1200               | S7-1500                   | S7-1200                      | S7-1500                      |   |
| IN        | Input       | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Wert, der verschoben wird.                              |
| N         | Input       | USINT, UINT, UDINT    | USINT, UINT, UDINT, ULINT | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Anzahl der Bitstellen, um die der Wert verschoben wird. |
| OUT       | Output      | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L                | E, A, M, D, L                | Ergebnis der Anweisung                                  |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0011 1111 1010 1111 |
| N         | Tag_Number   | 4                   |
| OUT       | TagOut_Value | 1111 1010 1111 0000 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Links schieben" ausgeführt. Der Inhalt des Operanden "TagIn\_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## ROR: Rechts rotieren

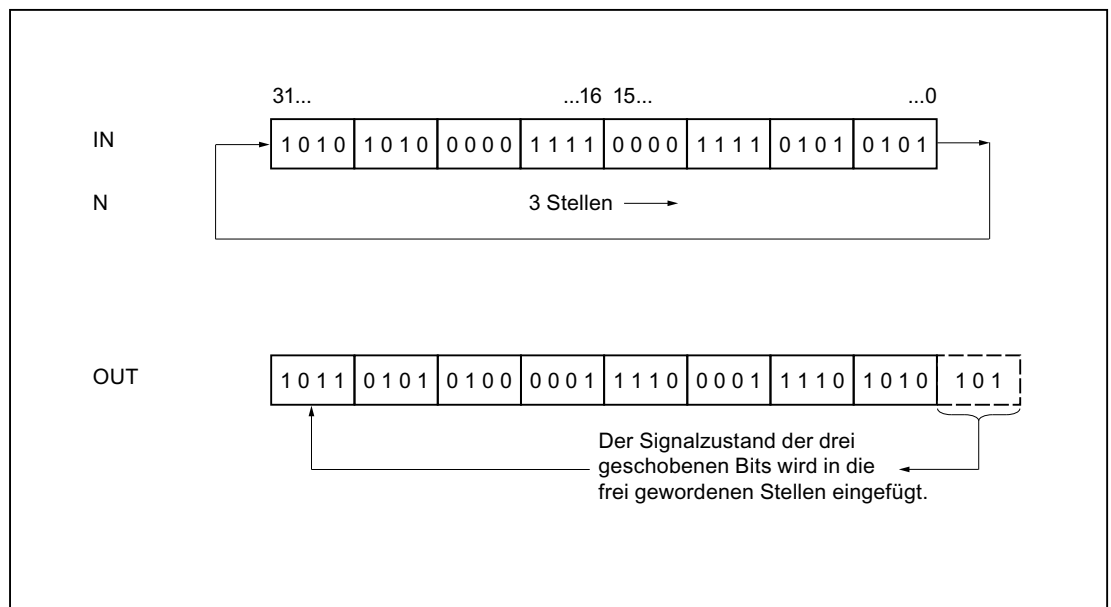
### Beschreibung

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach rechts und fragen das Ergebnis am Ausgang OUT ab. Mit dem Eingang N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen auf der linken Seite werden mit den hinausgeschobenen Bitstellen von der rechten Seite positionsgetreu aufgefüllt.

Wenn der Wert am Eingang N "0" ist, wird der Wert am Eingang IN unverändert in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts rotieren":

| Parameter | Deklaration | Datentyp |         | Speicherbereich |                     | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|---------------------|-----------------|
|           |             | S7-1200  | S7-1500 | S7-1200         | S7-1500             |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L, T, Z | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L       | Freigabeausgang |

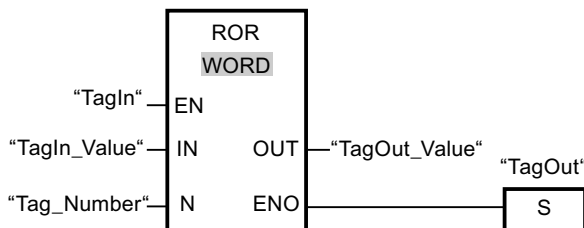
| Parameter | Deklaration | Datentyp              |                           | Speicherbereich              |                              | Beschreibung  |
|-----------|-------------|-----------------------|---------------------------|------------------------------|------------------------------|---|
|           |             | S7-1200               | S7-1500                   | S7-1200                      | S7-1500                      |   |
| IN        | Input       | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Wert, der rotiert wird                              |
| N         | Input       | USINT, UINT, UDINT    | USINT, UINT, UDINT, ULINT | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Anzahl der Bitstellen, um die der Wert rotiert wird |
| OUT       | Output      | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L                | E, A, M, D, L                | Ergebnis der Anweisung                              |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 0000 1111 1001 0101 |
| N         | Tag_Number   | 5                   |
| OUT       | TagOut_Value | 1010 1000 0111 1100 |

Wenn der Operand "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Rechts rotieren" ausgeführt. Der Inhalt des Operanden "TagIn\_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)



## ROL: Links rotieren

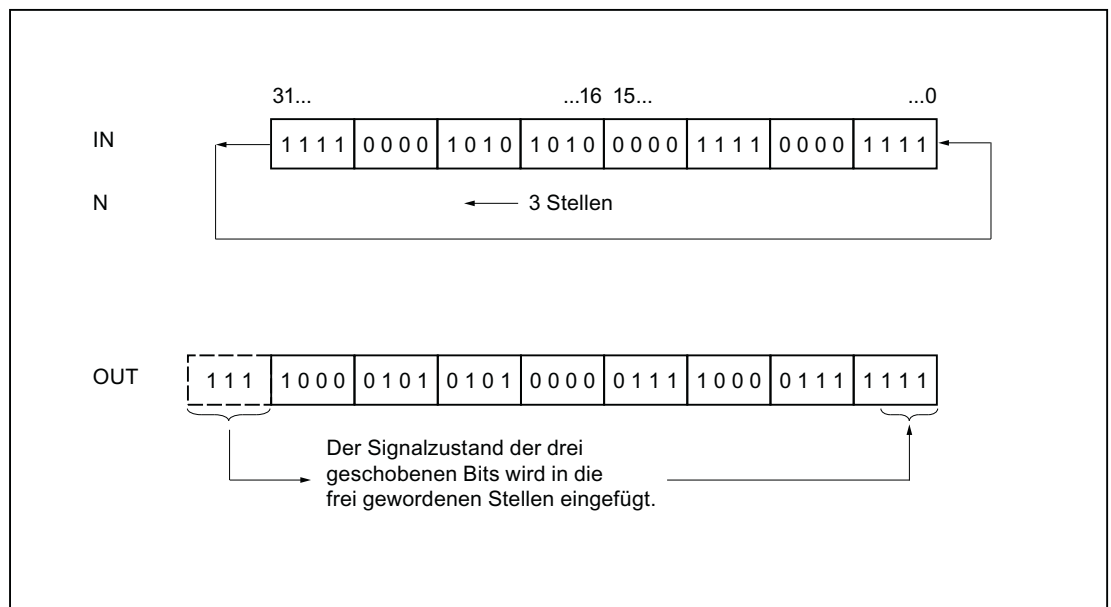
### Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Operanden am Eingang IN bitweise nach links und fragen das Ergebnis am Ausgang OUT ab. Mit dem Eingang N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen auf der rechten Seite werden mit den hinausgeschobenen Bitstellen von der linken Seite positionsgetreu aufgefüllt.

Wenn der Wert am Eingang N "0" ist, wird der Wert am Eingang IN in den Operanden am Ausgang OUT kopiert.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links rotieren":

| Parameter | Deklaration | Datentyp |         | Speicherbereich |                     | Beschreibung    |
|-----------|-------------|----------|---------|-----------------|---------------------|-----------------|
|           |             | S7-1200  | S7-1500 | S7-1200         | S7-1500             |                 |
| EN        | Input       | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L, T, Z | Freigabeeingang |
| ENO       | Output      | BOOL     | BOOL    | E, A, M, D, L   | E, A, M, D, L       | Freigabeausgang |

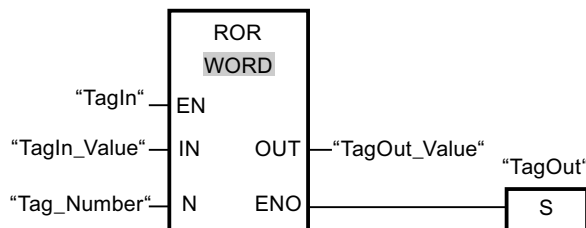
| Parameter | Deklaration | Datentyp              |                           | Speicherbereich              |                              | Beschreibung   |
|-----------|-------------|-----------------------|---------------------------|------------------------------|------------------------------|--|
|           |             | S7-1200               | S7-1500                   | S7-1200                      | S7-1500                      |  |
| IN        | Input       | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Wert, der rotiert wird.                              |
| N         | Input       | USINT, UINT, UDINT    | USINT, UINT, UDINT, ULINT | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Anzahl der Bitstellen, um die der Wert rotiert wird. |
| OUT       | Output      | Bitfolgen, Ganzzahlen | Bitfolgen, Ganzzahlen     | E, A, M, D, L                | E, A, M, D, L                | Ergebnis der Anweisung                               |

Aus der Klappliste "<???" der Anweisungsbox können Sie den Datentyp der Anweisung wählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand      | Wert                |
|-----------|--------------|---------------------|
| IN        | TagIn_Value  | 1010 1000 1111 0110 |
| N         | Tag_Number   | 5                   |
| OUT       | TagOut_Value | 0001 1110 1101 0101 |

Wenn der Eingang "TagIn" den Signalzustand "1" liefert, wird die Anweisung "Links rotieren" ausgeführt. Der Inhalt des Operanden "TagIn\_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis wird am Ausgang "TagOut\_Value" ausgegeben. Wenn keine Fehler während der Bearbeitung der Anweisung auftreten, liefert der Freigabeausgang ENO den Signalzustand "1" und der Ausgang "TagOut" wird gesetzt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## Weitere Anweisungen

### DRUM: Schrittschaltwerk realisieren

#### Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT\_WORD) mit den programmierten Werten des Parameters OUT\_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S\_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S\_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST\_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S\_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT\_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT\_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT\_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT\_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT\_WORD).

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

| Parameter               | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-------------------------|-------------|----------|------------------------------|---|
| EN                      | Input       | BOOL     | E, A, M, D, L, T, Z          | Freigabeeingang   |
| ENO                     | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang   |
| RESET                   | Input       | BOOL     | E, A, M, D, L oder Konstante | Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.  |
| JOG                     | Input       | BOOL     | E, A, M, D, L oder Konstante | Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.                        |
| DRUM_EN                 | Input       | BOOL     | E, A, M, D, L oder Konstante | Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen. |
| LST_STEP                | Input       | BYTE     | E, A, M, D, L oder Konstante | Nummer des zuletzt programmierten Schritts  |
| EVENT(i),<br>1 ≤ i ≤ 16 | Input       | BOOL     | E, A, M, D, L oder Konstante | Ereignisbit (i); Anfänglicher Signalzustand ist "1".  |
| OUT(j),<br>1 ≤ j ≤ 16   | Output      | BOOL     | E, A, M, D, L                | Ausgabebit (j)  |
| Q                       | Output      | BOOL     | E, A, M, D, L                | Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.                           |
| OUT_WORD                | Output      | WORD     | E, A, M, D, L, P             | Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.  |
| ERR_CODE                | Output      | WORD     | E, A, M, D, L, P             | Fehlerinformation   |
| JOG_HIS                 | Static      | BOOL     | E, A, M, D, L                | Verlaufsbit zum Parameter JOG   |

| Parameter | Deklaration | Datentyp      | Speicherbereich              | Beschreibung   |
|-----------|-------------|---------------|------------------------------|--|
| EOD       | Static      | BOOL          | E, A, M, D, L                | Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.              |
| DSP       | Static      | BYTE          | E, A, M, D, L, P             | Voreingestellter Schritt des Schrittschaltwerks  |
| DSC       | Static      | BYTE          | E, A, M, D, L, P             | Aktueller Schritt des Schrittschaltwerks   |
| DCC       | Static      | DWORD         | E, A, M, D, L, P             | Aktueller Zählwert des Schrittschaltwerks  |
| DTBP      | Static      | WORD          | E, A, M, D, L, P             | Voreingestellte Zeitbasis des Schrittschaltwerks   |
| PREV_TIME | Static      | DWORD         | E, A, M, D, L oder Konstante | Vorherige Systemzeit   |
| S_PRESET  | Static      | ARRAY of WORD | E, A, M, D, L                | Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.                             |
| OUT_VAL   | Static      | ARRAY of BOOL | E, A, M, D, L                | Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].   |
| S_MASK    | Static      | ARRAY of BOOL | E, A, M, D, L                | Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1". |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

| ERR_CODE*  | Erläuterung   |
|--|---|
| W#16#0000  | Kein Fehler   |
| W#16#000B  | Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.                        |
| W#16#000C  | Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP. |
| W#16#000D  | Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.           |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## DCAT: Diskreter Steuerungszeitalarm

### Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O\_FB bzw. C\_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD\_HIS, ET (nur wenn  $ET < PT$  ist), OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD\_HIS werden auf "1" gesetzt
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET (nur wenn  $ET < PT$  ist), OA, CA und CMD\_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD\_HIS der Signalzustand "1" anliegt und der Parameter O\_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Signalzustände der Parameter CMD, CMD\_HIS und O\_FB auf "1" gesetzt sind und der Parameter C\_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O\_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.

- Wenn die Parameter CMD, CMD\_HIS und C\_FB den Signalzustand "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD\_HIS und O\_FB den Signalzustand "0" liefern und der Parameter C\_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C\_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O\_FB und C\_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Diskreter Steuerungszeitalarm":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z          | Freigabeeingang  |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang  |
| CMD       | Input       | BOOL     | E, A, M, D, L oder Konstante | Der Signalzustand "0" gibt den Befehl "Schließen" an. Der Signalzustand "1" gibt den Befehl "Öffnen" an. |
| O_FB      | Input       | BOOL     | E, A, M, D, L oder Konstante | Feedback-Eingang beim Öffnen   |
| C_FB      | Input       | BOOL     | E, A, M, D, L oder Konstante | Feedback-Eingang beim Schließen  |
| Q         | Output      | BOOL     | E, A, M, D, L                | Zeigt den Status des Parameters CMD  |
| OA        | Output      | BOOL     | E, A, M, D, L                | Alarmausgang beim Öffnen   |
| CA        | Output      | BOOL     | E, A, M, D, L                | Alarmausgang beim Schließen  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                                 |
|-----------|-------------|----------|-----------------|--|
| ET        | Static      | DINT     | D, L            | Aktuelle, abgelaufene Zeit; ein Takt = 1 ms. |
| PT        | Static      | DINT     | D, L            | Voreingestellter Zeitwert; ein Takt = 1 ms.  |
| PREV_TIME | Static      | DWORD    | D, L            | Vorherige Systemzeit                         |
| CMD_HIS   | Static      | BOOL     | D, L            | Verlaufsbitt zu CMD                          |

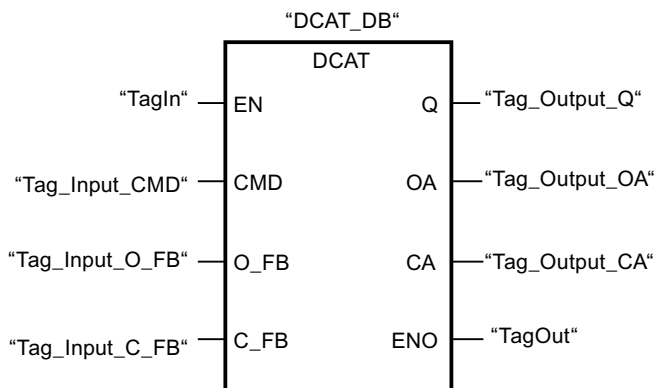
Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD\_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

#### Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.



Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

#### Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand        | Wert  |
|-----------|----------------|-------|
| CMD       | Tag_Input_CMD  | TRUE  |
| O_FB      | Tag_Input_O_FB | FALSE |
| C_FB      | Tag_Input_C_FB | FALSE |



| Parameter | Operand       | Wert  |
|-----------|---------------|-------|
| Q         | Tag_Output_Q  | FALSE |
| OA        | Tag_Output_OA | FALSE |
| CA        | Tag_Output_CA | FALSE |

Im Instanz-Datenbaustein "DCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| ET        | DBD4    | L#12  |
| PT        | DBD8    | L#222 |
| CMD_HIS   | DBX16.0 | FALSE |

### Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand       | Wert  |
|-----------|---------------|-------|
| Q         | Tag_Output_Q  | TRUE  |
| OA        | Tag_Output_OA | FALSE |
| CA        | Tag_Output_CA | FALSE |

Im Instanz-Datenbaustein "DCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert |
|-----------|---------|------|
| ET        | DBD4    | L#0  |
| CMD_HIS   | DBX16.0 | TRUE |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

## MCAT: Motorsteuerungszeitalarm

### Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der

Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Anweisung "Motorsteuerungszeitalarm" liefert keine Fehlerinformationen.

### Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

| Eingangsparameter |  |        |        |        |        |       |       | Ausgangsparameter |    |    |    |     |        |        |   |                   |
|-------------------|--|--------|--------|--------|--------|-------|-------|-------------------|----|----|----|-----|--------|--------|---|-------------------|
| ET                | O_H IS   | C_H IS | O_C MD | C_C MD | S_C MD | O_F B | C_F B | OO                | CO | OA | CA | ET  | O_H IS | C_H IS | Q | Zustand           |
| X                 | 1  | 1      | X      | X      | X      | X     | X     | 0                 | 0  | 1  | 1  | PT  | 0      | 0      | 0 | Alarm             |
| X                 | X  | X      | X      | X      | X      | 1     | 1     | 0                 | 0  | 1  | 1  | PT  | 0      | 0      | 0 | Alarm             |
| X                 | X  | X      | X      | X      | 1      | X     | X     | 0                 | 0  | 0  | 0  | X   | 0      | 0      | 1 | Stop              |
| X                 | X  | X      | 1      | 1      | X      | X     | X     | 0                 | 0  | 0  | 0  | X   | 0      | 0      | 1 | Stop              |
| X                 | 0  | X      | 1      | 0      | 0      | X     | X     | 1                 | 0  | 0  | 0  | 0   | 1      | 0      | 1 | Öffnen starten    |
| <PT               | 1  | 0      | X      | 0      | 0      | 0     | X     | 1                 | 0  | 0  | 0  | INC | 1      | 0      | 1 | Öffnen            |
| X                 | 1  | 0      | X      | 0      | 0      | 1     | 0     | 0                 | 0  | 0  | 0  | PT  | 1      | 0      | 1 | Geöffnet          |
| >= PT             | 1  | 0      | X      | 0      | 0      | 0     | X     | 0                 | 0  | 1  | 0  | PT  | 1      | 0      | 0 | Alarm öffnen      |
| X                 | X  | 0      | 0      | 1      | 0      | X     | X     | 0                 | 1  | 0  | 0  | 0   | 0      | 1      | 1 | Schließen starten |
| < PT              | 0  | 1      | 0      | X      | 0      | X     | 0     | 0                 | 1  | 0  | 0  | INC | 0      | 1      | 1 | Schließen         |
| X                 | 0  | 1      | 0      | X      | 0      | 0     | 1     | 0                 | 0  | 0  | 0  | PT  | 0      | 1      | 1 | Geschlossen       |
| >= PT             | 0  | 1      | 0      | X      | 0      | X     | 0     | 0                 | 0  | 0  | 1  | PT  | 0      | 1      | 0 | Alarm schließen   |
| X                 | 0  | 0      | 0      | 0      | 0      | X     | X     | 0                 | 0  | 0  | 0  | X   | 0      | 0      | 1 | Gestopt           |
| Legende:          |  |        |        |        |        |       |       |                   |    |    |    |     |        |        |   |                   |
| INC               | Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren |        |        |        |        |       |       |                   |    |    |    |     |        |        |   |                   |
| PT                | PT wird auf den gleichen Wert wie ET gesetzt                                 |        |        |        |        |       |       |                   |    |    |    |     |        |        |   |                   |
| X                 | Nicht anwendbar  |        |        |        |        |       |       |                   |    |    |    |     |        |        |   |                   |
| < PT              | ET < PT  |        |        |        |        |       |       |                   |    |    |    |     |        |        |   |                   |

| Eingangsparameter   |          | Ausgangsparameter |
|---|----------|-------------------|
| >= PT   | ET >= PT |                   |
| <p>Wenn die Eingangsparameter O_HIS und C_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O_HIS und C_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt:</p> <p>OO = FALSE<br/> CO = FALSE<br/> OA = FALSE<br/> CA = FALSE<br/> ET = PT<br/> Q = TRUE</p> |          |                   |

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z          | Freigabeeingang                                      |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang                                      |
| O_CMD     | Input       | BOOL     | E, A, M, D, L oder Konstante | Befehlseingabe "Öffnen"                              |
| C_CMD     | Input       | BOOL     | E, A, M, D, L oder Konstante | Befehlseingabe "Schließen"                           |
| S_CMD     | Input       | BOOL     | E, A, M, D, L oder Konstante | Befehlseingabe "Stoppen"                             |
| O_FB      | Input       | BOOL     | E, A, M, D, L oder Konstante | Feedback-Eingang beim Öffnen                         |
| C_FB      | Input       | BOOL     | E, A, M, D, L oder Konstante | Feedback-Eingang beim Schließen                      |
| OO        | Output      | BOOL     | E, A, M, D, L                | Ausgang "Öffnen"                                     |
| CO        | Output      | BOOL     | E, A, M, D, L                | Ausgang "Schließen"                                  |
| OA        | Output      | BOOL     | E, A, M, D, L                | Alarmausgang beim Öffnen                             |
| CA        | Output      | BOOL     | E, A, M, D, L                | Alarmausgang beim Schließen                          |
| Q         | Output      | BOOL     | E, A, M, D, L                | Der Signalzustand "0" zeigt eine Fehlerbedingung an. |
| ET        | Static      | DINT     | D, L                         | Aktuelle, abgelaufene Zeit; ein Takt = 1 ms          |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                               |
|-----------|-------------|----------|-----------------|--|
| PT        | Static      | DINT     | D, L            | Voreingestellter Zeitwert; ein Takt = 1 ms |
| PREV_TIME | Static      | DWORD    | D, L            | Vorherige Systemzeit                       |
| O_HIS     | Static      | BOOL     | D, L            | Verlaufsbit "Öffnen"                       |
| C_HIS     | Static      | BOOL     | D, L            | Verlaufsbit "Schließen"                    |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

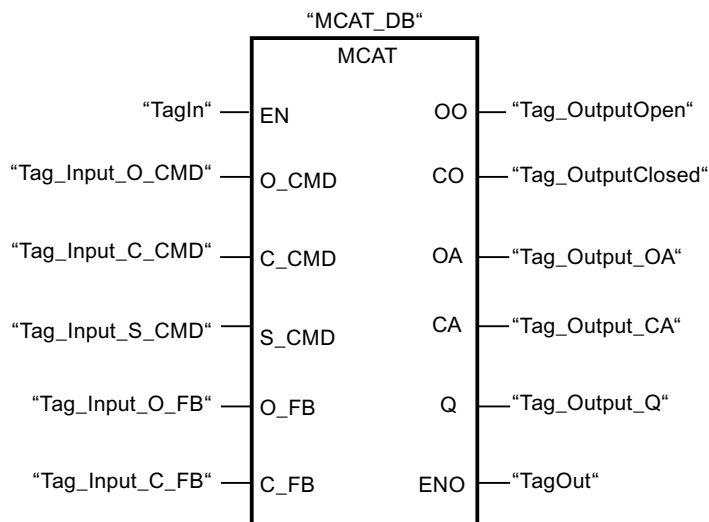
Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

---

#### Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

---



Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

#### Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand         | Wert  |
|-----------|-----------------|-------|
| O_CMD     | Tag_Input_O_CMD | TRUE  |
| C_CMD     | Tag_Input_C_CMD | FALSE |
| S_CMD     | Tag_Input_S_CMD | FALSE |
| O_FB      | Tag_Input_O_FB  | FALSE |

| Parameter | Operand          | Wert  |
|-----------|------------------|-------|
| C_FB      | Tag_Input_C_FB   | FALSE |
| OO        | Tag_OutputOpen   | FALSE |
| CO        | Tag_OutputClosed | FALSE |
| OA        | Tag_Output_OA    | FALSE |
| CA        | Tag_Output_CA    | FALSE |
| Q         | Tag_Output_Q     | FALSE |

Im Instanz-Datenbaustein "MCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| ET        | DBD4    | L#2   |
| PT        | DBD8    | L#22  |
| O_HIS     | DBX16.0 | TRUE  |
| C_HIS     | DBX16.1 | FALSE |

#### Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand          | Wert  |
|-----------|------------------|-------|
| OO        | Tag_OutputOpen   | TRUE  |
| CO        | Tag_OutputClosed | FALSE |
| OA        | Tag_Output_OA    | FALSE |
| CA        | Tag_Output_CA    | FALSE |
| Q         | Tag_Output_Q     | TRUE  |

Im Instanz-Datenbaustein "MCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| ET        | DBD4    | L#0   |
| O_HIS     | DBX16.0 | TRUE  |
| CMD_HIS   | DBX16.1 | FALSE |

#### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

#### IMC: Eingabebits mit den Bits einer Maske vergleichen

#### Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN\_BIT0 bis IN\_BIT15) mit dem

entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN\_BIT0 wird mit dem Wert der Maske CMP\_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP\_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP\_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR\_CODE wird eine Fehlermeldung ausgegeben.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung                                      |
|-----------|-------------|----------|------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z          | Freigabeeingang                                   |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang                                   |
| IN_BIT0   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 0 wird mit Bit 0 der Maske verglichen. |
| IN_BIT1   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 1 wird mit Bit 1 der Maske verglichen. |
| IN_BIT2   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 2 wird mit Bit 2 der Maske verglichen. |
| IN_BIT3   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 3 wird mit Bit 3 der Maske verglichen. |
| IN_BIT4   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 4 wird mit Bit 4 der Maske verglichen. |
| IN_BIT5   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 5 wird mit Bit 5 der Maske verglichen. |
| IN_BIT6   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 6 wird mit Bit 6 der Maske verglichen. |
| IN_BIT7   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 7 wird mit Bit 7 der Maske verglichen. |
| IN_BIT8   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 8 wird mit Bit 8 der Maske verglichen. |
| IN_BIT9   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 9 wird mit Bit 9 der Maske verglichen. |

| Parameter | Deklaration | Datentyp      | Speicherbereich                 | Beschreibung   |
|-----------|-------------|---------------|---------------------------------|--|
| IN_BIT10  | Input       | BOOL          | E, A, M, D, L oder Konstante    | Eingabebit 10 wird mit Bit 10 der Maske verglichen.  |
| IN_BIT11  | Input       | BOOL          | E, A, M, D, L oder Konstante    | Eingabebit 11 wird mit Bit 11 der Maske verglichen.  |
| IN_BIT12  | Input       | BOOL          | E, A, M, D, L oder Konstante    | Eingabebit 12 wird mit Bit 12 der Maske verglichen.  |
| IN_BIT13  | Input       | BOOL          | E, A, M, D, L oder Konstante    | Eingabebit 13 wird mit Bit 13 der Maske verglichen.  |
| IN_BIT14  | Input       | BOOL          | E, A, M, D, L oder Konstante    | Eingabebit 14 wird mit Bit 14 der Maske verglichen.  |
| IN_BIT15  | Input       | BOOL          | E, A, M, D, L oder Konstante    | Eingabebit 15 wird mit Bit 15 der Maske verglichen.  |
| CMP_STEP  | Input       | BYTE          | E, A, M, D, L, P oder Konstante | Schrittnummer der Maske, mit der verglichen wird.  |
| OUT       | Output      | BOOL          | E, A, M, D, L                   | Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde.<br>Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.            |
| ERR_CODE  | Output      | WORD          | E, A, M, D, L, P                | Fehlerinformation  |
| CMP_VAL   | Static      | ARRAY OF WORD | E, A, M, D, L                   | Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE ausgegeben werden:

| Fehlercode* (W#16#...)   | Erläuterung                                       |
|--|---|
| 0000   | Kein Fehler                                       |
| 000A   | Der Wert am Parameter CMP_STEP ist größer als 15. |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

**SMC: Matrixscanner**

**Beschreibung**

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN\_BIT0 bis IN\_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Übereinstimmung gefunden wird. Das Eingabebit des Parameters IN\_BIT0 wird mit dem Wert der Maske CMP\_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT\_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Übereinstimmung in den Parameter OUT\_STEP angegeben. Wird keine Übereinstimmung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT\_STEP um "1" größer als der Wert am Parameter LAST.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung                                      |
|-----------|-------------|----------|------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z          | Freigabeeingang                                   |
| ENO       | Output      | BOOL     | E, A, M, D, L                | Freigabeausgang                                   |
| IN_BIT0   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 0 wird mit Bit 0 der Maske verglichen. |
| IN_BIT1   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 1 wird mit Bit 1 der Maske verglichen. |
| IN_BIT2   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 2 wird mit Bit 2 der Maske verglichen. |
| IN_BIT3   | Input       | BOOL     | E, A, M, D, L oder Konstante | Eingabebit 3 wird mit Bit 3 der Maske verglichen. |



| Parameter | Deklaration | Datentyp | Speicherbereich                 | Beschreibung  |
|-----------|-------------|----------|---------------------------------|---|
| IN_BIT4   | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 4 wird mit Bit 4 der Maske verglichen.   |
| IN_BIT5   | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 5 wird mit Bit 5 der Maske verglichen.   |
| IN_BIT6   | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 6 wird mit Bit 6 der Maske verglichen.   |
| IN_BIT7   | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 7 wird mit Bit 7 der Maske verglichen.   |
| IN_BIT8   | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 8 wird mit Bit 8 der Maske verglichen.   |
| IN_BIT9   | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 9 wird mit Bit 9 der Maske verglichen.   |
| IN_BIT10  | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 10 wird mit Bit 10 der Maske verglichen.   |
| IN_BIT11  | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 11 wird mit Bit 11 der Maske verglichen.   |
| IN_BIT12  | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 12 wird mit Bit 12 der Maske verglichen.   |
| IN_BIT13  | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 13 wird mit Bit 13 der Maske verglichen.   |
| IN_BIT14  | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 14 wird mit Bit 14 der Maske verglichen.   |
| IN_BIT15  | Input       | BOOL     | E, A, M, D, L<br>oder Konstante | Eingabebit 15 wird mit Bit 15 der Maske verglichen.   |
| OUT       | Output      | BOOL     | E, A, M, D, L                   | Der Signalzustand "1" zeigt an, dass eine Übereinstimmung gefunden wurde.<br>Der Signalzustand "0" zeigt an, dass keine Übereinstimmung gefunden wurde.                           |
| ERR_CODE  | Output      | WORD     | E, A, M, D, L, P                | Fehlerinformation   |
| OUT_STEP  | Output      | BYTE     | E, A, M, D, L, P                | Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Übereinstimmung gefunden wurde. |

| Parameter | Deklaration | Datentyp      | Speicherbereich  | Beschreibung   |
|-----------|-------------|---------------|------------------|--|
| LAST      | Static      | BYTE          | E, A, M, D, L, P | Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.   |
| CMP_VAL   | Static      | ARRAY OF WORD | E, A, M, D, L    | Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

| Fehlercode*<br>(W#16#...)  | Erläuterung                                   |
|--|---|
| 0000   | Kein Fehler                                   |
| 000E   | Der Wert am Parameter LAST ist größer als 15. |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### LEAD\_LAG: Lead- und Lag-Algorithmus

#### Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[ \frac{LG\_TIME}{LG\_TIME + SAMPLE\_T} \right] PREV\_OUT + GAIN \left[ \frac{LD\_TIME + SAMPLE\_T}{LG\_TIME + SAMPLE\_T} \right] IN - GAIN \left[ \frac{LD\_TIME}{LG\_TIME + SAMPLE\_T} \right] * PREV\_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD\_TIME, LG\_TIME und SAMPLE\_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei  $LG\_TIME > 4 + SAMPLE\_T$  an folgende Funktion an:

$$\text{OUT} = \text{GAIN} * ((1 + \text{LD\_TIME} * s) / (1 + \text{LG\_TIME} * s)) * \text{IN}$$

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR\_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Lead- und Lag-Algorithmus":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung  |
|-----------|-------------|----------|------------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z                | Freigabeeingang   |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang   |
| IN        | Input       | REAL     | E, A, M, D, L, P<br>oder Konstante | Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird. |
| SAMPLE_T  | Input       | INT      | E, A, M, D, L, P<br>oder Konstante | Abtastzeit  |
| OUT       | Output      | REAL     | E, A, M, D, L, P                   | Ergebnis der Anweisung  |
| ERR_CODE  | Output      | WORD     | E, A, M, D, L, P                   | Fehlerinformation   |
| LD_TIME   | Static      | REAL     | E, A, M, D, L, P<br>oder Konstante | Voreilzeit in der gleichen Einheit wie die Abtastzeit.                  |
| LG_TIME   | Static      | REAL     | E, A, M, D, L, P<br>oder Konstante | Nacheilzeit in der gleichen Einheit wie die Abtastzeit                  |

| Parameter | Deklaration | Datentyp | Speicherbereich                 | Beschreibung   |
|-----------|-------------|----------|---------------------------------|--|
| GAIN      | Static      | REAL     | E, A, M, D, L, P oder Konstante | Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand). |
| PREV_IN   | Static      | REAL     | E, A, M, D, L, P oder Konstante | Vorheriger Eingang   |
| PREV_OUT  | Static      | REAL     | E, A, M, D, L, P oder Konstante | Vorheriger Ausgang   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

| Fehlercode* (W#16#...) | Erläuterung  |
|------------------------|--|
| 0000                   | Kein Fehler  |
| 0009                   | Der Wert am Parameter GAIN ist kleiner oder gleich Null. |

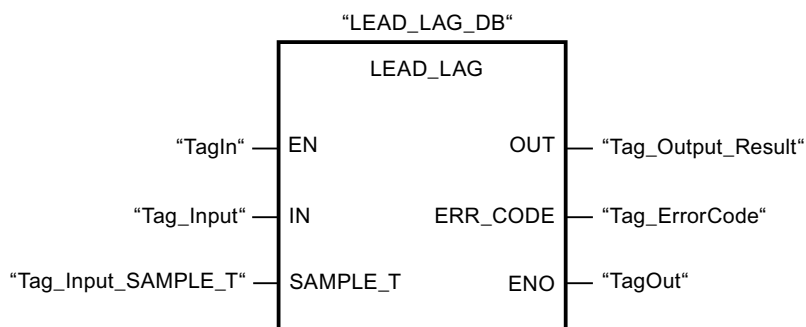
\*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

#### Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.



Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

#### Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand             | Wert |
|-----------|---------------------|------|
| IN        | Tag_Input           | 2.0  |
| SAMPLE_T  | Tag_InputSampleTime | 10   |

Im Instanz-Datenbaustein "LEAD\_LAG\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert |
|-----------|---------|------|
| LD_TIME   | DBD12   | 2.0  |
| LG_TIME   | DBD16   | 2.0  |
| GAIN      | DBD20   | 1.0  |
| PREV_IN   | DBD24   | 6.0  |
| PREV_OUT  | DBD28   | 6.0  |

#### Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand           | Wert |
|-----------|-------------------|------|
| OUT       | Tag_Output_Result | 2.0  |

Im Instanz-Datenbaustein "LEAD\_LAD\_DB" der Anweisung werden die folgenden Werte gespeichert:

| Parameter | Operand | Wert |
|-----------|---------|------|
| PREV_IN   | DBD24   | 2.0  |
| PREV_OUT  | DBD28   | 2.0  |

#### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### SEG: Bitmuster für 7-Segment-Anzeige erzeugen

#### Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

| Eingangsziffer<br>r<br>(Binär) | Belegung der<br>Segmente<br>- g f e d c b a | Anzeige<br>(Hexadezimal) | Sieben-Segment-Anzeige |
|--------------------------------|---|--------------------------|------------------------|
| 0000                           | 00111111                                    | 0                        |                        |
| 0001                           | 00000110                                    | 1                        |                        |
| 0010                           | 01011011                                    | 2                        |                        |
| 0011                           | 01001111                                    | 3                        |                        |
| 0100                           | 01100110                                    | 4                        |                        |
| 0101                           | 01101101                                    | 5                        |                        |
| 0110                           | 01111101                                    | 6                        |                        |
| 0111                           | 00000111                                    | 7                        |                        |
| 1000                           | 01111111                                    | 8                        |                        |
| 1001                           | 01100111                                    | 9                        |                        |
| 1010                           | 01110111                                    | A                        |                        |
| 1011                           | 01111100                                    | B                        |                        |
| 1100                           | 00111001                                    | C                        |                        |
| 1101                           | 01011110                                    | D                        |                        |
| 1110                           | 01111001                                    | E                        |                        |
| 1111                           | 01110001                                    | F                        |                        |

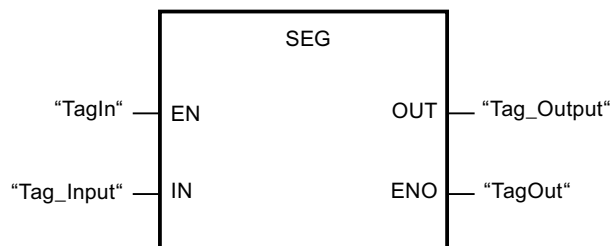
**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung                             |
|-----------|-------------|----------|------------------------------------|--|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z                | Freigabeeingang                          |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang                          |
| IN        | Input       | WORD     | E, A, M, D, L, P<br>oder Konstante | Quellwort mit vier<br>Hexadezimalziffern |
| OUT       | Output      | DWORD    | E, A, M, D, L, P                   | Bitmuster für die 7-<br>Segment-Anzeige  |

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

| Parameter   | Operand    | Wert          |   |
|-------------|------------|---------------|---|
| Hexadezimal | Binär      |               |   |
| IN          | Tag_Input  | W#16#1234     | 0001 0010 0011 0100                                     |
| OUT         | Tag_Output | DW#16065B4F66 | 00000110 01011011 01001111<br>01100110<br>Anzeige: 1234 |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### BCDCPL: Zehnerkomplement erzeugen

#### Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Parameter IN angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

$$\begin{array}{r}
 10000000 \text{ (als BCD)} \\
 - \text{ 7-stelliger BCD-Wert} \\
 \hline
 \text{Zehnerkomplement (als BCD)}
 \end{array}$$

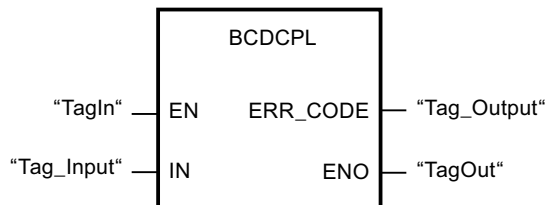
#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung           |
|-----------|-------------|----------|------------------------------------|------------------------|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z                | Freigabeeingang        |
| ENO       | Output      | BOOL     | E, A, M, D, L                      | Freigabeausgang        |
| IN        | Input       | DWORD    | E, A, M, D, L, P<br>oder Konstante | 7-stellige BCD-Zahl    |
| ERR_CODE  | Output      | DWORD    | E, A, M, D, L, P                   | Ergebnis der Anweisung |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

| Parameter  | Operand    | Wert*          |
|--|------------|----------------|
| IN   | Tag_Input  | DW#16#01234567 |
| ERR_CODE   | Tag_Output | DW#16#08765433 |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |            |                |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### BITSUM: Anzahl der gesetzten Bits zählen

#### Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bits eines Operanden, die auf den Signalzustand "1" gesetzt sind. Der Operand, dessen Bits gezählt werden, geben Sie am Parameter IN an. Das Ergebnis der Anweisung wird am Parameter RET\_VAL ausgegeben.

#### Parameter

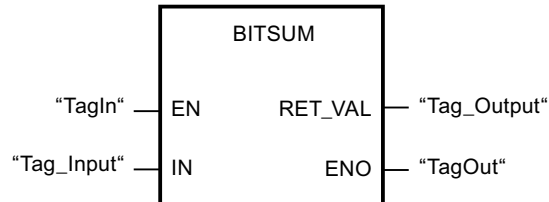
Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

| Parameter | Deklaration | Datentyp | Speicherbereich                 | Beschreibung                                  |
|-----------|-------------|----------|---------------------------------|---|
| EN        | Input       | BOOL     | E, A, M, D, L, T, Z             | Freigabeeingang                               |
| ENO       | Output      | BOOL     | E, A, M, D, L                   | Freigabeausgang                               |
| IN        | Input       | DWORD    | E, A, M, D, L, P oder Konstante | Operand, dessen gesetzte Bits gezählt werden. |
| RET_VAL   | Output      | INT      | E, A, M, D, L, P                | Anzahl der gesetzten Bits                     |



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:



Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

| Parameter  | Operand    | Wert*               |
|--|------------|---------------------|
| IN   | Tag_Input  | DW#16#12345678      |
| RET_VAL  | Tag_Output | W#16#000D (13 Bits) |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |            |                     |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Grundlagen zum EN-/ENO-Mechanismus (Seite 1180)

### 9.7.2.3 SCL

## Bitverknüpfungen

### R\_TRIG: Variable bei positiver Signalfanke setzen

## Beschreibung

Mit der Anweisung "Variable bei positiver Signalfanke setzen" setzen Sie eine angegebene Variable im Instanz-DB, wenn eine Änderung im Verknüpfungsergebnis (VKE) von "0" auf "1" vorliegt. Die Anweisung vergleicht das aktuelle VKE am Eingang CLK mit dem VKE der vorherigen Abfrage, die im angegebenen Instanz-DB gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "0" auf "1" erkennt, liegt eine positive Signalfanke vor.

Wenn eine positive Signalfanke erfasst wird, wird die Variable im Instanz-DB auf den Signalzustand "1" gesetzt und der Ausgang Q liefert den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der Flankenmerker in einem eigenen Datenbaustein (Einzelnstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

### Syntax

Für die Anweisung "Variable bei positiver Signalfanke setzen" wird die folgende Syntax verwendet:

```
SCL
<Instanz_DB>(CLK := <Operand>,
             Q => <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Variable bei positiver Signalfanke setzen":

| Parameter | Deklaration | Datentyp | Beschreibung                                     |
|-----------|-------------|----------|--|
| CLK       | Input       | BOOL     | Eingehendes Signal, dessen Flanke abgefragt wird |
| Q         | Output      | BOOL     | Ergebnis der Flankenauswertung                   |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"R_TRIG_DB"(CLK := "TagIn",
            Q => "TagOut");
```

In der Variablen im Instanz-DB "R\_TRIG\_DB" wird das VKE der vorherigen Abfrage gespeichert. Wenn im Operand "TagIn" eine Änderung im Signalzustand des VKE von "0" auf "1" erfasst wird, liefert der Ausgang "TagOut" den Signalzustand "1".

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## F\_TRIG: Variable bei negativer Signalflanke setzen

### Beschreibung

Mit der Anweisung "Variable bei negativer Signalflanke setzen" setzen Sie eine angegebene Variable im Instanz-DB, wenn eine Änderung im Verknüpfungsergebnis (VKE) von "1" auf "0" vorliegt. Die Anweisung vergleicht das aktuelle VKE am Eingang CLK mit dem VKE der vorherigen Abfrage, die im angegebenen Instanz-DB gespeichert ist. Wenn die Anweisung einen Wechsel im VKE von "1" auf "0" erkennt, liegt eine negative Signalflanke vor.

Wenn eine negative Signalflanke erfasst wird, wird die Variable im Instanz-DB auf den Signalzustand "1" gesetzt und der Ausgang Q liefert den Signalzustand "1". In allen anderen Fällen ist der Signalzustand am Ausgang der Anweisung "0".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der Flankenmerker in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

### Syntax

Für die Anweisung "Variable bei negativer Signalflanke setzen" wird die folgende Syntax verwendet:

```
SCL
<Instanz_DB>(CLK := <Operand>,
             Q => <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Variable bei negativer Signalflanke setzen":

| Parameter | Deklaration | Datentyp | Beschreibung                                     |
|-----------|-------------|----------|--|
| CLK       | Input       | BOOL     | Eingehendes Signal, dessen Flanke abgefragt wird |
| Q         | Output      | BOOL     | Ergebnis der Flankenauswertung                   |

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"F_TRIG_DB"(CLK := "TagIn",
            Q => "TagOut");
```

In der Variablen im Instanz-DB "F\_TRIG\_DB" wird das VKE der vorherigen Abfrage gespeichert. Wenn im Operand "TagIn" eine Änderung im Signalzustand des VKE von "1" auf "0" erfasst wird, liefert der Ausgang "TagOut" den Signalzustand "1".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

### Zeiten

#### IEC-Zeiten

#### TP: Impuls erzeugen

#### Beschreibung

Mit der Anweisung "Impuls erzeugen" setzen Sie den Parameter Q für die Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Der Parameter Q wird für die Zeitdauer PT gesetzt, unabhängig vom weiteren Verlauf des Eingangssignals. Auch die Erfassung einer neuen positiven Signalflanke beeinflusst den Signalzustand am Parameter Q nicht, solange die Zeitdauer PT läuft.

Am Parameter ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Wenn die Zeitdauer PT erreicht ist und der Signalzustand am Parameter IN "0" ist, wird der Parameter ET zurückgesetzt.

---

#### Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

Jedem Aufruf der Anweisung "Impuls erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

## Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TP\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME im Abschnitt "Static" eines Bausteins (z. B. #MyTP\_TIMER)

## Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TP\_TIME oder TP\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TP\_TIME oder TP\_LTIME im Abschnitt "Static" eines Bausteins (z. B. #MyTP\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstantz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

## Syntax

Für die Anweisung "Impuls erzeugen" wird die folgende Syntax verwendet:

- Datenbaustein vom Systemdatentyp IEC\_Zeit (Shared DB):

```
SCL  
<IEC_Zeit_DB>.TP(IN := <Operand>,  
                PT := <Operand>,  
                Q => <Operand>,  
                ET => <Operand>)
```

- Lokale Variable:

```
SCL  
#myLocal_timer(IN := <Operand>,  
              PT := <Operand>,  
              Q => <Operand>,  
              ET => <Operand>)
```

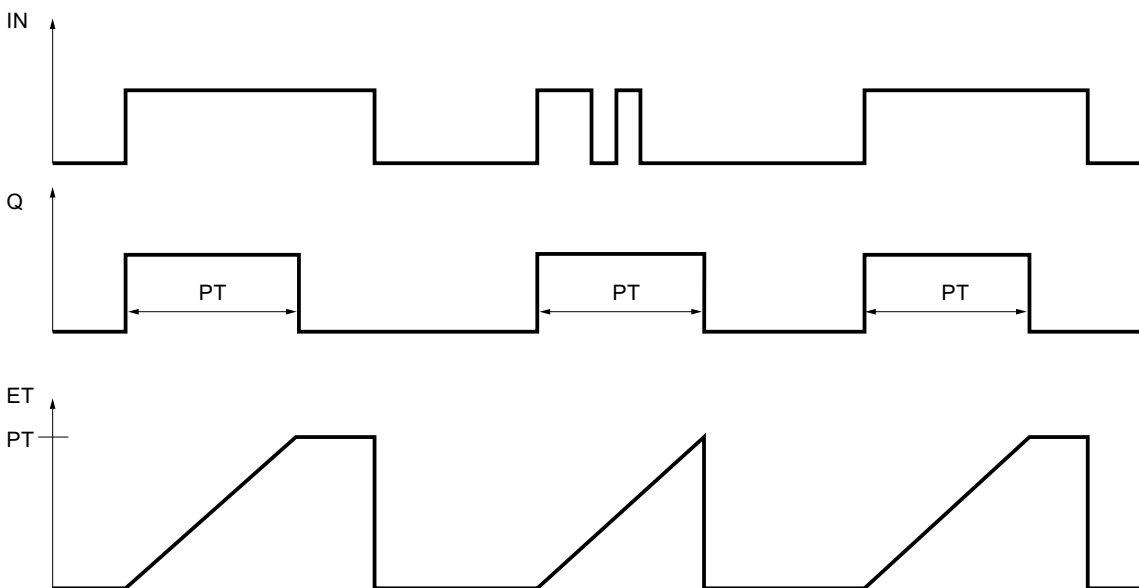
Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp |             | Beschreibung   |
|-----------|-------------|----------|-------------|--|
|           |             | S7-1200  | S7-1500     |  |
| IN        | Input       | BOOL     | BOOL        | Starteingang   |
| PT        | Input       | TIME     | TIME, LTIME | Zeitdauer des Impulses.<br>Der Wert am Parameter PT muss positiv sein. |
| Q         | Output      | BOOL     | BOOL        | Operand, der für die Zeitdauer PT gesetzt wird.                        |
| ET        | Output      | TIME     | TIME, LTIME | Aktueller Zeitwert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Impuls erzeugen":



### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"TP_DB".TP(IN := "Tag_Start",
           PT := "Tag_PresetTime",
           Q => "Tag_Status",
           ET => "Tag_ElapsedTime");
    
```

Wenn der Signalzustand des Operanden "Tag\_Start" von "0" auf "1" wechselt, wird die programmierte Zeitdauer am Parameter PT gestartet und der Operand "Tag\_Status" auf "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag\_ElapsedTime" gespeichert.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## TON: Einschaltverzögerung erzeugen

### Beschreibung

Mit der Anweisung "Einschaltverzögerung erzeugen" verzögern Sie das Setzen des Parameters Q um die programmierte Zeitdauer PT. Die Anweisung wird gestartet, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalflanke). Mit dem Start der Anweisung läuft die programmierte Zeitdauer PT ab. Wenn die Zeitdauer PT abgelaufen ist, liefert der Parameter Q den Signalzustand "1". Der Parameter Q bleibt so lange gesetzt, wie der Starteingang noch "1" führt. Wenn der Signalzustand am Parameter IN von "1" auf "0" wechselt, wird der Parameter Q zurückgesetzt. Die Zeitfunktion wird wieder gestartet, wenn eine neue positive Signalflanke am Parameter IN erfasst wird.

Am Parameter ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeit PT erreicht ist. Der Parameter ET wird zurückgesetzt, sobald der Signalzustand am Parameter IN auf "0" wechselt.

---

### Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

Jedem Aufruf der Anweisung "Einschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

## Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TON\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME im Abschnitt "Static" eines Bausteins (z. B. #MyTON\_TIMER)

### Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TON\_TIME oder TON\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TON\_TIME oder TON\_LTIME im Abschnitt "Static" eines Bausteins (z. B. #MyTON\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

### Syntax

Für die Anweisung "Einschaltverzögerung erzeugen" wird die folgende Syntax verwendet:

- Datenbaustein vom Systemdatentyp IEC\_Zeit (Shared DB):

**SCL**

```
<IEC_Zeit_DB>.TON(IN := <Operand>,
                 PT := <Operand>,
                 Q => <Operand>,
                 ET => <Operand>)
```

- Lokale Variable:

**SCL**

```
#myLocal_timer(IN := <Operand>,
               PT := <Operand>,
               Q => <Operand>,
               ET => <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp |             | Beschreibung   |
|-----------|-------------|----------|-------------|--|
|           |             | S7-1200  | S7-1500     |  |
| IN        | Input       | BOOL     | BOOL        | Starteingang   |
| PT        | Input       | TIME     | TIME, LTIME | Zeitdauer der Einschaltverzögerung.<br>Der Wert am Parameter PT muss positiv sein. |

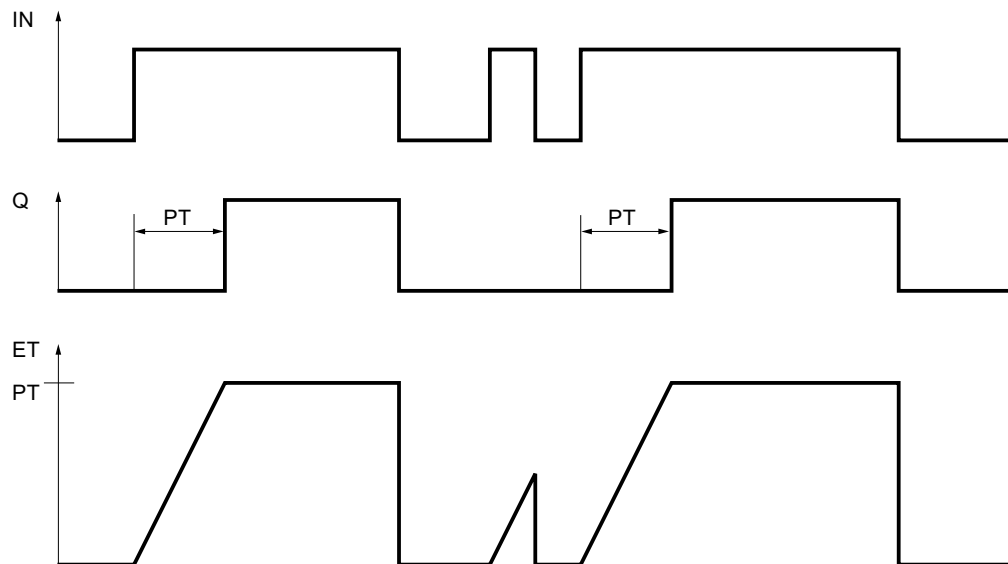


| Parameter | Deklaration | Datentyp |             | Beschreibung   |
|-----------|-------------|----------|-------------|--|
|           |             | S7-1200  | S7-1500     |  |
| Q         | Output      | BOOL     | BOOL        | Operand, der nach dem Ablauf der Zeit PT gesetzt wird. |
| ET        | Output      | TIME     | TIME, LTIME | Aktueller Zeitwert                                     |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Einschaltverzögerung erzeugen":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Wenn der Signalzustand des Operanden "Tag\_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Nach dem Ablauf der Zeit wird der Operand "Tag\_Status" auf den Signalzustand "1" gesetzt. Der Operand "Tag\_Status" bleibt solange auf "1" gesetzt, wie der Operand "Tag\_Start" den Signalzustand "1" führt. Der aktuelle Zeitwert wird im Operanden "Tag\_ElapsedTime" abgelegt. Wenn der Signalzustand am Operand "Tag\_Start" von "1" auf "0" wechselt, wird der Operand "Tag\_Status" zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## TOF: Ausschaltverzögerung erzeugen

### Beschreibung

Mit der Anweisung "Ausschaltverzögerung erzeugen" verzögern Sie das Zurücksetzen des Parameters Q um die programmierte Zeitdauer PT. Der Parameter Q wird gesetzt, wenn das Verknüpfungsergebnis (VKE) am Parameter IN von "0" auf "1" wechselt (positive Signalfanke). Wenn der Signalzustand am Parameter IN wieder auf "0" wechselt, läuft die programmierte Zeitdauer PT ab. Der Parameter Q bleibt gesetzt, solange die Zeitdauer PT läuft. Nach dem Ablauf der Zeit PT wird der Parameter Q zurückgesetzt. Falls der Signalzustand am Parameter IN auf "1" wechselt, bevor die Zeitdauer PT abgelaufen ist, wird die Zeit zurückgesetzt. Der Signalzustand am Parameter Q bleibt weiterhin auf "1" gesetzt.

Am Parameter ET kann der aktuelle Zeitwert abgefragt werden. Der Zeitwert beginnt bei T#0s und endet, wenn der Wert der Zeitdauer PT erreicht ist. Nach dem Ablauf der Zeitdauer PT bleibt der Parameter ET so lange auf dem aktuellen Wert stehen, bis der Parameter IN wieder auf "1" wechselt. Wenn der Parameter IN vor dem Ablauf der Zeit PT auf "1" wechselt, wird der Parameter ET auf den Wert T#0s zurückgesetzt.

---

#### Hinweis

Wenn die Zeit im Programm nicht aufgerufen wird, weil sie z. B. übersprungen wird, liefert der Ausgang ET, sobald die Zeit abgelaufen ist, einen Konstantenwert.

---

Jedem Aufruf der Anweisung "Ausschaltverzögerung erzeugen" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

### Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TOF\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME im Abschnitt "Static" eines Bausteins (z. B. #MyTOF\_TIMER)

## Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TOF\_TIME oder TOF\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TOF\_TIME oder TOF\_LTIME im Abschnitt "Static" eines Bausteins (z. B. #MyTOF\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstantz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

## Syntax

Für die Anweisung "Ausschaltverzögerung erzeugen" wird die folgende Syntax verwendet:

- Datenbaustein vom Systemdatentyp IEC\_Zeit (Shared DB):

### SCL

```
<IEC_Zeit_DB>.TOF(IN := <Operand>,
                 PT := <Operand>,
                 Q => <Operand>,
                 ET => <Operand>)
```

- Lokale Variable:

### SCL

```
#myLocal_timer(IN := <Operand>,
               PT := <Operand>,
               Q => <Operand>,
               ET => <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

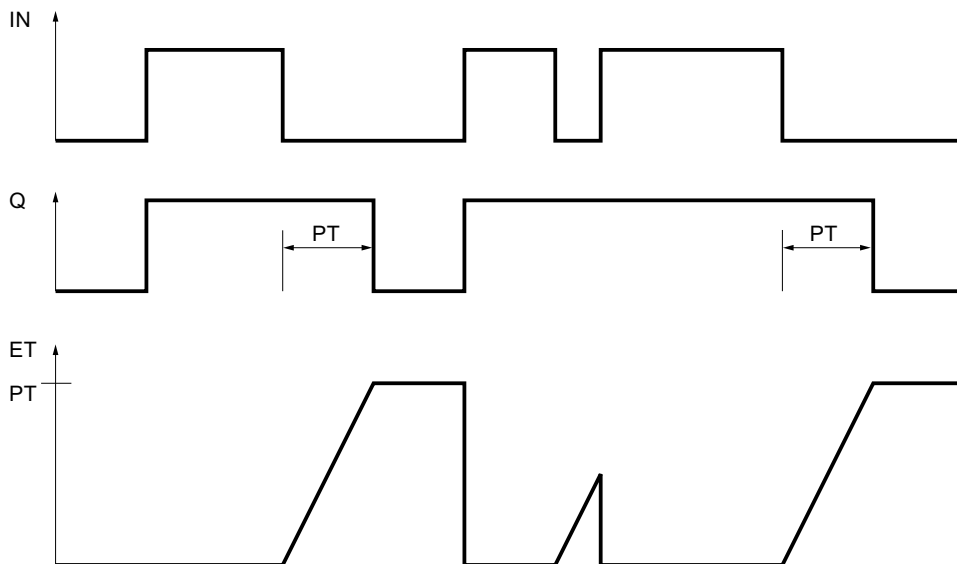
| Parameter | Deklaration | Datentyp |             | Beschreibung   |
|-----------|-------------|----------|-------------|--|
|           |             | S7-1200  | S7-1500     |  |
| IN        | Input       | BOOL     | BOOL        | Starteingang   |
| PT        | Input       | TIME     | TIME, LTIME | Zeitdauer der Ausschaltverzögerung.<br>Der Wert am Parameter PT muss positiv sein. |

| Parameter | Deklaration | Datentyp |             | Beschreibung   |
|-----------|-------------|----------|-------------|--|
|           |             | S7-1200  | S7-1500     |  |
| Q         | Output      | BOOL     | BOOL        | Operand, der nach dem Ablauf der Zeit PT zurückgesetzt wird. |
| ET        | Output      | TIME     | TIME, LTIME | Aktueller Zeitwert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Ausschaltverzögerung erzeugen":



### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"TOF_DB".TOF(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
    
```

Bei einem Wechsel im Signalzustand des Operanden "Tag\_Start" von "0" auf "1" wird der Operand "Tag\_Status" gesetzt. Wenn der Signalzustand des Operanden "Tag\_Start" von "1" auf "0" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Solange die Zeit läuft, bleibt der Operand "Tag\_Status" gesetzt. Nach dem Ablauf der Zeit wird der Operand "Tag\_Status" zurückgesetzt. Der aktuelle Zeitwert wird im Operanden "Tag\_ElapsedTime" abgelegt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## TONR: Zeit akkumulieren

### Beschreibung

Mit der Anweisung "Zeit akkumulieren" akkumulieren Sie Zeitwerte innerhalb eines durch den Parameter PT gesetzten Zeitraums. Wenn der Signalzustand am Parameter IN auf "1" wechselt, wird die Anweisung ausgeführt und die Zeitdauer PT gestartet. Während des Ablaufs der Zeitdauer PT werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" am Parameter IN erfasst werden. Die akkumulierte Zeit wird am Parameter ET ausgegeben und kann an diesem abgefragt werden. Wenn die Zeitdauer PT erreicht ist, liefert der Parameter Q den Signalzustand "1". Der Parameter Q bleibt auf "1" gesetzt, auch wenn der Signalzustand am Parameter IN auf "0" wechselt.

Der Parameter R setzt die Parameter ET und Q unabhängig vom Signalzustand am Parameter IN zurück.

Jedem Aufruf der Anweisung "Zeit akkumulieren" muss eine IEC-Zeit zugeordnet werden, in der die Anweisungsdaten gespeichert werden.

### Für CPU S7-1200

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER oder TONR\_TIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME im Abschnitt "Static" eines Bausteins (z. B. #MyTONR\_TIMER)

### Für CPU S7-1500

Eine IEC-Zeit ist eine Struktur vom Datentyp IEC\_TIMER, IEC\_LTIMER, TONR\_TIME oder TONR\_LTIME, die Sie wie folgt deklarieren können:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_TIMER oder IEC\_LTIMER (z. B. "MyIEC\_TIMER")
- Deklaration als lokale Variable vom Typ TONR\_TIME oder TONR\_LTIME im Abschnitt "Static" eines Bausteins (z. B. #MyTONR\_TIMER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die IEC-Zeit in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine. Weitere Informationen dazu finden Sie unter "Siehe auch".

Die Aktualisierung der Anweisungsdaten geschieht sowohl bei einem Aufruf der Anweisung als auch bei einem Zugriff auf die Ausgänge Q oder ET.

### Syntax

Für die Anweisung "Zeit akkumulieren" wird die folgende Syntax verwendet:

- Datenbaustein vom Systemdatentyp IEC\_Zeit (Shared DB):

**SCL**

```
<IEC_Zeit_DB>.TONR(IN := <Operand>,
                   R := <Operand>,
                   PT := <Operand>,
                   Q => <Operand>,
                   ET => <Operand>)
```

- Lokale Variable:

**SCL**

```
#myLocal_timer(IN := <Operand>,
               R := <Operand>,
               PT := <Operand>,
               Q => <Operand>,
               ET => <Operand>)
```

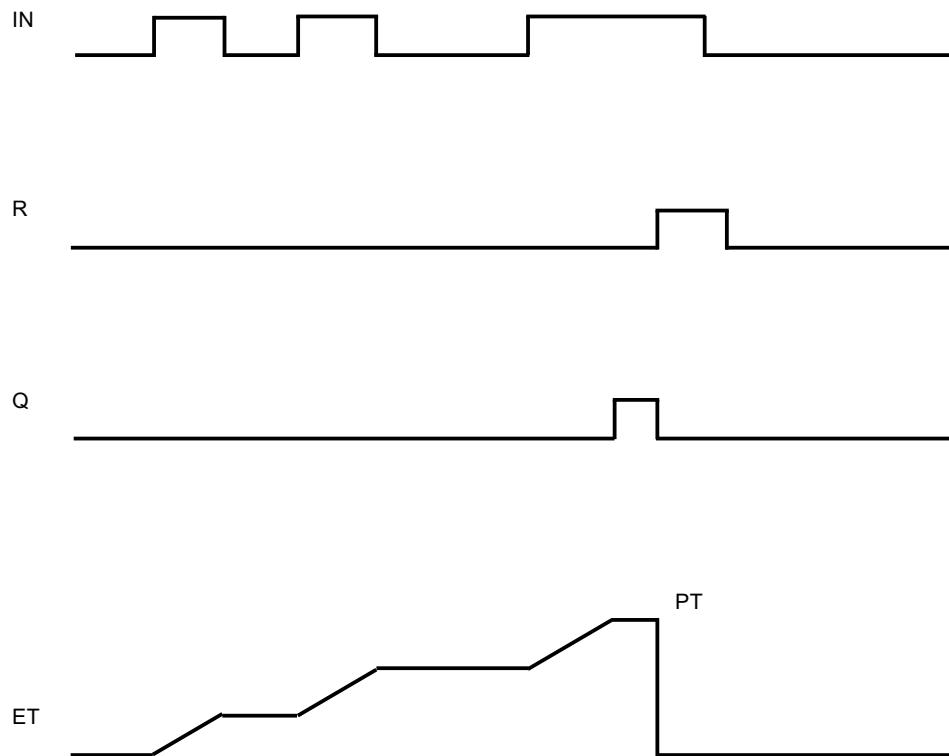
Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp |             | Beschreibung  |
|-----------|-------------|----------|-------------|---|
|           |             | S7-1200  | S7-1500     |   |
| IN        | Input       | BOOL     | BOOL        | Starteingang  |
| R         | Input       | BOOL     | BOOL        | Rücksetzen der Parameter ET und Q   |
| PT        | Input       | TIME     | TIME, LTIME | Maximale Dauer der Zeiterfassung. Der Wert am Parameter PT muss positiv sein. |
| Q         | Output      | BOOL     | BOOL        | Operand, der nach dem Ablauf der Zeit PT gesetzt bleibt.                      |
| ET        | Output      | TIME     | TIME, LTIME | Akkumulierte Zeit   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit akkumulieren":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"TONR_DB".TONR(IN := "Tag_Start",
               R := "Tag_Reset",
               PT := "Tag_PresetTime",
               Q => "Tag_Status",
               ET => "Tag_Time");
```

Wenn der Signalzustand des Operanden "Tag\_Start" von "0" auf "1" wechselt, wird die programmierte Zeit am Parameter PT gestartet. Während des Ablaufs der Zeit werden die Zeitwerte akkumuliert, die bei einem Signalzustand "1" des Operanden "Tag\_Start" erfasst werden. Die akkumulierte Zeit wird im Operanden "Tag\_Time" gespeichert. Wenn der am Parameter PT angegebene Zeitwert erreicht ist, wird der Operand "Tag\_Status" auf den Signalzustand "1" gesetzt. Der aktuelle Zeitwert wird im Operanden "Tag\_Time" abgelegt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**RESET\_TIMER: Zeit rücksetzen**

**Beschreibung**

Mit der Anweisung "Zeit rücksetzen" setzen Sie eine IEC-Zeit auf "0" zurück. Die Strukturkomponenten der Zeit im angegebenen Datenbaustein werden auf "0" zurückgesetzt.

Die Anweisung beeinflusst das VKE nicht. Am Parameter TIMER wird der Anweisung "Zeit rücksetzen" eine im Programm deklarierte IEC-Zeit zugewiesen.

Die Aktualisierung der Anweisungsdaten geschieht nur bei einem Aufruf der Anweisung und nicht bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage der Daten ist nur gleich vom Aufruf der Anweisung bis zum nächsten Aufruf der Anweisung.

**Syntax**

Für die Anweisung "Zeit rücksetzen" wird die folgende Syntax verwendet:

```
SCL
RESET_TIMER(TIMER := <IEC-Zeit>)
```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeit rücksetzen":

| Parameter  | Deklaration | Datentyp  | Beschreibung                     |
|------------|-------------|---|----------------------------------|
| <IEC-Zeit> | Output      | IEC_TIMER,<br>IEC_LTIMER,<br>TP_TIME, TP_LTIME,<br>TON_TIME,<br>TON_LTIME,<br>TOF_TIME,<br>TOF_LTIME,<br>TONR_TIME,<br>TONR_LTIME | IEC-Zeit, die zurückgesetzt wird |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

### SCL

```
IF #started = false THEN
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
#started := true;
END_IF;

IF "TON_DB".ET < T#25s THEN
RESET_TIMER(TIMER := "TON_DB");
#started := false;
END_IF;
```

Wenn die Variable #started den Signalzustand "0" führt, dann wird die Anweisung "Einschaltverzögerung erzeugen" bei einer positiven Signalflanke am Operanden "Tag\_Start" ausgeführt. Die im Instanz-Datenbaustein "TON\_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag\_PresetTime" vorgegeben ist. Der Operand "Tag\_Status" wird gesetzt, wenn die durch den Operanden "Tag\_PresetTime" vorgegebene Zeitdauer abgelaufen ist. Der Parameter Q bleibt so lange gesetzt, wie der Operand "Tag\_Start" noch den Signalzustand "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Operand am Parameter Q zurückgesetzt.

Wenn die abgelaufene Zeit der IEC-Zeit "TON\_DB" kleiner als 25s ist, wird die Anweisung "Zeit rücksetzen" ausgeführt und die im Instanz-Datenbaustein "TON\_DB" abgelegte Zeit zurückgesetzt.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## PRESET\_TIMER: Zeitdauer laden

### Beschreibung

Mit der Anweisung "Zeitdauer laden" stellen Sie die Zeitdauer einer IEC-Zeit ein. Die Anweisung wird in jedem Zyklus ausgeführt, wenn das Verknüpfungsergebnis (VKE) am

Eingang der Anweisung den Signalzustand "1" führt. Die Anweisung schreibt die angegebene Zeitdauer in die Struktur der angegebenen IEC-Zeit.

**Hinweis**

Wenn die angegebene IEC-Zeit während der Ausführung der Anweisung läuft, überschreibt die Anweisung die aktuelle Zeitdauer der angegebenen IEC-Zeit. Dadurch kann sich der Zeitstatus der IEC-Zeit ändern.

Der Anweisung "Zeitdauer laden" müssen Sie eine im Programm deklarierte IEC-Zeit zuweisen.

Die Aktualisierung der Anweisungsdaten geschieht bei einem Aufruf der Anweisung und bei jedem Zugriff auf die zugeordnete IEC-Zeit. Die Abfrage auf Q oder ET (z. B. "MyTimer".Q bzw. "MyTimer".ET) aktualisiert die IEC\_TIMER-Struktur.

**Syntax**

Für die Anweisung "Zeitdauer laden" wird die folgende Syntax verwendet:

```
SCL
PRESET_TIMER (PT := <Operand>,
              TIMER := <IEC-Zeit>)
```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Zeitdauer laden":

| Parameter   | Deklaration | Datentyp  | Beschreibung                            |
|-------------|-------------|---|---|
| <Zeitdauer> | Input       | TIME, LTIME   | Zeitdauer, mit der die IEC-Zeit läuft.  |
| <IEC-Zeit>  | Output      | IEC_TIMER,<br>IEC_LTIMER,<br>TP_TIME, TP_LTIME,<br>TON_TIME,<br>TON_LTIME,<br>TOF_TIME,<br>TOF_LTIME,<br>TONR_TIME,<br>TONR_LTIME | IEC-Zeit, deren Dauer eingestellt wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
IF #started = false THEN
```

**SCL**

```
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");

#started := true;
#preset = true
END_IF;

IF "TON_DB".ET < T#10s AND #preset = true THEN
PRESET_TIMER(PT := T#25s,
             TIMER := "TON_DB");
#preset := false;
END_IF;
```

Wenn die Variable #started den Signalzustand "0" und am Operanden "Tag\_Start" eine positive Signalfanke anliegt, dann wird die Anweisung "Einschaltverzögerung erzeugen" ausgeführt. Die im Instanz-Datenbaustein "TON\_DB" abgelegte IEC-Zeit wird mit der Zeitdauer gestartet, die durch den Operanden "Tag\_PresetTime" vorgegeben ist. Der Operand "Tag\_Status" wird gesetzt, wenn die durch den Operanden "Tag\_PresetTime" vorgegebene Zeitdauer PT abgelaufen ist. Der Parameter Q bleibt so lange gesetzt, wie der Operand "Tag\_Start" noch den Signalzustand "1" führt. Wenn der Signalzustand am Starteingang von "1" auf "0" wechselt, wird der Operand am Parameter Q zurückgesetzt.

Wenn die abgelaufene Zeit der IEC-Zeit "TON\_DB" kleiner als 10s ist und die Variable #preset den Signalzustand "1" führt, wird die Anweisung "Zeitdauer laden" ausgeführt. Die Anweisung schreibt die Zeitdauer, die am Parameter PT angegeben ist, in den Instanz-Datenbaustein "TON\_DB" und überschreibt dabei den Zeitwert des Operanden "Tag\_PresetTime" innerhalb des Instanz-Datenbausteins. Dadurch kann sich der Signalzustand des Zeitstatus bei der nächsten Abfrage bzw. beim nächsten Zugriff auf "TON\_DB".Q oder "TON\_DB".ET ändern.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**SIMATIC-Zeiten****S\_PULSE: Zeit als Impuls parametrieren und starten****Beschreibung**

Mit der Anweisung "Zeit als Impuls parametrieren und starten" starten Sie die am Parameter T\_NO programmierte Zeit, wenn ein Wechsel im Verknüpfungsergebnis (VKE) von "0" auf "1" (positive Signalfanke) am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, solange der Signalzustand am Parameter S "1" ist.

Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird die Zeit angehalten und der Parameter Q auf "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Eingang R keine Auswirkungen.

Der Parameter Q liefert den Signalzustand "1", solange die Zeit läuft und der Signalzustand am Parameter S "1" ist. Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, liefert der Parameter Q den Signalzustand "0". Wenn die Zeit durch den Parameter R zurückgesetzt wird oder die Zeit abgelaufen ist, liefert der Parameter Q ebenfalls den Signalzustand "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Syntax**

Für die Anweisung "Zeit als Impuls parametrieren und starten" wird die folgende Syntax verwendet:

**SCL**

```
S_PULSE(T_NO := <Operand>,
        S := <Operand>,
        TV := <Operand>,
        R := <Operand>,
        Q => <Operand>,
        BI => <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

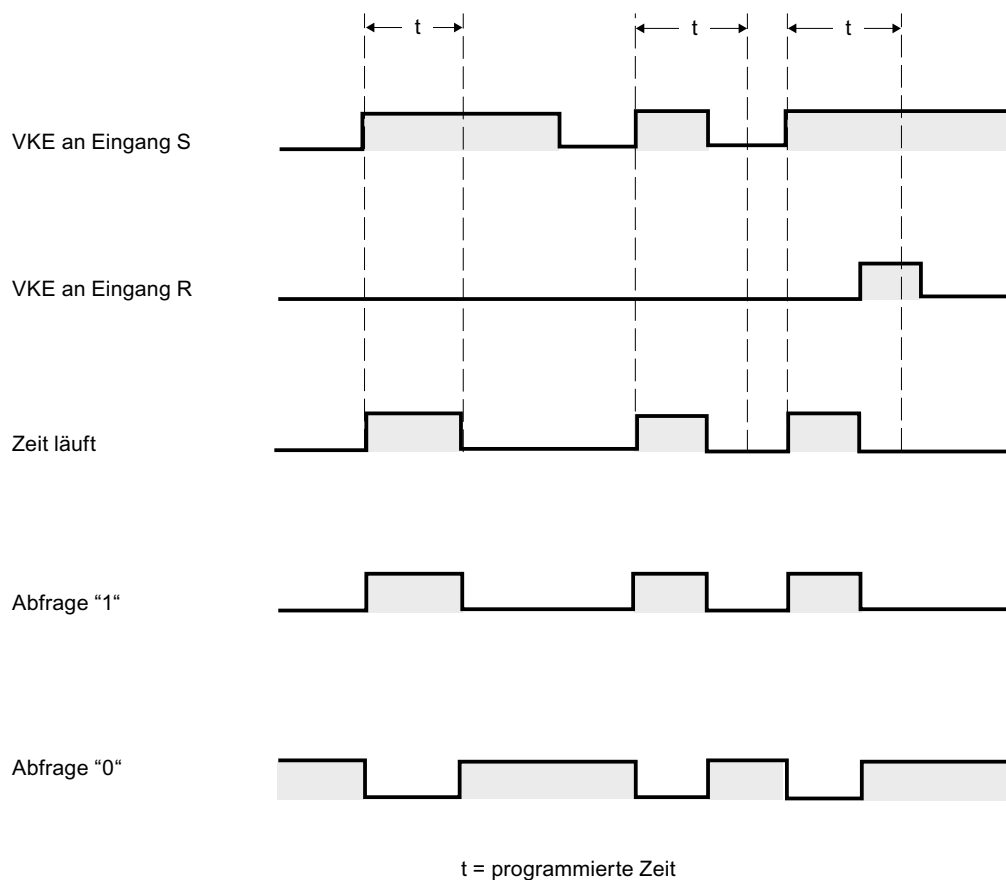
| Parameter | Deklaration | Datentyp        | Beschreibung   |
|-----------|-------------|-----------------|--|
| T_NO      | Input       | TIMER, INT      | Zeit, die gestartet wird.<br>Anzahl der Zeiten ist von der CPU abhängig. |
| S         | Input       | BOOL            | Starteingang   |
| TV        | Input       | S5TIME,<br>WORD | Voreingestellter Zeitwert  |

| Parameter     | Deklaration | Datentyp | Beschreibung                   |
|---------------|-------------|----------|--------------------------------|
| R             | Input       | BOOL     | Rücksetzeingang                |
| Q             | Output      | BOOL     | Status der Zeit                |
| BI            | Output      | WORD     | Aktueller Zeitwert dualcodiert |
| Funktionswert |             | S5TIME   | Aktueller Zeitwert             |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Impuls parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result" := S_PULSE(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",

```

**SCL**

```
R := "Tag_Reset",  
Q := "Tag_Status",  
BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit dem Zeitwert des Operanden "Tag\_Number" ab, solange der Operand "Tag\_1" den Signalzustand "1" liefert.

Wenn der Signalzustand am Parameter S auf "0" wechselt, bevor die programmierte Zeitdauer abgelaufen ist, wird der Operand "Tag\_Status" auf "0" zurückgesetzt. Wenn die Zeit durch den Parameter R zurückgesetzt wird oder die Zeit abgelaufen ist, liefert der Operand "Tag\_Status" ebenfalls den Signalzustand "0".

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag\_Value" abgelegt als auch als Funktionswert zurückgeliefert.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**S\_PEXT: Zeit als verlängerten Impuls parametrieren und starten**

**Beschreibung**

Mit der Anweisung "Zeit als verlängerten Impuls parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine positive Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, auch wenn der Signalzustand am Parameter S auf "0" wechselt. Solange die Zeit läuft, liefert der Parameter Q den Signalzustand "1".

Wenn die Zeit abgelaufen ist, wird der Parameter Q auf "0" zurückgesetzt. Wechselt der Signalzustand am Parameter S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der am Parameter TV programmierten Zeitdauer neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Parameter R auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Wenn die Zeit nicht läuft, hat der Signalzustand "1" am Parameter R keine Auswirkungen.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

## Syntax

Für die Anweisung "Zeit als verlängerten Impuls parametrieren und starten" wird die folgende Syntax verwendet:

### SCL

```
S_PEXT(T_NO := <Operand>,
        S := <Operand>,
        TV := <Operand>,
        R := <Operand>,
        Q => <Operand>,
        BI =><Operand>)
```

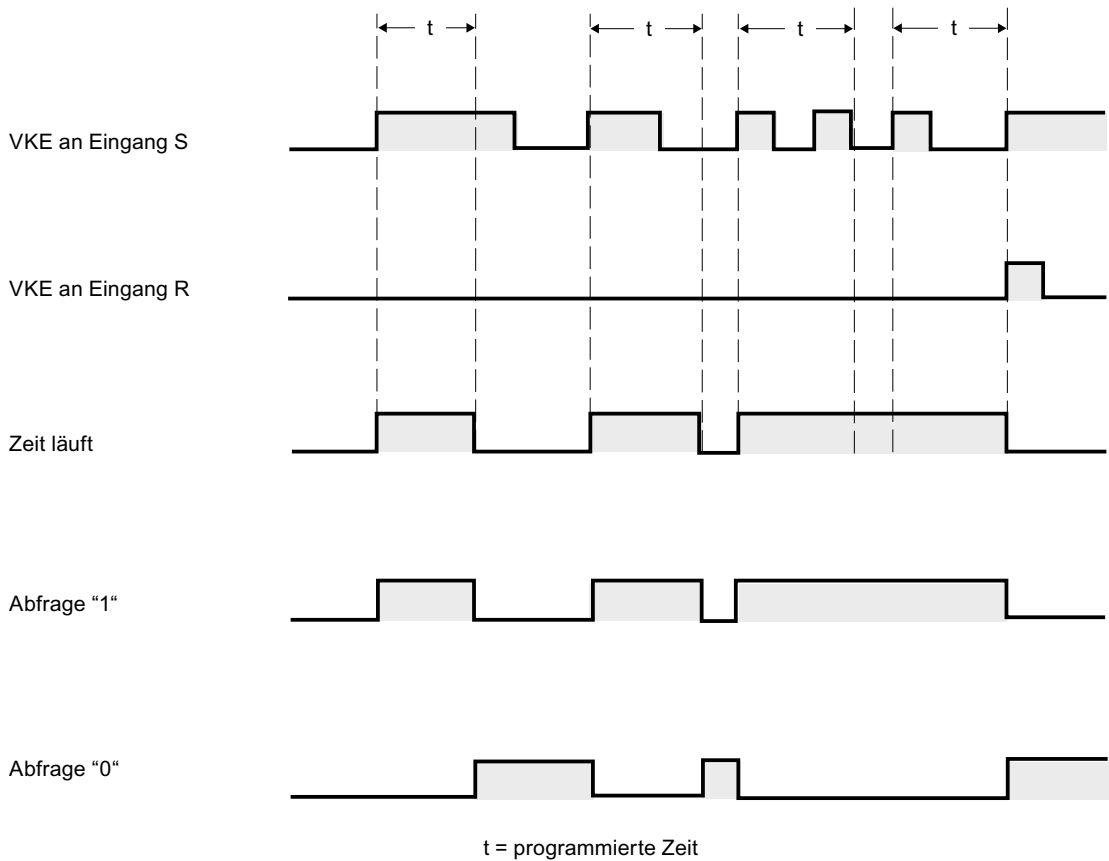
Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp        | Beschreibung   |
|---------------|-------------|-----------------|--|
| T_NO          | Input       | TIMER, INT      | Zeit, die gestartet wird.<br>Anzahl der Zeiten ist von der CPU abhängig. |
| S             | Input       | BOOL            | Starteingang   |
| TV            | Input       | S5TIME,<br>WORD | Voreingestellter Zeitwert  |
| R             | Input       | BOOL            | Rücksetzeingang  |
| Q             | Output      | BOOL            | Status der Zeit  |
| BI            | Output      | WORD            | Aktueller Zeitwert dualcodiert   |
| Funktionswert |             | S5TIME          | Aktueller Zeitwert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als verlängerten Impuls parametrieren und starten":



### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result" := S_PEXT(T_NO := "Timer_1",
    S := "Tag_1",
    TV := "Tag_Number",
    R := "Tag_Reset",
    Q := "Tag_Status",
    BI := "Tag_Value");
    
```

Wenn der Signalzustand des Operanden "Tag\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Solange die Zeit läuft, liefert der Operand "Tag\_Status" den Signalzustand "1". Wenn die Zeit abgelaufen ist, wird der Operand "Tag\_Status" auf "0" zurückgesetzt. Wechselt der Signalzustand am Eingang S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der Zeitdauer "Tag\_Number" neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag\_Value" abgelegt als auch als Funktionswert zurückgeliefert.



## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## S\_ODT: Zeit als Einschaltverzögerung parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit als Einschaltverzögerung, wenn eine positive Signalfanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, solange der Signalzustand am Parameter S "1" ist.

Wenn die Zeit ordnungsgemäß abgelaufen ist und am Parameter S der Signalzustand "1" noch anliegt, liefert der Parameter Q den Signalzustand "1". Wechselt der Signalzustand am Parameter S von "1" auf "0" während die Zeit läuft, wird die Zeit angehalten. Der Ausgang Q wird in diesem Fall auf den Signalzustand "0" zurückgesetzt.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Wenn die Zeit läuft und der Signalzustand am Eingang R von "0" auf "1" wechselt, werden der aktuelle Zeitwert und die Zeitbasis auch auf Null gesetzt. Der Signalzustand am Parameter Q ist in diesem Fall "0". Die Zeit wird bei einem Signalzustand "1" am Parameter R zurückgesetzt, auch wenn die Zeit nicht läuft und das Verknüpfungsergebnis (VKE) am Parameter S "1" ist.

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

---

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

### Syntax

Für die Anweisung "Zeit als Einschaltverzögerung parametrieren und starten" wird die folgende Syntax verwendet:

```
SCL  
S_ODT(T_NO := <Operand>,  
      S := <Operand>,
```

**SCL**

```

TV := <Operand>,
R := <Operand>,
Q => <Operand>,
BI =><Operand>
    
```

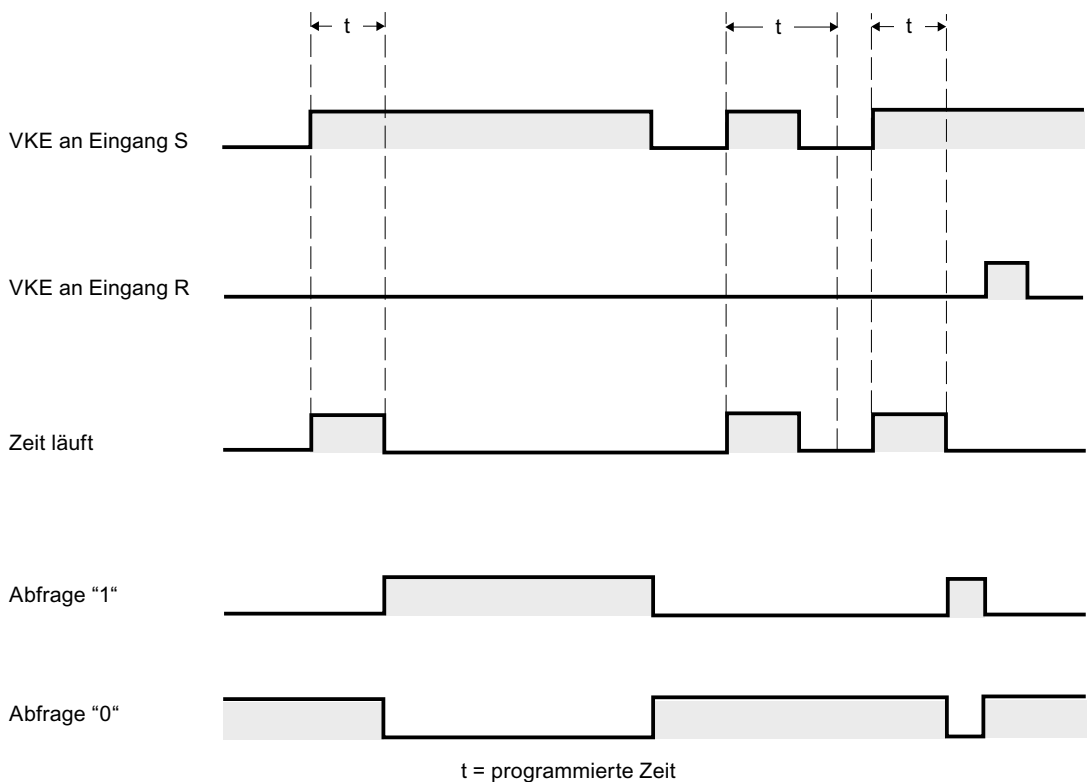
Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp        | Beschreibung   |
|---------------|-------------|-----------------|--|
| T_NO          | Input       | TIMER, INT      | Zeit, die gestartet wird.<br>Anzahl der Zeiten ist von der CPU abhängig. |
| S             | Input       | BOOL            | Starteingang   |
| TV            | Input       | S5TIME,<br>WORD | Voreingestellter Zeitwert  |
| R             | Input       | BOOL            | Rücksetzeingang  |
| Q             | Output      | BOOL            | Status der Zeit  |
| BI            | Output      | WORD            | Aktueller Zeitwert dualcodiert   |
| Funktionswert |             | S5TIME          | Aktueller Zeitwert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Impulsdiagramm**

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Einschaltverzögerung parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL  
"Tag_Result" := S_ODT(T_NO := "Timer_1",  
                      S := "Tag_1",  
                      TV := "Tag_Number",  
                      R := "Tag_Reset",  
                      Q := "Tag_Status",  
                      BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag\_Number" ab, solange der Signalzustand des Operanden "Tag\_1" "1" ist.

Wenn die Zeit ordnungsgemäß abgelaufen ist und der Operand "Tag\_Status" den Signalzustand "1" hat, wird der Operand "Tag\_Status" auf "1" gesetzt. Wechselt der Signalzustand des Operanden "Tag\_1" von "1" auf "0" während die Zeit läuft, wird die Zeit angehalten. Der Operand "Tag\_Status" liefert in diesem Fall den Signalzustand "0".

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag\_Value" abgelegt als auch als Funktionswert zurückgeliefert.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## S\_ODTS: Zeit als speichernde Einschaltverzögerung parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine positive Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab, auch wenn der Signalzustand am Parameter S auf "0" wechselt.

Wenn die Zeit abgelaufen ist, liefert der Parameter Q den Signalzustand "1" unabhängig vom Signalzustand am Parameter S. Wechselt der Signalzustand am Parameter S von "0" auf "1" während die Zeit läuft, wird die Zeit mit der programmierten Zeitdauer TV neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Der Signalzustand "1" am Parameter R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück, unabhängig vom Signalzustand am Parameter S. Der Signalzustand am Parameter Q ist in diesem Fall "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

**Hinweis**

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

**Syntax**

Für die Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten" wird die folgende Syntax verwendet:

```
SCL
S_ODTS (T_NO := <Operand>,
        S := <Operand>,
        TV := <Operand>,
        R := <Operand>,
        Q => <Operand>,
        BI =><Operand>)
```

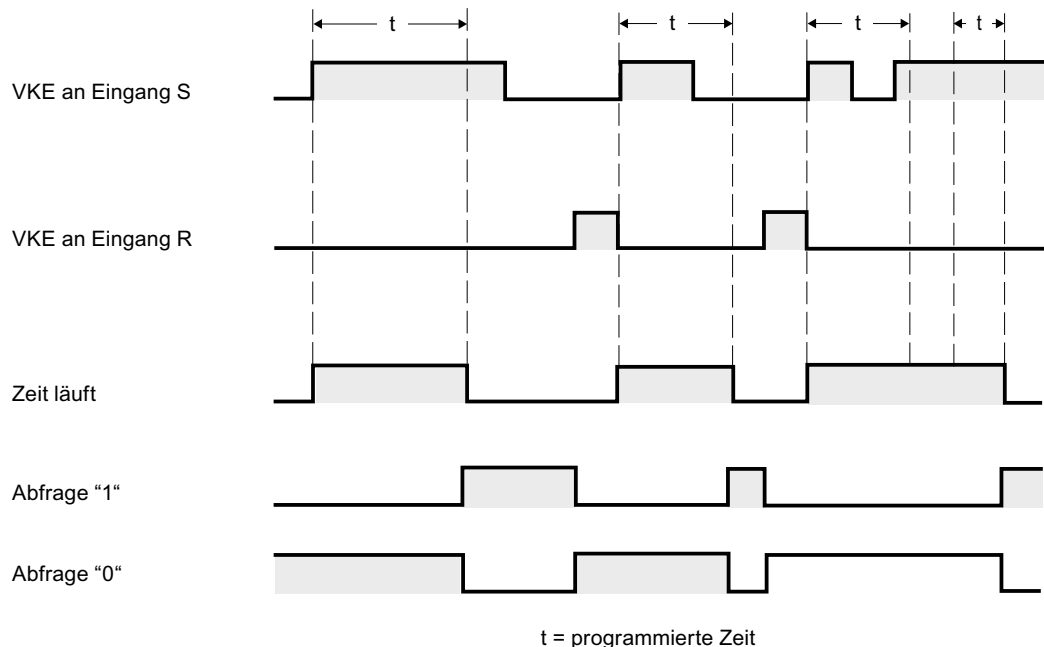
Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp        | Beschreibung   |
|---------------|-------------|-----------------|--|
| T_NO          | Input       | TIMER, INT      | Zeit, die gestartet wird.<br>Anzahl der Zeiten ist von der CPU abhängig. |
| S             | Input       | BOOL            | Starteingang   |
| TV            | Input       | S5TIME,<br>WORD | Voreingestellter Zeitwert  |
| R             | Input       | BOOL            | Rücksetzeingang  |
| Q             | Output      | BOOL            | Status der Zeit  |
| BI            | Output      | WORD            | Aktueller Zeitwert dualcodiert   |
| Funktionswert |             | S5TIME          | Aktueller Zeitwert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Impulsdiagramm

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als speichernde Einschaltverzögerung parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := S_ODTS(T_NO := "Timer_1",
                      S := "Tag_1",
                      TV := "Tag_Number",
                      R := "Tag_Reset",
                      Q := "Tag_Status",
                      BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag\_Number" ab.

Wenn die Zeit abgelaufen ist, liefert der Operand "Tag\_Status" den Signalzustand "1" unabhängig vom Signalzustand des Operanden "Tag\_1". Wechselt der Signalzustand des Operanden "Tag\_1" von "0" auf "1" während die Zeit läuft, wird die Zeit mit der Zeitdauer "Tag\_Number" neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag\_Value" abgelegt als auch als Funktionswert zurückgeliefert.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## S\_OFFDT: Zeit als Ausschaltverzögerung parametrieren und starten

### Beschreibung

Mit der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" starten Sie eine programmierte Zeit, wenn eine negative Signalflanke am Parameter S erfasst wird. Die Zeit läuft mit der programmierten Zeitdauer TV ab. Solange die Zeit läuft oder der Parameter S den Signalzustand "1" liefert, führt der Parameter Q den Signalzustand "1".

Wenn die Zeit abgelaufen ist und der Signalzustand "0" ist, wird der Parameter Q auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand am Parameter S von "0" auf "1", während die Zeit läuft, wird die Zeit angehalten. Erst wenn eine negative Signalflanke am Parameter S erfasst wird, wird die Zeit neu gestartet.

Die Zeitdauer setzt sich intern aus einem Zeitwert und einer Zeitbasis zusammen und wird am Parameter TV programmiert. Wenn die Anweisung gestartet wird, wird der programmierte Zeitwert rückwärts gegen Null gezählt. Mit welcher Zeitperiode der Zeitwert verändert wird, gibt dabei die Zeitbasis an. Der aktuelle Zeitwert wird am Parameter BI ausgegeben.

Der Signalzustand "1" am Parameter R setzt den aktuellen Zeitwert und die Zeitbasis auf "0" zurück. Der Signalzustand am Parameter Q ist in diesem Fall "0".

Die Anweisungsdaten werden bei jedem Zugriff aktualisiert. Dadurch kann es vorkommen, dass die Abfrage der Daten am Zyklusanfang andere Werte als am Zyklusende liefert.

---

### Hinweis

In der Zeitzelle vermindert das Betriebssystem den Zeitwert in einem, von der Zeitbasis festgelegten, Intervall um jeweils eine Einheit, bis der Zeitwert gleich "0" ist. Die Verminderung erfolgt asynchron zum Anwenderprogramm. Dadurch ist die resultierende Zeit maximal bis zu einem Zeitintervall der Zeitbasis kürzer als der gewünschte Zeitwert.

---

Ein Beispiel, wie eine Zeitzelle aufgebaut sein kann, finden Sie unter: Siehe auch "L: Zeitwert laden".

### Syntax

Für die Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten" wird die folgende Syntax verwendet:

#### SCL

```
S_OFFDT(T_NO := <Operand>,  
        S := <Operand>,  
        TV := <Operand>,  
        R := <Operand>,
```

**SCL**

Q => <Operand>,  
 BI =><Operand>

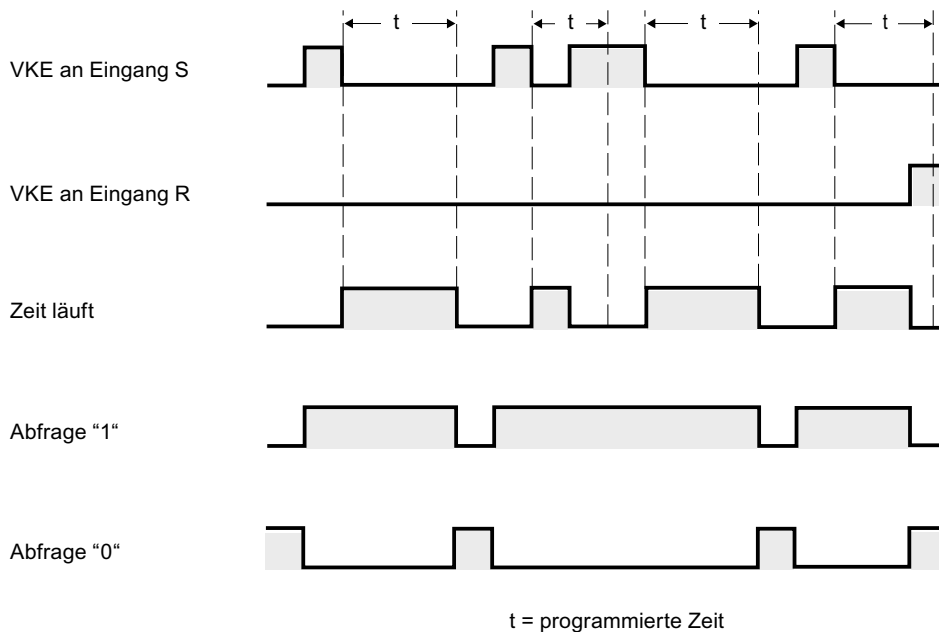
Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp        | Beschreibung   |
|---------------|-------------|-----------------|--|
| T_NO          | Input       | TIMER, INT      | Zeit, die gestartet wird.<br>Anzahl der Zeiten ist von der CPU abhängig. |
| S             | Input       | BOOL            | Starteingang   |
| TV            | Input       | S5TIME,<br>WORD | Voreingestellter Zeitwert  |
| R             | Input       | BOOL            | Rücksetzeingang  |
| Q             | Output      | BOOL            | Status der Zeit  |
| BI            | Output      | WORD            | Aktueller Zeitwert dualcodiert   |
| Funktionswert |             | S5TIME          | Aktueller Zeitwert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Impulsdiagramm**

Das folgende Bild zeigt das Impulsdiagramm der Anweisung "Zeit als Ausschaltverzögerung parametrieren und starten":



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL  
"Tag_Result" := S_OFFDT(T_NO := "Timer_1",  
                        S := "Tag_1",  
                        TV := "Tag_Number",  
                        R := "Tag_Reset",  
                        Q := "Tag_Status",  
                        BI := "Tag_Value");
```

Wenn der Signalzustand des Operanden "Tag\_1" von "0" auf "1" wechselt, wird die Zeit "Timer\_1" gestartet. Die Zeit läuft mit der Zeitdauer "Tag\_Number" ab. Solange die Zeit läuft oder der Operand "Tag\_1" den Signalzustand "1" liefert, führt der Operand "Tag\_Status" den Signalzustand "1".

Wenn die Zeit abgelaufen ist und der Signalzustand des Operanden "Tag\_1" "0" ist, wird der Operand "Tag\_Status" auf den Signalzustand "0" zurückgesetzt. Wechselt der Signalzustand des Operanden "Tag\_1" von "0" auf "1" während die Zeit läuft, wird die Zeit zurückgesetzt. Erst wenn eine negative Flanke am Parameter S erfasst wird, wird die Zeit neu gestartet.

Der aktuelle Zeitwert wird sowohl dualcodiert im Operanden "Tag\_Value" abgelegt als auch als Funktionswert zurückgeliefert.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## Zähler

### IEC-Zähler

#### CTU: Vorwärts zählen

### Beschreibung

Mit der Anweisung "Vorwärts zählen" zählen Sie den Wert am Parameter CV hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalfanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins erhöht. Bei der ersten Ausführung der Anweisung ist der aktuelle Zählwert am Parameter CV auf Null gestellt. Der Zählwert wird bei jeder Erfassung einer positiven Signalfanke erhöht, bis er den oberen Grenzwert des am Parameter CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, hat der Signalzustand am Parameter CU keine Wirkung mehr auf die Anweisung.



Am Parameter Q können Sie den Zählerstatus abfragen. Der Signalzustand am Parameter Q wird durch den Parameter PV bestimmt. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert des Parameters CV wird auf Null zurückgesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat der Signalzustand am Parameter CU keine Wirkung auf die Anweisung.

---

#### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

Jedem Aufruf der Anweisung "Vorwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

#### Für CPU S7-1200

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTU_SINT / CTU_USINT</li> <li>• CTU_INT / CTU_UINT</li> <li>• CTU_DINT / CTU_UDINT</li> </ul> |

#### Für CPU S7-1500

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTU_SINT / CTU_USINT</li> <li>• CTU_INT / CTU_UINT</li> <li>• CTU_DINT / CTU_UDINT</li> <li>• CTU_LINT / CTU_ULINT</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTU im Abschnitt "Static" eines Bausteins (z. B. #MyCTU\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

### Syntax

Für die Anweisung "Vorwärts zählen" wird die folgende Syntax verwendet:

Tabelle 9-28 Datenbaustein vom Systemdatentyp IEC\_Zähler (Shared DB)

```
SCL
<IEC_Zähler_DB>.CTU(CU := <Operand>,
                    R := <Operand>,
                    PV := <Operand>,
                    Q => <Operand>,
                    CV => <Operand>)
```

Tabelle 9-29 Lokale Variable

```
SCL
#myLocal_counter(CU := <Operand>,
                 R := <Operand>,
                 PV := <Operand>,
                 Q => <Operand>,
                 CV => <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp               | Beschreibung                             |
|-----------|-------------|------------------------|--|
| CU        | Input       | BOOL                   | Zähleingang                              |
| R         | Input       | BOOL                   | Rücksetzeingang                          |
| PV        | Input       | Ganzzahlen             | Wert, bei dem der Ausgang Q gesetzt wird |
| Q         | Output      | BOOL                   | Zählerstatus                             |
| CV        | Output      | Ganzzahlen, CHAR, DATE | Aktueller Zählwert                       |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"IEC_COUNTER_DB".CTU(CU := "Tag_Start",
                    R := "Tag_Reset",
                    PV := "Tag_PresetValue",
                    Q => "Tag_Status",
                    CV => "Tag_CounterValue");
```

Wenn der Signalzustand des Operanden "Tag\_Start" von "0" auf "1" wechselt, wird die Anweisung "Vorwärts zählen" ausgeführt und der aktuelle Zählwert des Operanden "Tag\_CounterValue" um eins erhöht. Bei jeder weiteren positiven Signalflanke wird der Zählwert erhöht, bis der obere Grenzwert des angegebenen Datentyps (INT = 32767) erreicht ist.

Der Ausgang "Tag\_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag\_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag\_Status" den Signalzustand "0". Der aktuelle Zählwert wird im Operanden "Tag\_CounterValue" abgelegt.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

### CTD: Rückwärts zählen

#### Beschreibung

Mit der Anweisung "Rückwärts zählen" zählen Sie den Wert am Parameter CV runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird die Anweisung ausgeführt und der aktuelle Zählwert am Parameter CV um eins verringert. Bei der ersten Ausführung der Anweisung wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Der Zählwert wird bei jeder Erfassung einer positiven Signalflanke verringert, bis er den unteren Grenzwert des angegebenen Datentyps erreicht. Wenn der untere Grenzwert erreicht ist, hat der Signalzustand am Parameter CD keine Wirkung mehr auf die Anweisung.

Am Parameter Q können Sie den Zählerstatus abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter Q auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter Q "0". Am Parameter PV können Sie auch eine Konstante angeben.

Der Wert am Parameter CV wird auf den Wert des Parameters PV gesetzt, wenn der Signalzustand am Parameter LD auf "1" wechselt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand am Parameter CD keine Wirkung auf die Anweisung.

---

#### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

Jedem Aufruf der Anweisung "Rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

Für CPU S7-1200

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTD_SINT / CTD_USINT</li> <li>• CTD_INT / CTD_UINT</li> <li>• CTD_DINT / CTD_UDINT</li> </ul> |

Für CPU S7-1500

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTD_SINT / CTD_USINT</li> <li>• CTD_INT / CTD_UINT</li> <li>• CTD_DINT / CTD_UDINT</li> <li>• CTD_LINT / CTD_ULINT</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTD im Abschnitt "Static" eines Bausteins (z. B. #MyCTD\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Syntax

Für die Anweisung "Rückwärts zählen" wird die folgende Syntax verwendet:

Tabelle 9-30 Datenbaustein vom Systemdatentyp IEC\_Zähler (Shared DB)

```

SCL
<IEC_Zähler_DB>.CTD(CD := <Operand>,
                    LD := <Operand>,
                    PV := <Operand>,
                    Q => <Operand>,
                    CV => <Operand>)
    
```

Tabelle 9-31 Lokale Variable

```

SCL
#myLocal_counter(CD := <Operand>,
                 LD := <Operand>,
    
```

**SCL**

```
PV := <Operand>,
Q => <Operand>,
CV => <Operand>
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp               | Beschreibung                             |
|-----------|-------------|------------------------|--|
| CD        | Input       | BOOL                   | Zähleingang                              |
| LD        | Input       | BOOL                   | Ladeeingang                              |
| PV        | Input       | Ganzzahlen             | Wert, bei dem der Ausgang Q gesetzt wird |
| Q         | Output      | BOOL                   | Zählerstatus                             |
| CV        | Output      | Ganzzahlen, CHAR, DATE | Aktueller Zählwert                       |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

**SCL**

```
"IEC_SCOUNTER_DB".CTD(CD := "Tag_Start",
    LD := "Tag_Load",
    PV := "Tag_PresetValue",
    Q => "Tag_Status",
    CV => "Tag_CounterValue");
```

Wenn der Signalzustand des Operanden "Tag\_Start" von "0" auf "1" wechselt, wird die Anweisung "Rückwärts zählen" ausgeführt und der Wert des Operanden "Tag\_CounterValue" um eins verringert. Bei jeder weiteren positiven Signalflanke wird der Zählwert verringert, bis der untere Grenzwert des angegebenen Datentyps (-128) erreicht ist.

Der Operand "Tag\_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag\_Status" den Signalzustand "0". Der aktuelle Zählwert wird im Operanden "Tag\_CounterValue" abgelegt.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## CTUD: Vorwärts und rückwärts zählen

### Beschreibung

Mit der Anweisung "Vorwärts und rückwärts zählen" zählen Sie den Zählwert am Parameter CV hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert am Parameter CV um eins erhöht. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert am Parameter CV um eins verringert. Wenn in einem Programmzyklus an den Eingängen CU und CD eine positive Signalflanke vorliegt, bleibt der aktuelle Zählwert am Parameter CV unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert des am Parameter CV angegebenen Datentyps erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert des angegebenen Datentyps erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter LD auf "1" wechselt, wird der Zählwert am Parameter CV auf den Wert des Parameters PV gesetzt. Solange am Parameter LD der Signalzustand "1" ansteht, hat der Signalzustand an den Parametern CU und CD keine Wirkung auf die Anweisung.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand der Parameter CU, CD und LD keine Wirkung auf die Anweisung "Vorwärts und rückwärts zählen".

Am Parameter QU können Sie den Status des Vorwärtszählers abfragen. Wenn der aktuelle Zählwert größer oder gleich dem Wert des Parameters PV ist, wird der Parameter QU auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QU "0". Am Parameter PV können Sie auch eine Konstante angeben.

Am Parameter QD können Sie den Status des Rückwärtszählers abfragen. Wenn der aktuelle Zählwert kleiner oder gleich Null ist, wird der Parameter QD auf den Signalzustand "1" gesetzt. In allen anderen Fällen ist der Signalzustand am Parameter QD "0".

---

### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

Jedem Aufruf der Anweisung "Vorwärts und rückwärts zählen" muss ein IEC-Zähler zugeordnet werden, in dem die Anweisungsdaten gespeichert werden. Ein IEC-Zähler ist eine Struktur mit einem der folgenden Datentypen:

### Für CPU S7-1200

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTUD_SINT / CTUD_USINT</li> <li>• CTUD_INT / CTUD_UINT</li> <li>• CTUD_DINT / CTUD_UDINT</li> </ul> |

## Für CPU S7-1500

| Datenbaustein vom Systemdatentyp IEC_Zähler (Shared DB)  | Lokale Variable  |
|--|--|
| <ul style="list-style-type: none"> <li>• IEC_SCOUNTER / IEC_USCOUNTER</li> <li>• IEC_COUNTER / IEC_UCOUNTER</li> <li>• IEC_DCOUNTER / IEC_UDCOUNTER</li> <li>• IEC_LCOUNTER / IEC_ULCOUNTER</li> </ul> | <ul style="list-style-type: none"> <li>• CTUD_SINT / CTUD_USINT</li> <li>• CTUD_INT / CTUD_UINT</li> <li>• CTUD_DINT / CTUD_UDINT</li> <li>• CTUD_LINT / CTUD_ULINT</li> </ul> |

Sie können einen IEC-Zähler wie folgt deklarieren:

- Deklaration eines Datenbausteins vom Systemdatentyp IEC\_COUNTER (z. B. "MyIEC\_COUNTER")
- Deklaration als lokale Variable vom Typ CTUD im Abschnitt "Static" eines Bausteins (z. B. #MyCTUD\_COUNTER)

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob der IEC-Zähler in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt wird. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

## Syntax

Für die Anweisung "Vorwärts und rückwärts zählen" wird die folgende Syntax verwendet:

Tabelle 9-32 Datenbaustein vom Systemdatentyp IEC\_Zähler (Shared DB)

## SCL

```
<IEC_Zähler_DB>.CTUD(CU := <Operand>,
                    CD := <Operand>,
                    R := <Operand>,
                    LD := <Operand>,
                    PV := <Operand>,
                    QU => <Operand>,
                    QD := <Operand>,
                    CV => <Operand>)
```

Tabelle 9-33 Lokale Variable

## SCL

```
myLocal_counter(CU := <Operand>,
                CD := <Operand>,
                R := <Operand>,
                LD := <Operand>,
                PV := <Operand>,
                QU => <Operand>,
                QD := <Operand>,
                CV => <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp               | Beschreibung                               |
|-----------|-------------|------------------------|--|
| CU        | Input       | BOOL                   | Vorwärtszähleingang                        |
| CD        | Input       | BOOL                   | Rückwärtszähleingang                       |
| R         | Input       | BOOL                   | Rücksetzeingang                            |
| LD        | Input       | BOOL                   | Ladeeingang                                |
| PV        | Input       | Ganzzahlen             | Wert, bei dem der Ausgang QU gesetzt wird. |
| QU        | Output      | BOOL                   | Status des Vorwärtszählers                 |
| QD        | Output      | BOOL                   | Status des Rückwärtszählers                |
| CV        | Output      | Ganzzahlen, CHAR, DATE | Aktueller Zählwert                         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"IEC_COUNTER_DB".CTUD(CU := "Tag_Start1",
                      CD := "Tag_Start2",
                      LD := "Tag_Load",
                      R := "Tag_Reset",
                      PV := "Tag_PresetValue",
                      QU => "Tag_CU_Status",
                      QD => "Tag_CD_Status",
                      CV => "Tag_CounterValue");
    
```

Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag\_Start1" vorliegt, wird der aktuelle Zählwert um eins erhöht und im Operanden "Tag\_CounterValue" abgelegt. Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag\_Start2" vorliegt, wird der Zählwert um eins verringert und ebenfalls im Operanden "Tag\_CounterValue" abgelegt. Der Zählwert wird bei einer positiven Signalflanke am Parameter CU so lange erhöht, bis er den oberen Grenzwert des angegebenen Datentyps (INT) erreicht. Bei einer positiven Signalflanke am Parameter CD wird der Zählwert so lange verringert, bis er den unteren Grenzwert des angegebenen Datentyps (INT) erreicht.

Der Operand "Tag\_CU\_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert größer oder gleich dem Wert des Operanden "Tag\_PresetValue" ist. In allen anderen Fällen führt der Ausgang "Tag\_CU\_Status" den Signalzustand "0".

Der Operand "Tag\_CD\_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert kleiner oder gleich Null ist. In allen anderen Fällen führt der Ausgang "Tag\_CD\_Status" den Signalzustand "0".



## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## SIMATIC-Zähler

### S\_CU: Parametrieren und vorwärts zählen

#### Beschreibung

Mit der Anweisung "Parametrieren und vorwärts zählen" zählen Sie den Wert eines Zählers hoch. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Der Zählwert kann so lange erhöht werden, bis er den Grenzwert "999" erreicht. Wenn der Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) am Eingang CU "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Parametern CU und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

---

#### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

## Syntax

Für die Anweisung "Parametrieren und vorwärts zählen" wird die folgende Syntax verwendet:

```
SCL  
S_CU(C_NO := <Operand>,  
      CU := <Operand>,  
      S := <Operand>,  
      PV := <Operand>,  
      R := <Operand>,  
      Q => <Operand>,  
      CV => <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp           | Beschreibung  |
|---------------|-------------|--------------------|---|
| C_NO          | Input       | COUNTER, INT       | Zähler<br>Anzahl der Zähler ist von der CPU abhängig.   |
| CU            | Input       | BOOL               | Vorwärtszähleingang                                     |
| S             | Input       | BOOL               | Eingang zum Voreinstellen des Zählers                   |
| PV            | Input       | WORD               | Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format |
| R             | Input       | BOOL               | Rücksetzeingang   |
| Q             | Output      | BOOL               | Status des Zählers                                      |
| CV            | Output      | WORD, S5TIME, DATE | Aktueller Zählwert                                      |
| Funktionswert |             | WORD, S5TIME, DATE | Aktueller Zählwert im BCD-Format                        |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

#### SCL

```
"Tag_Result" := S_CU(C_NO := "Counter_1",
                    CU := "Tag_Start",
                    S := "Tag_1",
                    PV := "Tag_PresetValue",
                    R := "Tag_Reset",
                    Q => "Tag_Status",
                    CV => "Tag_Value");
```

Wenn der Signalzustand am Parameter "Tag\_Start" von "0" auf "1" wechselt (positive Signalflanke) und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wechselt der Signalzustand am Eingang "Tag\_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag\_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag\_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird hexadezimal im Operanden "Tag\_Value" gespeichert.

Der Ausgang "Tag\_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag\_Value" abgelegt als auch als Funktionswert zurückgeliefert.

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## S\_CD: Parametrieren und rückwärts zählen

### Beschreibung

Mit der Anweisung "Parametrieren und rückwärts zählen" zählen Sie den Wert eines Zählers runter. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins verringert. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Der Zählwert kann so lange verringert werden, bis er den unteren Grenzwert "0" erreicht. Wenn der untere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr verringert.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) am Parameter CD "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat eine Änderung im Signalzustand an den Parametern CD und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

---

### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

### Syntax

Für die Anweisung "Parametrieren und rückwärts zählen" wird die folgende Syntax verwendet:

```
SCL
S_CD(C_NO := <Operand>,
      CD := <Operand>,
      S := <Operand>,
      PV := <Operand>,
      R := <Operand>,
      Q => <Operand>,
      CV => <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp     | Beschreibung  |
|-----------|-------------|--------------|---|
| C_NO      | Input       | COUNTER, INT | Zähler<br>Anzahl der Zähler ist von der CPU abhängig. |
| CD        | Input       | BOOL         | Rückwärtszähleingang                                  |

| Parameter     | Deklaration | Datentyp           | Beschreibung  |
|---------------|-------------|--------------------|---|
| S             | Input       | BOOL               | Eingang zum Voreinstellen des Zählers                   |
| PV            | Input       | WORD               | Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format |
| R             | Input       | BOOL               | Rücksetzeingang   |
| Q             | Output      | BOOL               | Status des Zählers                                      |
| CV            | Output      | WORD, S5TIME, WORD | Aktueller Zählwert                                      |
| Funktionswert |             | WORD, S5TIME, DATE | Aktueller Zählwert im BCD-Format                        |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result" := S_CD(C_NO := "Counter_1",
                    CD := "Tag_Start",
                    S := "Tag_1",
                    PV := "Tag_PresetValue",
                    R := "Tag_Reset",
                    Q => "Tag_Status",
                    CV => "Tag_Value");
    
```

Wenn der Signalzustand des Operanden "Tag\_Start" von "0" auf "1" wechselt (positive Signalfanke) und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert. Wechselt der Signalzustand des Operanden "Tag\_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag\_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag\_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird im Operanden "Tag\_Value" gespeichert.

Der Operand "Tag\_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag\_Value" abgelegt als auch als Funktionswert zurückgeliefert.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## S\_CUD: Parametrieren und vorwärts/rückwärts zählen

### Beschreibung

Mit der Anweisung "Parametrieren und vorwärts/rückwärts zählen" zählen Sie den Wert eines Zählers hoch und runter. Wenn der Signalzustand am Parameter CU von "0" auf "1" wechselt (positive Signalflanke), wird der aktuelle Zählwert um eins erhöht. Wenn der Signalzustand am Parameter CD von "0" auf "1" wechselt (positive Signalflanke), wird der Zählwert um eins verringert. Der aktuelle Zählwert wird am Parameter CV ausgegeben. Wenn in einem Programmzyklus an den Parametern CU und CD eine positive Signalflanke vorliegt, bleibt der Zählwert unverändert.

Der Zählwert kann so lange erhöht werden, bis er den oberen Grenzwert "999" erreicht. Wenn der obere Grenzwert erreicht ist, wird der Zählwert bei einer positiven Signalflanke nicht mehr hoch gezählt. Wenn der untere Grenzwert "0" erreicht ist, wird der Zählwert nicht mehr verringert.

Wenn der Signalzustand am Parameter S von "0" auf "1" wechselt, wird der Zählwert auf den Wert des Parameters PV gesetzt. Wird der Zähler gesetzt und das Verknüpfungsergebnis (VKE) an den Parametern CU und CD "1" ist, so zählt der Zähler einmalig im nächsten Zyklus, auch wenn kein Flankenwechsel erfasst wurde.

Der Zählwert wird auf Null gesetzt, wenn der Signalzustand am Parameter R auf "1" wechselt. Solange am Parameter R der Signalzustand "1" ansteht, hat die Abwicklung des Signalzustands an den Parametern CU, CD und S keine Wirkung auf den Zählwert.

Der Signalzustand am Parameter Q ist "1", wenn der Zählwert größer als Null ist. Wenn der Zählwert gleich Null ist, liefert der Parameter Q den Signalzustand "0".

---

### Hinweis

Verwenden Sie einen Zähler nur an einer Stelle im Programm, um Zählfehler zu vermeiden.

---

### Syntax

Für die Anweisung "Parametrieren und vorwärts/rückwärts zählen" wird die folgende Syntax verwendet:

```
SCL
S_CUD(C_NO := <Operand>,
      CU := <Operand>,
      CD := <Operand>,
      S := <Operand>,
      PV := <Operand>,
      R := <Operand>,
      Q => <Operand>,
      CV => <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp           | Beschreibung  |
|---------------|-------------|--------------------|---|
| C_NO          | Input       | COUNTER, INT       | Zähler<br>Anzahl der Zähler ist von der CPU abhängig.   |
| CU            | Input       | BOOL               | Vorwärtszähleingang                                     |
| CD            | Input       | BOOL               | Rückwärtszähleingang                                    |
| S             | Input       | BOOL               | Eingang zum Voreinstellen des Zählers                   |
| PV            | Input       | WORD               | Voreingestellter Zählwert (C#0 bis C#999) im BCD-Format |
| R             | Input       | BOOL               | Rücksetzeingang   |
| Q             | Output      | BOOL               | Status des Zählers                                      |
| CV            | Output      | WORD, S5TIME, DATE | Aktueller Zählwert (Hexadezimal)                        |
| Funktionswert |             | WORD, S5TIME, DATE | Aktueller Zählwert im BCD-Format                        |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result" := S_CD(C_NO := "Counter_1",
                    CU := "Tag_CU",
                    CD := "Tag_CD",
                    S := "Tag_1",
                    PV := "Tag_PresetValue",
                    R := "Tag_Reset",
                    Q => "Tag_Status",
                    CV => "Tag_Value");
    
```

Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag\_CU" erfasst wird und der aktuelle Zählwert kleiner als "999" ist, wird der Zählwert um eins erhöht. Wenn eine positive Signalflanke im Signalzustand des Operanden "Tag\_CD" erfasst wird und der aktuelle Zählwert größer als "0" ist, wird der Zählwert um eins verringert.

Wechselt der Signalzustand des Operanden "Tag\_1" von "0" auf "1", wird der Zählwert im BCD-Format auf den Wert des Operanden "Tag\_PresetValue" gesetzt. Der Zählwert wird auf "0" zurückgesetzt, wenn der Operand "Tag\_Reset" den Signalzustand "1" führt.

Der aktuelle Zählwert wird im Operanden "Tag\_Value" gespeichert.

Der Operand "Tag\_Status" liefert den Signalzustand "1", solange der aktuelle Zählwert ungleich "0" ist. Der aktuelle Zählwert wird sowohl im Operanden "Tag\_Value" abgelegt als auch als Funktionswert zurückgeliefert.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**Mathematische Funktionen****ABS: Absolutwert bilden****Beschreibung**

Mit der Anweisung "Absolutwert bilden" berechnen Sie den Absolutbetrag eines Eingangswertes und speichern das Ergebnis im angegebenen Operanden.

**Syntax**

Für die Anweisung "Absolutwert bilden" wird die folgende Syntax verwendet:

**SCL**

```
ABS (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp                             |   | Beschreibung                     |
|---------------|-------------|--------------------------------------|---|----------------------------------|
|               |             | S7-1200                              | S7-1500                                       |                                  |
| <Ausdruck>    | Input       | SINT, INT, DINT,<br>Gleitpunktzahlen | SINT, INT, DINT,<br>LINT,<br>Gleitpunktzahlen | Eingangswert                     |
| Funktionswert |             | SINT, INT, DINT,<br>Gleitpunktzahlen | SINT, INT, DINT,<br>LINT,<br>Gleitpunktzahlen | Absolutwert des<br>Eingangswerts |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

**SCL**

```
"Tag_Result1" := ABS("Tag_Value");
"Tag_Result2" := ABS("Tag_Value1"*"Tag_Value2");
```

Der Absolutbetrag des Eingangswerts wird im Format des Eingangswerts als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert |
|-------------|------|
| Tag_Value   | -2   |
| Tag_Result1 | 2    |
| Tag_Value1  | 4    |
| Tag_Value2  | -1   |
| Tag_Result2 | 4    |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

### MIN: Minimum ermitteln

#### Beschreibung

Mit der Anweisung "Minimum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und geben den kleinsten Wert als Ergebnis aus.

An der Anweisung können mindestens zwei und maximal 32 Eingänge angegeben werden.

Das Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

#### Syntax

Für die Anweisung "Minimum ermitteln" wird die folgende Syntax verwendet:

```
SCL  
MIN(IN1 := <Operand>,  
    IN2 := <Operand>)
```



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Minimum ermitteln":

| Parameter     | Deklaration | Datentyp  |   | Beschreibung  |
|---------------|-------------|---|---|---|
|               |             | S7-1200   | S7-1500   |   |
| IN1           | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Erster<br>Eingangswert  |
| IN2           | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Zweiter<br>Eingangswert   |
| INn           | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Zusätzlich<br>eingefügte<br>Eingänge, deren<br>Werte verglichen<br>werden |
| Funktionswert |             | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Ergebnis der<br>Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result" := MIN(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");

```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand    | Wert  |
|---------------|------------|-------|
| IN1           | Tag_Value1 | 12222 |
| IN2           | Tag_Value2 | 14444 |
| IN3           | Tag_Value3 | 13333 |
| Funktionswert | Tag_Result | 12222 |

Die Anweisung vergleicht die Werte der verfügbaren Eingänge und kopiert den kleinsten Wert ("Tag\_Value1") in den Operanden "Tag\_Result".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

### MAX: Maximum ermitteln

#### Beschreibung

Mit der Anweisung "Maximum ermitteln" vergleichen Sie die Werte der verfügbaren Eingänge und geben den größten Wert als Ergebnis aus.

An der Anweisung können mindestens zwei und maximal 32 Eingänge angegeben werden.

Das Ergebnis ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Die implizite Konvertierung der Datentypen schlägt während der Bearbeitung der Anweisung fehl.
- Eine Gleitpunktzahl weist einen ungültigen Wert auf.

#### Syntax

Für die Anweisung "Maximum ermitteln" wird die folgende Syntax verwendet:

##### SCL

```
MAX(IN1 := <Operand>,  
    IN2 := <Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Maximum ermitteln":

| Parameter     | Deklaration | Datentyp  |   | Beschreibung  |
|---------------|-------------|---|---|---|
|               |             | S7-1200   | S7-1500   |   |
| IN1           | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Erster<br>Eingangswert  |
| IN2           | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Zweiter<br>Eingangswert   |
| INn           | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Zusätzlich<br>eingefügte<br>Eingänge, deren<br>Werte verglichen<br>werden |
| Funktionswert |             | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Ergebnis der<br>Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result" := MAX(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
    
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand    | Wert   |
|---------------|------------|--------|
| IN1           | Tag_Value1 | 12 222 |
| IN2           | Tag_Value2 | 14 444 |
| IN3           | Tag_Value3 | 13 333 |
| Funktionswert | Tag_Result | 14 444 |

Die Anweisung vergleicht die Werte der angegebenen Operanden und kopiert den größten Wert ("Tag\_Value2") in den Operanden "Tag\_Result".

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

### LIMIT: Limitieren

#### Beschreibung

Mit der Anweisung "Limitieren" begrenzen Sie den Wert des Parameters IN auf die Werte der Parameter MN und MX. Der Wert des Parameters MN darf dabei nicht größer als der Wert des Parameters MX sein.

Wenn der Wert des Parameters IN die Bedingung  $MN \leq IN \leq MX$  erfüllt, wird er als Ergebnis der Anweisung ausgegeben. Wenn die Bedingung nicht erfüllt ist und der Eingangswert IN die Untergrenze MN unterschreitet, wird der Wert des Parameters MN als Ergebnis ausgegeben. Bei einer Überschreitung der Obergrenze MX wird der Wert des Parameters MX als Ergebnis ausgegeben.

Wenn der Wert am Eingang MN größer als am Eingang MX ist, dann ist das Ergebnis undefiniert.

Die Ausführung der Anweisung setzt voraus, dass die Operanden aller Parameter vom gleichen Datentyp sind.

#### Syntax

Für die Anweisung "Limitieren" wird die folgende Syntax verwendet:

```
SCL  
LIMIT (MN := <Operand>,  
       IN := <Operand>,  
       MX := <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp  |   | Beschreibung              |
|---------------|-------------|---|---|---------------------------|
|               |             | S7-1200   | S7-1500   |                           |
| MN            | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Untergrenze               |
| IN            | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Eingangswert              |
| MX            | Input       | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Obergrenze                |
| Funktionswert |             | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, DTL | Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>TOD, LTOD,<br>DATE, LDT, DT,<br>DTL | Ergebnis der<br>Anweisung |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

**SCL**

```
"Tag_Result" := LIMIT(MN := "Tag_Minimum",
                      IN := "Tag_Value",
                      MX := "Tag_Maximum");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand     | Wert   |
|---------------|-------------|--------|
| MN            | Tag_Minimum | 12 000 |
| IN            | Tag_Value   | 8 000  |
| MX            | Tag_Maximum | 16 000 |
| Funktionswert | Tag_Result  | 12 000 |

Der Wert des Operanden "Tag\_Value" wird mit den Werten der Operanden "Tag\_Minimum" und "Tag\_Maximum" verglichen. Da der Wert des Operanden "Tag\_Value" kleiner als der untere Grenzwert ist, wird der Wert des Operanden "Tag\_Minimum" in den Operanden "Tag\_Result" kopiert.

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

### SQR: Quadrat bilden

#### Beschreibung

Mit der Anweisung "Quadrat bilden" quadrieren Sie den Eingangswert und speichern das Ergebnis im angegebenen Operanden.

#### Syntax

Für die Anweisung "Quadrat bilden" wird die folgende Syntax verwendet:

```
SCL
SQR (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung              |
|---------------|-------------|------------------|---------------------------|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Eingangswert              |
| Funktionswert |             | Gleitpunktzahlen | Quadrat des Eingangswerts |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

#### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := SQR("Tag_Value");
"Tag_Result2" := SQR(SQR("Tag_Value1"))*"Tag_Value2";
```

Das Quadrat des Eingangswerts wird im Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert   |
|-------------|--------|
| Tag_Value   | 2.5    |
| Tag_Result1 | 6.25   |
| Tag_Value1  | 6.0    |
| Tag_Value2  | 2.0    |
| Tag_Result2 | 5184.0 |

## Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## SQRT: Quadratwurzel bilden

### Beschreibung

Mit der Anweisung "Quadratwurzel bilden" ziehen Sie aus dem Eingangswert die Quadratwurzel und speichern das Ergebnis im angegebenen Operanden. Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück. Wenn der Eingangswert "-0" ist, ist das Ergebnis auch "-0".

### Syntax

Für die Anweisung "Quadratwurzel bilden" wird die folgende Syntax verwendet:

|                   |
|-------------------|
| <b>SCL</b>        |
| SQRT (<Ausdruck>) |

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung                    |
|---------------|-------------|------------------|---------------------------------|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Eingangswert                    |
| Funktionswert |             | Gleitpunktzahlen | Quadratwurzel des Eingangswerts |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result1" := Sqrt("Tag_Value");
"Tag_Result2" := Sqrt((Sqr("Tag_Value1"))+"Tag_Value2");
    
```

Die Quadratwurzel des Eingangswerts wird im Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert |
|-------------|------|
| Tag_Value   | 4.0  |
| Tag_Result1 | 2.0  |
| Tag_Value1  | 3.0  |
| Tag_Value2  | 16.0 |
| Tag_Result2 | 5.0  |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## LN: Natürlichen Logarithmus bilden

### Beschreibung

Mit der Anweisung "Natürlichen Logarithmus bilden" berechnen Sie aus dem Eingangswert den natürlichen Logarithmus zur Basis e (e = 2.718282). Die Anweisung gibt ein positives Ergebnis aus, wenn der Eingangswert größer als Null ist. Bei Eingangswerten, die kleiner als Null sind, liefert die Anweisung eine ungültige Gleitpunktzahl zurück.

### Syntax

Für die Anweisung "Natürlichen Logarithmus bilden" wird die folgende Syntax verwendet:

```

SCL
LN(<Ausdruck>)
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:



| Parameter     | Deklaration | Datentyp             | Beschreibung                                 |
|---------------|-------------|----------------------|--|
| <Ausdruck>    | Input       | Gleitpunktzahle<br>n | Eingangswert                                 |
| Funktionswert |             | Gleitpunktzahle<br>n | Natürlicher Logarithmus des<br>Eingangswerts |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := LN("Tag_Value");
"Tag_Result2" := LN("Tag_Value1"+"Tag_Value2");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert  |
|-------------|-------|
| Tag_Value   | 2.5   |
| Tag_Result1 | 0.916 |
| Tag_Value1  | 1.5   |
| Tag_Value2  | 3.2   |
| Tag_Result2 | 1.548 |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## EXP: Exponentialwert bilden

### Beschreibung

Mit der Anweisung "Exponentialwert bilden" berechnen Sie die Potenz aus der Basis e (e = 2.718282) und dem Eingangswert und speichern das Ergebnis im angegebenen Operanden.

### Syntax

Für die Anweisung "Exponentialwert bilden" wird die folgende Syntax verwendet:

```
SCL
EXP (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung                      |
|---------------|-------------|------------------|-----------------------------------|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Eingangswert                      |
| Funktionswert |             | Gleitpunktzahlen | Exponentialwert des Eingangswerts |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := EXP("Tag_Value");
"Tag_Result2" := EXP("Tag_Value1"/"Tag_Value2");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert        |
|-------------|-------------|
| Tag_Value   | 20.5        |
| Tag_Result1 | 799 902 200 |
| Tag_Value1  | 15.5        |
| Tag_Value2  | 30.2        |
| Tag_Result2 | 1.671       |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## SIN: Sinuswert bilden

### Beschreibung

Mit der Anweisung "Sinuswert bilden" berechnen Sie den Sinus des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

### Syntax

Für die Anweisung "Sinuswert bilden" wird die folgende Syntax verwendet:

|                 |
|-----------------|
| <b>SCL</b>      |
| SIN(<Ausdruck>) |

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung                                   |
|---------------|-------------|------------------|--|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Eingangswert (Größe eines Winkels im Bogenmaß) |
| Funktionswert |             | Gleitpunktzahlen | Ergebnis der Anweisung                         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

|                                   |
|-----------------------------------|
| <b>SCL</b>                        |
| "Tag_Result" := SIN("Tag_Value"); |

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert                  |
|------------|-----------------------|
| Tag_Value  | +1.570796 ( $\pi/2$ ) |
| Tag_Result | 1.0                   |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**COS: Cosinuswert bilden**

**Beschreibung**

Mit der Anweisung "Cosinuswert bilden" berechnen Sie den Cosinus des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

**Syntax**

Für die Anweisung "Cosinuswert bilden" wird die folgende Syntax verwendet:

```
SCL
COS (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung                                   |
|---------------|-------------|------------------|--|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Eingangswert (Größe eines Winkels im Bogenmaß) |
| Funktionswert |             | Gleitpunktzahlen | Ergebnis der Anweisung                         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := COS("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert                  |
|------------|-----------------------|
| Tag_Value  | +1.570796 ( $\pi/2$ ) |
| Tag_Result | 0                     |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## TAN: Tangenswert bilden

### Beschreibung

Mit der Anweisung "Tangenswert bilden" berechnen Sie den Tangens des Eingangswerts. Der Eingangswert muss im Bogenmaß angegeben werden.

### Syntax

Für die Anweisung "Tangenswert bilden" wird die folgende Syntax verwendet:

| SCL             |
|-----------------|
| TAN(<Ausdruck>) |

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung                                   |
|---------------|-------------|------------------|--|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Eingangswert (Größe eines Winkels im Bogenmaß) |
| Funktionswert |             | Gleitpunktzahlen | Ergebnis der Anweisung                         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

| SCL                               |
|-----------------------------------|
| "Tag_Result" := TAN("Tag_Value"); |

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert                |
|------------|---------------------|
| Tag_Value  | +3.141593 ( $\pi$ ) |
| Tag_Result | 0                   |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**ASIN: Arcussinuswert bilden**

**Beschreibung**

Mit der Anweisung "Arcussinuswert bilden" berechnen Sie aus einem Sinuswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von  $-\pi/2$  bis  $+\pi/2$  liegen.

**Syntax**

Für die Anweisung "Arcussinuswert bilden" wird die folgende Syntax verwendet:

```

SCL
ASIN (<Ausdruck>)
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung                  |
|---------------|-------------|------------------|-------------------------------|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Sinuswert                     |
| Funktionswert |             | Gleitpunktzahlen | Größe des Winkels im Bogenmaß |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := ASIN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert                  |
|------------|-----------------------|
| Tag_Value  | 1.0                   |
| Tag_Result | +1.570796 ( $\pi/2$ ) |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## ACOS: Arcuscosinuswert bilden

### Beschreibung

Mit der Anweisung "Arcuscosinuswert bilden" berechnen Sie aus einem Cosinuswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen angegeben werden, die in einem Wertebereich von -1 bis +1 liegen. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von 0 bis  $+\pi$  liegen.

### Syntax

Für die Anweisung "Arcuscosinuswert bilden" wird die folgende Syntax verwendet:

```
SCL
ACOS (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung                  |
|---------------|-------------|------------------|-------------------------------|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Cosinuswert                   |
| Funktionswert |             | Gleitpunktzahlen | Größe des Winkels im Bogenmaß |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

scl
"Tag_Result" := ACOS("Tag_Value");
    
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert                  |
|------------|-----------------------|
| Tag_Value  | 0                     |
| Tag_Result | +1.570796 ( $\pi/2$ ) |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

### ATAN: Arcustangenswert bilden

#### Beschreibung

Mit der Anweisung "Arcustangenswert bilden" berechnen Sie aus einem Tangenswert die Größe des Winkels, der diesem Wert entspricht. Als Eingangswerte dürfen nur gültige Gleitpunktzahlen angegeben werden. Die berechnete Winkelgröße wird im Bogenmaß ausgegeben und kann in einem Wertebereich von  $-\pi/2$  bis  $+\pi/2$  liegen.



## Syntax

Für die Anweisung "Arcustangenswert bilden" wird die folgende Syntax verwendet:

```
SCL
ATAN (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp         | Beschreibung                  |
|---------------|-------------|------------------|-------------------------------|
| <Ausdruck>    | Input       | Gleitpunktzahlen | Tangenswert                   |
| Funktionswert |             | Gleitpunktzahlen | Größe des Winkels im Bogenmaß |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := ATAN("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert                  |
|------------|-----------------------|
| Tag_Value  | 1.0                   |
| Tag_Result | +0.785398 ( $\pi/4$ ) |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## FRAC: Nachkommastellen ermitteln

### Beschreibung

Mit der Anweisung "Nachkommastellen ermitteln" geben Sie die Nachkommastellen eines Wertes als Ergebnis aus. Beispielsweise liefert der Eingangswert 123.4567 als Ergebnis den Wert 0.4567.

### Syntax

Für die Anweisung "Nachkommastellen ermitteln" wird die folgende Syntax verwendet:

```
SCL
FRAC (<Ausdruck>)
FRAC_<Datentyp> (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp                              | Beschreibung  |
|---------------|-------------|---------------------------------------|---|
| <Ausdruck>    | Input       | Gleitpunktzahlen                      | Eingangswert  |
| _<Datentyp>   |             | Gleitpunktzahlen<br>Vorbelegung: REAL | Datentyp des Funktionswerts: <ul style="list-style-type: none"> <li>• Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe.</li> <li>• Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.</li> </ul> |
| Funktionswert |             | Gleitpunktzahlen                      | Nachkommastellen des Eingangswerts  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := FRAC("Tag_Value");
"Tag_Result2" := FRAC_LREAL("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert  |         |
|-------------|-------|---------|
| Tag_Value   | 2.555 | -1.4421 |
| Tag_Result1 | 0.555 | -0.4421 |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## Verschieben

### MOVE\_BLK: Bereich kopieren

#### Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des ersten Elements des Quellbereichs festgelegt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn die folgende Bedingung zutrifft:

- Es werden mehr Daten kopiert als am Parameter IN oder am Parameter OUT zur Verfügung gestellt werden.

#### Syntax

Für die Anweisung "Bereich kopieren" wird die folgende Syntax verwendet:

##### SCL

```
MOVE_BLK(IN := <Operand>,
          COUNT := <Operand>,
          OUT => <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

| Parameter   | Deklaration | Datentyp   |   | Beschreibung  |
|---|-------------|--|---|---|
|   |             | S7-1200  | S7-1500   |   |
| IN <sup>1)</sup>  | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten, DATE,<br>CHAR, TOD,<br>LTOD | Das erste Element<br>des<br>Quellbereichs,<br>das kopiert wird                                      |
| COUNT   | Input       | USINT, UINT,<br>UDINT  | USINT, UINT,<br>UDINT, ULINT  | Anzahl der<br>Elemente, die aus<br>dem Quellbereich<br>in den Zielbereich<br>kopiert werden         |
| OUT <sup>1)</sup>   | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten, DATE,<br>CHAR, TOD,<br>LTOD | Das erste Element<br>des Zielbereichs,<br>in den die Inhalte<br>des Quellbereichs<br>kopiert werden |
| <sup>1)</sup> Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden. |             |  |   |   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCI
MOVE_BLK(IN := #a_array[2],
         COUNT := "Tag_Count",
         OUT => #b_array[1]);
    
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a\_array und kopiert deren Inhalte in die Ausgangsvariable #b\_array, beginnend bei dem zweiten Element.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## MOVE\_BLK\_VARIANT: Bereich kopieren

### Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Sie können Elemente eines Arrays in ein anderes Array des selben Datentyps kopieren. Die Größen (Anzahl der Elemente) von Quell- und Ziel-Array dürfen unterschiedlich sein. Sie können mehrere Elemente innerhalb eines Arrays umkopieren oder einzelne Elemente kopieren.

Zum Zeitpunkt der Bausteinerstellung, wenn Sie die Anweisung verwenden, muss der Array noch nicht bekannt sein, da die Quelle und das Ziel per VARIANT übergeben werden.

Die Zählung an den Parametern SRC\_INDEX und DEST\_INDEX beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des Arrays.

Die Anweisung wird nicht ausgeführt, wenn mehr Daten kopiert werden, als zur Verfügung gestellt werden.

### Syntax

Für die Anweisung "Bereich kopieren" wird die folgende Syntax verwendet:

#### SCL

```
MOVE_BLK_VARIANT(SRC := <Operand>,
                  COUNT := <Operand>,
                  SRC_INDEX := <Operand>,
                  DEST_INDEX := <Operand>,
                  DEST := <Operand>,
                  RET_VAL => <Operand>)
```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

| Parameter  | Deklaration | Datentyp                                  | Beschreibung  |
|------------|-------------|---|---|
| SRC        | Input       | VARIANT<br>(Array oder einzelnes Element) | Das erste Element des Quellbereichs, das kopiert wird   |
| COUNT      | Input       | UDINT                                     | Gibt die Anzahl der Elemente, die kopiert werden, an.<br><br>Geben Sie dem Parameter COUNT den Wert „1“, wenn am Parameter SRC oder am Parameter DEST kein Array angegeben ist.   |
| SRC_INDEX  | Input       | UDINT                                     | <ul style="list-style-type: none"> <li>• Wenn am Parameter SRC ein Array angegeben ist, dann gibt der Parameter SRC_INDEX das erste Element am Parameter SRC an, das kopiert werden soll.</li> <li>• Wenn am Parameter SRC kein Array angegeben ist, dann geben Sie dem Parameter SRC_INDEX den Wert „0“.</li> </ul>      |
| DEST_INDEX | Input       | UDINT                                     | <ul style="list-style-type: none"> <li>• Wenn am Parameter DEST ein Array angegeben ist, dann gibt der Parameter DEST_INDEX das erste Element am Parameter DEST an, das kopiert werden soll.</li> <li>• Wenn am Parameter DEST kein Array angegeben ist, dann geben Sie dem Parameter DEST_INDEX den Wert „0“.</li> </ul> |
| DEST       | InOut       | VARIANT                                   | Das erste Element des Quellbereichs, das kopiert wird   |
| RET_VAL    | Output      | INT                                       | Fehlerinformationen:<br><br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameter RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 8082   | Datentypen stimmen nicht überein   |
| 80B4   | Datentypen stimmen nicht überein   |
| 8151   | Codeerzeugungsfehler am Parameter SRC  |
| 8152   | Codeerzeugungsfehler am Parameter SRC  |
| 8153   | Codeerzeugungsfehler am Parameter SRC  |
| 8251   | Ungültiger Wert am Parameter COUNT   |
| 8254   | Ungültiger Wert am Parameter COUNT   |
| 8281   | Ungültiger Wert am Parameter COUNT   |
| 8282   | Ungültiger Wert am Parameter COUNT   |
| 8382   | Der Wert am Parameter SRC_INDEX befindet sich außerhalb der Grenzwerte des VARIANT         |
| 8383   | Der Wert am Parameter SRC_INDEX befindet sich außerhalb des oberen Grenzwerts des Arrays.  |
| 8482   | Der Wert am Parameter DEST_INDEX befindet sich außerhalb der Grenzwerte des VARIANT        |
| 8483   | Der Wert am Parameter DEST_INDEX befindet sich außerhalb des oberen Grenzwerts des Arrays. |
| 8534   | Der Parameter DEST ist schreibgeschützt  |
| 8551   | Codeerzeugungsfehler am Parameter DEST   |
| 8552   | Codeerzeugungsfehler am Parameter DEST   |
| 8553   | Codeerzeugungsfehler am Parameter DEST   |
| 85A2   | Der Parameter DEST ist schreibgeschützt  |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

### SCL

```
MOVE_BLK_VARIANT(SRC := #SrcField,
                  COUNT := "Tag_Count",
                  SRC_INDEX := "Tag_Src_Index",
                  DEST_INDEX := "Tag_Dest_Index",
                  DEST := #DestField,
                  RET_VAL := "Tag_Result");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter  | Operand        | Wert   |
|------------|----------------|--|
| SRC        | #SrcField      | Der lokale Operand #SrcField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekanntes, UDT.  |
| COUNT      | Tag_Count      | 3  |
| SRC_INDEX  | Tag_Src_Index  | 4  |
| DEST_INDEX | Tag_Dest_Index | 2  |
| DEST       | #DestField     | Der lokale Operand #DestField verwendet einen, zum Zeitpunkt der Programmierung des Bausteins noch unbekanntes, UDT. |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**UMOVE\_BLK: Bereich ununterbrechbar kopieren**

**Beschreibung**

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich) ununterbrechbar. Die Anzahl der Elemente, die in den Zielbereich kopiert werden, legen Sie mit dem Parameter COUNT fest. Die Breite der zu kopierenden Elemente wird durch die Breite des ersten Elements des Quellbereichs festgelegt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

---

**Hinweis**

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

---

Der Wert am Ausgang OUT ist ungültig, wenn die folgende Bedingung zutrifft:



- Es werden mehr Daten kopiert als am Parameter IN oder am Parameter OUT zur Verfügung gestellt werden.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

## Syntax

Für die Anweisung "Bereich ununterbrechbar kopieren" wird die folgende Syntax verwendet:

```
SCL
UMOVE_BLK(IN := <Operand>,
          COUNT := <Operand>,
          OUT => <Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

| Parameter         | Deklaration | Datentyp   |   | Beschreibung  |
|-------------------|-------------|--|---|---|
|                   |             | S7-1200  | S7-1500   |   |
| IN <sup>1)</sup>  | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten, DATE,<br>CHAR, TOD,<br>LTOD | Das erste Element<br>des<br>Quellbereichs,<br>das kopiert wird                                      |
| COUNT             | Input       | USINT, UINT,<br>UDINT  | USINT, UINT,<br>UDINT, ULINT  | Anzahl der<br>Elemente, die aus<br>dem Quellbereich<br>in den Zielbereich<br>kopiert werden         |
| OUT <sup>1)</sup> | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten, DATE,<br>CHAR, TOD,<br>LTOD | Das erste Element<br>des Zielbereichs,<br>in den die Inhalte<br>des Quellbereichs<br>kopiert werden |

<sup>1)</sup> Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
UMOVE_BLK(IN := #a_array[2],
          COUNT := "Tag_Count",
          OUT => #b_array[1]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Die Anweisung selektiert ab dem dritten Element drei INT-Elemente aus der Variablen #a\_array und kopiert deren Inhalte in die Ausgangsvariable #b\_array, beginnend bei dem zweiten Element. Der Kopiervorgang kann durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**FILL\_BLK: Bereich befüllen**

**Beschreibung**

Mit der Anweisung "Bereich befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich). Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Quellbereich selektiert und so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

Der Wert am Ausgang OUT ist ungültig, wenn die folgende Bedingung zutrifft:

- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

**Syntax**

Für die Anweisung "Bereich befüllen" wird die folgende Syntax verwendet:

```
SCL
FILL_BLK(IN := <Operand>,
```

**SCL**

```
COUNT := <Operand>,
OUT => <Operand>
```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

| Parameter  | Deklaration | Datentyp   |   | Beschreibung  |
|--|-------------|--|---|---|
|  |             | S7-1200  | S7-1500   |   |
| IN <sup>1)</sup>   | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, CHAR | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten, DATE,<br>CHAR, TOD,<br>LTOD | Element, mit dem<br>der Zielbereich<br>befüllt wird |
| COUNT  | Input       | USINT, UINT,<br>UDINT  | USINT, UINT,<br>UDINT, ULINT  | Anzahl der<br>Kopierwiederholun<br>gen              |
| OUT <sup>2)</sup>  | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, CHAR | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten DATE,<br>CHAR, TOD,<br>LTOD  | Adresse im<br>Zielbereich, ab der<br>befüllt wird   |
| <sup>1)</sup> Die angegebenen Datentypen können auch als Elemente einer ARRAY-Struktur verwendet werden. |             |  |   |   |
| <sup>2)</sup> Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.  |             |  |   |   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

**SCL**

```
FILL_BLK(IN := #a_array[2],
COUNT := "Tag_Count",
OUT => #b_array[1]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Die Anweisung kopiert drei Mal das dritte Element (#a\_array[2]) der Variablen #a\_array in die Ausgangsvariable #b\_array.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**UFILL\_BLK: Bereich ununterbrechbar befüllen**

**Beschreibung**

Mit der Anweisung "Bereich ununterbrechbar befüllen" befüllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) ununterbrechbar. Die Anzahl der Kopierwiederholungen wird mit dem Parameter COUNT festgelegt. Bei der Ausführung der Anweisung wird der Wert am Eingang IN selektiert und so oft in den Zielbereich kopiert, wie der Wert am Parameter COUNT vorgibt.

Die Bearbeitung der Anweisung setzt voraus, dass der Quellbereich und der Zielbereich vom gleichen Datentyp sind.

---

**Hinweis**

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Aus diesem Grund können sich die Alarmreaktionszeiten der CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar befüllen" erhöhen.

---

Der Wert des Ausgangs OUT ist ungültig, wenn die folgende Bedingung zutrifft:

- Es werden mehr Daten kopiert als am Eingang IN oder am Ausgang OUT zur Verfügung gestellt werden.

Mit der Anweisung "Bereich ununterbrechbar befüllen" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

## Syntax

Für die Anweisung "Bereich ununterbrechbar befüllen" wird die folgende Syntax verwendet:

```
SCL
UFILL_BLK(IN := <Operand>,
          COUNT := <Operand>,
          OUT => <Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar befüllen":

| Parameter  | Deklaration | Datentyp   |   | Beschreibung  |
|--|-------------|--|---|---|
|  |             | S7-1200  | S7-1500   |   |
| IN <sup>1)</sup>   | Input       | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE,<br>CHAR, TOD | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten, DATE,<br>CHAR, TOD,<br>LTOD | Element, mit dem<br>der Zielbereich<br>befüllt wird |
| COUNT  | Input       | USINT, UINT,<br>UDINT  | USINT, UINT,<br>UDINT, ULINT  | Anzahl der<br>Kopierwiederholun<br>gen              |
| OUT <sup>2)</sup>  | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD,<br>DATE, CHAR | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeiten, DATE,<br>CHAR, TOD,<br>LTOD | Adresse im<br>Zielbereich, ab der<br>befüllt wird   |
| <sup>1)</sup> Die angegebenen Datentypen können auch als Elemente einer ARRAY-Struktur verwendet werden. |             |  |   |   |
| <sup>2)</sup> Die angegebenen Datentypen können nur als Elemente einer ARRAY-Struktur verwendet werden.  |             |  |   |   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
UFILL_BLK(IN := #a_array[2],
          COUNT := "Tag_Count",
          OUT => #b_array[1]);
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert  |
|-----------|------------|---|
| IN        | a_array[2] | Der Operand "a_array" ist vom Datentyp Array [0..5] of INT. Er besteht aus 6 Elementen des Datentyps INT. |
| COUNT     | Tag_Count  | 3   |
| OUT       | b_array[1] | Der Operand "b_array" ist vom Datentyp Array [0..6] of INT. Er besteht aus 7 Elementen des Datentyps INT. |

Die Anweisung kopiert drei Mal das dritte Element (#a\_array[2]) der Variablen #a\_array in die Ausgangsvariable #b\_array. Der Kopiervorgang kann dabei durch andere Tätigkeiten des Betriebssystems nicht unterbrochen werden.

**Siehe auch**

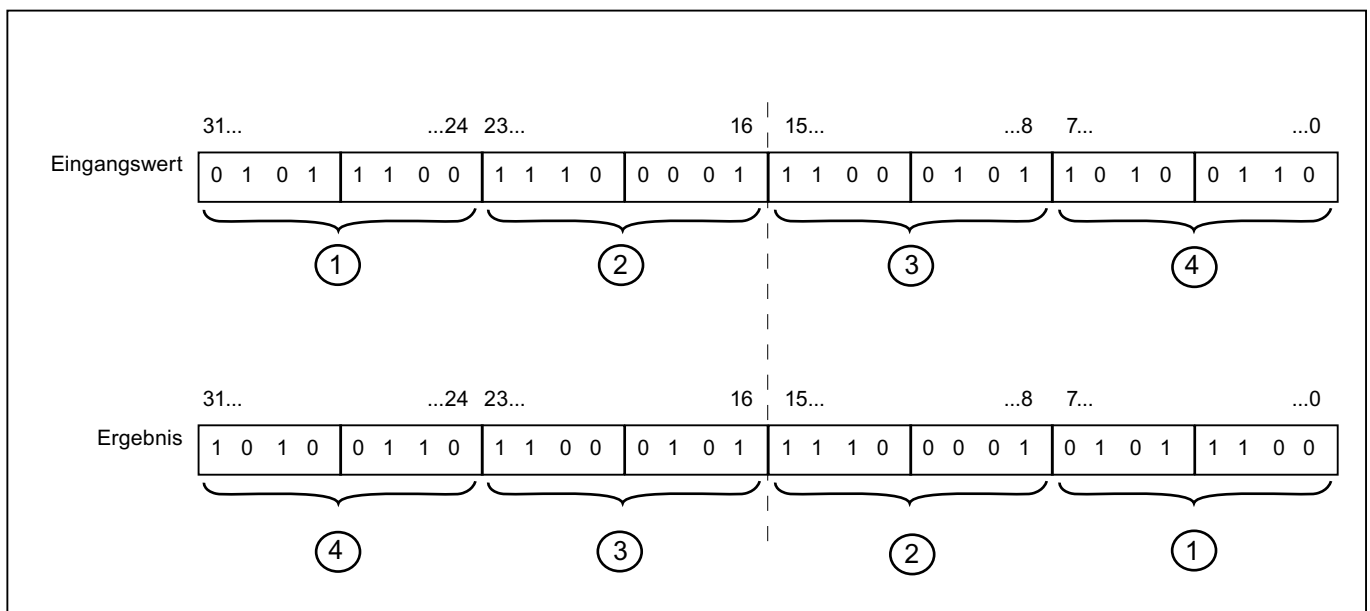
- Übersicht über die gültigen Datentypen (Seite 1087)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**SWAP: Anordnung ändern**

**Beschreibung**

Mit der Anweisung "Anordnung ändern" ändern Sie die Anordnung der Bytes eines Eingangswerts und speichern das Ergebnis im angegebenen Operanden.

Das folgende Bild zeigt, wie die Bytes eines Operanden vom Datentyp DWORD mit der Anweisung "Anordnung ändern" vertauscht werden:



## Syntax

Für die Anweisung "Anordnung ändern" wird die folgende Syntax verwendet:

```

SCL
SWAP (<Ausdruck>)

```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp    |                    | Beschreibung           |
|---------------|-------------|-------------|--------------------|------------------------|
|               |             | S7-1200     | S7-1500            |                        |
| <Ausdruck>    | Input       | WORD, DWORD | WORD, DWORD, LWORD | Eingangswert           |
| Funktionswert |             | WORD, DWORD | WORD, DWORD, LWORD | Ergebnis der Anweisung |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result" := SWAP("Tag_Value");

```

Das Ergebnis der Anweisung wird als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert                |
|------------|---------------------|
| Tag_Value  | 0000 1111 0101 0101 |
| Tag_Result | 0101 0101 0000 1111 |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## Array-Datenbausteine

### ReadFromArrayDB: Aus Array-Datenbaustein lesen

#### Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein lesen" lesen Sie Daten aus einem Array-DB.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert "0", unabhängig von der späteren Deklaration des Arrays.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein lesen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, aus dem gelesen wird  |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element, das gelesen wird  |
| VALUE     | InOut       | VARIANT  | E, A, M, D, L   | Wert, der gelesen wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".



## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 80B4   | Datentypen stimmen nicht überein   |
| 80B5   | Kopiervorgang wurde abgebrochen  |
| 8131   | Der Datenbaustein existiert nicht  |
| 8132   | Der Datenbaustein ist zu kurz  |
| 8133   | Der Datenbaustein liegt im Ladespeicher  |
| 8134   | Der Datenbaustein ist schreibgeschützt   |
| 8135   | Der Datenbaustein ist kein Array-DB  |
| 8282   | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350   | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352   | Codeerzeugungsfehler   |
| 8353   | Codeerzeugungsfehler   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## WriteToArrayDB: In Array-Datenbaustein schreiben

### Beschreibung

Mit der Anweisung "In Array-Datenbaustein schreiben" schreiben Sie Daten in einen Array-DB.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein schreiben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, in den geschrieben wird   |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element im DB, in das geschrieben wird   |
| VALUE     | Input       | VARIANT  | E, A, M, D, L   | Wert, der geschrieben wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Parameter RET\_VAL**

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 80B4   | Datentypen stimmen nicht überein   |
| 80B5   | Kopiervorgang wurde abgebrochen  |
| 8131   | Der Datenbaustein existiert nicht  |
| 8132   | Der Datenbaustein ist zu kurz  |
| 8133   | Der Datenbaustein liegt im Ladespeicher  |
| 8134   | Der Datenbaustein ist schreibgeschützt   |
| 8135   | Der Datenbaustein ist kein Array-DB  |
| 8282   | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350   | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352   | Codeerzeugungsfehler   |
| 8353   | Codeerzeugungsfehler   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## ReadFromArrayDBL: Aus Array-Datenbaustein im Ladespeicher lesen

### Beschreibung

Mit der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen" lesen Sie Daten aus einem Array-DB im Ladespeicher.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Wenn der Array-DB mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalfanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalfanke erfasst wird, ist die Anweisung beendet. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1" und innerhalb dieses Zyklus wird der ausgelesene Wert am Parameter VALUE ausgegeben. Bei allen anderen Programmzyklen wird der Wert am Parameter VALUE nicht verändert.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Aus Array-Datenbaustein im Ladespeicher lesen":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L   | REQ = "1": Mit dem Auslesen des Array-DBs beginnen   |
| DB        | Input       | DB_ANY   | E, A, M, D, L   | Datenbaustein, aus dem gelesen wird  |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element, das gelesen wird  |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | BUSY = "1": Das Auslesen des Array-DBs ist noch nicht abgeschlossen  |
| DONE      | Output      | BOOL     | E, A, M, D, L   | DONE = "1": Die Anweisung wurde erfolgreich ausgeführt   |
| VALUE     | InOut       | VARIANT  | E, A, M, D, L   | Wert, der gelesen wird   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Parameter**

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode* (W#16#...) | Erläuterung                       |
|------------------------|-----------------------------------|
| 0000                   | Kein Fehler                       |
| 80B4                   | Datentypen stimmen nicht überein  |
| 80B5                   | Kopiervorgang wurde abgebrochen   |
| 8131                   | Der Datenbaustein existiert nicht |
| 8132                   | Der Datenbaustein ist zu kurz     |

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 8133   | Der Datenbaustein liegt nicht im Ladespeicher                                  |
| 8134   | Der Datenbaustein ist schreibgeschützt   |
| 8135   | Der Datenbaustein ist kein Array-DB  |
| 8282   | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350   | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352   | Codeerzeugungsfehler   |
| 8353   | Codeerzeugungsfehler   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## WriteToArrayDBL: In Array-Datenbaustein im Ladespeicher schreiben

### Beschreibung

Mit der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben" schreiben Sie Daten in einen Array-DB im Ladespeicher.

Ein Array-DB ist ein Datenbaustein, der genau aus einem Array of [Datentyp] besteht. Die Elemente des Arrays können vom Datentyp UDT oder jedes anderen elementaren Datentyps sein. Die Zählung des Arrays beginnt immer mit dem unteren Grenzwert „0“, unabhängig von der späteren Deklaration des Arrays.

Wenn der Array-DB mit dem Bausteinattribut "Nur im Ladespeicher ablegen" gekennzeichnet worden ist, wird er nur im Ladespeicher abgelegt.

Wenn am Parameter REQ eine positive Signalfanke erfasst wird, wird die Anweisung ausgeführt. Der Parameter BUSY hat dann den Signalzustand "1". Wenn am Parameter BUSY eine negative Signalfanke erfasst wird, ist die Anweisung beendet und der Wert des Parameters VALUE wird in den Datenbaustein geschrieben. Einen Programmzyklus lang hat der Parameter DONE den Signalzustand "1".

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Der Freigabeausgang ENO liefert den Signalzustand "0", wenn eine der folgenden Bedingungen zutrifft:

- Der Freigabeeingang EN liefert den Signalzustand "0".
- Wenn ein Fehler während der Bearbeitung der Anweisung auftritt.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "In Array-Datenbaustein im Ladespeicher schreiben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L   | REQ = "1": Mit dem Schreiben in den Array-DB beginnen  |
| DB        | Input       | DB_ANY   | D               | Datenbaustein, in den geschrieben wird   |
| INDEX     | Input       | DINT     | E, A, M, D, L   | Element im DB, in das geschrieben wird   |
| VALUE     | Input       | VARIANT  | E, A, M, D, L   | Wert, der geschrieben wird   |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | BUSY = "1": Das Schreiben in den Array-DB ist noch nicht abgeschlossen   |
| DONE      | Output      | BOOL     | E, A, M, D, L   | DONE = "1": Die Anweisung wurde erfolgreich ausgeführt   |
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 80B4   | Datentypen stimmen nicht überein   |
| 80B5   | Kopiervorgang wurde abgebrochen  |
| 8131   | Der Datenbaustein existiert nicht  |
| 8132   | Der Datenbaustein ist zu kurz  |
| 8133   | Der Datenbaustein liegt nicht im Ladespeicher                                  |
| 8134   | Der Datenbaustein ist schreibgeschützt   |
| 8135   | Der Datenbaustein ist kein Array-DB  |
| 8282   | Der Wert am Parameter INDEX befindet sich außerhalb der Grenzwerte des Arrays. |
| 8350   | Der Datentyp VARIANT enthält keinen Wert                                       |
| 8352   | Codeerzeugungsfehler   |
| 8353   | Codeerzeugungsfehler   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## Lese-/Schreibzugriff

### PEEK: Speicheradresse lesen

#### Beschreibung

Mit der Anweisung "Speicheradresse lesen" lesen Sie eine Speicheradresse ohne Angabe eines Datentyps aus einem Speicherbereich aus.

Wenn Sie am Parameter AREA den Bereich 16#84 für einen Datenbaustein angeben, dann können Sie nur auf Datenbausteine mit der Bausteineigenschaft "Standard" zugreifen.

**Syntax**

Für die Anweisung "Speicheradresse lesen" wird die folgende Syntax verwendet:

**SCL**

```
PEEK (AREA := <Operand>,
      DBNUMBER := <Operand>,
      BYTEOFFSET := <Operand>)
```

**SCL**

```
PEEK_<Datentyp> (AREA := <Operand>,
                DBNUMBER := <Operand>,
                BYTEOFFSET := <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp                       | Beschreibung  |
|---------------|-------------|--------------------------------|---|
| AREA          | Input       | BYTE                           | Folgende Bereiche können ausgewählt werden: <ul style="list-style-type: none"> <li>• 16#81: Input</li> <li>• 16#82: Output</li> <li>• 16#83: Merker</li> <li>• 16#84: DB</li> <li>• 16#1: Peripherieeingang (nur S7-1500)</li> </ul>  |
| DBNUMBER      | Input       | DINT, DB_ANY                   | Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"  |
| BYTEOFFSET    | Input       | DINT                           | Adresse, an der ausgelesen wird   |
| _<Datentyp>   |             | Bitfolgen<br>Vorbelegung: BYTE | Datentyp des Funktionswerts: <ul style="list-style-type: none"> <li>• Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe.</li> <li>• Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.</li> </ul> |
| Funktionswert |             | Bitfolgen                      | Ergebnis der Anweisung  |



Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

---

### Hinweis

Wenn Sie die Speicheradresse aus den Bereichen Input, Output oder Merker auslesen, müssen Sie den Parameter "DBNUMBER" mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

---

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

#### SCL

```
"Tag_Result1" := PEEK(AREA := "Tag_Area",
                      DBNUMBER := "Tag_DBNumber",
                      BYTEOFFSET := "Tag_Byte");
```

#### SCL

```
"Tag_Result2" := PEEK_WORD(AREA := "Tag_Area",
                             DBNUMBER := "Tag_DBNumber",
                             BYTEOFFSET := "Tag_Byte");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand      | Wert                 |
|---------------|--------------|----------------------|
| AREA          | Tag_Area     | 16#84                |
| DBNUMBER      | Tag_DBNumber | 5                    |
| BYTEOFFSET    | Tag_Byte     | 20                   |
| Funktionswert | Tag_Result1  | Wert des Bytes "20"  |
| Funktionswert | Tag_Result2  | Wert des Wortes "20" |

Die Anweisung liest den Wert an der Adresse "20" aus dem Operanden "Tag\_Byte" aus dem Datenbaustein "5" aus und liefert das Ergebnis am Operanden "Tag\_Result" als Funktionswert zurück.

### Siehe auch

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

Übersicht über die gültigen Datentypen (Seite 1087)

## PEEK\_BOOL: Speicherbit lesen

### Beschreibung

Mit der Anweisung "Speicherbit lesen" lesen Sie ein Speicherbit ohne Angabe eines Datentyps aus einem Speicherbereich aus.

Wenn Sie am Parameter AREA den Bereich 16#84 für einen Datenbaustein angeben, dann können Sie nur auf Datenbausteine mit der Bausteineigenschaft "Standard" zugreifen.

---

#### Hinweis

Mit der Anweisung können Sie nur auf "Standard" Speicherbereiche zugreifen.

---

### Syntax

Für die Anweisung "Speicherbit lesen" wird die folgende Syntax verwendet:

#### SCL

```
PEEK_BOOL (AREA := <Operand>,
           DBNumber := <Operand>,
           BYTEOFFSET := <Operand>,
           BITOFFSET := <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp     | Beschreibung   |
|---------------|-------------|--------------|--|
| AREA          | Input       | BYTE         | Folgende Bereiche können ausgewählt werden: <ul style="list-style-type: none"> <li>• 16#81: Input</li> <li>• 16#82: Output</li> <li>• 16#83: Merker</li> <li>• 16#84: DB</li> <li>• 16#1: Peripherieeingang (nur S7-1500)</li> </ul> |
| DBNUMBER      | Input       | DINT, DB_ANY | Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"   |
| BYTEOFFSET    | Input       | DINT         | Adresse, an der ausgelesen wird  |
| BITOFFSET     | Input       | INT          | Bit, an dem ausgelesen wird  |
| Funktionswert |             | BOOL         | Ergebnis der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

---

### Hinweis

Wenn Sie das Speicherbit aus den Bereichen Input, Output oder Merker auslesen, müssen Sie den Parameter "DBNUMBER" mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

---

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

#### SCL

```
"Tag_Result" := PEEK_BOOL(AREA := "Tag_Area",
                          DBNUMBER := "Tag_DBNumber",
                          BYTEOFFSET := "Tag_Byte",
                          BITOFFSET := "Tag_Bit");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand      | Wert  |
|---------------|--------------|-------|
| AREA          | Tag_Area     | 16#84 |
| DBNUMBER      | Tag_DBNumber | 5     |
| BYTEOFFSET    | Tag_Byte     | 20    |
| BITOFFSET     | Tag_Bit      | 3     |
| Funktionswert | Tag_Result   | 3     |

Die Anweisung liest den Wert am Speicherbit "3" aus dem Operanden "Tag\_Bit" aus dem Datenbaustein "5" aus dem Byte "20" aus und liefert das Ergebnis am Operanden "Tag\_Result" als Funktionswert zurück.

### Siehe auch

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

Übersicht über die gültigen Datentypen (Seite 1087)

### POKE: Speicheradresse schreiben

#### Beschreibung

Mit der Anweisung "Speicheradresse schreiben" schreiben Sie eine Speicheradresse ohne Angabe eines Datentyps in einen Speicherbereich.

Wenn Sie am Parameter AREA den Bereich 16#84 für einen Datenbaustein angeben, dann können Sie nur auf Datenbausteine mit der Bausteineigenschaft "Standard" zugreifen.

**Hinweis**

Mit der Anweisung können Sie nur auf "Standard" Speicherbereiche zugreifen.

**Syntax**

Für die Anweisung "Speicheradresse schreiben" wird die folgende Syntax verwendet:

**SCI**

```
POKE (AREA := <Operand>,
      DBNUMBER := <Operand>,
      BYTEOFFSET := <Operand>,
      VALUE := <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter  | Deklaration | Datentyp     | Beschreibung   |
|------------|-------------|--------------|--|
| AREA       | Input       | BYTE         | Folgende Bereiche können ausgewählt werden: <ul style="list-style-type: none"> <li>• 16#81: Input</li> <li>• 16#82: Output</li> <li>• 16#83: Merker</li> <li>• 16#84: DB</li> <li>• 16#2: Peripherieausgang (nur S7-1500)</li> </ul> |
| DBNUMBER   | Input       | DINT, DB_ANY | Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"   |
| BYTEOFFSET | Input       | DINT         | Adresse, die geschrieben wird  |
| VALUE      | Input       | Bitfolgen    | Wert, der geschrieben wird   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Hinweis**

Wenn Sie die Speicheradresse in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter "DBNUMBER" mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL  
POKE (AREA := "Tag_Area",  
      DBNUMBER := "Tag_DBNumber",  
      BYTEOFFSET := "Tag_Byte"),  
      VALUE := "Tag_Value";
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter  | Operand      | Wert  |
|------------|--------------|-------|
| AREA       | Tag_Area     | 16#84 |
| DBNUMBER   | Tag_DBNumber | 5     |
| BYTEOFFSET | Tag_Byte     | 20    |
| VALUE      | Tag_Value    | 16#11 |

Die Anweisung überschreibt die Speicheradresse "20" im Datenbaustein "5" mit dem Wert "16#11".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## POKE\_BOOL: Speicherbit schreiben

### Beschreibung

Mit der Anweisung "Speicherbit schreiben" schreiben Sie ein Speicherbit ohne Angabe eines Datentyps in einen Speicherbereich.

Wenn Sie am Parameter AREA den Bereich 16#84 für einen Datenbaustein angeben, dann können Sie nur auf Datenbausteine mit der Bausteineigenschaft "Standard" zugreifen.

---

#### Hinweis

Mit der Anweisung können Sie nur auf "Standard" Speicherbereiche zugreifen.

---

**Syntax**

Für die Anweisung "Speicherbit schreiben" wird die folgende Syntax verwendet:

```

SCL
POKE_BOOL (AREA := <Operand>,
           DBNUMBER := <Operand>,
           BYTEOFFSET := <Operand>,
           BITOFFSET := <Operand>,
           VALUE := <Operand>)
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter  | Deklaration | Datentyp     | Beschreibung   |
|------------|-------------|--------------|--|
| AREA       | Input       | BYTE         | Folgende Bereiche können ausgewählt werden: <ul style="list-style-type: none"> <li>• 16#81: Input</li> <li>• 16#82: Output</li> <li>• 16#83: Merker</li> <li>• 16#84: DB</li> <li>• 16#2: Peripherieausgang (nur S7-1500)</li> </ul> |
| DBNUMBER   | Input       | DINT, DB_ANY | Nummer des Datenbausteins, wenn AREA = DB, ansonsten "0"   |
| BYTEOFFSET | Input       | DINT         | Adresse, die geschrieben wird  |
| BITOFFSET  | Input       | INT          | Bit, das geschrieben wird  |
| VALUE      | Input       | BOOL         | Wert, der geschrieben wird   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Hinweis**

Wenn Sie das Speicherbit in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter "DBNUMBER" mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
POKE_BOOL (AREA := "Tag_Area",
    
```

**SCL**

```

DBNUMBER := "Tag_DBNumber",
BYTEOFFSET := "Tag_Byte",
BITOFFSET := "Tag_Bit",
VALUE := "Tag_Value");

```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter  | Operand      | Wert  |
|------------|--------------|-------|
| AREA       | Tag_Area     | 16#84 |
| DBNUMBER   | Tag_DBNumber | 5     |
| BYTEOFFSET | Tag_Byte     | 20    |
| BITOFFSET  | Tag_Bit      | 3     |
| VALUE      | Tag_Value    | M0.0  |

Die Anweisung überschreibt das Speicherbit "3" im Datenbaustein "5" im Byte "20" mit dem Wert "M0.0".

**Siehe auch**

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

Übersicht über die gültigen Datentypen (Seite 1087)

**POKE\_BLK: Speicherbereich schreiben****Beschreibung**

Mit der Anweisung "Speicherbereich schreiben" schreiben Sie einen Speicherbereich ohne Angabe eines Datentyps in einen anderen Speicherbereich.

Wenn Sie am Parameter AREA den Bereich 16#84 für einen Datenbaustein angeben, dann können Sie nur auf Datenbausteine mit der Bausteineigenschaft "Standard" zugreifen.

**Hinweis**

Mit der Anweisung können Sie nur auf "Standard" Speicherbereiche zugreifen.

**Syntax**

Für die Anweisung "Speicherbereich schreiben" wird die folgende Syntax verwendet:

**SCL**

```

POKE_BLK(AREA_SRC := <Operand>,
          DBNUMBER_SRC := <Operand>,
          BYTEOFFSET_SRC := <Operand>);

```

**SCL**

```

AREA_DEST := <Operand>,
DBNUMBER_DEST := <Operand>,
BYTEOFFSET_DEST := <Operand>,
COUNT := <Operand>
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter       | Deklaration | Datentyp     | Beschreibung  |
|-----------------|-------------|--------------|---|
| AREA_SRC        | Input       | BYTE         | Folgende Bereiche können Sie im Quellspeicherbereich auswählen: <ul style="list-style-type: none"> <li>• 16#81: Input</li> <li>• 16#82: Output</li> <li>• 16#83: Merker</li> <li>• 16#84: DB</li> </ul> |
| DBNUMBER_SRC    | Input       | DINT, DB_ANY | Nummer des Datenbausteins im Quellspeicherbereich, wenn AREA = DB, ansonsten "0"  |
| BYTEOFFSET_SRC  | Input       | DINT         | Adresse im Quellspeicherbereich, die geschrieben wird   |
| AREA_DEST       | Input       | BYTE         | Folgende Bereiche können Sie im Zielspeicherbereich auswählen: <ul style="list-style-type: none"> <li>• 16#81: Input</li> <li>• 16#82: Output</li> <li>• 16#83: Merker</li> <li>• 16#84: DB</li> </ul>  |
| DBNUMBER_DEST   | Input       | DINT, DB_ANY | Nummer des Datenbausteins im Zielspeicherbereich, wenn AREA = DB, ansonsten "0"   |
| BYTEOFFSET_DEST | Input       | DINT         | Adresse im Zielspeicherbereich, die geschrieben wird  |
| COUNT           | Input       | DINT         | Anzahl der Bytes, die kopiert werden  |



Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Hinweis

Wenn Sie die Speicheradresse in die Bereiche Input, Output oder Merker schreiben, müssen Sie den Parameter "DBNUMBER" mit dem Wert "0" versorgen, da die Anweisung sonst fehlerhaft ist.

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

#### SCL

```
POKE_BLK (AREA_SRC := "Tag_Source_Area",
          DBNUMBER_SRC := "Tag_Source_DBNumber",
          BYTEOFFSET_SRC := "Tag_Source_Byte"),
          AREA_DEST := "Tag_Destination_Area",
          DBNUMBER_DEST := "Tag_Destination_DBNumber",
          BYTEOFFSET_DEST := "Tag_Destination_Byte",
          COUNT := "Tag_Count");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter       | Operand                  | Wert  |
|-----------------|--------------------------|-------|
| AREA_SRC        | Tag_Source_Area          | 16#84 |
| DBNUMBER_SRC    | Tag_Source_DBNumber      | 5     |
| BYTEOFFSET_SRC  | Tag_Source_Byte          | 20    |
| AREA_DEST       | Tag_Destination_Area     | 16#83 |
| DBNUMBER_DEST   | Tag_Destination_DBNumber | 0     |
| BYTEOFFSET_DEST | Tag_Destination_Byte     | 30    |
| COUNT           | Tag_Count                | 100   |

Die Anweisung schreibt 100 Byte aus dem Datenbaustein "5" beginnend bei der Adresse "20" in den Merker-Speicherbereich beginnend bei der Adresse "30".

### Siehe auch

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

Übersicht über die gültigen Datentypen (Seite 1087)

## Weitere

### BLKMOV: Bereich kopieren

#### Beschreibung

Mit der Anweisung "Bereich kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

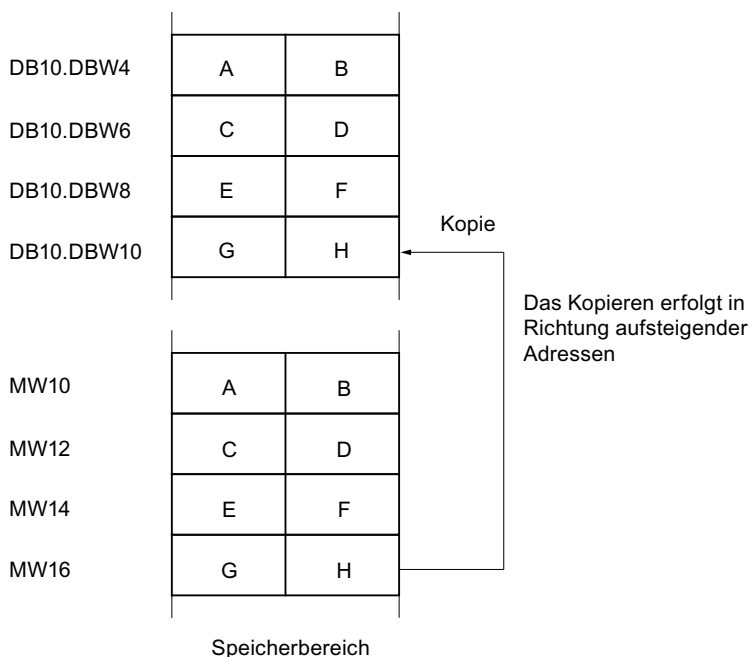
---

#### Hinweis

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

---

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:



#### Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich kopieren" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

## Unterbrechbarkeit

Es gibt keine Begrenzung der Schachtelungstiefe.

## Speicherbereiche

Mit der Anweisung "Bereich kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

## Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der Quell- und der Zielbereich unterschiedlich lang sind, wird nur bis zur Länge des kleineren Bereichs kopiert.

Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

## Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden ebenfalls in den Zielbereich geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt.

Falls Sie die Informationen über die maximale und tatsächliche Länge einer Zeichenkette kopieren wollen, geben Sie die Bereiche an den Parametern SRCBLK und DSTBLK in Bytes an.

## Syntax

Für die Anweisung "Bereich kopieren" wird die folgende Syntax verwendet:

```
SCL  
BLKMOV (SRCBLK := <Operand>,  
        DSTBLK => <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich kopieren":

| Parameter               | Deklaration | Datentyp | Beschreibung  |
|-------------------------|-------------|----------|---|
| SRCBLK                  | Input       | VARIANT  | Angabe des Speicherbereichs, der kopiert wird (Quellbereich).   |
| DSTBLK                  | Output      | VARIANT  | Angabe des Speicherbereichs, in den kopiert wird (Zielbereich). |
| Funktionswert (RET_VAL) |             | INT      | Fehlerinformationen   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode* (W#16#...)       | Erläuterung  |
|------------------------------|--|
| 0000                         | Kein Fehler  |
| 8091                         | Die zulässige Schachtelungstiefe wurde überschritten.  |
| 8092                         | Die Anweisung kann nicht ausgeführt werden, weil ein angegebener Datenbaustein schreibgeschützt, nicht ablauffähig oder nicht geladen ist. |
| allgemeine Fehlerinformation | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"   |

\*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2304)

### UBLKMOV: Bereich ununterbrechbar kopieren

#### Beschreibung

Mit der Anweisung "Bereich ununterbrechbar kopieren" kopieren Sie die Inhalte eines Speicherbereichs (Quellbereich) in einen anderen Speicherbereich (Zielbereich). Der

Kopiervorgang findet in Richtung aufsteigender Adressen statt. Den Quell- und Zielbereich definieren Sie durch VARIANT.

Der Kopiervorgang kann nicht durch andere Tätigkeiten des Betriebssystems unterbrochen werden. Dadurch kann sich die Alarmreaktionszeit Ihrer CPU während der Bearbeitung der Anweisung "Bereich ununterbrechbar kopieren" erhöhen.

---

**Hinweis**

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

---

## Speicherbereiche

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

## Allgemeine Regeln beim Kopieren

Bei der Ausführung der Anweisung "Bereich ununterbrechbar kopieren" dürfen sich der Quell- und der Zielbereich nicht überlappen. Wenn der Quellbereich kleiner als der Zielbereich ist, wird der Quellbereich komplett in den Zielbereich geschrieben. Die restlichen Bytes des Zielbereichs bleiben unverändert.

Wenn der Zielbereich kleiner als der Quellbereich ist, wird der Zielbereich komplett beschrieben. Die restlichen Bytes des Quellbereichs werden ignoriert.

Falls ein als Formalparameter definierter Ziel- oder Quellbereich kleiner als ein am Parameter SRCBLK oder DSTBLK angegebener Ziel- oder Quellbereich ist, werden keine Daten übertragen.

Wenn ein Bereich vom Datentyp BOOL kopiert wird, muss die Variable absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie maximal 16 KB kopieren. Beachten Sie dabei die CPU-spezifischen Einschränkungen.

## Regeln beim Kopieren von Zeichenketten

Mit der Anweisung "Bereich ununterbrechbar kopieren" können Sie auch Quell- und Zielbereiche vom Datentyp STRING kopieren. Wenn nur der Quellbereich vom Datentyp STRING ist, werden die Zeichen kopiert, die tatsächlich in der Zeichenkette enthalten sind. Informationen über tatsächliche und maximale Länge werden nicht in den Zielbereich

geschrieben. Wenn der Quell- und der Zielbereich jeweils vom Datentyp STRING sind, wird die aktuelle Länge der Zeichenkette im Zielbereich auf die Anzahl der tatsächlich kopierten Zeichen gesetzt. Wenn Bereiche vom Datentyp STRING kopiert werden, müssen Sie als Bereichslänge "1" angegeben werden.

### Syntax

Für die Anweisung "Bereich ununterbrechbar kopieren" wird die folgende Syntax verwendet:

#### SCL

```
UBLKMOV (SRCBLK := <Operand>,
        DSTBLK => <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich ununterbrechbar kopieren":

| Parameter               | Deklaration | Datentyp | Beschreibung  |
|-------------------------|-------------|----------|---|
| SRCBLK                  | Input       | VARIANT  | Angabe des Speicherbereichs, der kopiert wird (Quellbereich).   |
| DSTBLK                  | Output      | ANY      | Angabe des Speicherbereichs, in den kopiert wird (Zielbereich). |
| Funktionswert (RET_VAL) |             | VARIANT  | Fehlerinformationen   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode (W#16#...)  | Erläuterung   |
|--|---|
| 0000   | Kein Fehler   |
| 8091   | Der Quellbereich liegt in einem nicht ablaufrelevanten Datenbaustein. |
| allgemeine Fehlerinformation   | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"                    |
| *Die Fehlercodes im Programmmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2304)

**FILL: Bereich befüllen****Beschreibung**

Mit der Anweisung "Bereich befüllen" füllen Sie einen Speicherbereich (Zielbereich) mit dem Inhalt eines anderen Speicherbereichs (Quellbereich) auf. Die Anweisung "Bereich befüllen" kopiert den Inhalt des Quellbereichs in den Zielbereich so oft, bis der Zielbereich vollständig beschrieben ist. Der Kopiervorgang findet in Richtung aufsteigender Adressen statt.

Den Quell- und Zielbereich definieren Sie durch VARIANT.

---

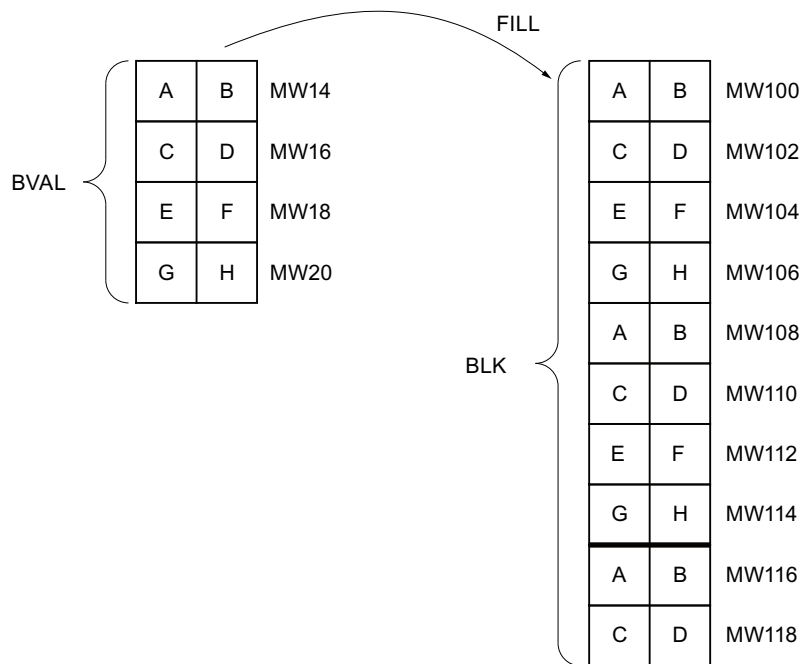
**Hinweis**

Die Variablen der Anweisung können Sie nur in Datenbausteinen verwenden, bei denen das Attribut "optimierter Bausteinzugriff" nicht gesetzt ist. Wenn die Variable mit der Remanenzeinstellung "Im IDB setzen" deklariert wurde, können Sie die Variablen auch "mit optimiertem Zugriff" verwenden.

Für Bausteine mit dem Attribut "optimierter Bausteinzugriff" können Sie die Anweisung "FILL\_BLK: Bereich befüllen" verwenden.

---

Das folgende Bild zeigt das Prinzip des Kopiervorgangs:



Beispiel: Der Inhalt des Bereichs MW100 bis MW118 soll mit dem Inhalt aus den Merkerworten MW14 bis MW20 vorbesetzt werden.

### Konsistenz der Quelldaten und der Zieldaten

Beachten Sie, dass während der Bearbeitung der Anweisung "Bereich befüllen" die Quelldaten unverändert bleiben, da sonst die Konsistenz der Zieldaten nicht gewährleistet ist.

### Speicherbereiche

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche kopieren:

- Bereiche eines Datenbausteins
- Merker
- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge

Mit der Anweisung "Bereich befüllen" können Sie die folgenden Speicherbereiche nicht beschreiben:

- Anweisungen, Systemdatenbausteine
- Zähler
- Zeiten
- Speicherbereiche des Peripheriebereichs.



## Allgemeine Regeln beim Kopieren

Quell- und Zielbereich dürfen sich nicht überlappen. Wenn der vorzubelegende Zielbereich kein ganzzahliges Vielfaches der Länge des Eingangsparameters BVAL ist, wird der Zielbereich trotzdem bis zum letzten Byte beschrieben.

Wenn der vorzubelegende Zielbereich kleiner als der Quellbereich ist, dann werden nur so viele Daten kopiert, wie der Zielbereich aufnehmen kann.

Falls der real vorhandene Ziel- oder Quellbereich kleiner als die Größe des parametrisierten Speicherbereichs für Quell- oder Zielbereich (Parameter BVAL, BLK) ist, werden keine Daten übertragen.

Ist der ANY-Pointer (Quelle oder Ziel) vom Datentyp BOOL, muss er absolut adressiert und die angegebene Länge des Bereichs durch 8 teilbar sein, da sonst die Anweisung nicht ausgeführt wird.

Falls der Zielbereich vom Datentyp STRING ist, beschreibt die Anweisung die gesamte Zeichenkette einschließlich der Verwaltungsinformation.

## Syntax

Für die Anweisung "Bereich befüllen" wird die folgende Syntax verwendet:

```
SCL
FILL(BVAL := <Operand>,
      BLK => <Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bereich befüllen":

| Parameter               | Deklaration | Datentyp | Beschreibung   |
|-------------------------|-------------|----------|--|
| BVAL                    | Input       | VARIANT  | Angabe des Speicherbereichs (Quellbereich), mit dessen Inhalt der Zielbereich am Parameter BLK befüllt wird. |
| BLK                     | Output      | VARIANT  | Angabe des Speicherbereichs, der mit dem Inhalt des Quellbereichs befüllt wird.                              |
| Funktionswert (RET_VAL) |             | INT      | Fehlerinformationen  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Parameter BVAL

Wenn Sie als Eingangsparameter eine Struktur übergeben, dann müssen Sie berücksichtigen, dass die Länge einer Struktur immer auf eine gerade Anzahl von Bytes ausgerichtet wird. Wenn Sie eine Struktur mit einer ungeraden Anzahl Bytes deklarieren, benötigt die Struktur ein Byte zusätzlichen Speicherplatz.

### Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode (W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| allgemeine Fehlerinformation   | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen" |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2304)

### Umwandler

#### CONVERT: Wert konvertieren

#### Beschreibung

Mit der Anweisung "Wert konvertieren" programmieren Sie explizite Konvertierungen. Die zu konvertierenden Datentypen bestimmen Sie in einem Dialog, der automatisch beim Einfügen der Anweisung geöffnet wird. Bei der Ausführung liest die Anweisung den Quellwert und konvertiert diesen in den angegebenen Zielwert.

Informationen zu den möglichen Konvertierungen finden Sie im Abschnitt "Explizite Konvertierung" unter "Siehe auch".

---

#### Hinweis

Für CPU S7-1500 gilt: Es können die Datentypen DWORD und LWORD ausgewählt werden, wenn als IN-Datentyp REAL oder LREAL ausgewählt wurde.

---

## Syntax

Für die Anweisung "Wert konvertieren" wird die folgende Syntax verwendet:

```

scl
<Zielwert> := <Konvertierungsfunktion> (<Quellwert>)
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter                | Deklaration         | Datentyp  |   | Beschreibung   |
|--------------------------|---------------------|---|---|--|
|                          |                     | S7-1200   | S7-1500   |  |
| <Quellwert>              | Input,<br>Konstante | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE,<br>TOD, DTL,<br>Zeichenfolgen | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>DATE, TOD,<br>LTOD, DTL,<br>Zeichenfolgen | Wert, der<br>konvertiert wird.                                       |
| <Konvertierungsfunktion> | -                   | -   | -   | Funktion, die die<br>zu<br>konvertierenden<br>Datentypen<br>vorgibt. |
| <Zielwert>               | Output              | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, DATE,<br>TOD, DTL,<br>Zeichenfolgen | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, LTIME,<br>DATE, TOD,<br>LTOD, DTL,<br>Zeichenfolgen | Ergebnis der<br>Konvertierung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

scl
"Tag_INT" := REAL_TO_INT("Tag_REAL");
    
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand  | Datentyp | Wert  |
|----------|----------|-------|
| Tag_REAL | REAL     | 20.56 |
| Tag_INT  | INT      | 21    |

Bei der Konvertierung wird der Wert des Operanden "Tag\_REAL" zur nächsten Ganzzahl gerundet und im Operanden "Tag\_INT" abgelegt.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**ROUND: Zahl runden**

**Beschreibung**

Mit der Anweisung "Zahl runden" runden Sie den Wert am Eingang IN zur nächsten Ganzzahl. Die Anweisung interpretiert den Wert am Eingang IN als Gleitpunktzahl und wandelt sie in eine Ganzzahl oder eine Gleitpunktzahl um. Liegt der Eingangswert zwischen einer geraden und einer ungeraden Zahl, wird die gerade Zahl gewählt.

**Syntax**

Für die Anweisung "Zahl runden" wird die folgende Syntax verwendet:

```
SCL
ROUND (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp                        | Beschreibung                     |
|---------------|-------------|---------------------------------|----------------------------------|
| <Ausdruck>    | Input       | Gleitpunktzahlen                | Eingangswert, der gerundet wird. |
| Funktionswert |             | Ganzzahlen,<br>Gleitpunktzahlen | Ergebnis der Rundung             |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := ROUND("Tag_Value");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert       |             |
|------------|------------|-------------|
| Tag_Value  | 1.50000000 | -1.50000000 |
| Tag_Result | 2          | -2          |

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## CEIL: Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen

### Beschreibung

Mit der Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" runden Sie einen Wert zur nächst höheren Ganzzahl. Die Anweisung interpretiert den Eingangswert als Gleitpunktzahl und wandelt diese in die nächst höhere ganze Zahl um. Der Funktionswert kann dabei größer oder gleich dem Eingangswert sein.

### Syntax

Für die Anweisung "Aus Gleitpunktzahl nächst höhere Ganzzahl erzeugen" wird die folgende Syntax verwendet:

| SCL                          |
|------------------------------|
| CEIL (<Ausdruck>)            |
| CEIL_<Datentyp> (<Ausdruck>) |

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp   | Beschreibung  |
|---------------|-------------|--|---|
| <Ausdruck>    | Input       | Gleitpunktzahlen                                     | Eingangswert  |
| _<Datentyp>   |             | Ganzzahlen,<br>Gleitpunktzahlen<br>Vorbelegung: DINT | Datentyp des Funktionswerts: <ul style="list-style-type: none"> <li>• Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe.</li> <li>• Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.</li> </ul> |
| Funktionswert |             | Ganzzahlen,<br>Gleitpunktzahlen                      | Aufgerundeter Eingangswert  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := CEIL("Tag_Value");
"Tag_Result2" := CEIL_REAL("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert |      |
|-------------|------|------|
| Tag_Value   | 0.5  | -0.5 |
| Tag_Result1 | 1    | 0    |
| Tag_Result2 | 1.0  | 0.0  |

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**FLOOR: Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen**

**Beschreibung**

Mit der Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" runden Sie einen Wert zur nächst niedrigeren Ganzzahl. Die Anweisung interpretiert den Eingangswert als Gleitpunktzahl und wandelt diese in die nächst niedrigere ganze Zahl um. Der Funktionswert kann dabei kleiner oder gleich dem Eingangswert sein.

**Syntax**

Für die Anweisung "Aus Gleitpunktzahl nächst niedere Ganzzahl erzeugen" wird die folgende Syntax verwendet:

```
SCL
FLOOR(<Ausdruck>)
FLOOR_<Datentyp>(<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp   | Beschreibung  |
|---------------|-------------|--|---|
| <Ausdruck>    | Input       | Gleitpunktzahlen                                     | Eingangswert  |
| _<Datentyp>   |             | Ganzzahlen,<br>Gleitpunktzahlen<br>Vorbelegung: DINT | Datentyp des Funktionswerts: <ul style="list-style-type: none"> <li>• Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe.</li> <li>• Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.</li> </ul> |
| Funktionswert |             | Ganzzahlen,<br>Gleitpunktzahlen                      | Abgerundeter Eingangswert   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result1" := FLOOR("Tag_Value");
"Tag_Result2" := FLOOR_REAL("Tag_Value");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert |      |
|-------------|------|------|
| Tag_Value   | 0.5  | -0.5 |
| Tag_Result1 | 0    | -1   |
| Tag_Result2 | 0.0  | -1.0 |

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## TRUNC: Ganzzahl erzeugen

### Beschreibung

Mit der Anweisung "Ganzzahl erzeugen" erzeugen Sie aus dem Eingangswert eine Ganzzahl ohne Rundung. Die Anweisung selektiert nur den ganzzahligen Anteil des Eingangswerts und gibt diesen ohne die Nachkommastellen als Funktionswert zurück.

### Syntax

Für die Anweisung "Ganzzahl erzeugen" wird die folgende Syntax verwendet:

**SCL**

```
TRUNC (<Ausdruck>)
TRUNC_<Datentyp> (<Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp   | Beschreibung  |
|---------------|-------------|--|---|
| <Ausdruck>    | Input       | Gleitpunktzahlen                                     | Eingangswert  |
| _<Datentyp>   |             | Ganzzahlen,<br>Gleitpunktzahlen<br>Vorbelegung: DINT | Datentyp des Funktionswerts: <ul style="list-style-type: none"> <li>• Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe.</li> <li>• Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.</li> </ul> |
| Funktionswert |             | Ganzzahlen,<br>Gleitpunktzahlen                      | Ganzzahliger Anteil des Eingangswerts   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

**SCL**

```
"Tag_Result1" := TRUNC("Tag_Value1");
"Tag_Result2" := TRUNC("Tag_Value2"+"Tag_Value3");
"Tag_Result3" := TRUNC_SINT("Tag_Value4");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:



| Operand     | Wert |
|-------------|------|
| Tag_Value1  | -1.5 |
| Tag_Result1 | -1   |
| Tag_Value2  | 2.1  |
| Tag_Value3  | 3.2  |
| Tag_Result2 | 5    |
| Tag_Result3 | 2    |
| Tag_Value4  | 2.4  |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

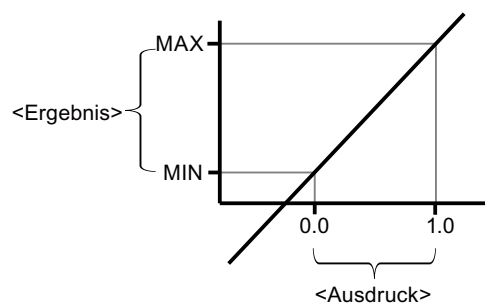
SCL-Anweisungen bearbeiten (Seite 1398)

## SCALE\_X: Skalieren

### Beschreibung

Mit der Anweisung "Skalieren" skalieren Sie eine Gleitpunktzahl, indem Sie diese auf einem bestimmten Wertebereich abbilden. Den Wertebereich legen Sie durch die Parameter MIN und MAX fest. Das Ergebnis der Skalierung ist eine Ganzzahl.

Das folgende Bild zeigt beispielhaft, wie Werte skaliert werden können:



Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

$$OUT = [VALUE * (MAX - MIN)] + MIN$$

### Syntax

Für die Anweisung "Skalieren" wird die folgende Syntax verwendet:

```

SCL
SCALE_X(MIN := <Operand>,
        VALUE := <Operand>,
        MAX := <Operand>)
SCALE_X_<Datentyp>(MIN := <Operand>,
                   VALUE := <Operand>,
                   MAX := <Operand>)
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp   | Beschreibung  |
|---------------|-------------|--|---|
| MIN           | Input       | Ganzzahlen, Gleitpunktzahlen                     | Untere Grenze des Wertebereichs   |
| VALUE         | Input       | Gleitpunktzahlen                                 | Wert, der skaliert wird.<br>Bei Angabe einer Konstante müssen Sie diese deklarieren.  |
| MAX           | Input       | Ganzzahlen, Gleitpunktzahlen                     | Obere Grenze des Wertebereichs  |
| _<Datentyp>   |             | Ganzzahlen, Gleitpunktzahlen<br>Vorbelegung: INT | Datentyp des Funktionswerts: <ul style="list-style-type: none"> <li>• Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe.</li> <li>• Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.</li> </ul> |
| Funktionswert |             | Ganzzahlen, Gleitpunktzahlen                     | Ergebnis der Skalierung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result1" := SCALE_X(MIN := "Tag_Value1",
    
```

**SCL**

```

VALUE := "Tag_Real",
MAX := "Tag_Value2");
"Tag_Result2" := SCALE_X_REAL(MIN := "Tag_Value1",
VALUE := "Tag_Real",
MAX := "Tag_Value2");

```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand     | Wert |
|-------------|------|
| Tag_Real    | 0.5  |
| Tag_Value1  | 10   |
| Tag_Value2  | 30   |
| Tag_Result1 | 20   |
| Tag_Result2 | 20.0 |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

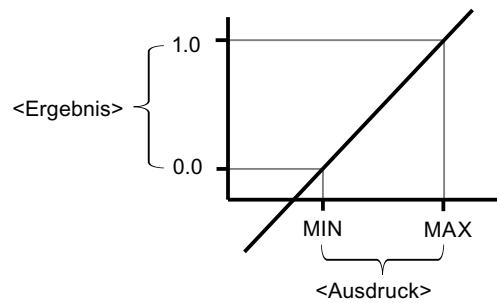
SCL-Anweisungen bearbeiten (Seite 1398)

Konstanten deklarieren (Seite 1202)

**NORM\_X: Normieren****Beschreibung**

Mit der Anweisung "Normieren" normieren Sie den Wert der Variablen am Eingang VALUE, indem Sie diesen auf einer linearen Skala abbilden. Mit den Parametern MIN und MAX definieren Sie die Grenzen eines Wertebereichs, der auf die Skala gespiegelt wird. Abhängig von der Lage des zu normierenden Werts in diesem Wertebereich wird das Ergebnis berechnet und am Ausgang OUT als Gleitpunktzahl abgelegt. Wenn der zu normierende Wert gleich dem Wert am Eingang MIN ist, liefert die Anweisung den Wert "0.0" als Ergebnis zurück. Wenn der zu normierende Wert gleich dem Wert am Eingang MAX ist, liefert die Anweisung den Wert "1.0" als Ergebnis zurück.

Das folgende Bild zeigt beispielhaft, wie Werte normiert werden können:



Die Anweisung "Normieren" arbeitet mit der folgenden Gleichung:

$$OUT = (VALUE - MIN) / (MAX - MIN)$$

### Syntax

Für die Anweisung "Normieren" wird die folgende Syntax verwendet:

```

SCL
NORM_X(MIN := <Operand>,
        VALUE := <Operand>,
        MAX := <Operand>)
NORM_X_<Datentyp>(MIN := <Operand>,
                  VALUE := <Operand>,
                  MAX := <Operand>)
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter           | Deklaration | Datentyp                        | Beschreibung                    |
|---------------------|-------------|---------------------------------|---------------------------------|
| MIN <sup>1)</sup>   | Input       | Ganzzahlen,<br>Gleitpunktzahlen | Untere Grenze des Wertebereichs |
| VALUE <sup>1)</sup> | Input       | Ganzzahlen,<br>Gleitpunktzahlen | Wert, der normiert wird.        |
| MAX <sup>1)</sup>   | Input       | Ganzzahlen,<br>Gleitpunktzahlen | Obere Grenze des Wertebereichs  |

| Parameter   | Deklaration | Datentyp                              | Beschreibung  |
|---|-------------|---------------------------------------|---|
| _<Datentyp>   |             | Gleitpunktzahlen<br>Vorbelegung: REAL | Datentyp des Funktionswerts: <ul style="list-style-type: none"> <li>• Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe.</li> <li>• Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.</li> </ul> |
| Funktionswert   |             | Gleitpunktzahlen                      | Ergebnis der Normierung   |
| <sup>1)</sup> Bei Verwendung von Konstanten an diesen drei Parametern genügt es, wenn Sie eine davon deklarieren. |             |                                       |   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Weitere Informationen zur Deklaration von Konstanten finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result1" := NORM_X(MIN := "Tag_Value1",
                        VALUE := "Tag_InputValue",
                        MAX := "Tag_Value2");
"Tag_Result2" := NORM_X_LREAL(MIN := "Tag_Value1",
                              VALUE := "Tag_InputValue",
                              MAX := "Tag_Value2");
    
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand        | Wert |
|----------------|------|
| Tag_InputValue | 20   |
| Tag_Value1     | 10   |
| Tag_Value2     | 30   |
| Tag_Result1    | 0.5  |
| Tag_Result2    | 0.5  |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- Konstanten deklarieren (Seite 1202)

**VARIANT**

**VARIANT\_TO\_DB\_ANY: VARIANT in DB\_ANY konvertieren**

**Beschreibung**

Mit der Anweisung "VARIANT in DB\_ANY konvertieren" lesen Sie die Nummer des Datenbausteins aus einem VARIANT und konvertieren sie in den Datentyp DB\_ANY.

**Voraussetzungen**

Wenn die Voraussetzungen erfüllt sind, wird die Anweisung ausgeführt. Wenn die Voraussetzungen nicht erfüllt sind, dann wird "0" als Datenbausteinnummer ausgegeben.

| Die Ausgangsvariable... | referenziert...  | Konvertierungsmöglichkeiten   |
|-------------------------|--|---|
| VARIANT                 | ...einen Datenbaustein, der ein Instanz-Datenbaustein eines UDTs oder eines SDTs ist | Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist möglich.  |
| VARIANT                 | ...einen Array-DB  | Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist möglich.  |
| VARIANT                 | ...ein Objekt mit einem elementaren Datentyp   | Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist nicht möglich, da ein Datenbaustein keinesfalls aus nur einem elementaren Datentyp besteht. |
| VARIANT                 | ...eine Struktur innerhalb eines Datenbausteins                                      | Die Konvertierung der Ausgangsvariablen in eine Datenbausteinnummer ist nicht möglich, da es sich nur um einen Teil innerhalb des Datenbausteins handelt.           |

## Syntax

Für die Anweisung "VARIANT in DB\_ANY konvertieren" wird die folgende Syntax verwendet:

```
SCL
VARIANT_TO_DB_ANY(IN := <Operand>,
                  ERR => <Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VARIANT in DB\_ANY konvertieren":

| Parameter     | Deklaration | Datentyp | Speicherbereich | Beschreibung                  |
|---------------|-------------|----------|-----------------|-------------------------------|
| IN            | Input       | VARIANT  | E, A, M, D, L   | Variable, die ausgelesen wird |
| ERR           | Output      | INT      | E, A, M, D, L   | Fehlerinformationen           |
| Funktionswert |             | DB_ANY   | E, A, M, D, L   | Ergebnis:<br>Nummer des DBs   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter ERR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR:

| Fehlercode*<br>(W#16#...)  | Erläuterung                               |
|--|---|
| 0000   | Kein Fehler                               |
| 252C   | Der Datentyp VARIANT liefert den Wert "0" |
| 8130   | Die Nummer des Datenbaustein ist "0"      |
| 8131   | Der Datenbaustein existiert nicht         |
| 8132   | Der Datenbaustein ist zu kurz             |
| 8134   | Der Datenbaustein ist schreibgeschützt    |
| 8150   | Der Datentyp VARIANT liefert den Wert "0" |
| 8154   | Der Datentyp ist nicht korrekt deklariert |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## DB\_ANY\_TO\_VARIANT: DB\_ANY in VARIANT konvertieren

### Beschreibung

Mit der Anweisung "DB\_ANY in VARIANT konvertieren" lesen Sie die Nummer des Datenbausteins aus einem DB\_ANY und konvertieren sie in den Datentyp VARIANT.

### Voraussetzungen

Wenn die Voraussetzungen erfüllt sind, wird die Anweisung ausgeführt. Wenn die Voraussetzungen nicht erfüllt sind oder der Datenbaustein nicht existiert, dann wird Wert "0" ausgegeben. Alle weiteren Zugriffe auf diesen Wert "0" schlagen fehl.

| Die Ausgangsvariable... | referenziert...  | Konvertierungsmöglichkeiten  |
|-------------------------|--|--|
| DB_ANY                  | ...einen Datenbaustein, der ein Instanz-Datenbaustein eines UDTs oder eines SDTs ist                               | Die Konvertierung der Ausgangsvariablen in den Datentyp VARIANT ist möglich.       |
| DB_ANY                  | ...einen Datenbaustein, der ein Array-DB ist   | Die Konvertierung der Ausgangsvariablen in den Datentyp VARIANT ist möglich.       |
| DB_ANY                  | ...einen Datenbaustein, der ein Instanz-Datenbaustein eines Funktionsbausteins oder ein globaler Datenbaustein ist | Die Konvertierung der Ausgangsvariablen in den Datentyp VARIANT ist nicht möglich. |

### Syntax

Für die Anweisung "DB\_ANY in VARIANT konvertieren" wird die folgende Syntax verwendet:

#### SCL

```
DB_ANY_TO_VARIANT(IN := <Operand>,
ERR => <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DB\_ANY in VARIANT konvertieren":

| Parameter     | Deklaration | Datentyp | Speicherbereich | Beschreibung                  |
|---------------|-------------|----------|-----------------|-------------------------------|
| IN            | Input       | DB_ANY   | E, A, M, D, L   | Variable, die ausgelesen wird |
| ERR           | Output      | INT      | E, A, M, D, L   | Fehlerinformationen           |
| Funktionswert |             | VARIANT  | L               | Ergebnis                      |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".



## Parameter ERR

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR:

| Fehlercode*<br>(W#16#...)  | Erläuterung                               |
|--|---|
| 0000   | Kein Fehler                               |
| 8130   | Die Nummer des Datenbaustein ist "0"      |
| 8131   | Der Datenbaustein existiert nicht         |
| 8132   | Der Datenbaustein ist zu kurz             |
| 8134   | Der Datenbaustein ist schreibgeschützt    |
| 8154   | Der Datentyp ist nicht korrekt deklariert |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## Weitere

## SCALE: Skalieren

### Beschreibung

Mit der Anweisung "Skalieren" wandeln Sie die Ganzzahl am Parameter IN in eine Gleitpunktzahl um, die in physikalischen Einheiten zwischen einem unteren und einem oberen Grenzwert skaliert wird. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert skaliert wird, legen Sie durch die Parameter LO\_LIM und HI\_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Skalieren" arbeitet mit der folgenden Gleichung:

$$\text{OUT} = [((\text{FLOAT}(\text{IN}) - \text{K1}) / (\text{K2} - \text{K1})) * (\text{HI\_LIM} - \text{LO\_LIM})) + \text{LO\_LIM}]$$

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert der Konstante "K2" ist, wird das Ergebnis der Anweisung auf den Wert des oberen Grenzwerts (HI\_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante "K1" ist, wird das Ergebnis der Anweisung auf den Wert des unteren Grenzwerts (LO\_LIM) gesetzt und ein Fehler ausgegeben.

Wenn der angegebene untere Grenzwert größer als der obere Grenzwert ist (LO\_LIM > HI\_LIM), wird das Ergebnis umgekehrt proportional zum Eingabewert skaliert.

### Syntax

Für die Anweisung "Skalieren" wird die folgende Syntax verwendet:

```

SCI
SCALE (IN := <Ausdruck>,
      HI_LIM := <Operand>,
      LO_LIM := <Operand>,
      BIPOLAR := <Operand>,
      OUT => <Operand>)
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter               | Deklaration | Datentyp | Beschreibung   |
|-------------------------|-------------|----------|--|
| IN                      | Input       | INT      | Eingabewert, der skaliert wird.  |
| HI_LIM                  | Input       | REAL     | Oberer Grenzwert   |
| LO_LIM                  | Input       | REAL     | Unterer Grenzwert  |
| BIPOLAR                 | Input       | BOOL     | Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen:<br>1: Bipolar<br>0: Unipolar |
| OUT                     | Output      | REAL     | Ergebnis der Anweisung   |
| Funktionswert (RET_VAL) |             | WORD     | Fehlerinformationen  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode (W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 0008   | Der Wert des Parameters IN ist größer als 27 648 oder kleiner als 0 (unipolar) bzw. -27 648 (bipolar). |
| allgemeine Fehlerinformation   | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
"Tag_ErrorCode" := SCALE(IN := "Tag_InputValue",
    HI_LIM := "Tag_HighLimit"
    LO_LIM := "Tag_LowLimit"
    BIPOLAR := "Tag_Bipolar",
    OUT => "Tag_Result");
```

Die Fehlerinformation wird am Operanden "Tag\_ErrorCode" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand        | Wert        |
|-----------|----------------|-------------|
| IN        | Tag_InputValue | 22          |
| HI_LIM    | Tag_HighLimit  | 100.0       |
| LO_LIM    | Tag_LowLimit   | 0.0         |
| BIPOLAR   | Tag_Bipolar    | 1           |
| OUT       | Tag_Result     | 50.03978588 |
| RET_VAL   | Tag_ErrorCode  | W#16#0000   |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

Ausdrücke (Seite 1364)

Operatoren und Operatorenrangfolge (Seite 1370)

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2304)

## UNSCALE: Deskalieren

### Beschreibung

Mit der Anweisung "Deskalieren" deskalieren Sie die Gleitpunktzahl am Parameter IN in physikalische Einheiten zwischen einem unteren und einem oberen Grenzwert und wandeln sie in eine Ganzzahl um. Den unteren und oberen Grenzwert des Wertebereichs, auf dem der Eingabewert deskaliert wird, legen Sie durch die Parameter LO\_LIM und HI\_LIM fest. Das Ergebnis der Anweisung wird am Parameter OUT ausgegeben.

Die Anweisung "Deskalieren" arbeitet mit der folgenden Gleichung:

$$OUT = [((IN-LO\_LIM)/(HI\_LIM-LO\_LIM)) * (K2-K1)] + K1$$

Die Werte der Konstanten "K1" und "K2" werden durch den Signalzustand am Parameter BIPOLAR bestimmt. Am Parameter BIPOLAR können die folgenden Signalzustände anstehen:

- Signalzustand "1": Es wird angenommen, dass der Wert am Parameter IN bipolar ist und in einem Wertebereich zwischen -27648 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "-27648,0" und die Konstante "K2" den Wert "+27648,0".
- Signalzustand "0": Es wird angenommen, dass der Wert am Parameter IN unipolar ist und in einem Wertebereich zwischen 0 und 27648 liegt. In diesem Fall hat die Konstante "K1" den Wert "0,0" und die Konstante "K2" den Wert "+27648,0".

Wenn der Wert am Parameter IN größer als der Wert des oberen Grenzwert (HI\_LIM) ist, wird das Ergebnis der Anweisung auf den Wert der Konstante "K2" gesetzt und ein Fehler ausgegeben.

Wenn der Wert am Parameter IN kleiner als der Wert der Konstante des unteren Grenzwert (LO\_LIM) ist, wird das Ergebnis der Anweisung auf den Wert der Konstante "K1" gesetzt und ein Fehler ausgegeben.

### Syntax

Für die Anweisung "Deskalieren" wird die folgende Syntax verwendet:

```

SCL
UNSCALE (IN := <Ausdruck>,
          HI_LIM := <Operand>,
          LO_LIM := <Operand>,
          BIPOLAR := <Operand>,
          OUT => <Operand>)
    
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp | Beschreibung   |
|-----------|-------------|----------|--|
| IN        | Input       | REAL     | Eingabewert, der in einen ganzzahligen Wert deskaliert wird. |
| HI_LIM    | Input       | REAL     | Oberer Grenzwert   |
| LO_LIM    | Input       | REAL     | Unterer Grenzwert  |

| Parameter               | Deklaration | Datentyp | Beschreibung   |
|-------------------------|-------------|----------|--|
| BIPOLAR                 | Input       | BOOL     | Gibt an, ob der Wert am Parameter IN als bipolar oder unipolar interpretiert wird. Der Parameter kann die folgenden Werte annehmen:<br>1: Bipolar<br>0: Unipolar |
| OUT                     | Output      | INT      | Ergebnis der Anweisung   |
| Funktionswert (RET_VAL) |             | WORD     | Fehlerinformationen  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode (W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 0008   | Der Wert des Parameters IN ist größer als der Wert des oberen Grenzwerts (HI_LIM) oder kleiner als der Wert des unteren Grenzwerts (LO_LIM). |
| allgemeine Fehlerinformation   | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_ErrorCode" := UNSCALE(IN := "Tag_InputValue",
                           HI_LIM := "Tag_HighLimit"
                           LO_LIM := "Tag_LowLimit"
                           BIPOLAR := "Tag_Bipolar",
                           OUT => "Tag_Result");

```

Die Fehlerinformation wird am Operanden "Tag\_ErrorCode" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand        | Wert        |
|-----------|----------------|-------------|
| IN        | Tag_InputValue | 50.03978588 |
| HI_LIM    | Tag_HighLimit  | 100.0       |

| Parameter | Operand       | Wert      |
|-----------|---------------|-----------|
| LO_LIM    | Tag_LowLimit  | 0.0       |
| BIPOLAR   | Tag_Bipolar   | 1         |
| OUT       | Tag_Result    | 22        |
| RET_VAL   | Tag_ErrorCode | W#16#0000 |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- Ausdrücke (Seite 1364)
- Operatoren und Operatorenrangfolge (Seite 1370)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2304)

**Programmsteuerung**

**IF: Bedingt ausführen**

**Beschreibung**

Mit der Anweisung "Bedingt ausführen" verzweigen Sie den Programmfluss abhängig von einer Bedingung. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

**Syntax**

Je nach Verzweigungsart können Sie die folgenden Formen der Anweisung programmieren:

- Verzweigung durch IF:

```
SCL
IF <Bedingung> THEN <Anweisungen>
END_IF;
```

Wenn die Bedingung erfüllt ist, werden die Anweisungen ausgeführt, die nach THEN programmiert sind. Wenn die Bedingung nicht erfüllt ist, wird die Programmbearbeitung mit der nächsten Anweisung nach END\_IF fortgesetzt.

- Verzweigung durch IF und ELSE:

```
SCL
IF <Bedingung> THEN <Anweisungen1>
ELSE <Anweisungen0>
END_IF;
```

Wenn die Bedingung erfüllt ist, werden die Anweisungen ausgeführt, die nach THEN programmiert sind. Wenn die Bedingung nicht erfüllt ist, werden die Anweisungen ausgeführt, die nach ELSE programmiert sind. Danach wird die Programmbearbeitung mit der nächsten Anweisung nach END\_IF fortgesetzt.

- Verzweigung durch IF, ELSIF und ELSE:

```
SCL
IF <Bedingung1> THEN <Anweisungen1>
ELSIF <Bedingung2> THEN <Anweisung2>
ELSE <Anweisungen0>
END_IF;
```

Wenn die erste Bedingung (<Bedingung1>) erfüllt ist, werden die Anweisungen (<Anweisungen1>) nach THEN ausgeführt. Nach der Bearbeitung der Anweisungen wird die Programmbearbeitung nach END\_IF fortgesetzt.

Wenn die erste Bedingung nicht erfüllt ist, wird die zweite Bedingung (<Bedingung2>) geprüft. Wenn die zweite Bedingung (<Bedingung2>) erfüllt ist, werden die Anweisungen <Anweisungen2> nach THEN ausgeführt. Nach der Bearbeitung der Anweisungen wird die Programmbearbeitung nach END\_IF fortgesetzt.

Wenn keine der Bedingungen erfüllt ist, werden die Anweisungen (<Anweisungen0>) nach ELSE ausgeführt und danach wird die Programmbearbeitung nach END\_IF fortgesetzt.

Sie können beliebig viele Kombinationen von ELSIF und THEN innerhalb der IF-Anweisung schachteln. Die Programmierung eines ELSE-Zweigs ist optional.

Die Syntax der IF-Anweisung besteht aus den folgenden Teilen:

| Parameter     | Datentyp | Beschreibung   |
|---------------|----------|--|
| <Bedingung>   | BOOL     | Ausdruck, der ausgewertet wird.  |
| <Anweisungen> | -        | Anweisungen, die bei erfüllter Bedingung ausgeführt werden. Eine Ausnahme sind die Anweisungen, die nach ELSE programmiert sind. Diese werden ausgeführt, wenn keine Bedingung innerhalb der Programmschleife erfüllt ist. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
IF "Tag_1" = 1
```

```

SCL
THEN "Tag_Value" := 10;
ELSIF "Tag_2" = 1
THEN "Tag_Value" := 20;
ELSIF "Tag_3" = 1
THEN "Tag_Value" := 30;
ELSE "Tag_Value" := 0;
END_IF;
    
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand   | Wert |    |    |   |
|-----------|------|----|----|---|
|           | 1    | 0  | 0  | 0 |
| Tag_1     | 1    | 0  | 0  | 0 |
| Tag_2     | 0    | 1  | 0  | 0 |
| Tag_3     | 0    | 0  | 1  | 0 |
| Tag_Value | 10   | 20 | 30 | 0 |

**Siehe auch**

- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- Übersicht über die gültigen Datentypen (Seite 1087)

**CASE: Mehrfach verzweigen**

**Beschreibung**

Mit der Anweisung "Mehrfach verzweigen" bearbeiten Sie abhängig vom Wert eines numerischen Ausdrucks eine von mehreren Anweisungsfolgen.

Der Wert des Ausdrucks muss eine Ganzzahl sein. Bei der Ausführung der Anweisung wird der Wert des Ausdrucks mit den Werten mehrerer Konstanten verglichen. Wenn der Wert des Ausdrucks mit dem Wert einer Konstante übereinstimmt, werden die Anweisungen ausgeführt, die direkt nach dieser Konstante programmiert sind. Die Konstanten können dabei die folgenden Werte annehmen:

- Eine Ganzzahl (z. B. 5)
- Ein Bereich aus Ganzzahlen (z. B. 15..20)
- Eine Aufzählung aus Ganzzahlen und Bereichen (z. B. 10,11,15..20)



## Syntax

Für die Anweisung "Mehrfach verzweigen" wird die folgende Syntax verwendet:

### SCL

```
CASE <Ausdruck> OF
<Konstante1>: <Anweisungen1>
<Konstante2>: <Anweisungen2>
<KonstanteX>: <AnweisungenX>; // X >= 3
ELSE <Anweisungen0>
END_CASE;
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter   | Datentyp   | Beschreibung  |
|-------------|------------|---|
| <Ausdruck>  | Ganzzahlen | Wert, der mit den programmierten Konstantenwerten verglichen wird.  |
| <Konstante> | Ganzzahlen | Konstantenwerte, die die Bedingung zur Ausführung einer Anweisungsfolge bilden. Die Konstanten können die folgenden Werte annehmen: <ul style="list-style-type: none"> <li>• Eine Ganzzahl (z. B. 5)</li> <li>• Ein Bereich aus Ganzzahlen (z. B. 15..20)</li> <li>• Eine Aufzählung aus Ganzzahlen und Bereichen (z. B. 10,11,15..20)</li> </ul> |
| <Anweisung> | -          | Beliebige Anweisungen, die ausgeführt werden, wenn der Wert des Ausdrucks mit dem Wert einer Konstanten übereinstimmt. Eine Ausnahme sind die Anweisungen, die nach ELSE programmiert sind. Diese werden ausgeführt, wenn die Werte nicht übereinstimmen.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn der Wert des Ausdrucks mit dem Wert der ersten Konstante (<Konstante1>) übereinstimmt, werden die Anweisungen (<Anweisungen1>) ausgeführt, die direkt nach der ersten Konstante programmiert sind. Danach wird die Programmbearbeitung nach END\_CASE fortgesetzt.

Wenn der Wert des Ausdrucks mit dem Wert der ersten Konstante (<Konstante1>) nicht übereinstimmt, wird dieser mit dem Wert der Konstante verglichen, die als nächste programmiert ist. Auf diese Weise wird die CASE-Anweisung ausgeführt, bis eine Übereinstimmung der Werte vorliegt. Wenn der Wert des Ausdrucks keinem der programmierten Konstantenwerte entspricht, werden die Anweisungen (<Anweisungen0>) ausgeführt, die nach ELSE programmiert sind. ELSE ist ein optionaler Teil der Syntax und kann entfallen.

Die CASE-Anweisung kann auch geschachtelt werden, indem ein Anweisungsblock durch CASE ersetzt wird. END\_CASE bildet den Abschluss der CASE-Anweisung.

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
CASE "Tag_Value" OF
  0 :
    "Tag_1" := 1;
  1,3,5 :
    "Tag_2" :=1;
  6..10 :
    "Tag_3" := 1;
  16,17,20..25 :
    "Tag_4" := 1;
ELSE "Tag_5" := 1;
END_CASE;
    
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand   | Werte |         |                |                               |   |
|-----------|-------|---------|----------------|-------------------------------|---|
|           | 0     | 1, 3, 5 | 6, 7, 8, 9, 10 | 16,17, 20, 21, 22, 23, 24, 25 | 2 |
| Tag_Value | 0     | 1, 3, 5 | 6, 7, 8, 9, 10 | 16,17, 20, 21, 22, 23, 24, 25 | 2 |
| Tag_1     | 1     | -       | -              | -                             | - |
| Tag_2     | -     | 1       | -              | -                             | - |
| Tag_3     | -     | -       | 1              | -                             | - |
| Tag_4     | -     | -       | -              | 1                             | - |
| Tag_5     | -     | -       | -              | -                             | 1 |

1: Der Operand wird auf den Signalzustand "1" gesetzt.  
 -: Der Signalzustand des Operanden bleibt unverändert.

**Siehe auch**

- CONTINUE: Schleifenbedingung erneut prüfen (Seite 2290)
- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- EXIT: Schleife sofort verlassen (Seite 2291)

**FOR: In Zählschleife ausführen**

**Beschreibung**

Mit der Anweisung "In Zählschleife ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Laufvariable innerhalb eines angegebenen Wertebereichs liegt.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung "Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

## Grenzwerte für FOR-Anweisungen

Um "sichere" FOR-Anweisungen, die nicht endlos laufen, zu programmieren, beachten Sie folgende Regel und Grenzwerte:

### Regel

```
FOR ii := anfang TO ende BY schritt DO
```

| Wenn...                       | ...dann                             | Bemerkung                                   |
|-------------------------------|-------------------------------------|---|
| anfang < ende                 | ende < (P <sub>MAX</sub> - schritt) | Laufvariable ii läuft in positiver Richtung |
| anfang > ende AND schritt < 0 | ende > (N <sub>MAX</sub> - schritt) | Laufvariable ii läuft in negativer Richtung |

### Grenzwerte

Für die möglichen Datentypen gelten unterschiedliche Grenzwerte:

| Datentyp        | P <sub>MAX</sub>    | N <sub>MAX</sub>     |
|-----------------|---------------------|----------------------|
| ii vom Typ SINT | 127                 | -128                 |
| ii vom Typ INT  | 32767               | -32768               |
| ii vom Typ DINT | 2147483647          | -2147483648          |
| ii vom Typ LINT | 9223372036854775807 | -9223372036854775808 |

## Syntax

Für die Anweisung "In Zählschleife ausführen" wird die folgende Syntax verwendet:

### SCL

```
FOR <Laufvariable> := <Anfangswert> TO <Endwert> BY <Schrittweite> DO
<Anweisungen>
END_FOR;
```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "In Zählschleife ausführen":

| Parameter          | Datentyp        |                          | Beschreibung  |
|--------------------|-----------------|--------------------------|---|
|                    | S7-1200         | S7-1500                  |   |
| <Laufvariable<br>> | SINT, INT, DINT | SINT, INT, DINT,<br>LINT | Operand, dessen Wert bei der Schleifenbearbeitung ausgewertet wird. Der Datentyp der Laufvariablen bestimmt den Datentyp der restlichen Parameter.  |
| <Anfangswert<br>>  | SINT, INT, DINT | SINT, INT, DINT,<br>LINT | Ausdruck, dessen Wert bei Beginn der Schleifenbearbeitung der Laufvariablen zugewiesen wird.  |
| <Endwert>          | SINT, INT, DINT | SINT, INT, DINT,<br>LINT | Ausdruck, dessen Wert den letzten Durchlauf der Programmschleife bestimmt. Vor jedem Schleifendurchlauf wird überprüft, welchen Wert die Laufvariable hat: <ul style="list-style-type: none"> <li>• Endwert nicht erreicht:<br/>Die Anweisungen nach DO werden ausgeführt</li> <li>• Endwert erreicht:<br/>Die FOR-Schleife wird ein letztes Mal ausgeführt</li> <li>• Endwert überschritten:<br/>Die FOR-Schleife wird beendet</li> </ul> Eine Veränderung des Endwerts während der Ausführung der Anweisung ist nicht zulässig. |
| <Schrittweite>     | SINT, INT, DINT | SINT, INT, DINT,<br>LINT | Ausdruck, um dessen Wert die Laufvariable nach jedem Schleifendurchlauf erhöht (positive Schrittweite) oder verringert (negative Schrittweite) wird. Die Angabe der Schrittweite ist optional. Wenn keine Schrittweite angegeben ist, wird der Wert der Laufvariablen nach jedem Schleifendurchlauf um 1 erhöht.<br><br>Eine Veränderung der Schrittweite während der Ausführung der Anweisung ist nicht zulässig.  |
| <Anweisung<br>n>   | -               | -                        | Anweisungen, die bei jedem Schleifendurchlauf ausgeführt werden, solange der Wert der Laufvariablen im Wertebereich liegt. Der Wertebereich wird durch den Anfangs- und Endwert bestimmt.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
FOR i := 2 TO 8 BY 2
```

**SCL**

```
DO "a_array[i] := "Tag_Value"*"b_array[i]";  
END_FOR;
```

Der Operand "Tag\_Value" wird mit den Elementen (2, 4, 6, 8) der ARRAY-Variablen "b\_array" multipliziert. Das Ergebnis wird in den Elementen (2, 4, 6, 8) der ARRAY-Variablen "a\_array" eingelesen.

**Siehe auch**

CONTINUE: Schleifenbedingung erneut prüfen (Seite 2290)

EXIT: Schleife sofort verlassen (Seite 2291)

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**WHILE: Bei erfüllter Bedingung ausführen****Beschreibung**

Mit der Anweisung "Bei erfüllter Bedingung ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Durchführungsbedingung erfüllt ist. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung "Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

**Syntax**

Für die Anweisung "Bei erfüllter Bedingung ausführen" wird die folgende Syntax verwendet:

```
SCL  
WHILE <Bedingung> DO <Anweisungen>  
END_WHILE;
```

Die Syntax der WHILE-Anweisung besteht aus den folgenden Teilen:

| Parameter     | Datentyp | Beschreibung   |
|---------------|----------|--|
| <Bedingung>   | BOOL     | Ausdruck, der vor jedem Schleifendurchlauf ausgewertet wird.   |
| <Anweisungen> | -        | Anweisungen, die bei erfüllter Bedingung ausgeführt werden. Wenn die Bedingung nicht erfüllt ist, wird die Programmbearbeitung nach END_WHILE fortgesetzt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
WHILE
    "Tag_Value1" <> "Tag_Value2"
    DO "Tag_Result"
        := "Tag_Input";
    END_WHILE;
    
```

Solange die Werte der Operanden "Tag\_Value1" und "Tag\_Value2" ungleich sind, wird dem Operanden "Tag\_Result" der Wert des Operanden "Tag\_Input" zugewiesen.

### Siehe auch

- EXIT: Schleife sofort verlassen (Seite 2291)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- CONTINUE: Schleifenbedingung erneut prüfen (Seite 2290)
- Übersicht über die gültigen Datentypen (Seite 1087)

### REPEAT: Bei nicht erfüllter Bedingung ausführen

#### Beschreibung

Mit der Anweisung "Bei nicht erfüllter Bedingung ausführen" bewirken Sie, dass eine Programmschleife wiederholt bearbeitet wird, solange eine Abbruchbedingung nicht erfüllt ist. Die Bedingung ist ein Ausdruck mit einem booleschen Wert (TRUE oder FALSE). Als Bedingung können logische Ausdrücke oder Vergleichsausdrücke angegeben werden.

Bei der Ausführung der Anweisung werden die angegebenen Ausdrücke ausgewertet. Wenn der Wert eines Ausdrucks TRUE ist, so gilt die Bedingung als erfüllt, bei FALSE gilt die Bedingung als nicht erfüllt.

Die Anweisungen werden einmal ausgeführt, auch wenn die Abbruchbedingung erfüllt ist.

Programmschleifen können auch geschachtelt werden. Innerhalb einer Programmschleife können weitere Programmschleifen mit anderen Laufvariablen programmiert werden.

Der aktuelle Durchlauf einer Programmschleife kann durch die Anweisung "Schleifenbedingung erneut prüfen" (CONTINUE) beendet werden. Durch die Anweisung "Schleife sofort verlassen" (EXIT) können Sie die gesamte Schleifenbearbeitung beenden. Weitere Informationen dazu finden Sie unter "Siehe auch".

## Syntax

Für die Anweisung "Bei nicht erfüllter Bedingung ausführen" wird die folgende Syntax verwendet:

| SCL                           |  |
|-------------------------------|--|
| REPEAT <Anweisungen>          |  |
| UNTIL <Bedingung> END_REPEAT; |  |

Die Syntax der REPEAT-Anweisung besteht aus den folgenden Teilen:

| Parameter     | Datentyp | Beschreibung  |
|---------------|----------|---|
| <Anweisungen> | -        | Anweisungen, die solange ausgeführt werden, wie die programmierte Bedingung den Wert FALSE hat. Die Anweisungen werden einmal ausgeführt, auch wenn die Abbruchbedingung erfüllt ist.   |
| <Bedingung>   | BOOL     | Ausdruck, der nach jedem Schleifendurchlauf ausgewertet wird. Wenn der Ausdruck den Wert FALSE hat, wird die Programmschleife erneut bearbeitet. Wenn der Ausdruck den Wert TRUE hat, wird die Programmbearbeitung nach END_REPEAT fortgesetzt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

| SCL                 |  |
|---------------------|--|
| REPEAT "Tag_Result" |  |
| := "Tag_Value";     |  |
| UNTIL "Tag_Error"   |  |
| END_REPEAT;         |  |

Solange der Wert des Operanden "Tag\_Error" den Signalzustand "0" hat, wird dem Operanden "Tag\_Result" der Wert des Operanden "Tag\_Value" zugewiesen.

### Siehe auch

- CONTINUE: Schleifenbedingung erneut prüfen (Seite 2290)
- EXIT: Schleife sofort verlassen (Seite 2291)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- Übersicht über die gültigen Datentypen (Seite 1087)

## CONTINUE: Schleifenbedingung erneut prüfen

### Beschreibung

Mit der Anweisung "Schleifenbedingung erneut prüfen" beenden Sie den aktuellen Programmdurchlauf einer FOR-, WHILE- oder REPEAT-Schleife.

Nach der Ausführung der Anweisung werden die Bedingungen zur Fortsetzung der Programmschleife erneut ausgewertet. Die Anweisung hat Auswirkung auf die Programmschleife, die die Anweisung unmittelbar umgibt.

### Syntax

Für die Anweisung "Schleifenbedingung erneut prüfen" wird die folgende Syntax verwendet:

```
SCL
CONTINUE;
```

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
FOR i
    := 1 TO 15 BY 2 DO
    IF (i < 5) THEN
        CONTINUE;
    END_IF;
    "DB10".Test[i] := 1;
END_FOR;
```

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn die Bedingung  $i < 5$  erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] := 1) nicht bearbeitet. Die Laufvariable (i) wird um die Schrittweite "2" erhöht und geprüft, ob ihr aktueller Wert im programmierten Wertebereich liegt. Wenn die Laufvariable im Wertebereich liegt, wird wieder die IF-Bedingung ausgewertet.



Wenn die Bedingung  $i < 5$  nicht erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] := 1) bearbeitet und ein neuer Schleifendurchlauf gestartet. Die Laufvariable wird ebenfalls um die Schrittweite "2" erhöht und geprüft.

## Siehe auch

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

EXIT: Schleife sofort verlassen (Seite 2291)

Übersicht über die gültigen Datentypen (Seite 1087)

## EXIT: Schleife sofort verlassen

### Beschreibung

Mit der Anweisung "Schleife sofort verlassen" brechen Sie die Bearbeitung einer FOR-, WHILE- oder REPEAT-Schleife, unabhängig von Bedingungen, an beliebiger Stelle ab. Die Programmbearbeitung wird nach dem Abschluss der Schleife (END\_FOR, END\_WHILE, END\_REPEAT) fortgesetzt.

Die Anweisung hat Auswirkung auf die Programmschleife, die die Anweisung unmittelbar umgibt.

### Syntax

Für die Anweisung "Schleife sofort verlassen" wird die folgende Syntax verwendet:

```
SCL
EXIT;
```

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
FOR i := 15 TO 1 BY -2 DO
IF (i < 5)
THEN EXIT;
END_IF;
"DB10".Test[i] := 1;
END_FOR;
```

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Wenn die Bedingung  $i < 5$  erfüllt ist, wird die Bearbeitung der Schleife abgebrochen. Die Programmbearbeitung wird nach END\_FOR fortgesetzt.

Wenn die Bedingung  $i < 5$  nicht erfüllt ist, wird die nachfolgende Wertzuweisung ("DB10".Test[i] :=1) bearbeitet und ein neuer Schleifendurchlauf gestartet. Die Laufvariable (i) wird um die Schrittweite "-2" verringert und geprüft, ob ihr aktueller Wert im programmierten Wertebereich liegt. Wenn die Laufvariable (i) im Wertebereich liegt, wird wieder die IF-Bedingung ausgewertet.

### Siehe auch

- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- CONTINUE: Schleifenbedingung erneut prüfen (Seite 2290)
- Übersicht über die gültigen Datentypen (Seite 1087)

## GOTO: Springen

### Beschreibung

Mit der Anweisung "Springen" setzen Sie die Programmbearbeitung an einer Stelle fort, die durch eine Sprungmarke gekennzeichnet ist.

Die Sprungmarke und die Anweisung "Springen" müssen im gleichen Baustein liegen. Die Bezeichnung einer Sprungmarke darf innerhalb eines Bausteins nur einmal vergeben werden. Jede Sprungmarke kann von mehreren Sprunganweisungen angesprungen werden.

Ein Sprung von "außen" in eine Programmschleife ist nicht zulässig, aber ein Sprung von einer Programmschleife nach "außen" ist möglich.

### Syntax

Für die Anweisung "Springen" wird die folgende Syntax verwendet:

```
SCL
GOTO <Sprungmarke>
...
<Sprungmarke>: <Anweisungen>
```

Die Syntax der GOTO-Anweisung besteht aus den folgenden Teilen:

| Parameter     | Datentyp | Beschreibung  |
|---------------|----------|---|
| <Sprungmarke> | -        | Sprungmarke, zu der gesprungen wird                 |
| <Anweisungen> | -        | Anweisungen, die nach dem Sprung ausgeführt werden. |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
CASE "Tag_Value" OF
1 : GOTO MyLABEL1;
2 : GOTO MyLABEL2;
3 : GOTO MyLABEL3;
ELSE GOTO MyLABEL4;
END_CASE;
MyLABEL1: "Tag_1" := 1;
MyLABEL2: "Tag_2" := 1;
MyLABEL3: "Tag_3" := 1;
MyLABEL4: "Tag_4" := 1;
```

Abhängig vom Wert des Operanden "Tag\_Value" wird die Programmbearbeitung an der Stelle fortgesetzt, die durch die entsprechende Sprungmarke gekennzeichnet ist. Wenn z. B. der Operand "Tag\_Value" den Wert "2" hat, wird die Programmbearbeitung bei der Sprungmarke "MyLABEL2" fortgesetzt. Die Programmzeile, die durch die Sprungmarke "MyLABEL1" gekennzeichnet ist, wird in diesem Fall übersprungen.

## Siehe auch

[Operatoren und Operatorenrangfolge \(Seite 1370\)](#)

[SCL-Anweisungen eingeben \(Seite 1380\)](#)

[SCL-Anweisungen bearbeiten \(Seite 1398\)](#)

[Übersicht über die gültigen Datentypen \(Seite 1087\)](#)

## RETURN: Baustein verlassen

### Beschreibung

Mit der Anweisung "Baustein verlassen" beenden Sie die Programmbearbeitung im aktuell bearbeiteten Baustein und setzen sie im aufrufenden Baustein fort.

Am Bausteinende kann die Anweisung entfallen.

### Syntax

Für die Anweisung "Baustein verlassen" wird die folgende Syntax verwendet:

```
SCL
RETURN;
```

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
IF "Tag_Error" <>0 THEN RETURN;
END_IF;
```

Wenn der Signalzustand des Operanden "Tag\_Error" ungleich Null ist, wird die Programmbearbeitung im aktuell bearbeiteten Baustein beendet.

### Siehe auch

- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- Übersicht über die gültigen Datentypen (Seite 1087)

### (\*...\*): Kommentarabschnitt einfügen

#### Beschreibung

Mit der Anweisung "Kommentarabschnitt einfügen" fügen Sie einen Kommentarabschnitt ein. Der Text, der innerhalb der Klammern "(\*...\*)" steht, wird wie ein Kommentar behandelt.

#### Syntax

Für die Anweisung "Kommentarabschnitt einfügen" wird die folgende Syntax verwendet:

```
SCL
(*...*)
```

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
(*Dies ist ein Kommentarabschnitt.*)
```

## Siehe auch

- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- Übersicht über die gültigen Datentypen (Seite 1087)

## Laufzeitsteuerung

### ENDIS\_PW: Passwort-Legitimierung einschränken und freigeben

#### Beschreibung

Mit der Anweisung "Passwort-Legitimierung einschränken und freigeben" legen Sie fest, ob für die CPU eine Legitimation erlaubt ist oder nicht. Sie können legitimierte Verbindungen verhindern, auch wenn das korrekte Passwort bekannt ist. Wenn Sie die Anweisung aufrufen und der Parameter REQ den Signalzustand "0" hat, wird lediglich der momentan eingestellte Zustand an den Ausgangsparametern angezeigt, aber keine Einstellung verändert. Wenn der Parameter REQ den Signalzustand "1" hat, wird der Signalzustand von den Eingangsparametern (F\_PWD, FULL\_PWD, R\_PWD, HMI\_PWD) übernommen. FALSE bedeutet, dass die Legitimation per Passwort nicht erlaubt ist, TRUE bedeutet, das Passwort ist verwendbar. Das Sperren bzw. Freigeben der Passwörter kann einzeln erlaubt oder untersagt werden. Es können z. B. alle Passwörter nicht erlaubt sein, außer dem Failsafe-Passwort. So können Sie die Zugriffsmöglichkeiten auf eine kleine Anwendergruppe begrenzen. Die Ausgangsparameter (F\_PWD\_ON, FULL\_PWD\_ON, R\_PWD\_ON, HMI\_PWD\_ON) zeigen immer den aktuellen Status der Passwortverwendung, unabhängig vom Parameter REQ.

Die gleiche Einstellung kann im Frontpanel der CPU vorgenommen werden und die CPU speichert die jeweils letzte Einstellung.

Um ein ungewolltes Aussperren zu verhindern, kann bei einer S7-1500-CPU der Schutz durch ein Umlegen des Betriebsartenschalters nach STOP außer Kraft gesetzt werden. Der Schutz wird nach dem Umlegen des Betriebsartenschalters in RUN automatisch wieder eingestellt, ohne dass die Anweisung "Passwort-Legitimierung einschränken und freigeben" erneut aufgerufen werden muss oder im Frontpanel weitere Aktionen notwendig sind.

Die Anweisung muss immer komplett durchführbar sein, d. h. es muss z. B. der Parameter F\_PWD immer den Signalzustand "0" haben, damit die Einstellungen abgespeichert werden können.

Unter den folgenden Bedingungen können gesperrte Passwörter wieder zugelassen werden:

- Die CPU wurde auf die Werkseinstellung zurückgesetzt.
- Das Frontpanel der S7-1500-CPU unterstützt einen Dialog, mit dessen Hilfe Sie zum passenden Menü navigieren und die Passwörter wieder zulassen können.
- Beim Aufruf der Anweisung "Passwort-Legitimierung einschränken und freigeben" hat der Eingangsparameter des gewünschten Passworts den Signalzustand "1".

- Den Betriebsartenschalter auf STOP setzen. Die Einschränkung der Passwort-Legitimierung wird wieder eingerichtet, sobald der Schalter wieder nach RUN bewegt wird.
- Das Stecken eines Transfermoduls in eine S7-1200-CPU.

---

**Hinweis**

Die Anweisung "Passwort-Legitimierung einschränken und freigeben" sperrt den Zugriff auf HMI Panels, wenn das HMI Passwort nicht freigegeben ist.

---

**Hinweis**

Bereits bestehende und legitimierte Verbindungen behalten ihre Zugriffs-Rechte und können über die Anweisung "Passwort-Legitimierung einschränken und freigeben" nicht eingeschränkt werden.

---

### Vorgehensweise bei verlorenem Passwort bei einer S7-1200-CPU

Wenn Sie das Passwort für eine passwortgeschützte CPU verloren haben, löschen Sie das passwortgeschützte Programm mit einer leeren Übertragungskarte. Die leere Übertragungskarte löscht den internen Ladespeicher der CPU. Dann können Sie ein neues Anwenderprogramm aus STEP 7 Basic in die CPU laden.

|  |
|--|
|  <b>WARNUNG</b> |
|--|

|                                  |
|----------------------------------|
| <b>Übertragungskarte stecken</b> |
|----------------------------------|

|  |
|--|
| Wenn Sie eine Übertragungskarte in eine laufende CPU stecken, geht die CPU in STOP. Steuerungen können bei unsicheren Betriebszuständen ausfallen und dadurch den unkontrollierten Betrieb der gesteuerten Geräte verursachen. Daraus resultiert ein unvorhersehbarer Betrieb des Automatisierungssystems, der zu tödlichen oder schweren Verletzungen und/oder Sachschaden führen kann. |
|--|

Sie müssen die Übertragungskarte ziehen, bevor Sie die CPU in RUN versetzen.

## Auswirkungen der Passwortverwendung auf die Betriebsarten

Die folgende Tabelle zeigt, welche Auswirkungen die Passwortverwendung durch die Anweisung "Passwort-Legitimierung einschränken und freigeben" auf die Betriebsarten und die entsprechenden Anwenderaktionen hat.

| Aktion  | Passwortschutz durch die Anweisung  |
|---|---|
| Grundzustand nach <ul style="list-style-type: none"> <li>• Betriebsartenschalter auf STOP</li> <li>• Speicher manuell zurücksetzen (PG, Schalter, Änderung des MC (Motion Control))</li> <li>• Zurücksetzen auf Werkseinstellung</li> </ul> | Nicht aktiviert<br>(Keine Einschränkungen)  |
| Grundzustand nach NETZ-EIN  | Aktiviert (wenn vor NETZ-AUS eine Sperre aktiviert war)<br>Die Möglichkeit, Passwörter nicht zu erlauben, ist remanent. |
| Betriebszustandsübergang RUN/ANLAUF/HALT -> STOP (durch das Beenden der Anweisung, einen Fehler oder Kommunikation) oder STOP -> ANLAUF/RUN/HALT  | Aktiviert<br>Die Passwörter dürfen weiterhin nicht verwendet werden.  |

## Syntax

Für die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird die folgende Syntax verwendet:

### SCL

```

ENDIS_PW(REQ := <Operand>,
         F_PWD := <Operand>,
         FULL_PWD := <Operand>,
         R_PWD := <Operand>,
         HMI_PWD := <Operand>,
         F_PWD_ON => <Operand>,
         FULL_PWD_ON => <Operand>,
         R_PWD_ON => <Operand>,
         HMI_PWD_ON => <Operand>,
         RET_VAL => <Operand>)

```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Passwort-Legitimierung einschränken und freigeben":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| REQ       | Input       | BOOL     | E, A, M, D, L   | Wenn der Parameter REQ den Signalzustand "0" führt, wird der aktuell eingestellte Signalzustand der Passwörter abgefragt.   |
| F_PWD     | Input       | BOOL     | E, A, M, D, L   | Lese-/Schreibzugriff inklusive Failsafe <ul style="list-style-type: none"> <li>• F_PWD = "0": Passwort nicht erlauben</li> <li>• F_PWD = "1": Passwort erlauben</li> </ul>            |
| FULL_PWD  | Input       | BOOL     | E, A, M, D, L   | Lese-/Schreibzugriff <ul style="list-style-type: none"> <li>• FULL_PWD = "0": Passwort nicht erlauben</li> <li>• FULL_PWD = "1": Passwort erlauben</li> </ul>                         |
| R_PWD     | Input       | BOOL     | E, A, M, D, L   | Lesezugriff <ul style="list-style-type: none"> <li>• R_PWD = "0": Passwort nicht erlauben</li> <li>• R_PWD = "1": Passwort erlauben</li> </ul>  |
| HMI_PWD   | Input       | BOOL     | E, A, M, D, L   | HMI-Zugriff <ul style="list-style-type: none"> <li>• HMI_PWD = "0": Passwort nicht erlauben</li> <li>• HMI_PWD = "1": Passwort erlauben</li> </ul>                                    |
| F_PWD_ON  | Output      | BOOL     | E, A, M, D, L   | Status Lese-/Schreibzugriff inklusive Failsafe <ul style="list-style-type: none"> <li>• F_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• F_PWD_ON = "1": Passwort erlaubt</li> </ul> |



| Parameter   | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-------------|-------------|----------|-----------------|--|
| FULL_PWD_ON | Output      | BOOL     | E, A, M, D, L   | Status Lese-/Schreibzugriff <ul style="list-style-type: none"> <li>• FULL_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• FULL_PWD_ON = "1": Passwort erlaubt</li> </ul> |
| R_PWD_ON    | Output      | BOOL     | E, A, M, D, L   | Status Lesezugriff <ul style="list-style-type: none"> <li>• R_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• R_PWD_ON = "1": Passwort erlaubt</li> </ul>                |
| HMI_PWD_ON  | Output      | BOOL     | E, A, M, D, L   | Status HMI-Zugriff <ul style="list-style-type: none"> <li>• HMI_PWD_ON = "0": Passwort nicht erlaubt</li> <li>• HMI_PWD_ON = "1": Passwort erlaubt</li> </ul>            |
| RET_VAL     | Output      | WORD     | E, A, M, D, L   | Fehlerinformation  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)  | Erläuterung  |
|--|--|
| 0000   | Kein Fehler  |
| 8090   | Die Anweisung "Passwort-Legitimierung einschränken und freigeben" wird nicht unterstützt |
| 80D0   | Das Passwort für Failsafe ist nicht konfiguriert   |
| 80D1   | Der Lese-/Schreibzugriff ist nicht konfiguriert  |
| 80D2   | Der Lesezugriff ist nicht konfiguriert   |
| 80D3   | Der HMI-Zugriff ist nicht konfiguriert   |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## RE\_TRIGR: Zyklusüberwachungszeit neu starten

### Beschreibung

Mit der Anweisung "Zyklusüberwachungszeit neu starten" starten Sie die Zykluszeitüberwachung der CPU neu. Die Zyklusüberwachungszeit läuft dann mit der Dauer neu an, die Sie bei der CPU-Konfiguration eingestellt haben.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann unabhängig von der Priorität in allen Bausteinen aufgerufen werden.

Wenn die Anweisung in einem Baustein mit höherer Priorität, z. B. einem Prozessalarm, einem Diagnosealarm oder einem Weckalarm, aufgerufen wird, wird die Anweisung nicht ausgeführt und der Freigabeausgang ENO auf den Signalzustand "0" gesetzt.

Die Anweisung "Zyklusüberwachungszeit neu starten" kann max. 10-mal in einem Programmzyklus aufgerufen werden.

---

### Hinweis

Wenn Sie die Anweisung "Zyklusüberwachungszeit neu starten" innerhalb einer LOOP-Schleife mehr als 30 Mal starten, geht die CPU aufgrund eines Laufzeitfehlers in den Betriebszustand STOP.

---

### Syntax

Für die Anweisung "Zyklusüberwachungszeit neu starten" wird die folgende Syntax verwendet:

```
SCL  
RE_TRIGR()
```

### Siehe auch

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## STP: Programm beenden

### Beschreibung

Mit der Anweisung "Programm beenden" versetzen Sie die CPU in den Betriebszustand STOPP und beenden damit die Programmbearbeitung. Die Auswirkungen beim Übergang von RUN in STOPP hängen von der CPU-Konfiguration ab.

## Syntax

Für die Anweisung "Programm beenden" wird die folgende Syntax verwendet:

|       |
|-------|
| SCL   |
| STP() |

## Siehe auch

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## GET\_ERROR: Fehler lokal abfragen

### Beschreibung

Mit der Anweisung "Fehler lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Wenn das System während der Bausteinbearbeitung Fehler meldet, gibt die Anweisung zum ersten aufgetretenen Fehler eine ausführliche Information aus.

Die Fehlerinformation kann nur in Operanden des Systemdatentyps "ErrorStruct" gespeichert werden. Der Systemdatentyp "ErrorStruct" gibt die genaue Struktur vor, in der die Informationen zum aufgetretenen Fehler gespeichert werden. Mithilfe weiterer Anweisungen können Sie diese Struktur auswerten und eine entsprechende Reaktion programmieren. Wenn der erste aufgetretene Fehler behoben ist, gibt die Anweisung die Fehlerinformation zum nächsten aufgetretenen Fehler aus.

### Hinweis

Die Anweisung "Fehler lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn "Fehler lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

## Syntax

Für die Anweisung "Fehler lokal abfragen" wird die folgende Syntax verwendet:

**SCL**

```
<Fehlerinformation> := GET_ERROR(<Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler lokal abfragen":

| Parameter     | Datentyp    | Beschreibung                         |
|---------------|-------------|--------------------------------------|
| Funktionswert | ErrorStruct | Information zu aufgetretenen Fehlern |

### Datentyp "ErrorStruct"

Die folgende Tabelle zeigt die Struktur des Datentyps "ErrorStruct":

| Strukturkomponente      |                | Datentyp              | Beschreibung  |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
|-------------------------|----------------|-----------------------|---|--------|--------|-----|-----|-----|-----|---|--|--|--|--|--|---|--|--|--|--|--------|---|--|--|------|--|--|---|----------|-------|--|--------|--|---|--|--|------|--|--------|---|--|--|------|--------|--------|---|----------------|--|------|--------|--------|---|----------------|-----------------------|------|--------|--------|
| ERROR_ID                |                | WORD                  | Fehler-ID   |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| FLAGS                   |                | BYTE                  | Zeigt, ob der Fehler während eines Bausteinaufrufs aufgetreten ist.<br>16#01: Fehler während eines Bausteinaufrufs<br>16#00: Kein Fehler während eines Bausteinaufrufs  |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| REACTION                |                | BYTE                  | Voreingestellte Reaktion:<br>0: Ignorieren (Schreibfehler),<br>1: Mit dem Ersatzwert "0" fortsetzen (Lesefehler),<br>2: Anweisung überspringen (Systemfehler)   |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| CODE_ADDRESS            |                | CREF                  | Informationen zur Adresse und Art des Bausteins   |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
|                         | BLOCK_TYPE     | BYTE                  | Bausteintyp, in dem der Fehler aufgetreten ist:<br>1: OB<br>2: FC<br>3: FB  |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
|                         | CB_NUMBER      | UINT                  | Nummer des Codebausteins  |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
|                         | OFFSET         | UDINT                 | Verweis auf den internen Speicher   |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| MODE                    |                | BYTE                  | Zugriffsart: Abhängig von der Zugriffsart können die folgenden Informationen ausgegeben werden:   |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
|                         |                |                       | <table border="1"> <thead> <tr> <th>Mode</th> <th>(A)</th> <th>(B)</th> <th>(C)</th> <th>(D)</th> <th>(E)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>1</td> <td></td> <td></td> <td></td> <td></td> <td>Offset</td> </tr> <tr> <td>2</td> <td></td> <td></td> <td>Area</td> <td></td> <td></td> </tr> <tr> <td>3</td> <td>Location</td> <td>Scope</td> <td></td> <td>Number</td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td>Area</td> <td></td> <td>Offset</td> </tr> <tr> <td>5</td> <td></td> <td></td> <td>Area</td> <td>DB no.</td> <td>Offset</td> </tr> <tr> <td>6</td> <td>PtrNo./<br/>Acc</td> <td></td> <td>Area</td> <td>DB no.</td> <td>Offset</td> </tr> <tr> <td>7</td> <td>PtrNo./<br/>Acc</td> <td>Slot<br/>No./<br/>Scope</td> <td>Area</td> <td>DB no.</td> <td>Offset</td> </tr> </tbody> </table> | Mode   | (A)    | (B) | (C) | (D) | (E) | 0 |  |  |  |  |  | 1 |  |  |  |  | Offset | 2 |  |  | Area |  |  | 3 | Location | Scope |  | Number |  | 4 |  |  | Area |  | Offset | 5 |  |  | Area | DB no. | Offset | 6 | PtrNo./<br>Acc |  | Area | DB no. | Offset | 7 | PtrNo./<br>Acc | Slot<br>No./<br>Scope | Area | DB no. | Offset |
| Mode                    | (A)            | (B)                   | (C)   | (D)    | (E)    |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| 0                       |                |                       |   |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| 1                       |                |                       |   |        | Offset |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| 2                       |                |                       | Area  |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| 3                       | Location       | Scope                 |   | Number |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| 4                       |                |                       | Area  |        | Offset |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| 5                       |                |                       | Area  | DB no. | Offset |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| 6                       | PtrNo./<br>Acc |                       | Area  | DB no. | Offset |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| 7                       | PtrNo./<br>Acc | Slot<br>No./<br>Scope | Area  | DB no. | Offset |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| OPERAND_NUMBER          |                | UINT                  | Operandnummer des Maschinenbefehls  |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| POINTER_NUMBER_LOCATION |                | UINT                  | (A) Interner Zeiger   |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| SLOT_NUMBER_SCOPE       |                | UINT                  | (B) Ablagebereich im internen Speicher  |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |
| DATA_ADDRESS            |                | NREF                  | Informationen zur Adresse eines Operanden   |        |        |     |     |     |     |   |  |  |  |  |  |   |  |  |  |  |        |   |  |  |      |  |  |   |          |       |  |        |  |   |  |  |      |  |        |   |  |  |      |        |        |   |                |  |      |        |        |   |                |                       |      |        |        |

| Strukturkomponente |           | Datentyp | Beschreibung   |
|--------------------|-----------|----------|--|
|                    | AREA      | BYTE     | (C) Speicherbereich:<br>L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE<br>E: 16#81<br>A: 16#82<br>M: 16#83<br>DB: 16#84, 85, 8A, 8B<br>Bereichsverletzungen bei einer direkt editierbaren Variablen vom Datentyp DINT: 16#04 |
|                    | DB_NUMBER | UINT     | (D) Nummer des Datenbausteins  |
|                    | OFFSET    | UDINT    | (E) Relative Adresse des Operanden   |

### Strukturkomponente "ERROR\_ID"

Die folgende Tabelle zeigt die Werte, die an der Strukturkomponente "ERROR\_ID" ausgegeben werden können:

| ID*<br>(Hexadezimal) | ID*<br>(Dezimal) | Beschreibung   |
|----------------------|------------------|--|
| 0                    | 0                | Kein Fehler  |
| 2503                 | 9475             | Ungültiger Zeiger  |
| 2520                 | 9504             | Ungültiger STRING  |
| 2522                 | 9506             | Lesefehler: Operand außerhalb des gültigen Bereichs                              |
| 2523                 | 9507             | Schreibfehler: Operand außerhalb des gültigen Bereichs                           |
| 2524                 | 9508             | Lesefehler: Ungültiger Operand   |
| 2525                 | 9509             | Schreibfehler: Ungültiger Operand  |
| 2528                 | 9512             | Lesefehler: Datenausrichtung   |
| 2529                 | 9513             | Schreibfehler: Datenausrichtung  |
| 252C                 | 9516             | Ungültiger Zeiger  |
| 2530                 | 9520             | Schreibfehler: Datenbaustein   |
| 2533                 | 9523             | Unzulässig verwendeter Zeiger  |
| 2538                 | 9528             | Zugriffsfehler: DB nicht vorhanden   |
| 2539                 | 9529             | Zugriffsfehler: Falscher DB verwendet  |
| 253A                 | 9530             | Globaler Datenbaustein existiert nicht   |
| 253C                 | 9532             | Fehlerhafte Angabe oder die Funktion existiert nicht                             |
| 253D                 | 9533             | Systemfunktion existiert nicht   |
| 253E                 | 9534             | Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht                    |
| 253F                 | 9535             | Systembaustein existiert nicht   |
| 2550                 | 9552             | Zugriffsfehler: DB nicht vorhanden   |
| 2551                 | 9553             | Zugriffsfehler: Falscher DB verwendet  |
| 2575                 | 9589             | Fehler in der Programmschachtelungstiefe   |
| 2576                 | 9590             | Fehler in der Lokaldatenverteilung   |
| 2577                 | 9591             | Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert. |

| ID*<br>(Hexadezimal)   | ID*<br>(Dezimal) | Beschreibung                       |
|--|------------------|------------------------------------|
| 25A0   | 9632             | Interner Fehler im TP              |
| 25A1   | 9633             | Variable ist schreibgeschützt      |
| 25A2   | 9634             | Ungültiger Zahlenwert der Variable |
| 2942   | 10562            | Lesefehler: Eingang                |
| 2943   | 10563            | Schreibfehler: Ausgang             |
| *Die Fehlercodes im Programmmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                  |                                    |

Die Anweisung "Fehler lokal abfragen" kann auch eingesetzt werden, um an den aufrufenden Baustein eine Meldung über den Fehlerzustand weiterzuleiten. Dafür muss die Anweisung am Ende des aufgerufenen Bausteins programmiert werden.

**Siehe auch**

- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**GET\_ERR\_ID: Fehler-ID lokal abfragen**

**Beschreibung**

Mit der Anweisung "Fehler-ID lokal abfragen" fragen Sie das Auftreten von Fehlern innerhalb eines Bausteins ab. Wenn das System während der Bausteinbearbeitung Fehler meldet, gibt die Anweisung die Fehler-ID des ersten aufgetretenen Fehlers aus. Die Fehler-ID kann nur in Operanden vom Datentyp WORD gespeichert werden. Wenn der erste aufgetretene Fehler behoben ist, gibt die Anweisung die Fehler-ID des nächsten aufgetretenen Fehlers aus.

Die Anweisung "Fehler-ID lokal abfragen" kann auch eingesetzt werden, um an den aufrufenden Baustein eine Meldung über den Fehlerzustand weiterzuleiten. Dafür muss die Anweisung am Ende des aufgerufenen Bausteins programmiert werden.

---

**Hinweis**

Die Anweisung "Fehler-ID lokal abfragen" aktiviert die lokale Fehlerbehandlung innerhalb eines Bausteins. Wenn die Anweisung "Fehler-ID lokal abfragen" im Programmcode eines Bausteins eingefügt ist, werden voreingestellte Systemreaktionen beim Auftreten von Fehlern ignoriert.

---

**Syntax**

Für die Anweisung "Fehler-ID lokal abfragen" wird die folgende Syntax verwendet:

**SCL**  
`<Fehler_ID> := GET_ERR_ID(<Operand>)`

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Fehler-ID lokal abfragen":

| Parameter     | Datentyp | Beschreibung |
|---------------|----------|--------------|
| Funktionswert | WORD     | Fehler-ID    |

## Fehler-ID

Die folgende Tabelle zeigt die Werte, die ausgegeben werden können:

| ID*<br>(Hexadezimal)  | ID*<br>(Dezimal) | Beschreibung   |
|---|------------------|--|
| 0   | 0                | Kein Fehler  |
| 2503  | 9475             | Ungültiger Zeiger  |
| 2520  | 9504             | Ungültiger STRING  |
| 2522  | 9506             | Lesefehler: Operand außerhalb des gültigen Bereichs                              |
| 2523  | 9507             | Schreibfehler: Operand außerhalb des gültigen Bereichs                           |
| 2524  | 9508             | Lesefehler: Ungültiger Operand   |
| 2525  | 9509             | Schreibfehler: Ungültiger Operand  |
| 2528  | 9512             | Lesefehler: Datenausrichtung   |
| 2529  | 9513             | Schreibfehler: Datenausrichtung  |
| 252C  | 9516             | Ungültiger Zeiger  |
| 2530  | 9520             | Schreibfehler: Datenbaustein   |
| 2533  | 9523             | Unzulässig verwendeter Zeiger  |
| 2538  | 9528             | Zugriffsfehler: DB nicht vorhanden   |
| 2539  | 9529             | Zugriffsfehler: Falscher DB verwendet  |
| 253A  | 9530             | Globaler Datenbaustein existiert nicht   |
| 253C  | 9532             | Fehlerhafte Angabe oder die Funktion existiert nicht                             |
| 253D  | 9533             | Systemfunktion existiert nicht   |
| 253E  | 9534             | Fehlerhafte Angabe oder der Funktionsbaustein existiert nicht                    |
| 253F  | 9535             | Systembaustein existiert nicht   |
| 2550  | 9552             | Zugriffsfehler: DB nicht vorhanden   |
| 2551  | 9553             | Zugriffsfehler: Falscher DB verwendet  |
| 2575  | 9589             | Fehler in der Programmschachtelungstiefe   |
| 2576  | 9590             | Fehler in der Lokaldatenverteilung   |
| 2577  | 9591             | Die Bausteineigenschaft "Parameterversorgung über Register" ist nicht aktiviert. |
| 25A0  | 9632             | Interner Fehler im TP  |
| 25A1  | 9633             | Variable ist schreibgeschützt  |
| 25A2  | 9634             | Ungültiger Zahlenwert der Variable   |
| 2942  | 10562            | Lesefehler: Eingang  |
| 2943  | 10563            | Schreibfehler: Ausgang   |
| *Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                  |  |

**Siehe auch**

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**INIT\_RD: Alle remanenten Daten zurücksetzen**

**Beschreibung**

Mit der Anweisung "Alle remanenten Daten zurücksetzen" setzen Sie gleichzeitig alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler zurück. Die Anweisung kann nur innerhalb eines Anlauf-OBs ausgeführt werden, da die Ausführung die Programmzyklusdauer überschreiten würde.

**Syntax**

Für die Anweisung "Alle remanenten Daten zurücksetzen" wird die folgende Syntax verwendet:

**SCL**

```
INIT_RD(REQ := <Operand>  
        RET_VAL := <Operand>)
```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Alle remanenten Daten zurücksetzen":

| Parameter | Deklaration | Datentyp | Beschreibung   |
|-----------|-------------|----------|--|
| REQ       | Input       | BOOL     | Wenn der Eingang REQ den Signalzustand "1" führt, werden alle remanenten Daten zurückgesetzt.  |
| RET_VAL   | Output      | INT      | Fehlerinformationen:<br>Am Parameter RET_VAL wird ein Fehlercode ausgegeben, wenn während der Bearbeitung der Anweisung ein Fehler auftritt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".



## Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode*<br>(W#16#...)    | Erläuterung   |
|------------------------------|---|
| 0000                         | Kein Fehler   |
| 80B5                         | Die Anweisung kann nicht ausgeführt werden, da sie nicht innerhalb eines Anlauf-OBs programmiert wurde. |
| allgemeine Fehlerinformation | Siehe auch: "GET_ERR_ID: Fehler-ID lokal abfragen"  |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
INIT_RD(REQ := "Tag_REQ",
        RET_VAL := "Tag_Result");
```

Wenn der Operanden "Tag\_REQ" den Signalzustand "1" liefert, wird die Anweisung ausgeführt. Alle remanenten Daten aller Datenbausteine, Merker und SIMATIC-Zeiten und -Zähler werden zurückgesetzt.

## Siehe auch

- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- Übersicht über die gültigen Datentypen (Seite 1087)
- GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2304)

## WAIT: Zeitverzögerung programmieren

### Beschreibung

Mit der Anweisung "Zeitverzögerung programmieren" halten Sie die Programmbearbeitung für eine vorgegebene Zeitdauer an. Die Zeitdauer geben Sie in Mikrosekunden am Parameter WT der Anweisung an.

Sie können Verzögerungszeiten bis zu 32 767 Mikrosekunden ( $\mu$ s) programmieren. Die minimale Verzögerungszeit hängt von der jeweiligen CPU ab und entspricht der Ausführungszeit der Anweisung "Zeitverzögerung programmieren".

Die Ausführung der Anweisung kann durch höherpriorie Ereignisse unterbrochen werden.

Die Anweisung "Zeitverzögerung programmieren" liefert keine Fehlerinformationen.

## Syntax

Für die Anweisung "Zeitverzögerung programmieren" wird die folgende Syntax verwendet:

```
SCL
WAIT(WT := <Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter | Deklaration | Datentyp | Beschreibung                           |
|-----------|-------------|----------|--|
| WT        | Input       | INT      | Verzögerungszeit in Mikrosekunden (µs) |

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## RUNTIME: Laufzeitmessung

### Beschreibung

Mit der Anweisung "Laufzeitmessung" messen Sie die Laufzeit des gesamten Programms, einzelner Bausteine oder Befehlssequenzen.

Wenn Sie die Laufzeit Ihres gesamten Programms messen wollen, dann rufen Sie die Anweisung "Laufzeitmessung" im OB1 auf. Der Startpunkt der Laufzeitmessung wird mit dem ersten Aufruf gesetzt und der Ausgang RET\_VAL liefert nach dem zweiten Aufruf die Laufzeit des Programms. Die gemessene Laufzeit enthält alle möglichen Vorgänge der CPU, die innerhalb des Programmablaufs stattgefunden haben, wie z. B. Unterbrechungen durch höher priorisierte Ereignisse oder Kommunikation. Die Anweisung "Laufzeitmessung" liest einen internen Zähler der CPU aus und schreibt dessen Wert in den Durchgangparameter. Die Anweisung berechnet die aktuelle Laufzeit des Programms gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET\_VAL.

Wenn Sie die Laufzeit einzelner Bausteine oder einzelner Befehlssequenzen messen wollen, dann benötigen Sie drei separate Netzwerke. Rufen Sie die Anweisung "Laufzeitmessung" in einem einzelnen Netzwerk innerhalb Ihres Programms auf. Mit diesem ersten Aufruf der Anweisung setzen Sie den Startpunkt der Laufzeitmessung. Anschließend rufen Sie im nächsten Netzwerk den gewünschten Programmbaustein oder die Befehlssequenz auf. Rufen Sie dann in einem weiteren Netzwerk die Anweisung "Laufzeitmessung" zum zweiten Mal auf und weisen Sie dem Durchgangparameter den gleichen Speicher wie beim ersten Aufruf der Anweisung zu. Die Anweisung "Laufzeitmessung" im dritten Netzwerk liest einen internen Zähler der CPU aus und berechnet die aktuelle Laufzeit des Programmbausteins oder der Befehlssequenz gemäß der internen Zählerfrequenz und schreibt diese in den Ausgang RET\_VAL.

Die Anweisung "Laufzeitmessung" verwendet einen internen hochfrequenten Zähler um die Zeit zu berechnen. Wenn der Zähler überläuft, gibt die Anweisung Werte  $\leq 0.0$  zurück. Diese Runtimewerte sind zu ignorieren.

### Hinweis

Da beim optimierten Übersetzen des Programms die Reihenfolge der Anweisungen innerhalb einer Befehlssequenz geändert wird, kann die Laufzeit einer Befehlssequenz nicht genau bestimmt werden.

Die Anweisung "Laufzeitmessung" liefert keine Fehlerinformationen.

## Syntax

Für die Anweisung "Laufzeitmessung" wird die folgende Syntax verwendet:

```
SCL
RUNTIME (<Operand>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp | Beschreibung                                  |
|---------------|-------------|----------|---|
| <Operand>     | InOut       | LREAL    | Speichert den Startpunkt der Laufzeitmessung. |
| Funktionswert |             | LREAL    | Liefert die gemessene Laufzeit in Sekunden    |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung anhand einer Berechnung der Laufzeit eines Programmbausteins:

```
SCL
"Tag_Result" := RUNTIME("Tag_Memory");

"Best_before_date_DB" ();

"Tag_Result" := RUNTIME("Tag_Memory");
```

Mit dem ersten Aufruf der Anweisung wird der Startpunkt für die Laufzeitmessung gesetzt und als Referenz für den zweiten Aufruf der Anweisung im Operanden "TagMemory" zwischengespeichert.

Im Anschluss wird der Programmbaustein FB1 "Best\_before\_date" aufgerufen.

Wenn der Programmbaustein FB1 abgearbeitet ist, wird die Anweisung zum zweiten Mal ausgeführt. Der zweite Aufruf der Anweisung berechnet die Laufzeit des Programmbausteins und schreibt das Ergebnis in den Ausgang "Tag\_Result".

**Siehe auch**

- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)
- Übersicht über die gültigen Datentypen (Seite 1087)

**Wortverknüpfungen**

**DECO: Decodieren**

**Beschreibung**

Mit der Anweisung "Decodieren" setzen Sie ein durch den Eingangswert vorgegebenes Bit im Ausgangswert.

Die Anweisung "Decodieren" liest den Wert des Parameters IN ab und setzt das Bit im Ausgangswert, dessen Bitposition dem abgelesenen Wert entspricht. Die übrigen Bits im Ausgangswert werden mit Nullen gefüllt. Wenn der Wert des Parameters IN größer als 31 ist, wird eine Modulo-32-Anweisung ausgeführt.

**Syntax**

Für die Anweisung "Decodieren" wird die folgende Syntax verwendet:

```
SCL
DECO(IN := <Ausdruck>)
DECO_WORD(IN := <Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter       | Deklaration | Datentyp                        | Beschreibung  |
|-----------------|-------------|---------------------------------|---|
| IN              | Input       | UINT                            | Position des Bits im Ausgangswert, das gesetzt wird.  |
| _<br><Datentyp> |             | Bitfolgen<br>Vorbelegung: DWORD | Datentyp des Funktionswerts: <ul style="list-style-type: none"> <li>• Wenn Sie den voreingestellten Datentyp verwenden wollen, entfällt die Angabe.</li> <li>• Wenn Sie einen der anderen zulässigen Datentypen verwenden wollen, müssen Sie ihn explizit angeben.</li> </ul> |
| Funktionswert   |             | Bitfolgen                       | Aktueller Ausgangswert  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

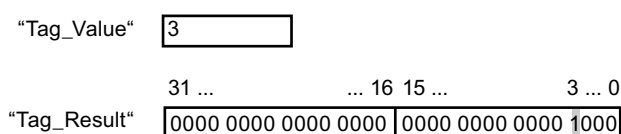
## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

### SCL

```
"Tag_Result" := DECO(IN := "Tag_Value");  
"Tag_Result2" := DECO_BYTE(IN := "Tag_Value2");
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:



Die Anweisung liest die Zahl "3" aus dem Wert des Operanden "Tag\_Value" ab und setzt das dritte Bit im Wert des Operanden "Tag\_Result".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## ENCO: Encodieren

### Beschreibung

Mit der Anweisung "Encodieren" lesen Sie die Bitnummer des niederwertigsten gesetzten Bits im Eingangswert ab und geben diese als Ergebnis aus.

### Syntax

Für die Anweisung "Encodieren" wird die folgende Syntax verwendet:

#### SCL

```
ENCO(IN := <Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp  | Beschreibung   |
|---------------|-------------|-----------|--|
| IN            | Input       | Bitfolgen | Eingangswert   |
| Funktionswert |             | INT       | Bitnummer des Bits im Eingangswert, das ausgelesen wird. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

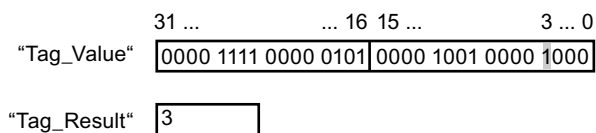
### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

**SCL**

```
"Tag_Result" := ENCO(IN := "Tag_Value");
```

Das folgende Bild zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:



Die Anweisung liest das niederwertigste gesetzte Bit des Operanden "Tag\_Value" ab und schreibt die Bitposition "3" in den Operanden "Tag\_Result".

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

### SEL: Selektieren

#### Beschreibung

Mit der Anweisung "Selektieren" wählen Sie abhängig von einem Schalter (Parameter G) einen der Parameter IN0 oder IN1 aus und geben dessen Inhalt als Ergebnis aus. Wenn der Parameter G den Signalzustand "0" hat, wird der Wert am Parameter IN0 kopiert. Wenn der Parameter G den Signalzustand "1" hat, wird der Wert am Parameter IN1 kopiert und als Funktionswert zurückgeliefert.

Die Ausführung der Anweisung setzt voraus, dass die Variablen an allen Parametern vom Datentyp der gleichen Klasse sind.

## Syntax

Für die Anweisung "Selektieren" wird die folgende Syntax verwendet:

```
SCL
SEL(G := <Ausdruck>,
     IN0 := <Ausdruck>,
     IN1 := <Ausdruck>)
```

Die Syntax der Anweisung besteht aus den folgenden Teilen:

| Parameter     | Deklaration | Datentyp   |  | Beschreibung              |
|---------------|-------------|--|--|---------------------------|
|               |             | S7-1200  | S7-1500  |                           |
| G             | Input       | BOOL   | BOOL   | Schalter                  |
| IN0           | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD, DATE,<br>CHAR, DT | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>CHAR, Zeiten,<br>DATE, TOD, LTOD,<br>DT, LDT | Erster Eingangswert       |
| IN1           | Input       | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD, DATE,<br>CHAR, DT | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>CHAR, Zeiten,<br>DATE, TOD, LTOD,<br>DT, LDT | Zweiter Eingangswert      |
| Funktionswert |             | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>TIME, TOD, DATE,<br>CHAR, DT | Bitfolgen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>CHAR, Zeiten,<br>DATE, TOD, LTOD,<br>DT, LDT | Ergebnis der<br>Anweisung |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := SEL(G := "Tag_Value",
                   IN0 := "Tag_0",
                   IN1 := "Tag_1");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert      |         |
|------------|-----------|---------|
| Tag_Value  | 0         | 1       |
| Tag_0      | W#16#0000 | W#16#4C |
| Tag_1      | W#16#FFFF | D#16#5E |
| Tag_Result | W#16#0000 | D#16#5E |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**MUX: Multiplexen**

**Beschreibung**

Mit der Anweisung "Multiplexen" kopieren Sie den Wert eines ausgewählten Eingangsparameters und geben diesen aus. Durch den Parameter K bestimmen Sie die Nummer des Eingangsparameters, dessen Wert kopiert wird. Die Nummerierung fängt dabei mit IN0 an und wird mit jedem neuen Eingang aufsteigend fortgesetzt. Sie können maximal 32 Eingänge deklarieren.

Wenn der Wert des Parameters K größer als die Anzahl der Eingänge ist und der Parameter INELSE nicht gesetzt ist, wird der Freigabeausgang ENO auf "0" gesetzt.

An den Eingängen sind numerische Datentypen und Zeitdatentypen zugelassen. Alle parametrisierten Variablen müssen vom gleichen Datentyp sein.

Der Funktionswert ist ungültig, wenn eine der folgenden Bedingungen zutrifft:

- Der Wert des Parameters K ist größer als die Anzahl der verfügbaren Eingänge.
- Es treten Fehler während der Bearbeitung der Anweisung auf.

**Syntax**

Für die Anweisung "Multiplexen" wird die folgende Syntax verwendet:

```

SCL
MUX(K := <Ausdruck>,
    IN0 := <Ausdruck>,
    IN1 := <Ausdruck>,
    INELSE := <Ausdruck>)
    
```



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Multiplexen":

| Parameter     | Deklaration | Datentyp  |  | Beschreibung   |
|---------------|-------------|---|--|--|
|               |             | S7-1200   | S7-1500  |  |
| K             | Input       | Ganzzahlen  | Ganzzahlen   | Gibt den Parameter an, dessen Inhalt übertragen wird. <ul style="list-style-type: none"> <li>• Wenn K = 0 =&gt; Parameter IN0</li> <li>• Wenn K = 1 =&gt; Parameter IN1, etc.</li> </ul> |
| IN0           | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, DATE, TIME, DT | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT | Erster Eingangswert  |
| IN1           | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, DATE, TIME, DT | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT | Zweiter Eingangswert   |
| INn           | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, DATE, TIME, DT | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT | Optionale Eingangswerte  |
| INELSE        | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, DATE, TIME, DT | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT | Gibt den Wert an, der bei K <> n kopiert wird.   |
| Funktionswert |             | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, DATE, TIME, DT | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TOD, LTOD, DATE, Zeiten, DT, LDT | Ergebnis der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := MUX(K := "Tag_Number",
                    IN0 := "Tag_1",
                    IN1 := "Tag_2",
                    INELSE := "Tag_3");
```

Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Operand    | Wert           |                |
|------------|----------------|----------------|
| Tag_Number | 2              | 4              |
| Tag_1      | DW#16#00000000 | DW#16#00000000 |
| Tag_2      | DW#16#003E4A7D | DW#16#003E4A7D |
| Tag_3      | DW#16#FFFF0000 | DW#16#FFFF0000 |
| Tag_Result | DW#16#003E4A7D | DW#16#FFFF0000 |

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

**DEMUX: Demultiplexen**

**Beschreibung**

Mit der Anweisung "Demultiplexen" übertragen Sie den Wert des Eingangsparameters IN an einen ausgewählten Ausgangsparameter. Die Auswahl des Ausgangsparameters erfolgt abhängig vom Wert des Parameters K. Der Parameter K gibt die Nummer des Ausgangsparameters, in welchen der Wert des Eingangsparameters IN übertragen wird. Die anderen Ausgangsparameter werden nicht verändert. Die Nummerierung fängt mit OUT0 an und wird mit jedem neuen Ausgang aufsteigend fortgesetzt. Sie können maximal 32 Ausgangsparameter deklarieren.

Wenn der Wert des Parameters K größer als die Anzahl der Ausgangsparameter ist, wird der Wert des Eingangsparameters IN an den Ausgangsparameter OUTELSE übertragen.

## Syntax

Für die Anweisung "Demultiplexen" wird die folgende Syntax verwendet:

```

SCL
DEMUX(K := <Ausdruck>,
      IN := <Ausdruck>,
      OUT0 := <Operand>,
      OUT1 := <Operand>,
      OUTELSE := <Operand>)

```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Demultiplexen":

| Parameter | Deklaration | Datentyp  |  | Beschreibung   |
|-----------|-------------|---|--|--|
|           |             | S7-1200   | S7-1500  |  |
| K         | Input       | Ganzzahlen  | Ganzzahlen   | Gibt den Ausgang an, in welchen der Eingangswert (IN) kopiert wird. <ul style="list-style-type: none"> <li>• Wenn K = 0 =&gt; Parameter OUT0</li> <li>• Wenn K = 1 =&gt; Parameter OUT1, etc.</li> </ul> |
| IN        | Input       | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE, DT | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, DT, LDT | Eingangswert   |
| OUT0      | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE, DT | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, DT, LDT | Erster Ausgang   |
| OUT1      | Output      | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, TIME, TOD, DATE, DT | Binärzahlen, Ganzzahlen, Gleitpunktzahlen, Zeichenfolgen, Zeiten, TOD, LTOD, DATE, DT, LDT | Zweiter Ausgang  |

| Parameter | Deklaration | Datentyp   |  | Beschreibung   |
|-----------|-------------|--|--|--|
|           |             | S7-1200  | S7-1500  |  |
| OUTn      | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, TOD,<br>DATE, DT | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>Zeiten, TOD,<br>LTOD, DATE, DT,<br>LDT | Optionale<br>Ausgänge  |
| OUTELSE   | Output      | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>TIME, TOD,<br>DATE, DT | Binärzahlen,<br>Ganzzahlen,<br>Gleitpunktzahlen,<br>Zeichenfolgen,<br>Zeiten, TOD,<br>LTOD, DATE, DT,<br>LDT | Ausgang, in<br>welchen der Wert<br>des Eingangs IN<br>bei K > n kopiert<br>wird. |

Weitere Informationen zu den verfügbaren Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCI
DEMUX(K := "Tag_Number",
      IN := "Tag_Value",
      OUT0 := "Tag_1",
      OUT1 := "Tag_2",
      OUTELSE := "Tag_3");
    
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

Eingangswerte der Anweisung "Demultiplexen" vor dem Ausführen des Netzwerks

| Parameter | Operand    | Werte          |                |
|-----------|------------|----------------|----------------|
| K         | Tag_Number | 2              | 4              |
| IN        | Tag_Value  | DW#16#FFFFFFFF | DW#16#003E4A7D |

Ausgangswerte der Anweisung "Demultiplexen" nach dem Ausführen des Netzwerks

| Parameter | Operand | Werte          |                |
|-----------|---------|----------------|----------------|
| OUT0      | Tag_1   | unverändert    | unverändert    |
| OUT1      | Tag_2   | DW#16#FFFFFFFF | unverändert    |
| OUTELSE   | Tag_3   | unverändert    | DW#16#003E4A7D |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**Schieben und Rotieren****SHR: Rechts schieben****Beschreibung**

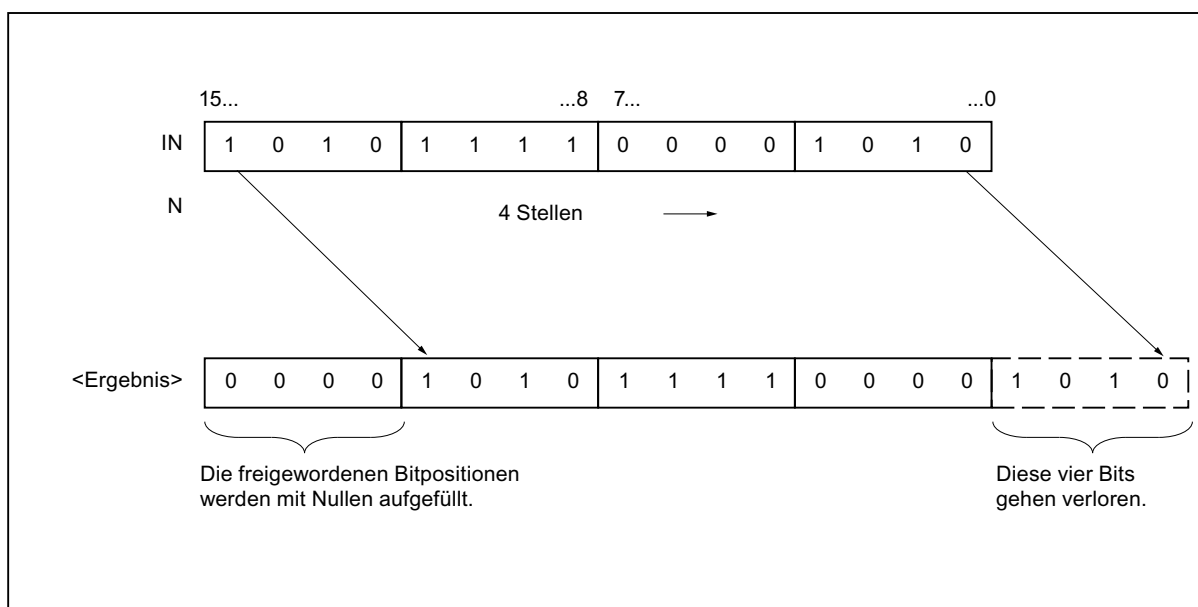
Mit der Anweisung "Rechts schieben" verschieben Sie den Inhalt des Parameters IN bitweise nach rechts und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert des Parameters IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Parameters IN um die verfügbaren Bitstellen nach rechts verschoben.

Die beim Schieben frei werdenden Bitstellen im linken Bereich des Operanden werden mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp Integer um vier Bitstellen nach rechts verschoben wird:



### Syntax

Für die Anweisung "Rechts schieben" wird die folgende Syntax verwendet:

```
SCL
SHR(IN := <Operand>,
     N := <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts schieben":

| Parameter     | Deklaration | Datentyp                 |                              | Beschreibung  |
|---------------|-------------|--------------------------|------------------------------|---|
|               |             | S7-1200                  | S7-1500                      |   |
| IN            | Input       | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | Wert, der<br>verschoben wird                                |
| N             | Input       | USINT, UINT,<br>UDINT    | USINT, UINT,<br>UDINT, ULINT | Anzahl der Bits, um<br>die der Wert (IN)<br>verschoben wird |
| Funktionswert |             | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | Ergebnis der<br>Anweisung                                   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := SHR(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand    | Wert                 |
|---------------|------------|----------------------|
| IN            | Tag_Value  | 0011 1111 1010 1111  |
| N             | Tag_Number | 3                    |
| Funktionswert | Tag_Result | 0000 0111 1111 010 1 |

Der Wert des Operanden "Tag\_Value" wird um drei Bitstellen nach rechts verschoben. Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

**SHL: Links schieben****Beschreibung**

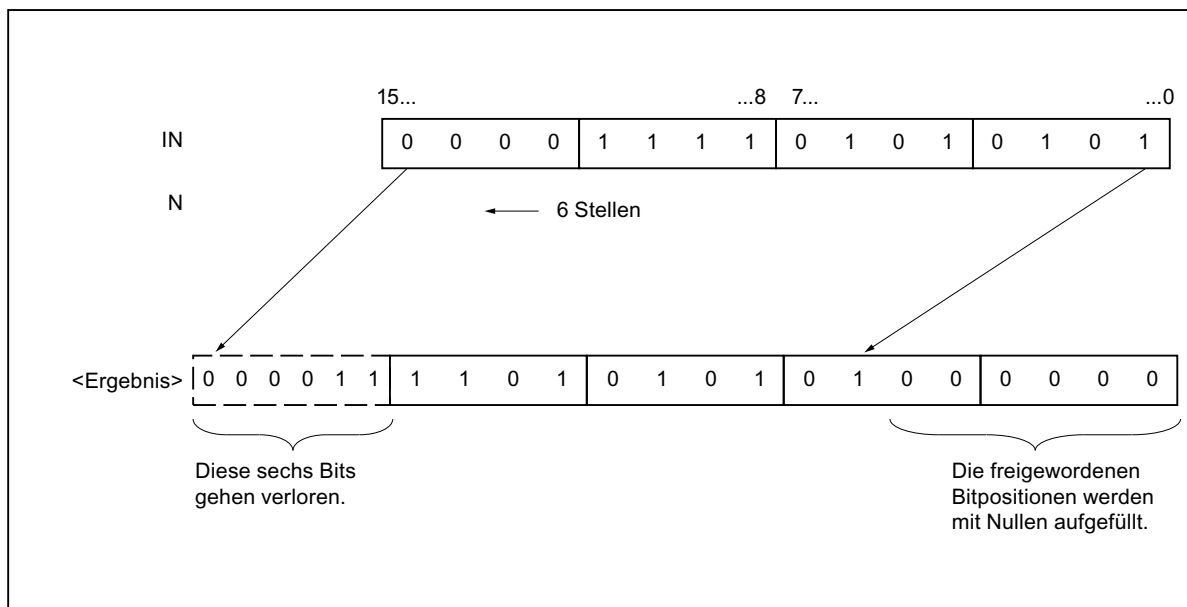
Mit der Anweisung "Links schieben" verschieben Sie den Inhalt des Parameters IN bitweise nach links und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert verschoben wird.

Wenn der Wert am Parameter N "0" ist, wird der Wert des Parameters IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Wert des Parameters IN um die verfügbaren Bitstellen nach links verschoben.

Die beim Schieben frei werdenden Bitstellen werden im Ergebniswert mit Nullen aufgefüllt.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp WORD um sechs Bitstellen nach links verschoben wird:



### Syntax

Für die Anweisung "Links schieben" wird die folgende Syntax verwendet:

```
SCL
SHL(IN := <Operand>,
     N := <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links schieben":

| Parameter     | Deklaration | Datentyp                 |                              | Beschreibung  |
|---------------|-------------|--------------------------|------------------------------|---|
|               |             | S7-1200                  | S7-1500                      |   |
| IN            | Input       | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | Wert, der<br>verschoben wird                                |
| N             | Input       | USINT, UINT,<br>UDINT    | USINT, UINT,<br>UDINT, ULINT | Anzahl der Bits, um<br>die der Wert (IN)<br>verschoben wird |
| Funktionswert |             | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | Ergebnis der<br>Anweisung                                   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := SHL(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand    | Wert                |
|---------------|------------|---------------------|
| IN            | Tag_Value  | 0011 1111 1010 1111 |
| N             | Tag_Number | 4                   |
| Funktionswert | Tag_Result | 1111 1010 1111 0000 |

Der Wert des Operanden "Tag\_Value" wird um vier Bitstellen nach links verschoben. Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.



**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

Operatoren und Operatorenrangfolge (Seite 1370)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

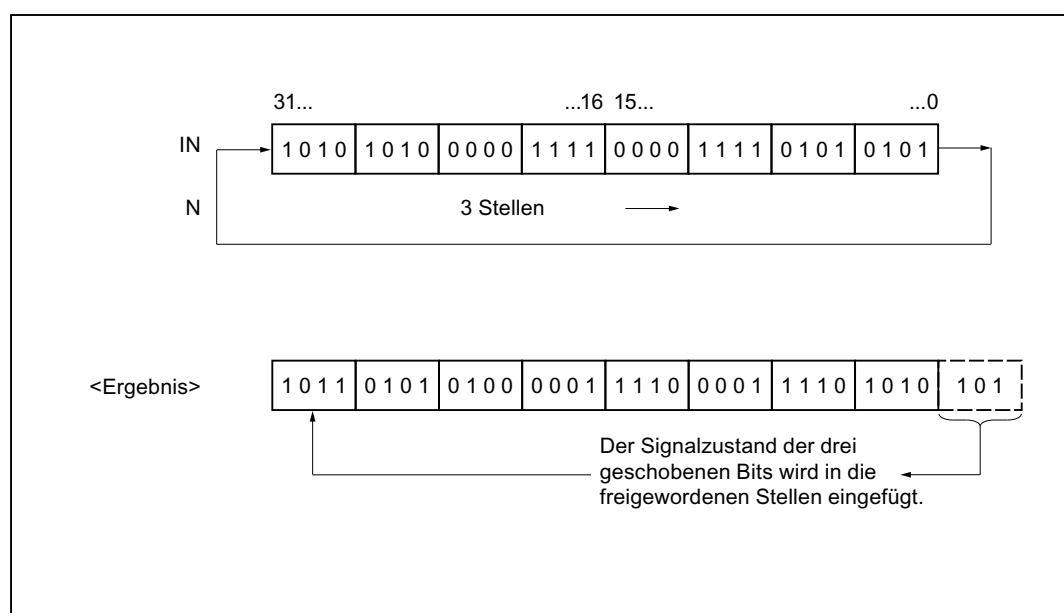
**ROR: Rechts rotieren****Beschreibung**

Mit der Anweisung "Rechts rotieren" rotieren Sie den Inhalt des Parameters IN bitweise nach rechts und weisen das Ergebnis dem angegebenen Operanden zu. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach rechts rotiert wird:

**Syntax**

Für die Anweisung "Rechts rotieren" wird die folgende Syntax verwendet:

**SCL**

```
ROR (IN := <Operand> ,
```

**SCL**

```
N := <Operand>
```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Rechts rotieren":

| Parameter     | Deklaration | Datentyp                 |                              | Beschreibung   |
|---------------|-------------|--------------------------|------------------------------|--|
|               |             | S7-1200                  | S7-1500                      |  |
| IN            | Input       | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | Wert, der rotiert wird                                   |
| N             | Input       | USINT, UINT,<br>UDINT    | USINT, UINT,<br>UDINT, ULINT | Anzahl der Bitstellen, um die der Wert (IN) rotiert wird |
| Funktionswert |             | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | Ergebnis der Anweisung                                   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

**SCL**

```
"Tag_Result" := ROR(IN := "Tag_Value",
                    N := "Tag_Number");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand    | Wert                |
|---------------|------------|---------------------|
| IN            | Tag_Value  | 0000 1111 1001 0101 |
| N             | Tag_Number | 5                   |
| Funktionswert | Tag_Result | 1010 1000 0111 1100 |

Der Inhalt des Operanden "Tag\_Value" wird um fünf Bitstellen nach rechts rotiert. Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

**Siehe auch**

- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## ROL: Links rotieren

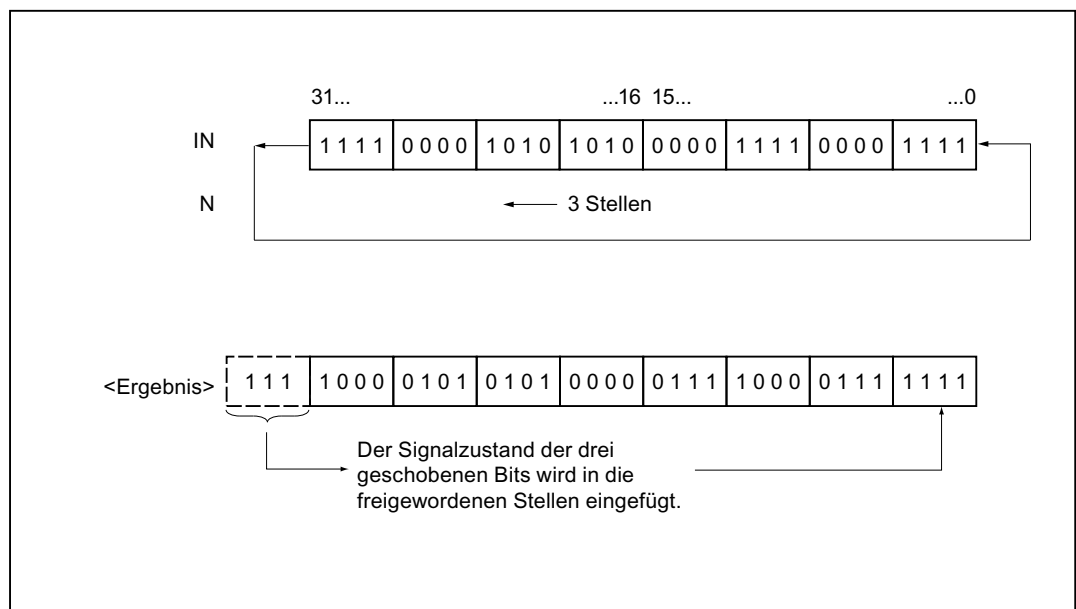
### Beschreibung

Mit der Anweisung "Links rotieren" rotieren Sie den Inhalt des Parameters IN bitweise nach links und liefern ihn als Funktionswert zurück. Mit dem Parameter N legen Sie die Anzahl der Bitstellen fest, um die der angegebene Wert rotiert wird. Die beim Rotieren frei werdenden Bitstellen werden mit den hinausgeschobenen Bitstellen aufgefüllt.

Wenn der Wert am Parameter N "0" ist, wird der Wert am Eingang IN als Ergebnis ausgegeben.

Wenn der Wert am Parameter N größer als die Anzahl der verfügbaren Bitstellen ist, wird der Operandenwert am Eingang IN trotzdem um die angegebene Anzahl der Bitstellen rotiert.

Das folgende Bild zeigt, wie der Inhalt eines Operanden vom Datentyp DWORD um drei Stellen nach links rotiert wird:



### Syntax

Für die Anweisung "Links rotieren" wird die folgende Syntax verwendet:

#### SCL

```
ROL (IN := <Operand>,
     N := <Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Links rotieren":

| Parameter     | Deklaration | Datentyp                 |                              | Beschreibung   |
|---------------|-------------|--------------------------|------------------------------|--|
|               |             | S7-1200                  | S7-1500                      |  |
| IN            | Input       | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | Wert, der rotiert wird                                   |
| N             | Input       | USINT, UINT,<br>UDINT    | USINT, UINT,<br>UDINT, ULINT | Anzahl der Bitstellen, um die der Wert (IN) rotiert wird |
| Funktionswert |             | Bitfolgen,<br>Ganzzahlen | Bitfolgen,<br>Ganzzahlen     | Ergebnis der Anweisung                                   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```

SCL
"Tag_Result" := ROL(IN := "Tag_Value",
                    N := "Tag_Number");
    
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter     | Operand    | Wert                |
|---------------|------------|---------------------|
| IN            | Tag_Value  | 1010 1000 1111 0110 |
| N             | Tag_Number | 5                   |
| Funktionswert | Tag_Result | 0001 1110 1101 0101 |

Der Inhalt des Operanden "Tag\_Value" wird um fünf Bitstellen nach links rotiert. Das Ergebnis der Anweisung wird am Operanden "Tag\_Result" als Funktionswert zurückgeliefert.

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- Operatoren und Operatorenrangfolge (Seite 1370)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## Weitere Anweisungen

### DRUM: Schrittschaltwerk realisieren

#### Beschreibung

Mit der Anweisung "Schrittschaltwerk realisieren" belegen Sie die programmierten Ausgabebits (OUT1 bis OUT16) und das Ausgabewort (OUT\_WORD) mit den programmierten Werten des Parameters OUT\_VAL des entsprechenden Schritts. Der jeweilige Schritt muss dabei die Bedingungen der am Parameter S\_MASK programmierten Freigabemaske erfüllen, während die Anweisung auf diesem Schritt bleibt. Die Anweisung geht zum nächsten Schritt über, wenn das Ereignis für den Schritt wahr wird und die programmierte Zeit für den aktuellen Schritt abläuft oder wenn der Wert am Parameter JOG von "0" auf "1" wechselt. Die Anweisung wird zurückgesetzt, wenn der Signalzustand am Parameter RESET auf "1" wechselt. Dadurch wird der aktuelle Schritt gleich dem voreingestellten Schritt (DSP) gesetzt.

Die Verweilzeit auf einem Schritt wird durch das Produkt aus der voreingestellten Zeitbasis (DTBP) und dem voreingestellten Zählwert (S\_PRESET) für jeden Schritt festgelegt. Zu Beginn eines neuen Schritts wird dieser berechnete Wert in den Parameter DCC geladen, der die verbleibende Zeit für den aktuellen Schritt enthält. Wenn beispielsweise der Wert am Parameter DTBP "2" und der voreingestellte Wert für den ersten Schritt "100" (100 ms) ist, dann liefert der Parameter DCC den Wert "200" (200 ms).

Ein Schritt kann mit einem Zeitwert, mit einem Ereignis oder mit beiden programmiert werden. Schritte, die mit einem Ereignisbit und dem Zeitwert "0" programmiert sind, gehen zum nächsten Schritt über, sobald der Signalzustand des Ereignisbits "1" ist. Schritte, die nur mit einem Zeitwert programmiert sind, starten die Zeit sofort. Schritte, die mit einem Ereignisbit und einem Zeitwert größer als "0" programmiert sind, starten die Zeit, wenn der Signalzustand des Ereignisbits "1" ist. Die Ereignisbits werden mit dem Signalzustand "1" initialisiert.

Wenn sich das Schrittschaltwerk auf dem letzten programmierten Schritt (LST\_STEP) befindet und die Zeit für diesen Schritt abgelaufen ist, dann wird der Signalzustand am Parameter Q auf "1" gesetzt, andernfalls wird er auf "0" gesetzt. Wenn der Parameter Q gesetzt ist, dann verweilt die Anweisung bis zum Zurücksetzen auf dem Schritt.

In der konfigurierbaren Maske (S\_MASK) können Sie die einzelnen Bits in dem Ausgabewort (OUT\_WORD) auswählen und die Ausgabebits (OUT1 bis OUT16) über die Ausgabewerte (OUT\_VAL) setzen bzw. rücksetzen. Wenn ein Bit der konfigurierbaren Maske im Signalzustand "1" ist, dann setzt oder rücksetzt der Wert OUT\_VAL das entsprechende Bit. Ist der Signalzustand eines Bits der konfigurierbaren Maske "0", dann wird das entsprechende Bit nicht verändert. Alle Bits der konfigurierbaren Maske aller 16 Schritte werden mit dem Signalzustand "1" initialisiert.

Das Ausgabebit am Parameter OUT1 entspricht dem niederwertigsten Bit des Ausgabeworts (OUT\_WORD). Das Ausgabebit am Parameter OUT16 entspricht dem höchstwertigen Bit des Ausgabeworts (OUT\_WORD).

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

**Syntax**

Für die Anweisung "Schrittschaltwerk realisieren" wird die folgende Syntax verwendet:

```

SCL
<Instanz>(RESET := <Operand>,
          JOG := <Operand>,
          DRUM_EN := <Operand>,
          LST_STEP := <Operand>,
          EVENT1 - 16 := <Operand>,
          OUT1 - 16 => <Operand>,
          Q => <Operand>,
          OUT_WORD => <Operand>,
          ERR_CODE => <Operand>)
    
```

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "Schrittschaltwerk realisieren":

| Parameter               | Deklaration | Datentyp | Beschreibung  |
|-------------------------|-------------|----------|---|
| RESET                   | Input       | BOOL     | Der Signalzustand "1" kennzeichnet eine Rücksetzbedingung.  |
| JOG                     | Input       | BOOL     | Wechselt der Signalzustand von "0" auf "1", wechselt die Anweisung zum nächsten Schritt.                        |
| DRUM_EN                 | Input       | BOOL     | Der Signalzustand "1" lässt das Schrittschaltwerk entsprechend dem Ereignis und den Zeitkriterien weiterzählen. |
| LST_STEP                | Input       | BYTE     | Nummer des zuletzt programmierten Schritts.   |
| EVENT(i),<br>1 ≤ i ≤ 16 | Input       | BOOL     | Ereignisbit (i); Anfänglicher Signalzustand ist "1".  |
| OUT(j),<br>1 ≤ j ≤ 16   | Output      | BOOL     | Ausgabebit (j)  |
| Q                       | Output      | BOOL     | Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.                           |
| OUT_WORD                | Output      | WORD     | Wortadresse, in die das Schrittschaltwerk die Ausgabewerte schreibt.  |
| ERR_CODE                | Output      | WORD     | Fehlerinformation   |

| Parameter | Deklaration | Datentyp      | Beschreibung   |
|-----------|-------------|---------------|--|
| JOG_HIS   | Static      | BOOL          | Verlaufsbit zum Parameter JOG  |
| EOD       | Static      | BOOL          | Der Signalzustand "1" zeigt an, dass die Zeit für den letzten Schritt abgelaufen ist.              |
| DSP       | Static      | BYTE          | Voreingestellter Schritt des Schrittschaltwerks  |
| DSC       | Static      | BYTE          | Aktueller Schritt des Schrittschaltwerks   |
| DCC       | Static      | DWORD         | Aktueller Zählwert des Schrittschaltwerks  |
| DTBP      | Static      | WORD          | Voreingestellte Zeitbasis des Schrittschaltwerks   |
| PREV_TIME | Static      | DWORD         | Vorherige Systemzeit   |
| S_PRESET  | Static      | ARRAY of WORD | Voreingestellter Zählwert für jeden Schritt [1 bis 16]; 1 Takt = 1 ms.                             |
| OUT_VAL   | Static      | ARRAY of BOOL | Ausgabewerte für jeden Schritt [1 bis 16, 0 bis 15].   |
| S_MASK    | Static      | ARRAY of BOOL | Konfigurierbare Maske für jeden Schritt [1 bis 16, 0 bis 15]. Anfängliche Signalzustände sind "1". |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

## Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

| ERR_CODE* | Erläuterung   |
|-----------|---|
| W#16#0000 | Kein Fehler   |
| W#16#000B | Der Wert am Parameter LST_STEP ist kleiner als 1 oder größer als 16.                        |
| W#16#000C | Der Wert am Parameter DSC ist kleiner als 1 oder größer als der Wert am Parameter LST_STEP. |
| W#16#000D | Der Wert am Parameter DSP ist kleiner als 1 oder größer als der Wert am LST_STEP.           |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## DCAT: Diskreter Steuerungszeitalarm

### Beschreibung

Mit der Anweisung "Diskreter Steuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem der Parameter CMD den Befehl zum Öffnen oder Schließen erteilt. Die Zeit wird akkumuliert, bis die voreingestellte Zeit (PT) überschritten wird oder die Information empfangen wird, dass das Gerät innerhalb der vorgeschriebenen Zeit geöffnet bzw. geschlossen wurde (O\_FB bzw. C\_FB). Wird die voreingestellte Zeit überschritten, bevor die Information über das Öffnen oder Schließen des Geräts empfangen wird, dann wird der entsprechende Alarm eingeschaltet. Wechselt der Signalzustand des Befehlseingangs vor der voreingestellten Zeit, dann wird die Zeit neu gestartet.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

Auf die Eingangsbedingungen hat die Anweisung "Diskreter Steuerungszeitalarm" die folgenden Reaktionen:

- Wenn der Signalzustand des Parameters CMD von "0" nach "1" wechselt, werden die Signalzustände der Parameter Q, CMD\_HIS, ET (nur wenn  $ET < PT$  ist), OA und CA wie folgt beeinflusst:
  - Die Parameter Q und CMD\_HIS werden auf "1" gesetzt
  - Die Parameter ET, OA und CA werden auf "0" zurückgesetzt
- Bei einem Wechsel im Signalzustand am Parameter CMD von "1" nach "0" werden die Parameter Q, ET (nur wenn  $ET < PT$  ist), OA, CA und CMD\_HIS auf "0" zurückgesetzt.
- Wenn an den Parametern CMD und CMD\_HIS den Signalzustand "1" anliegt und der Parameter O\_FB auf "0" gesetzt ist, wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert am Parameter ET addiert. Überschreitet der Wert des Parameters ET den Wert des Parameters PT, dann wird der Signalzustand am Parameter OA auf "1" gesetzt. Wenn der Wert des Parameters ET den Wert am Parameter PT nicht überschreitet, wird der Signalzustand am Parameter OA auf "0" zurückgesetzt. Der Wert am Parameter CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.



- Wenn die Signalzustände der Parameter CMD, CMD\_HIS und O\_FB auf "1" gesetzt sind und der Parameter C\_FB "0" liefert, dann wird der Signalzustand des Parameters OA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters O\_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD\_HIS und C\_FB "0" liefern, dann wird der Unterschied der Zeit (ms) seit der letzten Bearbeitung der Anweisung zu dem Wert des Parameters ET addiert. Wenn der Wert des Parameters ET den Wert des Parameters PT überschreitet, dann wird der Signalzustand des Parameters CA auf "1" gesetzt. Ist der Wert am Parameter PT nicht überschritten, liefert der Parameter CA den Signalzustand "0". Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter CMD, CMD\_HIS und O\_FB den Signalzustand "0" liefern und der Parameter C\_FB auf "1" gesetzt ist, dann wird der Parameter CA auf "0" gesetzt. Der Wert des Parameters ET wird auf den Wert des Parameters PT gesetzt. Wenn der Signalzustand des Parameters C\_FB auf "0" wechselt, wird der Alarm bei der nächsten Bearbeitung der Anweisung gesetzt. Der Wert des Parameters CMD\_HIS wird auf den Wert des Parameters CMD gesetzt.
- Wenn die Parameter O\_FB und C\_FB gleichzeitig den Signalzustand "1" liefern, werden die Signalzustände beider Alarmausgänge auf "1" gesetzt.

Die Anweisung "Diskreter Steuerungszeitalarm" liefert keine Fehlerinformationen.

## Syntax

Für die Anweisung "Diskreter Steuerungszeitalarm" wird die folgende Syntax verwendet:

### SCL

```
<Instanz>(CMD := <Operand>,
          O_FB := <Operand>,
          C_FB := <Operand>,
          Q => <Operand>,
          OA => <Operand>,
          CA => <Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung:

| Parameter | Deklaration | Datentyp | Beschreibung  |
|-----------|-------------|----------|---|
| CMD       | Input       | BOOL     | Der Signalzustand "0" gibt den Befehl "Schließen" an.<br>Der Signalzustand "1" gibt den Befehl "Öffnen" an. |
| O_FB      | Input       | BOOL     | Feedback-Eingang beim Öffnen  |
| C_FB      | Input       | BOOL     | Feedback-Eingang beim Schließen   |
| Q         | Output      | BOOL     | Zeigt den Status des Parameters CMD   |
| OA        | Output      | BOOL     | Alarmausgang beim Öffnen  |

| Parameter | Deklaration | Datentyp | Beschreibung                                 |
|-----------|-------------|----------|--|
| CA        | Output      | BOOL     | Alarmausgang beim Schließen                  |
| ET        | Static      | DINT     | Aktuelle, abgelaufene Zeit; ein Takt = 1 ms. |
| PT        | Static      | DINT     | Voreingestellter Zeitwert; ein Takt = 1 ms.  |
| PREV_TIME | Static      | DWORD    | Vorherige Systemzeit                         |
| CMD_HIS   | Static      | BOOL     | Verlaufsbit zu CMD                           |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

### Beispiel

Im folgenden Beispiel wechselt der Parameter CMD von "0" auf "1". Nach der Ausführung der Anweisung wird der Parameter Q auf "1" gesetzt und die beiden Alarmausgänge OA und CA erhalten den Signalzustand "0". Der Parameter CMD\_HIS des Instanz-Datenbausteins wird auf den Signalzustand "1" gesetzt und der Parameter ET auf "0" zurückgesetzt.

---

#### Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

---

#### SCI

```
"DCAT_DB" (CMD := "Tag_Input_CMD",
           O_FB := "Tag_Input_O_FB",
           C_FB := "Tag_Input_C_FB",
           Q => "Tag_Output_Q",
           OA => "Tag_Output_OA",
           CA => "Tag_Output_CA");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

#### Vor der Bearbeitung

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand        | Wert  |
|-----------|----------------|-------|
| CMD       | Tag_Input_CMD  | TRUE  |
| O_FB      | Tag_Input_O_FB | FALSE |
| C_FB      | Tag_Input_C_FB | FALSE |
| Q         | Tag_Output_Q   | FALSE |
| OA        | Tag_Output_OA  | FALSE |
| CA        | Tag_Output_CA  | FALSE |

Im Instanz-Datenbaustein "DCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| ET        | DBD4    | L#12  |
| PT        | DBD8    | L#222 |
| CMD_HIS   | DBX16.0 | FALSE |

### Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand       | Wert  |
|-----------|---------------|-------|
| Q         | Tag_Output_Q  | TRUE  |
| OA        | Tag_Output_OA | FALSE |
| CA        | Tag_Output_CA | FALSE |

Im Instanz-Datenbaustein "DCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert |
|-----------|---------|------|
| ET        | DBD4    | L#0  |
| CMD_HIS   | DBX16.0 | TRUE |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## MCAT: Motorsteuerungszeitalarm

### Beschreibung

Mit der Anweisung "Motorsteuerungszeitalarm" akkumulieren Sie die Zeit ab dem Zeitpunkt, ab dem einer der Befehlseingänge (Öffnen oder Schließen) eingeschaltet wird. Die Zeit wird akkumuliert, bis die voreingestellte Zeit überschritten wird oder der entsprechende Feedback-Eingang anzeigt, dass das Gerät die angeforderte Operation innerhalb der vorgeschriebenen Zeit ausgeführt hat. Wird die voreingestellte Zeit überschritten, bevor Feedback empfangen wird, dann wird der entsprechende Alarm ausgelöst.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

### Ausführung der Anweisung "Motorsteuerungszeitalarm"

Die folgende Tabelle zeigt die Reaktionen der Anweisung "Motorsteuerungszeitalarm" auf die verschiedenen Eingangsbedingungen:

| Eingangsparameter |        |        |        |        |        |       |       | Ausgangsparameter |    |    |    |     |        |        |   |                   |
|-------------------|--------|--------|--------|--------|--------|-------|-------|-------------------|----|----|----|-----|--------|--------|---|-------------------|
| ET                | O_H IS | C_H IS | O_C MD | C_C MD | S_C MD | O_F B | C_F B | OO                | CO | OA | CA | ET  | O_H IS | C_H IS | Q | Zustand           |
| X                 | 1      | 1      | X      | X      | X      | X     | X     | 0                 | 0  | 1  | 1  | PT  | 0      | 0      | 0 | Alarm             |
| X                 | X      | X      | X      | X      | X      | 1     | 1     | 0                 | 0  | 1  | 1  | PT  | 0      | 0      | 0 | Alarm             |
| X                 | X      | X      | X      | X      | 1      | X     | X     | 0                 | 0  | 0  | 0  | X   | 0      | 0      | 1 | Stop              |
| X                 | X      | X      | 1      | 1      | X      | X     | X     | 0                 | 0  | 0  | 0  | X   | 0      | 0      | 1 | Stop              |
| X                 | 0      | X      | 1      | 0      | 0      | X     | X     | 1                 | 0  | 0  | 0  | 0   | 1      | 0      | 1 | Öffnen starten    |
| <PT               | 1      | 0      | X      | 0      | 0      | 0     | X     | 1                 | 0  | 0  | 0  | INC | 1      | 0      | 1 | Öffnen            |
| X                 | 1      | 0      | X      | 0      | 0      | 1     | 0     | 0                 | 0  | 0  | 0  | PT  | 1      | 0      | 1 | Geöffnet          |
| >= PT             | 1      | 0      | X      | 0      | 0      | 0     | X     | 0                 | 0  | 1  | 0  | PT  | 1      | 0      | 0 | Alarm öffnen      |
| X                 | X      | 0      | 0      | 1      | 0      | X     | X     | 0                 | 1  | 0  | 0  | 0   | 0      | 1      | 1 | Schließen starten |
| < PT              | 0      | 1      | 0      | X      | 0      | X     | 0     | 0                 | 1  | 0  | 0  | INC | 0      | 1      | 1 | Schließen         |
| X                 | 0      | 1      | 0      | X      | 0      | 0     | 1     | 0                 | 0  | 0  | 0  | PT  | 0      | 1      | 1 | Geschlossen       |
| >= PT             | 0      | 1      | 0      | X      | 0      | X     | 0     | 0                 | 0  | 0  | 1  | PT  | 0      | 1      | 0 | Alarm schließen   |
| X                 | 0      | 0      | 0      | 0      | 0      | X     | X     | 0                 | 0  | 0  | 0  | X   | 0      | 0      | 1 | Gestoppt          |

Legende:

|       |  |
|-------|--|
| INC   | Unterschied der Zeit (ms) seit der letzten Bearbeitung des FB zu ET addieren |
| PT    | PT wird auf den gleichen Wert wie ET gesetzt                                 |
| X     | Nicht anwendbar  |
| < PT  | ET < PT  |
| >= PT | ET >= PT   |

Wenn die Eingangsparameter O\_HIS und C\_HIS beide den Signalzustand "1" führen, werden sie sofort auf den Signalzustand "0" gesetzt. In diesem Fall ist die letzte Zeile in der oben genannten Tabelle (X) gültig. Da es aus diesem Grund nicht möglich ist zu prüfen, ob die Eingangsparameter O\_HIS und C\_HIS den Signalzustand "1" führen, werden in diesem Fall die Ausgangsparameter wie folgt gesetzt:

- OO = FALSE
- CO = FALSE
- OA = FALSE
- CA = FALSE
- ET = PT
- Q = TRUE

## Syntax

Für die Anweisung "Motorsteuerungszeitalarm" wird die folgende Syntax verwendet:

```
SCL
<Instanz> (O_CMD := <Operand>,
           C_CMD := <Operand>,
           S_CMD := <Operand>,
           O_FB := <Operand>,
           C_FB := <Operand>,
           OO => <Operand>,
           CO => <Operand>,
           OA => <Operand>,
           CA => <Operand>,
           Q => <Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Motorsteuerungszeitalarm":

| Parameter | Deklaration | Datentyp | Beschreibung   |
|-----------|-------------|----------|--|
| O_CMD     | Input       | BOOL     | Befehlseingabe "Öffnen"                              |
| C_CMD     | Input       | BOOL     | Befehlseingabe "Schließen"                           |
| S_CMD     | Input       | BOOL     | Befehlseingabe "Stoppen"                             |
| O_FB      | Input       | BOOL     | Feedback-Eingang beim Öffnen                         |
| C_FB      | Input       | BOOL     | Feedback-Eingang beim Schließen                      |
| OO        | Output      | BOOL     | Ausgang "Öffnen"                                     |
| CO        | Output      | BOOL     | Ausgang "Schließen"                                  |
| OA        | Output      | BOOL     | Alarmausgang beim Öffnen                             |
| CA        | Output      | BOOL     | Alarmausgang beim Schließen                          |
| Q         | Output      | BOOL     | Der Signalzustand "0" zeigt eine Fehlerbedingung an. |
| ET        | Static      | DINT     | Aktuelle, abgelaufene Zeit; ein Takt = 1 ms          |
| PT        | Static      | DINT     | Voreingestellter Zeitwert; ein Takt = 1 ms           |
| PREV_TIME | Static      | DWORD    | Vorherige Systemzeit                                 |
| O_HIS     | Static      | BOOL     | Verlaufsbit "Öffnen"                                 |
| C_HIS     | Static      | BOOL     | Verlaufsbit "Schließen"                              |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

**Beispiel**

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

**Hinweis**

Statische Parameter können Sie im Datenbaustein initialisieren.

**SCL**

```
"MCAT_DB" (O_CMD := "Tag_Input_O_CMD",
           C_CMD := "Tag_Input_C_CMD",
           S_CMD := "Tag_Input_S_CMD",
           O_FB := "Tag_Input_O_FB",
           C_FB := "Tag_Input_C_FB",
           OO => "Tag_OutputOpen",
           CO => "Tag_OutputClosed",
           OA => "Tag_Output_OA",
           CA => "Tag_Output_CA",
           Q => "Tag_Output_Q");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

**Vor der Bearbeitung**

Für die Eingangs- und Ausgangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand          | Wert  |
|-----------|------------------|-------|
| O_CMD     | Tag_Input_O_CMD  | TRUE  |
| C_CMD     | Tag_Input_C_CMD  | FALSE |
| S_CMD     | Tag_Input_S_CMD  | FALSE |
| O_FB      | Tag_Input_O_FB   | FALSE |
| C_FB      | Tag_Input_C_FB   | FALSE |
| OO        | Tag_OutputOpen   | FALSE |
| CO        | Tag_OutputClosed | FALSE |
| OA        | Tag_Output_OA    | FALSE |
| CA        | Tag_Output_CA    | FALSE |
| Q         | Tag_Output_Q     | FALSE |

Im Instanz-Datenbaustein "MCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert |
|-----------|---------|------|
| ET        | DBD4    | L#2  |
| PT        | DBD8    | L#22 |

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| O_HIS     | DBX16.0 | TRUE  |
| C_HIS     | DBX16.1 | FALSE |

### Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand          | Wert  |
|-----------|------------------|-------|
| OO        | Tag_OutputOpen   | TRUE  |
| CO        | Tag_OutputClosed | FALSE |
| OA        | Tag_Output_OA    | FALSE |
| CA        | Tag_Output_CA    | FALSE |
| Q         | Tag_Output_Q     | TRUE  |

Im Instanz-Datenbaustein "MCAT\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert  |
|-----------|---------|-------|
| ET        | DBD4    | L#0   |
| O_HIS     | DBX16.0 | TRUE  |
| CMD_HIS   | DBX16.1 | FALSE |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## IMC: Eingabebits mit den Bits einer Maske vergleichen

### Beschreibung

Mit der Anweisung "Eingabebits mit den Bits einer Maske vergleichen" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN\_BIT0 bis IN\_BIT15) mit dem entsprechenden Bit einer Maske. Sie können maximal 16 Schritte mit Masken programmieren. Der Wert des Parameters IN\_BIT0 wird mit dem Wert der Maske CMP\_VAL[x,0] verglichen, wobei "x" die Schrittnummer angibt. Die Schrittnummer der Maske, die für den Vergleich verwendet wird, legen Sie am Parameter CMP\_STEP fest. Auf die gleiche Weise werden alle programmierten Werte verglichen. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE.

Wird im Vergleich eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt. Andernfalls wird der Parameter OUT auf "0" gesetzt.

Wenn der Wert des Parameters CMP\_STEP größer als 15 ist, wird die Anweisung nicht ausgeführt. Am Parameter ERR\_CODE wird eine Fehlermeldung ausgegeben.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelninstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

### Syntax

Für die Anweisung "Eingabebits mit den Bits einer Maske vergleichen" wird die folgende Syntax verwendet:

```

SCL
<Instanz>(IN_BIT0 - 15 := <Operand>,
          CMP_STEP := <Operand>,
          OUT => <Operand>,
          ERR_CODE => <Operand>)
    
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Eingabebits mit den Bits einer Maske vergleichen":

| Parameter | Deklaration | Datentyp | Beschreibung  |
|-----------|-------------|----------|---|
| IN_BIT0   | Input       | BOOL     | Eingabebit 0 wird mit Bit 0 der Maske verglichen.   |
| IN_BIT1   | Input       | BOOL     | Eingabebit 1 wird mit Bit 1 der Maske verglichen.   |
| IN_BIT2   | Input       | BOOL     | Eingabebit 2 wird mit Bit 2 der Maske verglichen.   |
| IN_BIT3   | Input       | BOOL     | Eingabebit 3 wird mit Bit 3 der Maske verglichen.   |
| IN_BIT4   | Input       | BOOL     | Eingabebit 4 wird mit Bit 4 der Maske verglichen.   |
| IN_BIT5   | Input       | BOOL     | Eingabebit 5 wird mit Bit 5 der Maske verglichen.   |
| IN_BIT6   | Input       | BOOL     | Eingabebit 6 wird mit Bit 6 der Maske verglichen.   |
| IN_BIT7   | Input       | BOOL     | Eingabebit 7 wird mit Bit 7 der Maske verglichen.   |
| IN_BIT8   | Input       | BOOL     | Eingabebit 8 wird mit Bit 8 der Maske verglichen.   |
| IN_BIT9   | Input       | BOOL     | Eingabebit 9 wird mit Bit 9 der Maske verglichen.   |
| IN_BIT10  | Input       | BOOL     | Eingabebit 10 wird mit Bit 10 der Maske verglichen. |
| IN_BIT11  | Input       | BOOL     | Eingabebit 11 wird mit Bit 11 der Maske verglichen. |



| Parameter | Deklaration | Datentyp      | Beschreibung   |
|-----------|-------------|---------------|--|
| IN_BIT12  | Input       | BOOL          | Eingabebit 12 wird mit Bit 12 der Maske verglichen.  |
| IN_BIT13  | Input       | BOOL          | Eingabebit 13 wird mit Bit 13 der Maske verglichen.  |
| IN_BIT14  | Input       | BOOL          | Eingabebit 14 wird mit Bit 14 der Maske verglichen.  |
| IN_BIT15  | Input       | BOOL          | Eingabebit 15 wird mit Bit 15 der Maske verglichen.  |
| CMP_STEP  | Input       | BYTE          | Schrittnummer der Maske, mit der verglichen wird.  |
| OUT       | Output      | BOOL          | Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde.<br>Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.                  |
| ERR_CODE  | Output      | WORD          | Fehlerinformation  |
| CMP_VAL   | Static      | ARRAY OF WORD | Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

## Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE ausgegeben werden:

| Fehlercode*<br>(W#16#...) | Erläuterung                                       |
|---------------------------|---|
| 0000                      | Kein Fehler                                       |
| 000A                      | Der Wert am Parameter CMP_STEP ist größer als 15. |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## SMC: Matrixscanner

### Beschreibung

Mit der Anweisung "Matrixscanner" vergleichen Sie den Signalzustand von bis zu 16 programmierten Eingabebits (IN\_BIT0 bis IN\_BIT15) mit den entsprechenden Bits der Vergleichsmasken zu jedem Schritt. Die Bearbeitung beginnt mit Schritt 1 und wird bis zum letzten programmierten Schritt (LAST) fortgesetzt bzw. bis eine Entsprechung gefunden wird. Das Eingabebit des Parameters IN\_BIT0 wird mit dem Wert der Maske CMP\_VAL[x,0] verglichen, wobei es sich bei "x" um die Schrittnummer handelt. Auf die gleiche Weise werden alle programmierten Werte verglichen. Wird eine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "1" gesetzt und die Schrittnummer mit der entsprechenden Maske in den Parameter OUT\_STEP geschrieben. Nicht programmierte Eingabebits oder nicht programmierte Bits der Maske haben den voreingestellten Signalzustand FALSE. Haben mehrere Schritte eine entsprechende Maske, dann wird nur die erste gefundene Entsprechung in den Parameter OUT\_STEP angegeben. Wird keine Entsprechung gefunden, dann wird der Signalzustand des Parameters OUT auf "0" gesetzt. In diesem Fall ist der Wert am Parameter OUT\_STEP um "1" größer als der Wert am Parameter LAST.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelinstanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

### Syntax

Für die Anweisung "Matrixscanner" wird die folgende Syntax verwendet:

```
SCL
<Instanz>(IN_BIT0 - 15 := <Operand>,
          OUT => <Operand>,
          OUT_STEP => <Operand>,
          ERR_CODE => <Operand>)
```

### Syntax

Die folgende Tabelle zeigt die Parameter der Anweisung "Matrixscanner":

| Parameter | Deklaration | Datentyp | Beschreibung                                      |
|-----------|-------------|----------|---|
| IN_BIT0   | Input       | BOOL     | Eingabebit 0 wird mit Bit 0 der Maske verglichen. |
| IN_BIT1   | Input       | BOOL     | Eingabebit 1 wird mit Bit 1 der Maske verglichen. |
| IN_BIT2   | Input       | BOOL     | Eingabebit 2 wird mit Bit 2 der Maske verglichen. |

| Parameter | Deklaration | Datentyp      | Beschreibung   |
|-----------|-------------|---------------|--|
| IN_BIT3   | Input       | BOOL          | Eingabebit 3 wird mit Bit 3 der Maske verglichen.  |
| IN_BIT4   | Input       | BOOL          | Eingabebit 4 wird mit Bit 4 der Maske verglichen.  |
| IN_BIT5   | Input       | BOOL          | Eingabebit 5 wird mit Bit 5 der Maske verglichen.  |
| IN_BIT6   | Input       | BOOL          | Eingabebit 6 wird mit Bit 6 der Maske verglichen.  |
| IN_BIT7   | Input       | BOOL          | Eingabebit 7 wird mit Bit 7 der Maske verglichen.  |
| IN_BIT8   | Input       | BOOL          | Eingabebit 8 wird mit Bit 8 der Maske verglichen.  |
| IN_BIT9   | Input       | BOOL          | Eingabebit 9 wird mit Bit 9 der Maske verglichen.  |
| IN_BIT10  | Input       | BOOL          | Eingabebit 10 wird mit Bit 10 der Maske verglichen.  |
| IN_BIT11  | Input       | BOOL          | Eingabebit 11 wird mit Bit 11 der Maske verglichen.  |
| IN_BIT12  | Input       | BOOL          | Eingabebit 12 wird mit Bit 12 der Maske verglichen.  |
| IN_BIT13  | Input       | BOOL          | Eingabebit 13 wird mit Bit 13 der Maske verglichen.  |
| IN_BIT14  | Input       | BOOL          | Eingabebit 14 wird mit Bit 14 der Maske verglichen.  |
| IN_BIT15  | Input       | BOOL          | Eingabebit 15 wird mit Bit 15 der Maske verglichen.  |
| OUT       | Output      | BOOL          | Der Signalzustand "1" zeigt an, dass eine Entsprechung gefunden wurde.<br>Der Signalzustand "0" zeigt an, dass keine Entsprechung gefunden wurde.                              |
| OUT_STEP  | Output      | BYTE          | Enthält die Schrittnummer mit der entsprechenden Maske oder die Schrittnummer, die um "1" größer ist als der Wert am Parameter LAST, sofern keine Entsprechung gefunden wurde. |
| ERR_CODE  | Output      | WORD          | Fehlerinformation  |
| LAST      | Static      | BYTE          | Gibt die Schrittnummer des letzten Schritts an, der nach der entsprechenden Maske abgefragt werden soll.   |
| CMP_VAL   | Static      | ARRAY OF WORD | Vergleichsmasken [0 bis 15, 0 bis 15]: Bei der ersten Nummer des Index handelt es sich um die Schrittnummer und bei der zweiten Nummer um die Bitnummer der Maske.             |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

## Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

| Fehlercode*<br>(W#16#...)  | Erläuterung                                   |
|--|---|
| 0000   | Kein Fehler                                   |
| 000E   | Der Wert am Parameter LAST ist größer als 15. |
| *Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## LEAD\_LAG: Lead- und Lag-Algorithmus

### Beschreibung

Mit der Anweisung "Lead- und Lag-Algorithmus" bearbeiten Sie mit einer analogen Variablen Signale. Der Wert für die Verstärkung am Parameter GAIN muss größer als Null sein. Das Ergebnis der Anweisung "Lead- und Lag-Algorithmus" wird mit der folgenden Gleichung berechnet:

$$OUT = \left[ \frac{LG\_TIME}{LG\_TIME + SAMPLE\_T} \right] PREV\_OUT + GAIN \left[ \frac{LD\_TIME + SAMPLE\_T}{LG\_TIME + SAMPLE\_T} \right] IN - GAIN \left[ \frac{LD\_TIME}{LG\_TIME + SAMPLE\_T} \right] * PREV\_IN$$

Die Anweisung "Lead- und Lag-Algorithmus" liefert nur bei einer Bearbeitung in festen Programmzyklen sinnvolle Ergebnisse. An den Parametern LD\_TIME, LG\_TIME und SAMPLE\_T müssen die gleichen Einheiten angegeben werden. Die Anweisung nähert sich bei  $LG\_TIME > 4 + SAMPLE\_T$  an folgende Funktion an:

$$OUT = GAIN * ((1 + LD\_TIME * s) / (1 + LG\_TIME * s)) * IN$$

Wenn der Wert des Parameters GAIN kleiner oder gleich Null ist, wird die Berechnung nicht ausgeführt und eine Fehlerinformation am Parameter ERR\_CODE ausgegeben.

Sie können die Anweisung "Lead- und Lag-Algorithmus" zusammen mit Schleifen zur Kompensation bei der dynamischen Vorwärtsregelung einsetzen. Die Anweisung besteht dabei aus zwei Operationen. Die Operation "Lead" verschiebt die Phase des Ausgangs OUT, sodass der Ausgang dem Eingang voreilt. Die Operation "Lag" hingegen verschiebt den Ausgang, sodass der Ausgang dem Eingang nacheilt. Da die Operation "Lag" mit einer Integration gleichzusetzen ist, kann sie als Entstörelement oder als Tiefpassfilter eingesetzt werden. Die Operation "Lead" entspricht einer Differenziation und kann deshalb als Hochpassfilter eingesetzt werden. Beide Operationen zusammen (Lead und Lag) führen dazu, dass die Ausgangsphase dem Eingang bei niederen Frequenzen nacheilt und ihm bei hohen Frequenzen voreilt. Deshalb kann die Anweisung "Lead- und Lag-Algorithmus" als Bandpassfilter eingesetzt werden.

Beim Einfügen der Anweisung im Programm wird automatisch der Dialog "Aufrufoptionen" geöffnet, in dem Sie festlegen können, ob die Bausteinparameter der Anweisung in einem eigenen Datenbaustein (Einzelnanz) oder als lokale Variable (Multiinstanz) in der Bausteinschnittstelle abgelegt werden. Falls Sie einen eigenen Datenbaustein erstellen, finden Sie diesen in der Projektnavigation im Ordner "Programmressourcen" unter "Programmbausteine > Systembausteine". Weitere Informationen dazu finden Sie unter "Siehe auch".

## Syntax

Für die Anweisung "Lead- und Lag-Algorithmus" wird die folgende Syntax verwendet:

### SCL

```
<Instanz>(IN := <Operand>,
          SAMPLE_T := <Operand>,
          OUT => <Operand>,
          ERR_CODE => <Operand>)
```

Die folgende Tabelle zeigt die Parameter der Anweisung:

| Parameter | Deklaration | Datentyp | Beschreibung  |
|-----------|-------------|----------|---|
| IN        | Input       | REAL     | Eingabewert der aktuellen Abtastzeit (Zykluszeit), die bearbeitet wird.<br>Am Parameter IN können auch Konstanten angegeben werden. |
| SAMPLE_T  | Input       | INT      | Abtastzeit<br>Am Parameter SAMPLE_T können auch Konstanten angegeben werden.  |
| OUT       | Output      | REAL     | Ergebnis der Anweisung  |
| ERR_CODE  | Output      | WORD     | Fehlerinformation   |
| LD_TIME   | Static      | REAL     | Voreilzeit in der gleichen Einheit wie die Abtastzeit.  |
| LG_TIME   | Static      | REAL     | Nacheilzeit in der gleichen Einheit wie die Abtastzeit  |
| GAIN      | Static      | REAL     | Verstärkung in % / % (Verhältnis von Ausgabeveränderung zu Eingabeveränderung als stetiger Zustand).                                |

| Parameter | Deklaration | Datentyp | Beschreibung       |
|-----------|-------------|----------|--------------------|
| PREV_IN   | Static      | REAL     | Vorheriger Eingang |
| PREV_OUT  | Static      | REAL     | Vorheriger Ausgang |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

Die statischen Parameter sind beim Aufruf der Anweisung im Programm nicht sichtbar. Diese werden in der Instanz der Anweisung abgelegt.

### Parameter ERR\_CODE

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters ERR\_CODE:

| Fehlercode*<br>(W#16#...) | Erläuterung  |
|---------------------------|--|
| 0000                      | Kein Fehler  |
| 0009                      | Der Wert am Parameter GAIN ist kleiner oder gleich Null. |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

#### Hinweis

Statische Parameter können Sie im Datenbaustein initialisieren.

#### SCL

```
"LEAD_LAG_DB" (IN := "Tag_Input",
                SAMPLE_T := "Tag_Input_SAMPLE_T",
                OUT => "Tag_Output_Result",
                ERR_CODE => "Tag_ErrorCode");
```

Die folgenden Tabellen zeigen die Funktionsweise der Anweisung anhand konkreter Werte.

#### Vor der Bearbeitung

Für die Eingangsparameter werden in diesem Beispiel die folgenden Werte verwendet:

| Parameter | Operand            | Wert |
|-----------|--------------------|------|
| IN        | Tag_Input          | 2.0  |
| SAMPLE_T  | Tag_Input_SAMPLE_T | 10   |

Im Instanz-Datenbaustein "LEAD\_LAG\_DB" der Anweisung sind die folgenden Werte gespeichert:

| Parameter | Adresse | Wert |
|-----------|---------|------|
| LD_TIME   | DBD12   | 2.0  |
| LG_TIME   | DBD16   | 2.0  |
| GAIN      | DBD20   | 1.0  |
| PREV_IN   | DBD24   | 6.0  |
| PREV_OUT  | DBD28   | 6.0  |

#### Nach der Bearbeitung

Nach der Ausführung der Anweisung werden in die Ausgangsparameter die folgenden Werte geschrieben:

| Parameter | Operand           | Wert |
|-----------|-------------------|------|
| OUT       | Tag_Output_Result | 2.0  |

Im Instanz-Datenbaustein "LEAD\_LAD\_DB" der Anweisung werden die folgenden Werte gespeichert:

| Parameter | Operand | Wert |
|-----------|---------|------|
| PREV_IN   | DBD24   | 2.0  |
| PREV_OUT  | DBD28   | 2.0  |

#### Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

### SEG: Bitmuster für 7-Segment-Anzeige erzeugen

#### Beschreibung

Mit der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wandeln Sie jede der vier Hexadezimalziffern des angegebenen Quellworts (IN) in ein äquivalentes Bitmuster für eine 7-Segment-Anzeige um. Das Ergebnis der Anweisung wird in das Doppelwort am Parameter OUT ausgegeben.

Zwischen den Hexadezimalziffern und der Belegung der 7 Segmenten (a, b, c, d, e, f, g) besteht die folgende Beziehung:

| Eingangsziffer<br>r<br>(Binär) | Belegung der<br>Segmente<br>- g f e d c b a | Anzeige<br>(Hexadezimal) | Sieben-Segment-Anzeige |
|--------------------------------|---|--------------------------|------------------------|
| 0000                           | 00111111                                    | 0                        |                        |
| 0001                           | 00000110                                    | 1                        |                        |
| 0010                           | 01011011                                    | 2                        |                        |
| 0011                           | 01001111                                    | 3                        |                        |
| 0100                           | 01100110                                    | 4                        |                        |
| 0101                           | 01101101                                    | 5                        |                        |
| 0110                           | 01111101                                    | 6                        |                        |
| 0111                           | 00000111                                    | 7                        |                        |
| 1000                           | 01111111                                    | 8                        |                        |
| 1001                           | 01100111                                    | 9                        |                        |
| 1010                           | 01110111                                    | A                        |                        |
| 1011                           | 01111100                                    | B                        |                        |
| 1100                           | 00111001                                    | C                        |                        |
| 1101                           | 01011110                                    | D                        |                        |
| 1110                           | 01111001                                    | E                        |                        |
| 1111                           | 01110001                                    | F                        |                        |

### Syntax

Für die Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen" wird die folgende Syntax verwendet:

```

SCL
SEG(IN := <Operand>,
    OUT => <Operand>)
    
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Bitmuster für 7-Segment-Anzeige erzeugen":

| Parameter     | Deklaration | Datentyp | Beschreibung                          |
|---------------|-------------|----------|---------------------------------------|
| IN            | Input       | WORD     | Quellwort mit vier Hexadezimalziffern |
| OUT           | Output      | DWORD    | Bitmuster für die 7-Segment-Anzeige   |
| Funktionswert |             | VOID     | Leerer Funktionswert                  |



## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
SEG(IN := "Tag_Input",
    OUT => "Tag_Output");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Operandenwerte:

| Parameter | Operand    | Wert          |   |
|-----------|------------|---------------|---|
|           |            | Hexadezimal   | Binär   |
| IN        | Tag_Input  | W#16#1234     | 0001 0010 0011 0100                                     |
| OUT       | Tag_Output | DW16#065B4F66 | 00000110 01011011 01001111<br>01100110<br>Anzeige: 1234 |

## Siehe auch

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

Übersicht über die gültigen Datentypen (Seite 1087)

## BCDCPL: Zehnerkomplement erzeugen

### Beschreibung

Mit der Anweisung "Zehnerkomplement erzeugen" erzeugen Sie das Zehnerkomplement einer siebenstelligen BCD-Zahl, die am Operanden angegeben wird. Die Anweisung rechnet mit folgender mathematischer Formel:

```
10000000 (als BCD)
– 7-stelliger BCD-Wert
-----
Zehnerkomplement (als BCD)
```

### Syntax

Für die Anweisung "Zehnerkomplement erzeugen" wird die folgende Syntax verwendet:

```
SCL
BCDCPL(<Operand>)
```

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Zehnerkomplement erzeugen":

| Parameter     | Deklaration | Datentyp | Beschreibung           |
|---------------|-------------|----------|------------------------|
| <Operand>     | Input       | DWORD    | 7-stellige BCD-Zahl    |
| Funktionswert |             | DWORD    | Ergebnis der Anweisung |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Siehe auch".

### Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := BCDCPL("Tag_Input");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

| Operand    | Wert*          |
|------------|----------------|
| Tag_Input  | DW#16#01234567 |
| Tag_Result | DW#16#08765433 |

\*Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Siehe auch

- Übersicht über die gültigen Datentypen (Seite 1087)
- SCL-Anweisungen eingeben (Seite 1380)
- SCL-Anweisungen bearbeiten (Seite 1398)

## **BITSUM: Anzahl der gesetzten Bits zählen**

### Beschreibung

Mit der Anweisung "Anzahl der gesetzten Bits zählen" zählen Sie die Anzahl der Bit eines Operanden, die auf den Signalzustand "1" gesetzt sind.

### Syntax

Für die Anweisung "Anzahl der gesetzten Bits zählen" wird die folgende Syntax verwendet:

```
SCL
BITSUM(<Operand>)
```

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Anzahl der gesetzten Bits zählen":

| Parameter     | Deklaration | Datentyp | Beschreibung                                |
|---------------|-------------|----------|---|
| <Operand>     | Input       | DWORD    | Operand, dessen gesetzte Bit gezählt werden |
| Funktionswert |             | INT      | Ergebnis der Anweisung                      |

## Beispiel

Das folgende Beispiel zeigt die Funktionsweise der Anweisung:

```
SCL
"Tag_Result" := BITSUM("Tag_Input");
```

Die folgende Tabelle zeigt die Funktionsweise der Anweisung anhand konkreter Werte:

| Operand    | Wert*               |
|------------|---------------------|
| Tag_Input  | DW#16#12345678      |
| Tag_Result | W#16#000D (13 Bits) |

\*Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## Siehe auch

Übersicht über die gültigen Datentypen (Seite 1087)

SCL-Anweisungen eingeben (Seite 1380)

SCL-Anweisungen bearbeiten (Seite 1398)

## 9.7.3 Erweiterte Anweisungen

### 9.7.3.1 Datum und Uhrzeit

#### T\_COMP: Zeitvariablen vergleichen

#### Beschreibung

Mit der Anweisung vergleichen Sie die Inhalte zweier Variablen der Datentypen "Zeiten" oder "Datum und Uhrzeit".

Die Anweisung unterstützt den Vergleich der folgenden Datentypen: DATE, TIME, LTIME, TOD (TIME\_OF\_DAY), LTOD (LTIME\_OF\_DAY), DT (DATE\_AND\_TIME), LDT (DATE\_AND\_LTIME), DTL. Um den Vergleich durchzuführen, müssen die Datentypen in Länge und Format übereinstimmen.

Das Vergleichsergebnis wird als Rückgabewert am Parameter OUT ausgegeben. Hierfür wird der Parameter OUT auf "1" gesetzt, wenn die verwendete Vergleichsbedingung erfüllt ist.

Die folgenden Vergleichsmöglichkeiten können verwendet werden:

| Symbol | Beschreibung  |
|--------|---|
| EQ     | Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 und Parameter IN2 gleich ist.                                   |
| NE     | Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 und Parameter IN2 ungleich ist.                                 |
| GE     | Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt Parameter IN1 größer (jünger) oder gleich dem Zeitpunkt am Parameter IN2 ist.    |
| LE     | Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 kleiner (älter) oder gleich dem Zeitpunkt am Parameter IN2 ist. |
| GT     | Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 größer (jünger) als am Parameter IN2 ist.                       |
| LT     | Der Rückgabewert führt Signalzustand "1", wenn der Zeitpunkt am Parameter IN1 kleiner (älter) als am Parameter IN2 ist.                       |

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T\_COMP":

| Parameter | Deklaration | Datentyp                                   | Speicherbereich              | Beschreibung                    |
|-----------|-------------|--|------------------------------|---------------------------------|
| IN1       | Input       | DATE, TIME, LTIME, TOD, LTOD, DT, LDT, DTL | E, A, M, D, L oder Konstante | Erster zu vergleichender Wert.  |
| IN2       | Input       | DATE, TIME, LTIME, TOD, LTOD, DT, LDT, DTL | E, A, M, D, L oder Konstante | Zweiter zu vergleichender Wert. |
| OUT       | Output      | BOOL                                       | E, A, M, D, L                | Rückgabewert                    |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### T\_CONV: Zeiten umwandeln und extrahieren

#### Beschreibung

Mit der Anweisung "T\_CONV" konvertieren Sie den Datentyp am Eingangsparameter IN in den Datentyp, der am Ausgang OUT ausgegeben wird. Über die Anweisungsboxen des Ein- und Ausgangs wählen Sie die Datenformate zur Konvertierung aus.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T\_CONV". Wird am Eingangs- und Ausgangsparameter der gleiche Datentyp verwendet, kopiert die Anweisung den entsprechenden Wert.

| Parameter | Deklaration | Datentyp                             |  | Speicherbereich              | Beschreibung               |
|-----------|-------------|--------------------------------------|--|------------------------------|----------------------------|
|           |             | S7-1200                              | S7-1500                                      |                              |                            |
| IN        | Input       | Ganzzahlen, TIME, Datum und Uhrzeit* | WORD, Ganzzahlen, Zeiten, Datum und Uhrzeit* | E, A, M, D, L oder Konstante | Zu konvertierender Wert    |
| OUT       | Return      | Ganzzahlen, TIME, Datum und Uhrzeit* | WORD, Ganzzahlen, Zeiten, Datum und Uhrzeit* | E, A, M, D, L                | Ergebnis der Konvertierung |

\* Der Umfang der unterstützten Datentypen ist abhängig von der CPU. Welche Datentypen von den Baugruppen S7-1200 und S7-1500 jeweils unterstützt werden, entnehmen Sie der Übersicht der jeweils gültigen Datentypen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## T\_ADD: Zeiten addieren

### Beschreibung

Mit der Anweisung addieren Sie die Zeitangabe am Eingangsparameter IN1 zu der Zeitangabe am Eingangsparameter IN2. Das Ergebnis können Sie am Ausgangsparameter OUT abfragen. Dabei ist die Addition folgender Formate möglich:

- Addition einer Zeitdauer zu einer anderen Zeitdauer.  
Beispiel: Addition eines Datentyps TIME zu einem weiteren Datentyp TIME.
- Addition einer Zeitdauer zu einem Zeitpunkt.  
Beispiel: Addition eines Datentyps TIME zu dem Datentyp DTL.

Den Datentyp für den Wert am Eingangsparameter IN1 und Ausgangsparameter OUT bestimmen Sie durch die Auswahl in den Anweisungsboxen des Ein- und Ausgangs. Am Eingangsparameter IN2 können Sie nur Zeitangaben des Formats TIME (bei S7-1500er Baugruppen auch LTIME) angeben.

**Parameter**

Die folgenden Tabellen zeigen die Parameter der Anweisung "T\_ADD" entsprechend der möglichen Konvertierungen:

Tabelle 9-34 Addition einer Zeitdauer zu einer anderen Zeitdauer

| Parameter | Deklaration | Datentyp               |              | Speicherbereich              | Beschreibung   |
|-----------|-------------|------------------------|--------------|------------------------------|--|
|           |             | S7-1200                | S7-1500      |                              |  |
| IN1       | Input       | TIME                   | TIME, LTIME  | E, A, M, D, L oder Konstante | Erster Summand   |
| IN2       | Input       | TIME                   | TIME, LTIME  | E, A, M, D, L oder Konstante | Zweiter Summand  |
| OUT       | Return      | DINT, DWORD, TIME, TOD | TIME, LTIME, | E, A, M, D, L                | Ergebnis der Addition<br>Die Auswahl der Datentypen ist abhängig von den gewählten Datentypen der Eingangsparameter IN1 und IN2. |

Tabelle 9-35 Addition einer Zeitdauer zu einem Zeitpunkt

| Parameter | Deklaration | Datentyp                           |                         | Speicherbereich              | Beschreibung   |
|-----------|-------------|------------------------------------|-------------------------|------------------------------|--|
|           |             | S7-1200                            | S7-1500                 |                              |  |
| IN1       | Input       | DTL, TOD                           | DT, TOD, LTOD, LDT, DTL | E, A, M, D, L oder Konstante | Erster Summand   |
| IN2       | Input       | TIME                               | TIME, LTIME             | E, A, M, D, L oder Konstante | Zweiter Summand  |
| OUT       | Return      | DINT, DWORD, TIME, TOD, UDINT, DTL | DT, DTL, LDT, TOD, LTOD | E, A, M, D, L                | Ergebnis der Addition<br>Die Auswahl der Datentypen ist abhängig von den gewählten Datentypen der Eingangsparameter IN1 und IN2. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## T\_SUB: Zeiten subtrahieren

### Beschreibung

Mit der Anweisung subtrahieren Sie die Zeitangabe am Eingangsparameter IN2 von der Zeitangabe am Eingangsparameter IN1. Die Differenz können Sie am Ausgangsparameter OUT abfragen. Dabei ist die Subtraktion folgender Formate möglich:

- Subtraktion einer Zeitdauer von einer anderen Zeitdauer  
Beispiel: Subtraktion einer Zeitdauer vom Datentyp TIME von einer anderen Zeitdauer vom Datentyp TIME. Das Ergebnis kann in eine Variable des Datentyps TIME ausgegeben werden.
- Subtraktion einer Zeitdauer von einem Zeitpunkt  
Beispiel: Subtraktion einer Zeitdauer vom Datentyp TIME von einem Zeitpunkt vom Datentyp DTL. Das Ergebnis kann in eine Variable des Datentyps DTL ausgegeben werden.

Die Formate der Werte am Eingangsparameter IN1 und Ausgangsparameter OUT bestimmen Sie durch die Auswahl der Datentypen für den Eingangsparameter und den Ausgangsparameter der Anweisung.

### Parameter

Die folgenden Tabellen zeigen die Parameter der Anweisung "T\_SUB" entsprechend der möglichen Konvertierungen:

Tabelle 9-36 Subtraktion einer Zeitdauer von einer anderen Zeitdauer

| Parameter | Deklaration | Datentyp                      |             | Speicherbereich              | Beschreibung             |
|-----------|-------------|-------------------------------|-------------|------------------------------|--------------------------|
|           |             | S7-1200                       | S7-1500     |                              |                          |
| IN1       | Input       | TIME                          | TIME, LTIME | E, A, M, D, L oder Konstante | Minuend                  |
| IN2       | Input       | TIME                          | TIME, LTIME | E, A, M, D, L oder Konstante | Subtrahend               |
| OUT       | Return      | DINT, DWORD, TIME, TOD, UDINT | TIME, LTIME | E, A, M, D, L                | Ergebnis der Subtraktion |

Tabelle 9-37 Subtraktion einer Zeitdauer von einem Zeitpunkt

| Parameter | Deklaration | Datentyp                           |                         | Speicherbereich              | Beschreibung             |
|-----------|-------------|------------------------------------|-------------------------|------------------------------|--------------------------|
|           |             | S7-1200                            | S7-1500                 |                              |                          |
| IN1       | Input       | DTL, TOD                           | TOD, LTOD, DTL, DT, LDT | E, A, M, D, L oder Konstante | Minuend                  |
| IN2       | Input       | TIME                               | TIME, LTIME             | E, A, M, D, L oder Konstante | Subtrahend               |
| OUT       | Return      | DTL, DINT, DWORD, TIME, TOD, UDINT | TOD, LTOD, DTL, DT, LDT | E, A, M, D, L                | Ergebnis der Subtraktion |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### T\_DIFF: Zeitdifferenz

#### Beschreibung

Mit der Anweisung subtrahieren Sie die Zeitangabe am Eingangsparameter IN2 von der Zeitangabe am Eingangsparameter IN1. Das Ergebnis wird am Ausgangsparameter OUT im Format TIME ausgegeben.

- Wenn die Zeitangabe am Eingangsparameter IN2 größer als die Zeitangabe am Eingangsparameter IN1 ist, wird als Ergebnis ein negativer Wert am Ausgangsparameter OUT ausgegeben.
- Wenn das Ergebnis der Subtraktion außerhalb des TIME-Zahlenbereichs liegt, wird das Ergebnis auf "0" (0:00) gesetzt und der Freigabeausgang ENO = "0".

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T\_DIFF":

| Parameter | Deklaration | Datentyp |                               | Speicherbereich              | Beschreibung             |
|-----------|-------------|----------|-------------------------------|------------------------------|--------------------------|
|           |             | S7-1200  | S7-1500                       |                              |                          |
| IN1       | Input       | DTL      | DTL, DATE, DT, TOD, LTOD, LDT | E, A, M, D, L oder Konstante | Minuend                  |
| IN2       | Input       | DTL      | DTL, DATE, DT, TOD, LTOD, LDT | E, A, M, D, L oder Konstante | Subtrahend               |
| OUT       | Return      | TIME     | TIME, LTIME, INT              | E, A, M, D, L                | Differenz im Format TIME |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".



## T\_COMBINE: Zeiten zusammenfassen

### Beschreibung

Die Anweisung fasst den Wert für ein Datum mit dem einer Uhrzeitangabe zusammen und wandelt diese in eine kombinierte Datums- und Uhrzeitangabe.

- Am Eingangsparameter IN1 wird die Datumsangabe eingegeben. Bei dem Datentyp DATE muss ein Wert zwischen 1990-01-01 und 2089-12-31 verwendet werden (wird nicht geprüft).
- Am Eingangswert IN2 wird die Uhrzeitangabe (Datentyp TOD/LTOD) eingegeben.
- Am Ausgangswert OUT wird der kombinierte Datentyp zur Datums- und Uhrzeitangabe ausgegeben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T\_COMBINE":

| Parameter | Deklaration | Datentyp |                             | Speicherbereich                 | Beschreibung                          |
|-----------|-------------|----------|-----------------------------|---------------------------------|---------------------------------------|
|           |             | S7-1200  | S7-1500                     |                                 |                                       |
| IN1       | Input       | DATE     | DATE,<br>WORD,<br>UINT, INT | E, A, M, D, L oder<br>Konstante | Eingangsvariable des Datums           |
| IN2       | Input       | TOD      | TOD, LTOD                   | E, A, M, D, L oder<br>Konstante | Eingangsvariable der Uhrzeit          |
| OUT       | Return      | DTL      | DT, DTL, LDT                | E, A, M, D, L                   | Rückgabewert von Datum und<br>Uhrzeit |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Uhrzeitfunktionen

#### WR\_SYS\_T: Uhrzeit stellen

### Beschreibung

Mit der Anweisung stellen Sie das Datum und die Uhrzeit der CPU-Uhr ein. Die Angaben zu Datum und Uhrzeit geben Sie am Eingangsparameter IN der Anweisung an. Am Ausgangsparameter RET\_VAL können Sie abfragen, ob Fehler während der Ausführung der Anweisung aufgetreten sind.

Angaben zur lokalen Zeitzone oder Sommerzeit können mit der Anweisung "WR\_SYS\_T" nicht übertragen werden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "WR\_SYS\_T":

| Parameter | Deklaration | Datentyp        |              | Speicherbereich              | Beschreibung         |
|-----------|-------------|-----------------|--------------|------------------------------|----------------------|
|           |             | S7-1200         | S7-1500      |                              |                      |
| IN        | Input       | DTL             | DT, DTL, LDT | E, A, M, D, L oder Konstante | Datum und Uhrzeit    |
| RET_VAL   | Return      | INT, REAL, DINT | INT          | E, A, M, D, L                | Status der Anweisung |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode* (W#16#....)   | Beschreibung          |
|---|-----------------------|
| 0000  | Kein Fehler           |
| 8080  | Fehler im Datum       |
| 8081  | Fehler in der Uhrzeit |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                       |

**RD\_SYS\_T: Uhrzeit lesen**

**Beschreibung**

Mit der Anweisung lesen Sie das aktuelle Datum und die aktuelle Uhrzeit der CPU-Uhr aus. Die ausgelesenen Daten werden am Ausgangsparameter OUT der Anweisung ausgegeben. Im ausgegebenen Wert sind dabei keine Angaben zur lokalen Zeitzone oder Sommerzeit enthalten. Am Ausgang RET\_VAL können Sie abfragen, ob Fehler während der Ausführung der Anweisung aufgetreten sind.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_SYS\_T":

| Parameter | Deklaration | Datentyp |              | Speicherbereich | Beschreibung              |
|-----------|-------------|----------|--------------|-----------------|---------------------------|
|           |             | S7-1200  | S7-1500      |                 |                           |
| RET_VAL   | Return      | INT      | INT          | E, A, M, D, L   | Status der Anweisung      |
| OUT       | Output      | DTL      | DT, DTL, LDT | E, A, M, D, L   | Datum und Uhrzeit der CPU |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 0000  | Kein Fehler  |
| 8081  | Angegebener Zeitwert am Parameter LOCTIME außerhalb des zulässigen Wertebereichs:<br>Bei DTL: min. DTL#1970-01-01-00:00:00.0, max. DTL#2200-12-31-23:59:59.999999999<br>Bei LDT: min. LDT#1970-1-1-0:0:0.000000000, max. LDT#2200-12-31-23:59:59.999999999 |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### RD\_LOC\_T: Lokalzeit lesen

#### Beschreibung

Mit der Anweisung lesen Sie die aktuelle Lokalzeit aus der CPU-Uhr aus und geben diese am Ausgang OUT aus. Zur Ausgabe der Lokalzeit werden die Angaben zur Zeitzone sowie zum Beginn der Sommer- und Winterzeit verwendet, die Sie bei der Konfiguration der CPU-Uhr eingestellt haben.

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_LOC\_T":

| Parameter | Deklaration | Datentyp |              | Speicherbereich | Beschreibung         |
|-----------|-------------|----------|--------------|-----------------|----------------------|
|           |             | S7-1200  | S7-1500      |                 |                      |
| RET_VAL   | Return      | INT      | INT          | M, D, L         | Status der Anweisung |
| OUT       | Output      | DTL      | DT, LDT, DTL | D               | Lokalzeit            |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode*<br>(W#16#...) | Beschreibung   |
|---------------------------|--|
| 0000                      | Kein Fehler  |
| 0001                      | Kein Fehler. Lokalzeit wird als Sommerzeit ausgegeben. |
| 8080                      | Die Lokalzeit ist nicht auslesbar.                     |

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 8081  | Angegebener Zeitwert am Parameter LOCTIME außerhalb des zulässigen Wertebereichs:<br><br>Bei DTL: min. DTL#1970-01-01-00:00:00.0, max. DTL#2554-12-31-23:59:59.999999999<br><br>Bei LDT: min. LDT#1970-1-1-0:0:0.000000000, max. LDT#2262-04-11-23:47:16.854775807 |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## WR\_LOC\_T: Lokalzeit schreiben

### Beschreibung

Mit der Anweisung "WR\_LOC\_T" stellen Sie Datum und Uhrzeit der CPU-Uhr ein. Die Angaben zu Datum und Uhrzeit geben Sie als Lokalzeit am Eingangsparameter LOCTIME der Anweisung an.

Die Anweisung berechnet anhand der Struktur "TimeTransformationRule" die Systemzeit. Die Granularität der Zeitangabe für die Lokalzeit und die Systemzeit ist produktspezifisch und beträgt mindestens eine Millisekunde. Eingangswerte am Parameter LOCTIME, die kleiner sind als von der CPU unterstützt, werden bei der Berechnung der Systemzeit gerundet.

Am Ausgangsparameter RET\_VAL können Sie abfragen, ob Fehler während der Ausführung der Anweisung aufgetreten sind.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR\_LOC\_T":

| Parameter | Deklaration | Datentyp | Speicherbereich               | Beschreibung   |
|-----------|-------------|----------|-------------------------------|--|
| LOCTIME   | Input       | DTL, LDT | E, A, M, D, L, oder Konstante | Lokalzeit  |
| DST       | Input       | BOOL     | E, A, M, D, L, oder Konstante | <b>Daylight Saving Time</b><br>Wird nur während der "doppelten Stunde" bei der Umschaltung zur Sommerzeit ausgewertet.<br><ul style="list-style-type: none"> <li>• TRUE = Sommerzeit (erste Stunde)</li> <li>• FALSE = Winterzeit (zweite Stunde)</li> </ul> |
| RET_VAL   | Return      | INT      | E, A, M, D, L                 | Fehlermeldung (siehe "Parameter RET_VAL")  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0000  | Kein Fehler.  |
| 8080  | Ungültiger Wert am Parameter LOCTIME.   |
| 8081  | Angegebener Zeitwert am Parameter LOCTIME außerhalb des zulässigen Wertebereichs: <ul style="list-style-type: none"> <li>• Bei DTL: min. DTL#1970-01-01-00:00:00.0, max. DTL#2554-12-31-23:59:59.999999999</li> <li>• Bei LDT: min. LDT#1970-1-1-0:0:0.000000000, max. LDT#2262-04-11-23:47:16.854775807</li> </ul> |
| 8082**  | Ungültiger Wert für die Angabe des Monats (Byte 2 im Format DTL).   |
| 8083**  | Ungültiger Wert für die Angabe des Tages (Byte 3 im Format DTL).  |
| 8084**  | Ungültiger Wert für die Angabe der Stunde (Byte 5 im Format DTL).   |
| 8085**  | Ungültiger Wert für die Angabe der Minute (Byte 5 im Format DTL).   |
| 8086**  | Ungültiger Wert für die Angabe der Sekunde (Byte 7 im Format DTL).  |
| 8087**  | Ungültiger Wert für die Angabe der Nanosekunde (Byte 8 bis 11 im Format DTL).   |
| 8089  | Zeitwert existiert nicht (abgelaufene Stunde bei der Umschaltung zur Sommerzeit).   |
| 80B0  | Die Echtzeituhr ist ausgefallen.  |
| 80B1  | Die Struktur "TimeTransformationRule" wurde nicht definiert.  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |
| ** Nur bei Lokalzeitangabe am Parameter LOCTIME im Format DTL.  |   |

## SET\_TIMEZONE: Zeitzone setzen

### Beschreibung

Mit der Anweisung berechnen Sie die Lokalzeit auf Basis der Baugruppenzeit. Die Baugruppenzeit der CPU ist die UTC-Zeit. Zur Kommunikation innerhalb des Systems wird ausschließlich die Baugruppenzeit verwendet. Die Regel zur Umrechnung in die Lokalzeit ist in dem Attribut "TimeTransformationRule" definiert, das Sie am Parameter TimeZone angeben. Die Regel definiert die Berechnung der Zeitzone, sowie die automatische Umschaltung zwischen Sommer- und Winterzeit.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET\_TIMEZONE":

| Parameter | Deklaration | Datentyp                             | Speicherbereich              | Beschreibung  |
|-----------|-------------|--------------------------------------|------------------------------|---|
| REQ       | Input       | BOOL                                 | E, A, M, D, L oder Konstante | REQ=1: Umwandlung der Baugruppenzeit in die Lokalzeit     |
| TimeZone  | Input       | Time TransformationRule (Seite 2360) | D                            | Regel zur Umwandlung der Baugruppenzeit in die Lokalzeit. |

| Parameter | Deklaration | Datentyp                 | Speicherbereich | Beschreibung  |
|-----------|-------------|--------------------------|-----------------|---|
| DONE      | Output      | BOOL                     | E, A, M, D, L   | <ul style="list-style-type: none"> <li>0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>1: Auftrag fehlerfrei ausgeführt</li> </ul>  |
| BUSY      | Output      | BOOL                     | E, A, M, D, L   | <ul style="list-style-type: none"> <li>0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.</li> </ul> |
| ERROR     | Output      | BOOL                     | E, A, M, D, L   | <ul style="list-style-type: none"> <li>0: Kein Fehler</li> <li>1: Fehler aufgetreten</li> </ul>   |
| STATUS    | Output      | DINT, DWORD, UDINT, WORD | E, A, M, D, L   | Fehlermeldung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

| Fehlercode*<br>(W#16#....)  | Beschreibung  |
|---|---|
| 7000  | Keine Auftragsbearbeitung aktiv.  |
| 7001  | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0                      |
| 7002  | Zwischenaufwurf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1". |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### TimeTransformationRule

#### Beschreibung

In der Struktur TimeTransformationRule werden die Zeitpunkte zur Umschaltung in Sommerzeit und Winterzeit definiert. Die Struktur hat den folgenden Aufbau:

| Name                   | Datentyp | Beschreibung   |
|------------------------|----------|--|
| TimeTransformationRule | STRUCT   |  |
| Bias                   | INT      | // Zeitunterschied zwischen Lokalzeit und UTC [min]  |
| DaylightBias           | INT      | // Zeitunterschied zwischen Winter- und Sommerzeit [min]   |
| DaylightStartMonth     | USINT    | // Monat der Umstellung auf Sommerzeit   |
| DaylightStartWeek      | USINT    | // Woche der Umstellung auf Sommerzeit:<br>// 1 = Erstes Auftreten des Wochentags im Monat, ...,<br>// 5 = Letztes Auftreten des Wochentags im Monat |
| DaylightStartWeekday   | USINT    | // Wochentag der Sommerzeitumstellung:<br>// 1 = Sonntag   |

| Name                 | Datentyp   | Beschreibung   |
|----------------------|------------|--|
| DaylightStartHour    | USINT      | // Stunde der Sommerzeitumstellung   |
| DaylightStartMinute  | USINT      | // Minute der Sommerzeitumstellung   |
| StandardStartMonth   | USINT      | // Monat der Umstellung auf Winterzeit   |
| StandardStartWeek    | USINT      | // Woche der Umstellung auf Winterzeit:<br>// 1 = Erstes Auftreten des Wochentags im Monat, ...,<br>// 5 = Letztes Auftreten des Wochentags im Monat |
| StandardStartWeekday | USINT      | // Wochentag der Winterzeitumstellung:<br>// 1 = Sonntag   |
| StandardStartHour    | USINT      | // Stunde der Winterzeitumstellung   |
| StandardStartMinute  | USINT      | // Minute der Winterzeitumstellung   |
| TimeZoneName         | STRING[80] | // Name der Zeitzone: "(GMT+01:00) Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna"   |

## SNC\_RTCB: Uhrzeitslaves synchronisieren

### Definition: Synchronisation von Uhrzeitslaves

Unter der Synchronisation von Uhrzeitslaves versteht man die Übertragung des Datums und der Uhrzeit vom Uhrzeitmaster eines Bussegments auf sämtliche Uhrzeitslaves dieses Bussegments.

### Beschreibung

Mit der Anweisung synchronisieren Sie alle an einem Bussegment vorhandenen Uhrzeitslaves unabhängig vom parametrisierten Synchronisationsintervall. Voraussetzung für die erfolgreiche Synchronisation ist, dass "SNC\_RTCB" auf einer CPU aufgerufen wird, deren Echtzeituhr für mindestens ein Bussegment als Uhrzeitmaster parametrisiert wurde.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SNC\_RTCB":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| RET_VAL   | Output      | INT      | E, A, M, D, L   | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Erläuterung   |
|---|---|
| 0000  | Bei der Synchronisation ist kein Fehler aufgetreten.                        |
| 0001  | Die vorhandene Uhr wurde für kein Bussegment als Uhrzeitmaster parametrier. |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)               |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### TIME\_TCK: Systemzeit lesen

#### Beschreibung

Mit der Anweisung "TIME\_TCK" lesen Sie die Systemzeit der CPU. Die Systemzeit ist ein Zeitzähler, der von 0 bis max. 2147483647 ms zählt. Bei einem Überlauf der Systemzeit wird wieder ab "0" gezählt. Das Zeitraster und die Genauigkeit der Systemzeit beträgt 1 ms. Die Systemzeit wird nur von den Betriebszuständen der CPU beeinflusst. Die Systemzeit können Sie zum Beispiel nutzen, um die Dauer von Vorgängen durch Differenzbildung der Ergebnisse von zwei "TIME\_TCK"-Aufrufen zu messen. Die Anweisung stellt keine Fehlerinformationen zur Verfügung.

Die folgende Tabelle gibt eine Übersicht, wie sich die Systemzeit abhängig von den Betriebszuständen der CPU ändert.

| Betriebszustand      | Systemzeit ...   |
|----------------------|--|
| Anlauf               | ... wird ständig aktualisiert.                             |
| RUN                  |  |
| STOP                 | ... wird angehalten und behält den aktuellen Wert.         |
| Neustart (Warmstart) | ... wird gelöscht und beginnt wieder von "0" an zu laufen. |

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TIME\_TCK":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| RET_VAL   | Return      | TIME     | E, A, M, D, L   | Der Parameter RET_VAL enthält die gelesene Systemzeit im Bereich von 0 bis $2^{31} - 1$ ms. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".



## RTM: Betriebsstundenzähler

### Beschreibung

Mit der Anweisung können Sie einen 32-Bit-Betriebsstundenzähler Ihrer CPU setzen, starten, stoppen und auslesen.

Beachten Sie, dass der Betriebsstundenzähler während der Ausführung des Anwenderprogramms auch gestoppt oder neu gestartet werden kann, wodurch die gespeicherten Werte verfälscht werden können.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RTM":

| Parameter | Deklaration | Datentyp   | Speicherbereich              | Beschreibung  |
|-----------|-------------|------------|------------------------------|---|
| NR        | Input       | RTM (UINT) | E, A, M, D, L oder Konstante | Nummer des Betriebsstundenzählers<br>Die Nummerierung beginnt mit 0.<br>Die Anzahl der Betriebsstundenzähler Ihrer CPU entnehmen Sie den Technischen Daten.   |
| MODE      | Input       | BYTE       | E, A, M, D, L oder Konstante | Auftragskennung: <ul style="list-style-type: none"> <li>• 0: auslesen (Status wird dann in CQ, aktueller Wert wird in CV eingetragen). Läuft der Betriebsstundenzähler länger als <math>(2^{31} - 1)</math> Stunden, dann bleibt er beim letzten darstellbaren Wert stehen und liefert die Fehlermeldung "Überlauf".</li> <li>• 1: starten (mit dem letzten Zählerstand)</li> <li>• 2: stoppen</li> <li>• 4: setzen (auf den in PV angegebenen Wert)</li> <li>• 5: setzen (auf den in PV angegebenen Wert) und starten</li> <li>• 6: setzen (auf den in PV angegebenen Wert) und stoppen</li> </ul> |
| PV        | Input       | DINT       | E, A, M, D, L oder Konstante | Neuer Wert für den Betriebsstundenzähler  |
| RET_VAL   | Return      | INT        | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.  |
| CQ        | Output      | BOOL       | E, A, M, D, L                | Status des Betriebsstundenzählers (1: läuft)  |
| CV        | Output      | DINT       | E, A, M, D, L                | Aktueller Wert des Betriebsstundenzählers   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode (W#16#...)        | Erläuterung   |
|------------------------------|---|
| 0000                         | Kein Fehler   |
| 8080                         | Falsche Nummer des Betriebsstundenzählers                   |
| 8081                         | Ein negativer Wert wurde dem Parameter PV übergeben.        |
| 8082                         | Überlauf des Betriebsstundenzählers                         |
| 8091                         | Der Eingangsparameter MODE enthält einen unzulässigen Wert. |
| allgemeine Fehlerinformation | Siehe auch: Fehlerauswertung über GET_ERR_ID (Seite 2304)   |

### 9.7.3.2 String + Char

#### S\_MOVE: Zeichenkette verschieben

##### Beschreibung

Mit der Anweisung können Sie den Inhalt einer Zeichenkette (STRING) verschieben. Die Zeichenkette am Eingangsparameter IN wird auf den Ausgangsparameter OUT kopiert.

Für die Anweisung "S\_MOVE" können Sie zusätzliche Ausgänge einfügen. Der Inhalt des Operanden am Eingangsparameter IN wird in diesem Fall zu allen verfügbaren Ausgängen übertragen.

Zum Kopieren von Variablen des Datentyps ARRAY können Sie die Anweisungen "MOVE\_BLK" und "UMOVE\_BLK" verwenden.

##### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "S\_MOVE":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung |
|-----------|-------------|----------|-----------------|--------------|
| IN        | Input       | STRING   | D, L            | Quellwert    |
| OUT       | Output      | STRING   | D, L            | Zieladresse  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

#### S\_COMP: Zeichenketten vergleichen

##### Beschreibung

Die Anweisung vergleicht die Inhalte zweier Variablen im Format STRING und gibt das Vergleichsergebnis als Rückgabewert aus. Die zu vergleichenden Variablen werden an den

Eingängen IN1 und IN2 verschaltet. Die Eingangsparameter können nur mit einer symbolisch definierten Variablen belegt werden.

Über die Anweisungsbox wählen Sie die Vergleichsbedingung aus. Ist die Vergleichsbedingung (z. B. größer oder gleich) erfüllt, wird am Ausgangsparameter OUT der Signalzustand auf "1" gesetzt.

Die folgenden Vergleichsbedingungen können verwendet werden:

| Symbol   | Beschreibung  |
|--|---|
| EQ   | Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 gleich der Zeichenkette am Parameter IN2 ist.              |
| NE   | Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 ungleich der Zeichenkette am Parameter IN2 ist.            |
| GT <sup>(1)</sup>  | Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 größer als die Zeichenkette am Parameter IN2 ist.          |
| LT <sup>(1)</sup>  | Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 kleiner als die Zeichenkette am Parameter IN2 ist.         |
| GE <sup>(1)</sup>  | Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 größer oder gleich der Zeichenkette am Parameter IN2 ist.  |
| LE <sup>(1)</sup>  | Der Rückgabewert führt Signalzustand "1", wenn die Zeichenkette am Parameter IN1 kleiner oder gleich der Zeichenkette am Parameter IN2 ist. |
| <sup>(1)</sup> Die Zeichen werden beginnend von links über ihre ASCII-Codierung verglichen (z. B. ist 'a' größer als 'A'). Das erste unterschiedliche Zeichen entscheidet über das Vergleichsergebnis. Ist der linke Teil der längeren Zeichenkette identisch mit der kürzeren Zeichenkette, gilt die längere Zeichenkette als größer. |   |

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "S\_COMP":

| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung                      |
|--|-------------|----------|-----------------|-----------------------------------|
| IN1  | Input       | STRING*  | D, L            | Eingangsvariable im Format STRING |
| IN2  | Input       | STRING*  | D, L            | Eingangsvariable im Format STRING |
| OUT  | Output      | BOOL     | E, A, M, D, L   | Vergleichsergebnis                |
| * Definieren Sie die maximale Länge der Zeichenkette, wenn Sie bei der Schnittstellendeklaration den Datentyp STRING für eine temporäre Variable verwenden (weitere Information finden Sie in der Beschreibung des Datentyps). |             |          |                 |                                   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## S\_CONV: Zeichenkette konvertieren

### Beschreibung

Mit der Anweisung konvertieren Sie den Wert am Eingang IN in das Datenformat, das Sie am Ausgang OUT angegeben haben. Dabei sind die folgenden Konvertierungen möglich:

- **Konvertierung einer Zeichenkette (STRING) in einen numerischen Wert:**  
Die Konvertierung wird für alle Zeichen der am Eingangsparameter IN angegebenen Zeichenkette ausgeführt. Erlaubte Zeichen sind die Ziffern von "0" bis "9", der Dezimalpunkt sowie das Plus- und Minus-Zeichen. Das erste Zeichen der Kette darf dabei eine gültige Ziffer oder ein Vorzeichen sein. Führende Leerzeichen und Exponentialdarstellungen werden ignoriert.  
Die Zeichenkonvertierung kann durch ungültige Zeichen unterbrochen werden. Das Ausgabeformat der Konvertierung bestimmen Sie durch die Auswahl eines Datentyps für den Ausgangsparameter OUT.
- **Konvertierung eines numerischen Werts in eine Zeichenkette (STRING):**  
Das Format des zu konvertierenden numerischen Werts bestimmen Sie durch die Auswahl eines Datentyps für den Eingang IN. Am Ausgang OUT muss eine gültige Variable des Datentyps STRING angegeben werden. Die Länge der Zeichenkette nach der Konvertierung hängt vom Wert am Eingang IN ab. Das Konvertierungsergebnis wird ab dem dritten Byte der Zeichenkette gespeichert. In dem ersten Byte der Zeichenkette wird die maximale Länge und im zweiten Byte die tatsächliche Länge der Zeichenkette erfasst. Positive numerische Werte werden ohne Vorzeichen ausgegeben.

---

#### Hinweis

Bei einer Konvertierung von Null (z. B. INT\_TO\_STRING(0)) hat die Zeichenkette im Ergebnis eine Länge von 6 Zeichen.

---

- **Kopieren einer Zeichenkette:**  
Wenn Sie am Eingangsparameter und am Ausgangsparameter der Anweisung den Datentyp STRING eingeben, wird die Zeichenkette am Eingang IN auf den Ausgang OUT kopiert. Wenn die tatsächliche Länge der Zeichenkette am Eingang IN die maximale Länge der Zeichenkette am Ausgang OUT überschreitet, wird derjenige Teil der Zeichenkette an IN kopiert, der gerade noch in die Zeichenkette von OUT passt.

---

#### Hinweis

##### Exponentielle Notation bei der Konvertierung von Gleitpunktzahlen

Verwenden Sie bei der Konvertierung von Gleitpunktzahlen mit der Anweisung "S\_CONV" keine exponentielle Notation ("e" oder "E"). Verwenden Sie für die Konvertierung von Gleitpunktzahlen mit exponentieller Notation stattdessen die Anweisung "STRG\_VAL (Seite 2367)". Über den Parameter FORMAT der Anweisung können Sie als Eingangsformat die exponentielle Notation auswählen.

---

## Parameter

Die folgenden Tabellen zeigen die Parameter der Anweisung "S\_CONV" entsprechend der möglichen Konvertierungen:

Tabelle 9-38 Parameter bei der Konvertierung einer Zeichenkette in einen numerischen Wert:

| Parameter | Deklaration | Datentyp  | Speicherbereich | Beschreibung               |
|-----------|-------------|---|-----------------|----------------------------|
| IN        | Input       | STRING  | D, L            | Zu konvertierender Wert    |
| OUT       | Output      | CHAR, USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, REAL, LREAL | E, A, M, D, L   | Ergebnis der Konvertierung |

Tabelle 9-39 Parameter bei der Konvertierung eines numerischen Werts in eine Zeichenkette:

| Parameter | Deklaration | Datentyp  | Speicherbereich              | Beschreibung               |
|-----------|-------------|---|------------------------------|----------------------------|
| IN        | Input       | CHAR, USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, REAL, LREAL | E, A, M, D, L oder Konstante | Zu konvertierender Wert    |
| OUT       | Output      | STRING  | D, L                         | Ergebnis der Konvertierung |

Tabelle 9-40 Parameter beim Kopieren einer Zeichenkette:

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                |
|-----------|-------------|----------|-----------------|-----------------------------|
| IN        | Input       | STRING   | D, L            | Zu kopierender Wert         |
| OUT       | Output      | STRING   | D, L            | Ergebnis des Kopiervorgangs |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## STRG\_VAL: Zeichenkette in Zahlenwert konvertieren

### Beschreibung

Die Anweisung "STRG\_VAL" wandelt eine Zahlenzeichenkette in die entsprechende Ganzzahlige-oder Gleitpunktdarstellung um:

- Die zu konvertierende Zeichenkette geben Sie am Eingangsparameter IN an.
- Das Format des Ausgabewerts bestimmen Sie durch die Auswahl eines Datentyps für den Ausgangsparameter OUT. Das Ergebnis können Sie am Ausgangsparameter OUT abfragen.

Erlaubte Zeichen für die Konvertierung sind die Ziffern "0" bis "9", der Dezimalpunkt, das Dezimalkomma, Notationen "E" und "e" sowie das Plus- und Minus-Zeichen. Durch ungültige Zeichen kann die Konvertierung unterbrochen werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "STRG\_VAL":

| Parameter | Deklaration | Datentyp   |   | Speicherbereich              | Beschreibung  |
|-----------|-------------|--|---|------------------------------|---|
|           |             | S7-1200  | S7-1500   |                              |   |
| IN        | Input       | STRING   | STRING  | D, L oder Konstante          | Zu konvertierende Zahlenzeichenkette  |
| FORMAT    | Input       | WORD   | WORD  | E, A, M, D, L oder Konstante | Ausgabeformat der Zeichen   |
| P         | Input       | UINT   | UINT  | E, A, M, D, L oder Konstante | Verweis zum ersten Zeichen, das umgewandelt werden soll (erstes Zeichen = 1, der Wert "0" oder ein Wert > Länge des Strings ist ungültig) |
| OUT       | Output      | USINT, SINT, UINT, INT, UDINT, DINT, REAL, LREAL | USINT, SINT, UINT, INT, UDINT, DINT, ULINT, LINT, REAL, LREAL | E, A, M, D, L                | Ergebnis der Konvertierung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter FORMAT

Mit dem Parameter FORMAT geben Sie vor, wie die Zeichen einer Zeichenkette zu interpretieren sind. Exponentialwerte können mit der Anweisung "STRG\_VAL" auch konvertiert und dargestellt werden. Am Parameter FORMAT können nur Variablen vom Datentyp USINT angegeben werden.

Die folgende Tabelle zeigt die möglichen Werte des Parameters FORMAT und deren Bedeutung:

| Wert (W#16#...) | Notation        | Dezimaldarstellung |
|-----------------|-----------------|--------------------|
| 0000            | Dezimalbruch    | " "                |
| 0001            |                 | " "                |
| 0002            | Exponentiell    | " "                |
| 0003            |                 | " "                |
| 0004 bis FFFF   | Ungültige Werte |                    |

### Parameter P

Die Konvertierung beginnt ab dem Zeichen, dessen Position Sie im Parameter P angegeben haben. Wenn beispielsweise am Parameter P der Wert "1" angegeben ist, wird die Konvertierung ab dem ersten Zeichen der angegebenen Zeichenkette ausgeführt.

## Beispiel

Die folgende Tabelle zeigt Beispiele für die Konvertierung einer Zeichenkette in einen numerischen Wert:

| IN (STRING)     | FORMAT (W#16#....) | OUT (Datentyp) | OUT (Wert)    | ENO-Status |
|-----------------|--------------------|----------------|---------------|------------|
| '123'           | 0000               | INT/DINT       | 123           | 1          |
| '-00456'        | 0000               | INT/DINT       | -456          | 1          |
| '123.45'        | 0000               | INT/DINT       | 123           | 1          |
| '+2345'         | 0000               | INT/DINT       | 2345          | 1          |
| '00123AB'       | 0000               | INT/DINT       | 123           | 1          |
| '123'           | 0000               | REAL           | 123.0         | 1          |
| '-00456'        | 0001               | REAL           | -456.0        | 1          |
| '+00456'        | 0001               | REAL           | 456.0         | 1          |
| '123.45'        | 0000               | REAL           | 123.45        | 1          |
| '123.45'        | 0001               | REAL           | 12345.0       | 1          |
| '123,45'        | 0000               | REAL           | 12345.0       | 1          |
| '123,45'        | 0001               | REAL           | 123.45        | 1          |
| '.00123AB'      | 0001               | REAL           | 123.0         | 1          |
| '1.23e-4'       | 0000               | REAL           | 1.23          | 1          |
| '1.23E-4'       | 0000               | REAL           | 1.23          | 1          |
| '1.23E-4'       | 0002               | REAL           | 1.23E-4       | 1          |
| '12,345.67'     | 0000               | REAL           | 12345.67      | 1          |
| '12,345.67'     | 0001               | REAL           | 12.345        | 1          |
| '3.4e39'        | 0002               | REAL           | W#16#7F800000 | 1          |
| '-3.4e39'       | 0002               | REAL           | W#16#FF800000 | 1          |
| '1.1754943e-38' | 0002               | REAL           | 0.0           | 1          |
| '12345'         | -/-                | SINT           | 0             | 0          |
| 'A123'          | -/-                | -/-            | 0             | 0          |
| '               | -/-                | -/-            | 0             | 0          |
| '++123'         | -/-                | -/-            | 0             | 0          |
| '+-123'         | -/-                | -/-            | 0             | 0          |

## VAL\_STRG: Zahlenwert in Zeichenkette konvertieren

### Beschreibung

Mit der Anweisung "VAL\_STRG" konvertieren Sie einen numerischen Wert in eine Zeichenkette.

- Den zu konvertierenden Wert geben Sie am Eingangsparameter IN an. Das Format des numerischen Werts bestimmen Sie durch die Auswahl eines Datentyps.
- Das Ergebnis der Konvertierung fragen Sie am Ausgangsparameter OUT ab.

Erlaubte Zeichen für die Konvertierung sind die Ziffern "0" bis "9", der Dezimalpunkt, das Dezimalkomma, Notationen "E" und "e" sowie das Plus- und Minus-Zeichen. Durch ungültige Zeichen kann die Konvertierung unterbrochen werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "VAL\_STRG":

| Parameter | Deklaration | Datentyp   |   | Speicherbereich              | Beschreibung                                   |
|-----------|-------------|--|---|------------------------------|--|
|           |             | S7-1200  | S7-1500   |                              |  |
| IN        | Input       | USINT, SINT, UINT, INT, UDINT, DINT, REAL, LREAL | USINT, SINT, UINT, INT, UDINT, DINT, ULINT, LINT, REAL, LREAL | E, A, M, D, L oder Konstante | Zu konvertierender Wert                        |
| SIZE      | Input       | USINT  | USINT   | E, A, M, D, L oder Konstante | Anzahl der Zeichenplätze                       |
| PREC      | Input       | USINT  | USINT   | E, A, M, D, L oder Konstante | Anzahl der Nachkommastellen                    |
| FORMAT    | Input       | WORD   | WORD  | E, A, M, D, L oder Konstante | Ausgabeformat der Zeichen                      |
| P         | InOut       | UINT   | UINT  | E, A, M, D, L oder Konstante | Zeichen, ab dem das Ergebnis geschrieben wird. |
| OUT       | Output      | STRING   | STRING  | D, L                         | Ergebnis der Konvertierung                     |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter P

Durch den Parameter P geben Sie an, ab welchem Zeichen der Zeichenkette das Ergebnis geschrieben wird. Wenn beispielsweise am Parameter P der Wert "2" angegeben ist, wird der konvertierte Wert ab dem zweiten Zeichen der Zeichenkette gespeichert.

### Parameter SIZE und P

Durch den Parameter SIZE legen Sie fest, wie viele Zeichen der Zeichenkette beschrieben werden. Gezählt wird dabei ab dem Zeichen, das am Parameter P angegeben ist. Wenn der Ausgabewert kürzer als die angegebene Länge ist, wird das Ergebnis rechtsbündig in die Zeichenkette geschrieben. Die leeren Zeichenstellen werden dabei mit Leerzeichen gefüllt.

### Parameter FORMAT

Durch den Parameter FORMAT geben Sie vor, wie der numerische Wert bei der Konvertierung interpretiert und in die Zeichenkette geschrieben wird. Am Parameter FORMAT können Sie nur Variablen vom Datentyp USINT angeben.

Die folgende Tabelle zeigt die möglichen Werte des Parameters FORMAT und deren Bedeutung:



| Wert (W#16#...) | Notation        | Vorzeichen  | Dezimaldarstellung |
|-----------------|-----------------|-------------|--------------------|
| 0000            | Dezimalbruch    | "-"         | "."                |
| 0001            |                 |             | ","                |
| 0002            | Exponentiell    |             | "."                |
| 0003            |                 |             | ","                |
| 0004            | Dezimalbruch    | "+" und "-" | "."                |
| 0005            |                 |             | ","                |
| 0006            | Exponentiell    |             | "."                |
| 0007            |                 |             | ","                |
| 0008 bis FFFF   | Ungültige Werte |             |                    |

## Parameter PREC

Durch den Parameter PREC definieren Sie die Anzahl der Nachkommastellen bei der Konvertierung von Gleitpunktzahlen. Maximal wird eine Genauigkeit von 7 Ziffern für numerische Werte des Datentyps REAL unterstützt. Wenn der zu konvertierende Wert eine Ganzzahl ist, legen Sie mit dem Parameter PREC die Stelle fest, an der ein Dezimalpunkt gesetzt wird.

## Beispiel

Die folgende Tabelle zeigt Beispiele für die Konvertierung von numerischen Werten in eine Zeichenkette.

| IN (Wert)          | IN (Datentyp) | P  | SIZE | FORMAT (W#16#...) | PREC | OUT (STRING)  | ENO-Status |
|--------------------|---------------|----|------|-------------------|------|---------------|------------|
| 123                | UINT          | 16 | 10   | 0000              | 0    | xxxxxxxx123 C | 1          |
| 0                  | UINT          | 16 | 10   | 0000              | 2    | xxxxxx0.00 C  | 1          |
| 12345678           | UDINT         | 16 | 10   | 0000              | 3    | x12345.678 C  | 1          |
| 12345678           | UDINT         | 16 | 10   | 0001              | 3    | x12345.678 C  | 1          |
| 123                | INT           | 16 | 10   | 0004              | 0    | xxxxxx+123 C  | 1          |
| -123               | INT           | 16 | 10   | 0004              | 0    | xxxxxx-123 C  | 1          |
| -0.00123           | REAL          | 16 | 10   | 0004              | 4    | xxx-0.0012 C  | 1          |
| -0.00123           | REAL          | 16 | 10   | 0006              | 4    | -1.2300E-3 C  | 1          |
| -Inf <sup>1)</sup> | REAL          | 16 | 10   | -/-               | 4    | xxxxxx-INF C  | 0          |
| +Inf <sup>2)</sup> | REAL          | 16 | 10   | -/-               | 4    | xxxxxx+INF C  | 0          |
| NaN <sup>3)</sup>  | REAL          | 16 | 10   | -/-               | 4    | xxxxxxNaN C   | 0          |
| 12345678           | UDINT         | 16 | 6    | -/-               | 3    | xxxxxxxxxxx C | 0          |

"x" repräsentiert Leerzeichen  
<sup>1)</sup>-Inf: Gleitpunktzahl, die einen negativen unendlichen Wert repräsentiert.  
<sup>2)</sup>+Inf: Gleitpunktzahl, die einen positiven unendlichen Wert repräsentiert.  
<sup>3)</sup>NaN: Wert, der als Ergebnis ungültiger Rechenoperationen zurückgegeben wird.

## Strg\_TO\_Chars: Zeichenkette in Array of CHAR konvertieren

### Beschreibung

Mit der Anweisung kopieren Sie Zeichen aus einer Zeichenkette STRING in ein Feld von mehreren Zeichen des Datentyps CHAR oder BYTE (Array of CHAR / BYTE).

- Die Zeichenkette, aus der Zeichen kopiert werden sollen, geben Sie am Eingangsparameter STRG an.
- Die Zeichen werden am Parameter CHARS in einen Datentyp Array of CHAR oder Array of BYTE geschrieben. Über den Parameter PCHARS geben Sie an, ab welcher Stelle die Zeichen in das Feld Array of CHAR / BYTE geschrieben werden sollen. Als Standard wird die Untergrenze des Arrays verwendet (Beispiel: "0", bei Array[1 .. 10] of CHAR).
- Die Anzahl der Zeichen in dem Feld Array of CHAR muss mindestens so viele Zeichen umfassen, wie aus der Zeichenkette STRING kopiert werden sollen.

Für Datentypen STRING, BYTE und CHAR sind ausschließlich ASCII-Zeichen gültig.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Strg\_TO\_Chars":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| STRG      | Input       | STRING   | D, L oder Konstante          | Quelle: Zeichenkette  |
| PCHARS    | Input       | DINT     | E, A, M, D, L oder Konstante | Position in der Ziel-Zeichenkette, ab der die Zeichen geschrieben werden.<br>PCHARS = "0" adressiert den ersten Index der Zeichenkette.     |
| CHARS     | InOut       | VARIANT  | D, L                         | Ziel: Feld, in das die Zeichen kopiert werden. Die Zeichen können in ein Feld vom Datentyp Array of CHAR oder Array of BYTE kopiert werden. |
| CNT       | Output      | UINT     | E, A, M, D, L                | Anzahl der kopieren Zeichen.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Chars\_TO\_Strg: Array of CHAR in Zeichenkette konvertieren

### Beschreibung

Mit der Anweisung "Chars\_TO\_Strg" kopieren Sie Zeichen aus einem Feld von mehreren Zeichen des Datentyps CHAR oder BYTE (Array of CHAR / BYTE) in eine Zeichenkette STRING.

- Die Zeichen des Felds Array of CHAR / BYTE, die in eine Zeichenkette kopiert werden sollen, geben Sie am Eingangsparameter CHARS an. Über den Parameter PCHARS geben Sie an, ab welcher Stelle die Zeichen des Array kopiert werden sollen. Als Standard wird die Untergrenze des Arrays verwendet (Beispiel: "0", bei Array[1 .. 10] of CHAR).
- Die Zeichen werden am Parameter STRG in einen Datentyp STRING geschrieben. Die Anzahl der Zeichen in der Zeichenkette STRING muss mindestens so viele Zeichen umfassen, wie aus dem Feld Array of CHAR kopiert werden sollen.

Für Datentypen STRING, CHAR und BYTE sind ausschließlich ASCII-Zeichen gültig.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Chars\_TO\_Strg":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| CHARS     | Input       | VARIANT  | D, L                         | Quelle: Feld, aus dem die Zeichen kopiert werden  |
| PCHARS    | Input       | DINT     | E, A, M, D, L oder Konstante | Position in dem Feld Array of CHAR / Array of BYTE, ab der die Zeichen kopiert werden.<br>PCHARS = "0" adressiert den ersten Index des Felds. |
| CNT       | Input       | UINT     | E, A, M, D, L oder Konstante | Anzahl der zu kopierenden Zeichen.<br>Bei "0" werden alle Zeichen kopiert.  |
| STRG      | Output      | STRING   | D, L                         | Ziel: Zeichenkette  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## MAX\_LEN: Maximale Länge einer Zeichenkette ermitteln

### Beschreibung

Eine Variable vom Datentyp STRING enthält zwei Längen: die maximale Länge und die aktuelle Länge (das ist die Anzahl der momentan gültigen Zeichen).

- Die Maximallänge der Zeichenkette wird für jede Variable am Schlüsselwort STRING in eckigen Klammern angegeben. Die Anzahl der durch eine Zeichenkette belegten Bytes ist um 2 größer als die maximale Länge.
- Die aktuelle Länge repräsentiert die Anzahl der tatsächlich belegten Zeichenplätze. Die aktuelle Länge ist kleiner oder gleich der maximalen Länge.

Mit der Anweisung "MAX\_LEN" fragen Sie die maximale Länge der am Eingangsparameter IN angegebenen Zeichenkette ab und geben diese als numerischen Wert am Ausgangsparameter OUT aus.

Wenn Fehler während der Bearbeitung der Anweisung auftreten, wird ein Leerstring ausgegeben.

---

**Hinweis**

**Auslesen der aktuellen Länge**

Mit der Anweisung "LEN (Seite 2378)" können Sie die aktuelle Länge einer Zeichenkette auslesen.

---

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "MAX\_LEN":

| Parameter | Deklaration | Datentyp | Speicherbereich     | Beschreibung                |
|-----------|-------------|----------|---------------------|-----------------------------|
| IN        | Input       | STRING   | D, L oder Konstante | Zeichenkette                |
| OUT       | Return      | DINT     | E, A, M, D, L       | Maximale Anzahl der Zeichen |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**ATH: ASCII-Zeichenkette in Hexadezimalzahl konvertieren**

**Beschreibung**

Mit der Anweisung "ATH" konvertieren Sie die am Eingangsparameter IN angegebene ASCII-Zeichenkette in eine Hexadezimalzahl. Das Ergebnis der Konvertierung wird am Ausgangsparameter OUT ausgegeben.

- Über den Zeiger am Parameter IN (ASCII) können Sie auf die folgenden Datentypen verweisen: STRING, Array of CHAR, Array of BYTE.
- Über den Zeiger am Parameter OUT (hexadezimal) können Sie auf die folgenden Datentypen verweisen: Bitfolgen, Ganzzahlen, STRING, Array of CHAR, Array of BYTE.

Die Anzahl der zu konvertierenden ASCII-Zeichen legen Sie durch den Parameter N fest. Maximal können 32767 gültige ASCII-Zeichen konvertiert werden. Nur die Ziffern "0" bis "9" die Großbuchstaben "A" bis "F" und die Kleinbuchstaben "a" bis "f" können interpretiert werden. Alle anderen Zeichen werden in Nullen umgewandelt.

Da ein ASCII-Zeichen 8 Bits benötigt und eine Hexadezimalziffer nur 4 Bits, ist das Ausgabewort nur halb so lang wie das Eingabewort. Die ASCII-Zeichen werden nach dem Umwandeln in dem Ausgang in der gleichen Reihenfolge angeordnet, in der sie eingelesen wurden. Wenn es sich um eine ungerade Anzahl von ASCII-Zeichen handelt, wird die

Hexadezimalzahl in dem Halbbyte rechts von der zuletzt umgewandelten Hexadezimalzahl mit Nullen aufgefüllt.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ATH":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung                                |
|-----------|-------------|----------|------------------------------|---|
| IN        | Input       | VARIANT  | E, A, M, D, L                | Zeiger auf ASCII-Zeichenkette               |
| N         | Input       | UINT     | E, A, M, D, L oder Konstante | Anzahl der zu konvertierenden ASCII-Zeichen |
| RET_VAL   | Return      | WORD     | E, A, M, D, L                | Status der Anweisung                        |
| OUT       | Output      | VARIANT  | E, A, M, D, L                | Hexadezimalzahl                             |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode (W#16#...)* | Beschreibung   |
|------------------------|--|
| 0000                   | Kein Fehler  |
| 0007                   | Ungültiges Zeichen. Nur die folgenden ASCII-Zeichen dürfen verwendet werden: Ziffern "0" bis "9", Großbuchstaben "A" bis "F", Kleinbuchstaben "a" bis "f". |
| 8101                   | Ungültiger Zeiger am Parameter IN, z.B. weil auf nicht vorhandenen Datenbaustein verwiesen wird.   |
| 8182                   | Eingangspuffer ist zu klein für Daten am Parameter N.  |
| 8120                   | Ungültiges Format am Parameter IN.   |
| 8151                   | Nicht unterstützter Datentyp am Parameter IN.  |
| 8401                   | Ungültiger Zeiger am Parameter OUT, z.B. weil auf nicht vorhandenen Datenbaustein verwiesen wird.  |
| 8482                   | Ausgangspuffer ist zu klein für Daten am Parameter N.  |
| 8420                   | Ungültiges Format am Parameter OUT.  |
| 8451                   | Nicht unterstützter Datentyp am Parameter OUT.   |

\* Die Fehlercodes im Programmmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## ASCII-Zeichen und Hexadezimalwerte

Die folgende Tabelle zeigt die ASCII-Zeichen und die entsprechenden Hexadezimalwerte:

|     | ASCII-codierter Hexadezimalwert | Hexadezimalziffer |
|-----|---------------------------------|-------------------|
| "0" | 30                              | 0                 |
| "1" | 31                              | 1                 |
| "2" | 32                              | 2                 |
| "3" | 33                              | 3                 |

|     | ASCII-codierter<br>Hexadezimalwert | Hexadezimalziffer |
|-----|------------------------------------|-------------------|
| "4" | 34                                 | 4                 |
| "5" | 35                                 | 5                 |
| "6" | 36                                 | 6                 |
| "7" | 37                                 | 7                 |
| "8" | 38                                 | 8                 |
| "9" | 39                                 | 9                 |
| "A" | 41                                 | A                 |
| "B" | 42                                 | B                 |
| "C" | 43                                 | C                 |
| D   | 44                                 | D                 |
| E   | 45                                 | E                 |
| F   | 46                                 | F                 |

### Beispiel

Die folgende Tabelle zeigt Beispiele für die Konvertierung von ASCII-Zeichenketten in Hexadezimalzahl:

| IN           | N  | OUT           | ENO-Status |
|--------------|----|---------------|------------|
| '0123'       | 4  | 16#0123       | 1          |
| '123AFx1a23' | 10 | 16#123AF01a23 | 0          |

### HTA: Hexadezimalzahl in ASCII-Zeichenkette konvertieren

#### Beschreibung

Mit der Anweisung "HTA" konvertieren Sie die am Eingang IN angegebene Hexadezimalzahl in eine ASCII-Zeichenkette. Das Ergebnis der Konvertierung wird an der am Parameter OUT angegebenen Adresse abgelegt.

- Über den Zeiger am Parameter IN (hexadezimal) können Sie auf die folgenden Datentypen verweisen: Bitfolgen, Ganzzahlen, STRING, Array of CHAR, Array of BYTE.
- Über den Zeiger am Parameter OUT (ASCII) können Sie auf die folgenden Datentypen verweisen: STRING, Array of CHAR, Array of BYTE.

Die Anzahl der zu konvertierenden Hexadezimalbytes legen Sie durch den Parameter N fest. Da ein ASCII-Zeichen 8 Bits benötigt und eine Hexadezimalziffer nur 4 Bits, ist der Ausgabewert doppelt so lang wie der Eingabewert. Jedes Halbbyte der Hexadezimalzahl wird in ein Zeichen unter Beibehaltung der ursprünglichen Reihenfolge umgewandelt.

In die ASCII-Zeichenkette können maximal 32767 Zeichen geschrieben werden. Das Ergebnis der Konvertierung wird mit den Ziffern "0" bis "9" und Großbuchstaben "A" bis "F" dargestellt.

Kann das vollständige Ergebnis der Konvertierung nicht am Parameter OUT angezeigt werden, wird das Ergebnis nur teilweise in den Parameter geschrieben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "HTA":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung                                   |
|-----------|-------------|----------|------------------------------|--|
| IN        | Input       | VARIANT  | E, A, M, D, L                | Anfangsadresse der Hexadezimalziffer           |
| N         | Input       | UINT     | E, A, M, D, L oder Konstante | Anzahl der zu konvertierenden Hexadezimalbytes |
| RET_VAL   | Return      | WORD     | E, A, M, D, L                | Fehlermeldung                                  |
| OUT       | Output      | VARIANT  | E, A, M, D, L                | Adresse, an der das Ergebnis abgelegt wird.    |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...) | Beschreibung  |
|---------------------------|---|
| 0000                      | Kein Fehler   |
| 8101                      | Ungültiger Zeiger am Parameter IN, z. B. weil auf nicht vorhandenen Datenbaustein verwiesen wird. |
| 8182                      | Eingangspuffer ist zu klein für Daten am Parameter N.   |
| 8120                      | Ungültiges Format am Parameter IN.  |
| 8151                      | Nicht unterstützter Datentyp am Parameter IN.   |
| 8401                      | Ungültiger Zeiger am Parameter OUT, z.B. weil auf nicht vorhandenen Datenbaustein verwiesen wird. |
| 8482                      | Ausgangspuffer ist zu klein für Daten am Parameter N.   |
| 8420                      | Ungültiges Format am Parameter OUT.   |
| 8451                      | Nicht unterstützter Datentyp am Parameter OUT.  |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## ASCII-Zeichen und Hexadezimalwerte

Die folgende Tabelle zeigt die ASCII-Zeichen und die entsprechenden Hexadezimalwerte:

| Hexadezimalziffer | ASCII-codierter Hexadezimalwert | ASCII-Zeichen |
|-------------------|---------------------------------|---------------|
| 0                 | 30                              | "0"           |
| 1                 | 31                              | "1"           |
| 2                 | 32                              | "2"           |
| 3                 | 33                              | "3"           |
| 4                 | 34                              | "4"           |
| 5                 | 35                              | "5"           |
| 6                 | 36                              | "6"           |

| Hexadezimalziffer | ASCII-codierter Hexadezimalwert | ASCII-Zeichen |
|-------------------|---------------------------------|---------------|
| 7                 | 37                              | "7"           |
| 8                 | 38                              | "8"           |
| 9                 | 39                              | "9"           |
| A                 | 41                              | "A"           |
| B                 | 42                              | "B"           |
| C                 | 43                              | "C"           |
| D                 | 44                              | "D"           |
| E                 | 45                              | "E"           |
| F                 | 46                              | "F"           |

**Beispiel**

Die folgende Tabelle zeigt Beispiele für die Konvertierung von Hexadezimalzahlen in ASCII-Zeichenketten:

| IN            | N | OUT        | ENO-Status |
|---------------|---|------------|------------|
| W#16#0123     | 2 | '0123'     | 1          |
| 16#123AF01023 | 4 | '123AF010' | 0          |

**Weitere Anweisungen**

**LEN: Länge einer Zeichenkette ermitteln**

**Beschreibung**

Eine Variable vom Datentyp STRING enthält zwei Längen: die maximale Länge und die aktuelle Länge (das ist die Anzahl der momentan gültigen Zeichen).

- Die Maximallänge der Zeichenkette wird für jede Variable am Schlüsselwort STRING in eckigen Klammern angegeben. Die Anzahl der durch eine Zeichenkette belegten Bytes ist um 2 größer als die maximale Länge.
- Die aktuelle Länge repräsentiert die Anzahl der tatsächlich belegten Zeichenplätze. Die aktuelle Länge ist kleiner oder gleich der maximalen Länge.

Mit der Anweisung "LEN" fragen Sie die aktuelle Länge der am Eingangsparameter IN angegebenen Zeichenkette ab und geben diese als numerischen Wert am Ausgangsparameter OUT aus. Ein Leerstring ("") hat die Länge Null.



Wenn Fehler während der Bearbeitung der Anweisung auftreten, wird ein Leerstring ausgegeben.

#### Hinweis

##### Auslesen der maximalen Länge

Mit der Anweisung "MAX\_LEN (Seite 2373)" können Sie die aktuelle Länge einer Zeichenkette auslesen.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LEN":

| Parameter | Deklaration | Datentyp                     | Speicherbereich     | Beschreibung                |
|-----------|-------------|------------------------------|---------------------|-----------------------------|
| IN        | Input       | STRING                       | D, L oder Konstante | Zeichenkette                |
| OUT       | Return      | INT, DINT,<br>REAL,<br>LREAL | E, A, M, D, L       | Anzahl der gültigen Zeichen |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## CONCAT: Zeichenketten zusammenfassen

### Beschreibung

Mit der Anweisung "CONCAT" verbinden Sie die Zeichenkette am Eingangsparameter IN1 mit der Zeichenkette am Eingangsparameter IN2. Das Ergebnis wird am Ausgangsparameter OUT in Format STRING ausgegeben. Wenn die Ergebniszeichenkette länger als die am Ausgangsparameter OUT angegebene Variable ist, wird die Ergebniszeichenkette auf die verfügbare Länge begrenzt.

Wenn Fehler während der Bearbeitung der Anweisung auftreten und der Ausgangsparameter OUT beschrieben werden kann, wird ein Leerstring ausgegeben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CONCAT":

| Parameter | Deklaration | Datentyp | Speicherbereich     | Beschreibung         |
|-----------|-------------|----------|---------------------|----------------------|
| IN1       | Input       | STRING   | D, L oder Konstante | Zeichenkette         |
| IN2       | Input       | STRING   | D, L oder Konstante | Zeichenkette         |
| OUT       | Return      | STRING   | D, L                | Ergebniszeichenkette |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## LEFT: Linke Zeichen einer Zeichenkette lesen

### Beschreibung

Mit der Anweisung "LEFT" extrahieren Sie eine Teilzeichenkette ab dem ersten Zeichen der Zeichenkette am Eingangsparameter IN. Die Anzahl der zu extrahierenden Zeichen legen Sie am Parameter L fest. Die extrahierten Zeichen werden am Ausgangsparameter OUT in Format STRING ausgegeben.

Wenn die Anzahl der zu extrahierenden Zeichen größer als die aktuelle Länge der Zeichenkette ist, liefert der Ausgangsparameter OUT die Eingangszeichenkette als Ergebnis zurück. Beim Wert "0" am Parameter L oder bei einem Leerstring als Eingangswert wird ein Leerstring zurückgeliefert. Wenn der Wert am Parameter L negativ ist, wird ein Leerstring ausgegeben.

Wenn Fehler während der Bearbeitung der Anweisung auftreten und der Ausgangsparameter OUT beschrieben werden kann, wird ein Leerstring ausgegeben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LEFT":

| Parameter | Deklaration | Datentyp               | Speicherbereich              | Beschreibung                         |
|-----------|-------------|------------------------|------------------------------|--------------------------------------|
| IN        | Input       | STRING                 | D, L oder Konstante          | Zeichenkette                         |
| L         | Input       | BYTE, INT, SINT, USINT | E, A, M, D, L oder Konstante | Anzahl der zu extrahierenden Zeichen |
| OUT       | Return      | STRING                 | D, L                         | Extrahierte Teilzeichenkette         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## RIGHT: Rechte Zeichen einer Zeichenkette lesen

### Beschreibung

Mit der Anweisung extrahieren Sie die letzten L Zeichen einer Zeichenkette am Eingangsparameter IN. Die Anzahl der zu extrahierenden Zeichen legen Sie am Parameter L fest. Die extrahierten Zeichen werden am Ausgangsparameter OUT in Format STRING ausgegeben.

Wenn die Anzahl der zu extrahierenden Zeichen größer als die aktuelle Länge der Zeichenkette ist, liefert der Ausgangsparameter OUT die Eingangszeichenkette als Ergebnis zurück. Beim Wert "0" am Parameter L oder bei einem Leerstring als Eingangswert wird ein Leerstring zurückgeliefert. Wenn der Wert am Parameter L negativ ist, wird ein Leerstring ausgegeben.

Wenn Fehler während der Bearbeitung der Anweisung auftreten und der Ausgangsparameter OUT beschrieben werden kann, wird ein Leerstring ausgegeben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RIGHT":

| Parameter | Deklaration | Datentyp               | Speicherbereich              | Beschreibung                         |
|-----------|-------------|------------------------|------------------------------|--------------------------------------|
| IN        | Input       | STRING                 | D, L oder Konstante          | Zeichenkette                         |
| L         | Input       | BYTE, INT, SINT, USINT | E, A, M, D, L oder Konstante | Anzahl der zu extrahierenden Zeichen |
| OUT       | Return      | STRING                 | D, L                         | Extrahierte Teilzeichenkette         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## MID: Mittlere Zeichen einer Zeichenkette lesen

### Beschreibung

Mit der Anweisung extrahieren Sie einen Teil der Zeichenkette am Eingangsparameter IN. Mit dem Parameter P legen Sie die Position des ersten zu extrahierenden Zeichens fest. Die Länge der zu extrahierenden Zeichenkette bestimmen Sie durch den Parameter L. Die extrahierte Teilzeichenkette wird am Ausgangsparameter OUT ausgegeben.

Bei der Ausführung der Anweisung sind die folgenden Regeln zu beachten:

- Wenn die Anzahl der zu extrahierenden Zeichen die aktuelle Länge der Zeichenkette am Eingangsparameter IN überschreitet, wird eine Teilzeichenkette ausgegeben, die an der Zeichenposition P beginnt und bis zum Ende der Zeichenkette fortgesetzt wird.
- Wenn die am Parameter P angegebene Zeichenposition außerhalb der aktuellen Länge der Zeichenkette am Eingangsparameter IN liegt, wird eine leere Zeichenkette am Ausgangsparameter OUT ausgegeben.
- Wenn der Wert des Parameters P oder L gleich Null oder negativ ist, wird eine leere Zeichenkette am Ausgangsparameter OUT ausgegeben.

Wenn Fehler während der Bearbeitung der Anweisung auftreten und der Ausgangsparameter OUT beschrieben werden kann, wird ein Leerstring ausgegeben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MID":

| Parameter | Deklaration | Datentyp               | Speicherbereich              | Beschreibung  |
|-----------|-------------|------------------------|------------------------------|---|
| IN        | Input       | STRING                 | D, L oder Konstante          | Zeichenkette  |
| L         | Input       | BYTE, INT, SINT, USINT | E, A, M, D, L oder Konstante | Länge der zu extrahierenden Zeichenkette                            |
| P         | Input       | BYTE, INT, SINT, USINT | E, A, M, D, L oder Konstante | Position des ersten zu extrahierenden Zeichens (erstes Zeichen = 1) |
| OUT       | Return      | STRING                 | D, L                         | Extrahierte Teilzeichenkette  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## DELETE: Zeichen in Zeichenkette löschen

### Beschreibung

Mit der Anweisung löschen Sie einen Teil der Zeichenkette am Eingangsparameter IN. Die Position des ersten zu löschenden Zeichens legen Sie mit dem Parameter P fest. Am Parameter L geben Sie die Anzahl der zu löschenden Zeichen an. Am Ausgangsparameter OUT wird die verbliebene Teilzeichenkette in Format STRING ausgegeben.

Bei der Ausführung der Anweisung sind die folgenden Regeln zu beachten:

- Wenn der Wert am Parameter P kleiner oder gleich Null ist, wird am Ausgangsparameter OUT eine leere Zeichenkette ausgegeben.
- Wenn der Wert am Parameter P größer als die aktuelle Länge der Zeichenkette am Eingang IN ist, wird die Eingangszeichenkette am Ausgangsparameter OUT zurückgeliefert.
- Wenn der Wert am Parameter L gleich Null ist, wird die Eingangszeichenkette am Ausgangsparameter OUT zurückgeliefert.
- Wenn die Anzahl der zu löschenden Zeichen am Parameter L größer als die Länge der Zeichenkette am Eingangsparameter IN ist, wird eine leere Zeichenkette ausgegeben.
- Wenn der Wert am Parameter L negativ ist, wird eine leere Zeichenkette ausgegeben.

Wenn Fehler während der Bearbeitung der Anweisung auftreten und der Ausgangsparameter OUT beschrieben werden kann, wird ein Leerstring ausgegeben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DELETE":

| Parameter | Deklaration | Datentyp               | Speicherbereich              | Beschreibung                               |
|-----------|-------------|------------------------|------------------------------|--|
| IN        | Input       | STRING                 | D, L oder Konstante          | Zeichenkette                               |
| L         | Input       | BYTE, INT, SINT, USINT | E, A, M, D, L oder Konstante | Anzahl der zu löschenden Zeichen           |
| P         | Input       | BYTE, INT, SINT, USINT | E, A, M, D, L oder Konstante | Position des ersten zu löschenden Zeichens |
| OUT       | Return      | STRING                 | D, L                         | Ergebniszeichenkette                       |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## INSERT: Zeichen in Zeichenkette einfügen

### Beschreibung

Mit der Anweisung fügen Sie die Zeichenkette am Eingangsparameter IN2 in die Zeichenkette am Eingangsparameter IN1 ein. Durch den Parameter P legen Sie die Position des Zeichens fest, ab dem die Zeichen eingefügt werden. Das Ergebnis wird am Ausgangsparameter OUT im Format STRING ausgegeben.

Bei der Ausführung der Anweisung sind die folgenden Regeln zu beachten:

- Wenn der Wert am Parameter P die aktuelle Länge der Zeichenkette am Eingangsparameter IN1 überschreitet, wird die Zeichenkette des Eingangsparameters IN2 an die Zeichenkette des Eingangsparameters IN1 angehängt.
- Wenn der Wert am Parameter P null ist, wird am Ausgang OUT die Zeichenkette am Parameter IN2 gefolgt von der Zeichenkette am Parameter IN1 ausgegeben.
- Wenn der Wert am Parameter P negativ ist, wird am Ausgang OUT eine leere Zeichenkette ausgegeben.
- Wenn die Ergebniszeichenkette länger ist als die am Ausgangsparameter OUT angegebene Variable, wird die Ergebniszeichenkette auf die verfügbare Länge begrenzt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "INSERT":

| Parameter | Deklaration | Datentyp                     | Speicherbereich                 | Beschreibung              |
|-----------|-------------|------------------------------|---------------------------------|---------------------------|
| IN1       | Input       | STRING                       | D, L oder Konstante             | Zeichenkette              |
| IN2       | Input       | STRING                       | D, L oder Konstante             | Einzufügende Zeichenkette |
| P         | Input       | BYTE,<br>INT, SINT,<br>USINT | E, A, M, D, L oder<br>Konstante | Einfügeposition           |
| OUT       | Return      | STRING                       | D, L                            | Ergebniszeichenkette      |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## REPLACE: Zeichen in Zeichenkette ersetzen

### Beschreibung

Mit der Anweisung ersetzen Sie einen Teil der Zeichenkette am Eingang IN1 durch die Zeichenkette am Eingangsparameter IN2. Die Position des ersten zu ersetzenden Zeichens legen Sie am Parameter P fest. Die Anzahl der zu ersetzenden Zeichen geben Sie am Parameter L an. Das Ergebnis wird am Ausgangsparameter OUT in Format STRING ausgegeben.

Bei der Ausführung der Anweisung sind die folgenden Regeln zu beachten:

- Wenn der Wert am Parameter P kleiner oder gleich Null ist, wird am Ausgangsparameter OUT eine leere Zeichenkette ausgegeben.
- Wenn der Wert am Parameter L kleiner als Null ist, wird am Ausgangsparameter OUT eine leere Zeichenkette ausgegeben.
- Wenn P gleich eins ist, wird die Zeichenkette am Eingang IN1 ab dem ersten Zeichen (einschließlich) ersetzt.
- Wenn der Wert am Parameter P die aktuelle Länge der Zeichenkette am Eingangsparameter IN1 überschreitet, wird die Zeichenkette des Eingangsparameters IN2 an die Zeichenkette des Eingangsparameters IN1 angehängt.
- Wenn die Ergebniszeichenkette länger ist als die am Ausgangsparameter OUT angegebene Variable, wird die Ergebniszeichenkette auf die verfügbare Länge begrenzt.
- Wenn der Wert am Parameter L gleich Null ist, werden keine Zeichen ersetzt, sondern eingefügt. Es gelten die Bedingungen wie bei der Anweisung INSERT. Siehe auch: INSERT: Zeichen in Zeichenkette einfügen (Seite 2383)

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "REPLACE":

| Parameter | Deklaration | Datentyp               | Speicherbereich              | Beschreibung                                  |
|-----------|-------------|------------------------|------------------------------|---|
| IN1       | Input       | STRING                 | D, L oder Konstante          | Zeichenkette, deren Zeichen ersetzt werden.   |
| IN2       | Input       | STRING                 | D, L oder Konstante          | Zeichenkette, deren Zeichen eingefügt werden. |
| L         | Input       | BYTE, INT, SINT, USINT | E, A, M, D, L oder Konstante | Anzahl der zu ersetzenden Zeichen             |
| P         | Input       | BYTE, INT, SINT, USINT | E, A, M, D, L oder Konstante | Position des ersten zu ersetzenden Zeichens   |
| OUT       | Return      | STRING                 | D, L                         | Ergebniszeichenkette                          |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## FIND: Zeichen in Zeichenkette finden

### Beschreibung

Mit der Anweisung können Sie die Zeichenkette am Eingangsparameter IN1 nach einem bestimmten Zeichen oder einer bestimmten Zeichenkette durchsuchen.

- Den zu suchenden Wert geben Sie am Eingangsparameter IN2 an. Die Suche erfolgt von links nach rechts.
- Am Ausgangsparameter OUT wird die Position des ersten Auftretens ausgegeben. Wenn die Suche keine Übereinstimmung ergibt, wird am Ausgangsparameter OUT der Wert "0" ausgegeben.

Wenn Fehler während der Bearbeitung der Anweisung auftreten, wird ein Leerstring ausgegeben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "FIND":

| Parameter | Deklaration | Datentyp                     | Speicherbereich                                   | Beschreibung                      |
|-----------|-------------|------------------------------|---|-----------------------------------|
| IN1       | Input       | STRING                       | D, L oder Konstante                               | Zeichenkette, die durchsucht wird |
| IN2       | Input       | STRING,<br>CHAR              | D, L oder Konstante<br>(Bei CHAR auch E,<br>A, M) | Zu suchende Zeichen               |
| OUT       | Return      | DINT, INT,<br>LREAL,<br>REAL | E, A, M, D, L                                     | Zeichenposition                   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### 9.7.3.3 Prozessabbild

#### UPDAT\_PI: Prozessabbild der Eingänge aktualisieren

##### Beschreibung

Mit der Anweisung aktualisieren Sie das OB 1-Prozessabbild (=Teilprozessabbild 0) der Eingänge oder ein per Projektierung definiertes Teilprozessabbild der Eingänge.

Falls Sie als Meldeverfahren für die systemseitige Prozessabbildaktualisierung das wiederholte Melden aller Peripheriezugriffsfehler projektiert haben, wird die Aktualisierung des ausgewählten Prozessabbilds stets durchgeführt.

Andernfalls wird diese Aktualisierung nur dann durchgeführt, wenn das ausgewählte Teilprozessabbild nicht systemseitig aktualisiert wird, d. h.

- wenn Sie dieses Teilprozessabbild keinem Alarm-OB zugeordnet haben, oder
- wenn Sie das Teilprozessabbild 0 ausgewählt und (per Projektierung) die Aktualisierung des OB 1-Teilprozessabbilds ausgeschaltet haben.

---

##### Hinweis

Jede logische Adresse, die Sie per Projektierung einem Teilprozessabbild der Eingänge zugeordnet haben, gehört nicht mehr zum OB 1-Prozessabbild der Eingänge. Ein Teilprozessabbild, das Sie mit "UPDAT\_PI" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "SYNC\_PI (Seite 2388)" aktualisieren.

---

Die systemseitige Aktualisierung des OB 1-Prozessabbilds der Eingänge und der Teilprozessabbilder der Eingänge, die Sie einem Alarm-OB zugeordnet haben, findet unabhängig von "UPDAT\_PI"-Aufrufen statt.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UPDAT\_PI":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| PART      | Input       | PIP      | E, A, M, D, L oder Konstante | Nummer des zu aktualisierenden Teilprozessabbilds der Eingänge. Maximaler Wertbereich (Wertbereich ist CPU-abhängig): 0 bis 31 (0 bedeutet OB 1-Prozessabbild, n mit $1 \leq n \leq 31$ bedeutet Teilprozessabbild n). |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlerinformation  |
| FLADDR    | Output      | WORD     | E, A, M, D, L                | Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Erläuterung  |
|---|--|
| 0000  | Es ist kein Fehler aufgetreten.  |
| 8090  | Unzulässiger Wert beim Parameter PART  |
| 8091  | Das angegebene Teilprozessabbild wurde noch nicht definiert oder befindet sich nicht im zulässigen Prozessabbildbereich der CPU. |
| 8092  | Das Teilprozessabbild wird systemseitig aktualisiert und kann für den Parameter PART nicht verwendet werden.                     |
| 8093  | Die Aktualisierung des Teilprozessabbilds wird momentan in einem anderen OB verarbeitet.   |
| 80A0  | Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.  |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Hinweis

Falls Sie die Anweisung für Prozessabbildteile von DP-Normslaves einsetzen, für die Sie Konsistenzbereiche größer als 32 Bytes definiert haben, sind auch die Fehlercodes der Anweisung "DPRD\_DAT (Seite 2436)" möglich.



## UPDAT\_PO: Prozessabbild der Ausgänge aktualisieren

### Beschreibung

Mit der Anweisung übertragen Sie die Signalzustände des OB 1-Prozessabbilds (=Teilprozessabbild 0) der Ausgänge oder eines per Projektierung definierten Teilprozessabbilds der Ausgänge zu den Ausgabebaugruppen.

Falls Sie für das ausgewählte Teilprozessabbild einen Konsistenzbereich definiert haben, werden die zugehörigen Daten konsistent an die entsprechende Peripheriebaugruppe übertragen.

### Hinweis

Jede logische Adresse, die Sie per Projektierung einem Teilprozessabbild der Ausgänge zugeordnet haben, gehört nicht mehr zum OB 1-Prozessabbild der Ausgänge. Ausgänge, die Sie mit "UPDAT\_PO" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "SYNC\_PO (Seite 2390)" aktualisieren.

Die systemseitige Übertragung des OB 1-Prozessabbilds der Ausgänge und der Teilprozessabbilder der Ausgänge, die Sie einem Alarm-OB zugeordnet haben, zu den Ausgabebaugruppen erfolgt unabhängig von "UPDAT\_PO"-Aufrufen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "UPDAT\_PO":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| PART      | Input       | PIP      | E, A, M, D, L oder Konstante | Nummer des zu übertragenden Teilprozessabbilds der Ausgänge.<br>Maximaler Wertbereich (Wertbereich ist CPU-abhängig): 0 bis 31.<br>(0 bedeutet OB 1-Prozessabbild, n mit $1 \leq n \leq 31$ bedeutet Teilprozessabbild n) |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlerinformation   |
| FLADDR    | Output      | WORD     | E, A, M, D, L                | Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Erläuterung  |
|---|--|
| 0000  | Es ist kein Fehler aufgetreten.  |
| 8090  | Unzulässiger Wert beim Parameter PART  |
| 8091  | Das angegebene Teilprozessabbild wurde noch nicht definiert oder befindet sich nicht im zulässigen Prozessabbildbereich der CPU. |
| 8092  | Das Teilprozessabbild wird systemseitig aktualisiert und kann für den Parameter PART nicht verwendet werden.                     |
| 8093  | Die Aktualisierung des Teilprozessabbildes wird in einem anderen OB bearbeitet.  |
| 80A0  | Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.  |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Hinweis

Falls Sie die Anweisung für Prozessabbildteile von DP-Normslaves einsetzen, für die Sie Konsistenzbereiche größer als 32 Bytes definiert haben, sind auch die Fehlercodes der Anweisung "DPWR\_DAT (Seite 2438)" möglich.

## SYNC\_PI: Prozessabbild der Eingänge synchronisieren

### Beschreibung

Mit der Anweisung "SYNC\_PI" aktualisieren Sie ein Teilprozessabbild der Eingänge takt synchron. Ein an einen DP-Takt bzw. PN-Sendetakt angebundenes Anwenderprogramm kann mit dieser Anweisung die erfassten Eingangsdaten in einem Teilprozessabbild der Eingänge takt synchron und konsistent aktualisieren.

### Aufruf

"SYNC\_PI" ist unterbrechbar und kann nur in den OBs 61, 62, 63 und 64 aufgerufen werden.

### Hinweis

Der Aufruf der Anweisung "SYNC\_PI" in den OBs 61 bis 64 ist nur dann erlaubt, wenn Sie in der Hardware-Konfiguration das betroffene Teilprozessabbild dem zugehörigen OB zugeordnet haben.

Ein Teilprozessabbild, das Sie mit "SYNC\_PI" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "UPDAT\_PI (Seite 2385)" aktualisieren.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SYNC\_PI":

| Parameter | Deklaration | Datentyp | Speicherbereich                 | Wertebereich | Beschreibung  |
|-----------|-------------|----------|---------------------------------|--------------|---|
| PART      | Input       | PIP      | E, A, M, D, L<br>oder Konstante | 1 bis 31     | Nummer des Teilprozessabbildes der Eingänge, welches taktsynchron aktualisiert werden soll. |
| RET_VAL   | Return      | INT      | E, A, M, D, L                   | -            | Fehlerinformation   |
| FLADDR    | Output      | WORD     | E, A, M, D, L                   |              | Adresse des ersten Fehler verursachenden Bytes, falls ein Zugriffsfehler auftrat.           |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Ereignisklasse<br>Fehlercode<br>(W#16#...)*   | Erläuterung  |
|---|--|
| 0000  | Es ist kein Fehler aufgetreten.  |
| 0001  | Konsistenzwarnung. Die Aktualisierung des Teilprozessabbildes wurde auf zwei DP- bzw. PN-Zyklen verteilt. Daten innerhalb eines Slaves bzw. IO-Devices sind jedoch konsistent übertragen worden.   |
| 8090  | Unzulässiger Wert am Parameter PART oder Aktualisierung des angegebenen Teilprozessabbildes der Eingänge ist in diesem OB nicht erlaubt. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.  |
| 8091  | Das angegebene Teilprozessabbild wurde noch nicht definiert oder befindet sich nicht im zulässigen Prozessabbildbereich der CPU. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.  |
| 8092  | Das Teilprozessabbild wird systemseitig aktualisiert und kann für den Parameter PART nicht verwendet werden.   |
| 8093  | Die Aktualisierung des Teilprozessabbildes wird in einem anderen OB bearbeitet.  |
| 80A0  | Bei der Aktualisierung wurde ein Zugriffsfehler erkannt. Betroffene Eingänge wurden auf "0" gesetzt.   |
| 80A1  | Aktualisierungszeitpunkt liegt nach dem zulässigen Zugriffsfenster. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.<br>Der DP- bzw. PN-Zyklus ist zu kurz, um ausreichend Zeit für die Bearbeitung der Anweisung zu gewährleisten. Sie müssen daher die Zeiten TDP (auch bekannt als T_DC), Ti und To vergrößern.   |
| 80A2  | Zugriffsfehler mit Konsistenzwarnung<br>Bei der Aktualisierung des angegebenen Teilprozessabbildes der Eingänge wurde ein Zugriffsfehler erkannt mit gleichzeitiger Konsistenzwarnung. <ul style="list-style-type: none"> <li>Die Daten der fehlerhaften Eingänge wurden nicht von der Peripherie gelesen. Im Teilprozessabbild der Eingänge werden die betroffenen Eingänge auf Null gesetzt.</li> <li>Die Aktualisierung der nicht vom Zugriffsfehler betroffenen Eingangsdaten wurde auf zwei DP- bzw. PN-Zyklen verteilt.</li> </ul> |
| 80C1  | Aktualisierungszeitpunkt liegt vor dem zulässigen Zugriffsfenster. Das Teilprozessabbild der Eingänge wurde nicht aktualisiert.  |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**Hinweis**

Falls Sie die Anweisung "SYNC\_PI" für Prozessabbildteile von DP-Normslaves einsetzen, für die Sie Konsistenzbereiche größer als 32 Bytes definiert haben, sind auch die Fehlercodes der Anweisung "DPRD\_DAT (Seite 2436)" möglich.

**SYNC\_PO: Prozessabbild der Ausgänge synchronisieren**

**Beschreibung**

Mit der Anweisung "SYNC\_PO" aktualisieren Sie ein Teilprozessabbild der Ausgänge takt synchron. Ein an einen DP-Takt bzw. PN-Sendetakt angebundenes Anwenderprogramm kann mit dieser Anweisung die berechneten Ausgangsdaten eines Teilprozessabbildes der Ausgänge takt synchron und konsistent in die Peripherie übertragen.

**Aufruf**

"SYNC\_PO" ist unterbrechbar und kann nur in den OBs 61, 62, 63 und 64 ausgerufen werden.

**Hinweis**

Der Aufruf der Anweisung "SYNC\_PO" in den OBs 61 bis 64 ist nur dann erlaubt, wenn Sie in der Hardware-Konfiguration das betroffene Teilprozessabbild dem zugehörigen OB zugeordnet haben.

Ein Teilprozessabbild, das Sie mit "SYNC\_PO" aktualisieren, dürfen Sie nicht gleichzeitig mit der Anweisung "UPDAT\_PO (Seite 2387)" aktualisieren.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "SYNC\_PO":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Wertebereich | Beschreibung   |
|-----------|-------------|----------|------------------------------|--------------|--|
| PART      | Input       | PIP      | E, A, M, D, L oder Konstante | 1 bis 31     | Nummer des Teilprozessabbildes der Ausgänge, welches takt synchron aktualisiert werden soll.           |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | -            | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. |
| FLADDR    | Output      | WORD     | E, A, M, D, L                | -            | Adresse des ersten Fehler verursachenden Bytes.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode (W#16#...)*  | Erläuterung  |
|---|--|
| 0000  | Es ist kein Fehler aufgetreten.  |
| 0001  | Konsistenzwarnung. Die Aktualisierung des Teilprozessabbildes wurde auf zwei DP- bzw. PN-Zyklen verteilt. Daten innerhalb eines Slaves bzw. IO-Devices sind jedoch konsistent übertragen worden.   |
| 8090  | Unzulässiger Wert am Parameter PART oder Aktualisierung des angegebenen Teilprozessabbildes der Ausgänge ist in diesem OB nicht erlaubt. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.  |
| 8091  | Das angegebene Teilprozessabbild wurde noch nicht definiert oder befindet sich nicht im zulässigen Prozessabbildbereich der CPU. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.  |
| 8092  | Das Teilprozessabbild wird systemseitig aktualisiert und kann für den Parameter PART nicht verwendet werden.   |
| 8093  | Die Aktualisierung des Teilprozessabbildes wird in einem anderen OB bearbeitet.  |
| 80A0  | Bei der Aktualisierung des angegebenen Teilprozessabbildes der Ausgänge wurde ein Zugriffsfehler erkannt. Fehlerhafte Ausgänge wurden nicht an die Peripherie übertragen. Im Teilprozessabbild der Ausgänge bleiben diese Ausgänge unverändert.  |
| 80A1  | Zugriffsfehler mit Konsistenzwarnung<br>Bei der Aktualisierung des angegebenen Teilprozessabbildes der Ausgänge wurde ein Zugriffsfehler erkannt mit gleichzeitiger Konsistenzwarnung. <ul style="list-style-type: none"> <li>Die Daten der fehlerhaften Ausgänge wurden nicht an die Peripherie übertragen. Im Teilprozessabbild der Ausgänge bleiben die betroffenen Ausgänge unverändert.</li> <li>Die Aktualisierung der nicht vom Zugriffsfehler betroffenen Ausgangsdaten wurde auf zwei DP- bzw. PN-Zyklen verteilt.</li> </ul> |
| 80A2  | Aktualisierungszeitpunkt liegt nach dem zulässigen Zugriffsfenster. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.   |
| 80C1  | Aktualisierungszeitpunkt liegt vor dem zulässigen Zugriffsfenster. Ausgänge wurden nicht an die Peripherie übertragen. Das Teilprozessabbild der Ausgänge bleibt unverändert.  |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**Hinweis**

Falls Sie die Anweisung "SYNC\_PO" für Prozessabbildteile von DP-Normslaves einsetzen, für die Sie Konsistenzbereiche größer als 32 Bytes definiert haben, sind auch die Fehlercodes der Anweisung "DPWR\_DAT (Seite 2438)" möglich.

### 9.7.3.4 Dezentrale Peripherie

#### RDREC: Datensatz lesen

##### Beschreibung

Mit der Anweisung "RDREC" lesen Sie den Datensatz mit der Nummer INDEX von der mittels ID adressierten Komponente. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Komponente (PROFIBUS DP oder PROFINET IO) handeln.

- Mit MLEN geben Sie vor, wie viele Bytes Sie maximal lesen möchten. Wenn Sie am Parameter MLEN als Länge "0" wählen, wird der vollständige Datensatz an dem Parameter RECORD geschrieben
- Den Zielbereich RECORD sollten Sie mindestens MLEN Bytes lang wählen.
- Der Wert TRUE des Ausgangsparameters VALID zeigt an, dass der Datensatz erfolgreich in den Zielbereich RECORD übertragen wurde. In diesem Fall enthält der Ausgangsparameter LEN die Länge der gelesenen Daten in Bytes.
- Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

---

##### Hinweis

Die Schnittstelle der Anweisung "RDREC" ist identisch mit der des in der Norm "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3" definierten FB "RDREC".

---

##### Arbeitsweise

"RDREC" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie "RDREC" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die mittleren zwei Bytes von STATUS dem Ausgangsparameter RET\_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

Die Datensatzübertragung ist abgeschlossen, wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RDREC":

| Parameter | Deklaration | Datentyp*                                | Speicherbereich                    | Beschreibung   |
|-----------|-------------|--|------------------------------------|--|
| REQ       | Input       | BOOL                                     | E, A, M, D, L, T, Z oder Konstante | REQ = 1: Datensatzübertragung durchführen  |
| ID        | Input       | HW_IO                                    | E, A, M, L oder Konstante          | Hardware-Kennung der Hardware-Komponente (DP/PROFINET IO)<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Komponente oder des Interface in der Hardware-Konfiguration hinterlegt. |
| INDEX     | Input       | BYTE, DINT, INT, SINT, UINT, USINT, WORD | E, A, M, D, L oder Konstante       | Datensatznummer  |
| MLEN      | Input       | BYTE, UINT, USINT                        | E, A, M, D, L oder Konstante       | Maximale Länge der zu lesenden Datensatzinformation in Byte.   |
| VALID     | Output      | BOOL                                     | E, A, M, D, L                      | Neuer Datensatz wurde empfangen und ist gültig   |
| BUSY      | Output      | BOOL                                     | E, A, M, D, L                      | BUSY = 1: Der Lesevorgang ist noch nicht beendet.  |
| ERROR     | Output      | BOOL                                     | E, A, M, D, L                      | ERROR = 1: Beim Lesevorgang trat ein Fehler auf.   |
| STATUS    | Output      | DWORD                                    | E, A, M, D, L                      | Bausteinstatus bzw. Fehlerinformation.   |
| LEN       | Output      | UINT                                     | E, A, M, D, L                      | Länge der gelesenen Datensatzinformation.  |
| RECORD    | InOut       | VARIANT                                  | E, A, M, D, L                      | Zielbereich für den gelesenen Datensatz.   |

\* Bei AWL erfolgt keine implizierte Konvertierung, weshalb der Umfang der zulässigen Datentypen eingeschränkt sein kann. Beachten Sie bei der Programmierung in AWL die jeweils zulässigen Datentypen im Tooltip des Parameters.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Hinweis

Falls Sie "RDREC" für das Lesen eines Datensatzes bei PROFINET IO einsetzen, werden negative Werte in den Parametern INDEX, MLEN und LEN als 16-Bit vorzeichenlose ganze Zahl interpretiert.

## Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Parameter STATUS (Seite 2402).

## Siehe auch

Grundlagen zum Bausteinzugriff (Seite 1034)

## WRREC: Datensatz schreiben

### Beschreibung

Mit der Anweisung "WRREC" übertragen Sie den Datensatz RECORD zu der mittels ID adressierten Komponente. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Komponente (PROFIBUS DP oder PROFINET IO) handeln.

- Mit LEN geben Sie die Länge des zu übertragenden Datensatzes in Bytes vor. Den Quellbereich RECORD sollten Sie daher mindestens LEN Bytes lang wählen.
- Der Wert TRUE des Ausgangsparameters DONE zeigt an, dass der Datensatz erfolgreich übertragen wurde.
- Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

---

### Hinweis

Die Schnittstelle der Anweisung "WRREC" ist identisch mit der des in der Norm "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3" definierten FB "WRREC".

---

### Arbeitsweise

"WRREC" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie "WRREC" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die mittleren zwei Bytes von STATUS dem Ausgangsparameter RET\_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

Beachten Sie, dass Sie dem Aktualparameter von RECORD bei allen zu ein und demselben Auftrag gehörenden Aufrufen von "WRREC" denselben Wert zuweisen. Dasselbe gilt für die Aktualparameter von LEN.

Die Datensatzübertragung ist abgeschlossen, wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat.



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WRREC":

| Parameter | Deklaration | Datentyp          | Speicherbereich                    | Beschreibung  |
|-----------|-------------|-------------------|------------------------------------|---|
| REQ       | Input       | BOOL              | E, A, M, D, L, T, Z oder Konstante | REQ = 1: Datensatzübertragung durchführen   |
| ID        | Input       | HW_IO             | E, A, M, L oder Konstante          | Identifikationsnummer der Hardware-Komponente (DP/PROFINET IO)<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Komponente oder des Interface in der Hardware-Konfiguration hinterlegt. |
| INDEX     | Input       | DINT              | E, A, M, D, L oder Konstante       | Datensatznummer   |
| LEN       | Input       | BYTE, UINT, USINT | E, A, M, D, L oder Konstante       | (ausgeblendet)<br>maximale Länge des zu übertragenden Datensatzes in Bytes  |
| DONE      | Output      | BOOL              | E, A, M, D, L                      | Datensatz wurde übertragen  |
| BUSY      | Output      | BOOL              | E, A, M, D, L                      | BUSY = 1: Der Schreibvorgang ist noch nicht beendet.  |
| ERROR     | Output      | BOOL              | E, A, M, D, L                      | ERROR = 1: Beim Schreibvorgang trat ein Fehler auf.   |
| STATUS    | Output      | DWORD             | E, A, M, D, L                      | Bausteinstatus bzw. Fehlerinformation<br>Zur Interpretation des Parameters STATUS siehe Parameter STATUS (Seite 2402).  |
| RECORD    | InOut       | VARIANT           | E, A, M, D, L                      | Datensatz   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Hinweis

Falls Sie "WRREC" für das Schreiben eines Datensatzes bei PROFINET IO einsetzen, werden negative Werte in den Parametern INDEX und LEN als 16-Bit vorzeichenlose ganze Zahl interpretiert.

## Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Parameter STATUS (Seite 2402).

## GETIO: Prozessabbild lesen

### Beschreibung

Mit der Anweisung "GETIO" lesen Sie alle Eingänge eines DP-Normslaves / PROFINET IO-Devices konsistent aus. Die Anweisung "GETIO" ruft dabei die Anweisung "DPRD\_DAT

(Seite 2436)" auf. Falls bei der Datenübertragung kein Fehler auftrat, werden die gelesenen Daten in den Zielbereich am Parameter INPUTS eingetragen.

Der Zielbereich muss dieselbe Länge aufweisen, die Sie für die selektierte Komponente projiziert haben.

Bei einem DP-Normslave mit modularem Aufbau bzw. mit mehreren DP-Kennungen können Sie mit einem "GETIO"-Aufruf jeweils nur auf die Daten einer Komponente / DP-Kennung unter der projizierten Anfangsadresse zugreifen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GETIO":

| Parameter | Deklaration | Datentyp         | Speicherbereich                 | Beschreibung   |
|-----------|-------------|------------------|---------------------------------|--|
| ID        | Input       | HW_SUBMOD<br>ULE | E, A, M, D, L oder<br>Konstante | Hardware-ID des DP-<br>Normslave / PROFINET IO-Device.   |
| STATUS    | Output      | DWORD            | E, A, M, D, L                   | Enthält die Fehlerinformation von<br>"DPRD_DAT (Seite 2436)" in der Form<br>DW#16#40xxx00.   |
| LEN       | Output      | INT              | E, A, M, D, L                   | Anzahl der gelesenen Daten in Bytes.   |
| INPUTS    | InOut       | VARIANT          | E, A, M, D                      | Zielbereich für die gelesenen Daten. Dieser<br>muss genauso lang sein, wie Sie für den<br>selektierten DP-Normslave / PROFINET IO-<br>Device projiziert haben. Es ist nur der<br>Datentyp BYTE zulässig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

Siehe auch: DPRD\_DAT: Konsistente Daten eines DP-Normslaves lesen (Seite 2436).

### SETIO: Prozessabbild übertragen

#### Beschreibung

Mit der Anweisung "SETIO" übertragen Sie die Daten aus dem durch den Parameter OUTPUTS definierten Quellbereich konsistent zum adressierten DP-Normslave / PROFINET IO-Device und ggf. ins Prozessabbild (nämlich dann, wenn Sie den betroffenen Adressbereich des DP-Normslave / PROFINET IO-Device als Konsistenzbereich in einem Prozessabbild projiziert haben). "SETIO" ruft dabei die Anweisung "DPWR\_DAT (Seite 2438)" auf.

Der Quellbereich muss dieselbe Länge aufweisen, die Sie für die selektierte Komponente projiziert haben.

Bei einem DP-Normslave / PROFINET IO-Device mit modularem Aufbau bzw. mit mehreren Kennungen können Sie mit einem "SETIO"-Aufruf nur auf eine DP-Kennung / Komponente zugreifen.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SETIO":

| Parameter | Deklaration | Datentyp         | Speicherbereich                 | Beschreibung  |
|-----------|-------------|------------------|---------------------------------|---|
| ID        | Input       | HW_SUBMODU<br>LE | E, A, M, D, L oder<br>Konstante | Hardware-ID des DP-Normslave / PROFINET IO-Device.  |
| STATUS    | Output      | DWORD            | E, A, M, D, L                   | Enthält die Fehlerinformation von "DPWR_DAT (Seite 2438)" in der Form DW#16#40xxx00.  |
| OUTPUTS   | InOut       | VARIANT          | E, A, M, D                      | Quellbereich für die zu schreibenden Daten. Er muss genauso lang sein, wie Sie für den selektierten DP-Normslave / PROFINET IO-Device projektiert haben. Es ist nur der Datentyp BYTE zulässig. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

Siehe auch: DPWR\_DAT: Konsistente Daten eines DP-Normslaves schreiben (Seite 2438).

## GETIO\_PART: Prozessabbildbereich lesen

### Beschreibung

Mit der Anweisung "GETIO\_PART" lesen Sie einen zusammenhängenden Teil der Eingänge einer EA-Baugruppe konsistent aus. "GETIO\_PART" ruft dabei die Anweisung "DPRD\_DAT (Seite 2436)" auf.

Über den Eingangsparameter ID wählen Sie die EA-Baugruppe anhand der Hardware-Kennung aus.

Über die Parameter OFFSET und LEN legen Sie die auszulesende Teilmenge des Prozessabbildbereichs fest. Falls der mittels OFFSET und LEN aufgespannte Eingangsbereich nicht komplett von der Baugruppe abgedeckt wird, liefert der Baustein den Fehlercode DW#16#4080B700.

Die Länge des Zielbereichs muss größer oder gleich als die Anzahl der zu lesenden Bytes sein:

- Falls bei der Datenübertragung kein Fehler auftrat, erhält ERROR den Wert FALSE. Die gelesenen Daten werden in den am Parameter INPUTS definierten Zielbereich geschrieben.
- Falls bei der Datenübertragung ein Fehler auftrat, erhält ERROR den Wert TRUE. Der Parameter STATUS erhält die Fehlerinformation von "DPRD\_DAT".
- Falls der Zielbereich größer ist als LEN, werden die ersten LEN Bytes des Zielbereichs beschrieben. ERROR erhält den Wert FALSE.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GETIO\_PART":

| Parameter | Deklaration | Datentyp         | Speicherbereich                 | Beschreibung  |
|-----------|-------------|------------------|---------------------------------|---|
| ID        | Input       | HW_SUBMOD<br>ULE | E, A, M, D, L oder<br>Konstante | Hardware-Kennung der Baugruppe  |
| OFFSET    | Input       | INT              | E, A, M, D, L oder<br>Konstante | Nummer des ersten zu lesenden Bytes im<br>Prozessabbild der Komponente<br>(kleinstmöglicher Wert: 0).   |
| LEN       | Input       | INT              | E, A, M, D, L oder<br>Konstante | Anzahl der zu lesenden Bytes.   |
| STATUS    | Output      | DWORD            | E, A, M, D, L                   | Enthält die Fehlerinformation von<br>"DPRD_DAT" in der Form<br>DW#16#40xxx00, falls ERROR = TRUE.   |
| ERROR     | Output      | BOOL             | E, A, M, D, L                   | Fehleranzeige:<br>ERROR = TRUE, falls beim Aufruf von<br>"DPRD_DAT" ein Fehler auftritt.  |
| INPUTS    | InOut       | VARIANT          | E, A, M, D                      | Zielbereich für die gelesenen Daten:<br>Falls der Zielbereich größer ist als LEN,<br>werden die ersten LEN Bytes des<br>Zielbereichs beschrieben. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

Siehe Parameter RET\_VAL der Anweisung "DPRD\_DAT (Seite 2436)".

## SETIO\_PART: Prozessabbildbereich übertragen

### Beschreibung

Mit der Anweisung "SETIO\_PART" schreiben Sie die Daten aus dem durch OUTPUTS aufgespannten Quellbereich konsistent auf die Ausgänge einer EA-Baugruppe. "SETIO\_PART" ruft dabei die Anweisung "DPWR\_DAT (Seite 2438)" auf.

Über den Eingangsparameter ID wählen Sie die EA-Baugruppe anhand der Hardware-Kennung aus.

Über die Parameter OFFSET und LEN legen Sie die zu beschreibende Teilmenge des Prozessabbildbereichs der über ID adressierten Komponente fest. Falls der mittels OFFSET und LEN aufgespannte Ausgangsbereich nicht komplett von der Baugruppe abgedeckt wird, liefert der Baustein den Fehlercode DW#16#4080B700.

Die Länge des Quellbereichs muss größer oder gleich als die Anzahl der zu schreibenden Bytes sein:

- Falls bei der Datenübertragung kein Fehler auftrat, erhält ERROR den Wert FALSE.
- Falls bei der Datenübertragung ein Fehler auftrat, erhält ERROR den Wert TRUE, und STATUS erhält die Fehlerinformation von "DPWR\_DAT".
- Falls der Quellbereich größer ist als LEN, werden die ersten LEN Bytes aus OUTPUTS übertragen. ERROR erhält den Wert FALSE.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SETIO\_PART":

| Parameter | Deklaration | Datentyp         | Speicherbereich                 | Beschreibung  |
|-----------|-------------|------------------|---------------------------------|---|
| ID        | Input       | HW_SUBMO<br>DULE | E, A, M, D, L oder<br>Konstante | Hardware-Kennung der EA-Baugruppe.  |
| OFFSET    | Input       | INT              | E, A, M, D, L oder<br>Konstante | Nummer des ersten zu beschreibenden Bytes im Prozessabbild der Komponente (kleinstmöglicher Wert: 0).   |
| LEN       | Input       | INT              | E, A, M, D, L oder<br>Konstante | Anzahl der zu schreibenden Bytes.   |
| STATUS    | Output      | DWORD            | E, A, M, D, L                   | Enthält die Fehlerinformation von "DPWR_DAT" in der Form DW#16#40xxxx00, falls ERROR = TRUE.  |
| ERROR     | Output      | BOOL             | E, A, M, D, L                   | Fehleranzeige:<br>ERROR = TRUE, falls beim Aufruf von "DPWR_DAT" ein Fehler auftritt.   |
| OUTPUTS   | InOut       | VARIANT          | E, A, M, D                      | Quellbereich für die zu schreibenden Daten:<br>Falls der Quellbereich größer ist als LEN, werden die ersten LEN Bytes aus OUTPUTS übertragen. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS und ERROR

Siehe Anweisung "DPWR\_DAT (Seite 2438)".

## RALRM: Alarm empfangen

### Beschreibung RALRM

#### Beschreibung

Die Anweisung empfängt einen Alarm samt der zugehörigen Information von einer Peripheriebaugruppe (zentraler Aufbau) bzw. eines Moduls eines DP-Slaves oder eines PROFINET IO-Devices und stellt diese Information an seinen Ausgangsparametern zur Verfügung.

Die Information in den Ausgangsparametern enthält sowohl die Startinformation des aufgerufenen OB als auch Informationen aus der Alarmquelle.

Der Aufbau der Datenstruktur des Zielbereichs AINFO entspricht für einen zentralen Aufbau der Datenstruktur von PROFINET IO.

Rufen Sie "RALRM" nur innerhalb desjenigen Alarm-OBs auf, den das Betriebssystem der CPU aufgrund des zu untersuchenden Alarms aus der Peripherie gestartet hat.

---

**Hinweis**

Wenn Sie "RALRM" in einem OB aufrufen, dessen Starterereignis kein Alarm aus der Peripherie ist, stellt Ihnen die Anweisung an ihren Ausgängen entsprechend weniger Information zur Verfügung.

Achten Sie darauf, dass Sie beim Aufruf von "RALRM" in verschiedenen OBs unterschiedliche Instanz-DBs verwenden. Falls Sie die aus einem "RALRM"-Aufruf resultierenden Daten außerhalb des zugehörigen Alarm-OB auswerten, sollten Sie sogar pro OB-Startereignis einen eigenen Instanz-DB benutzen.

---

**Hinweis**

Die Schnittstelle der Anweisung "RALRM" ist identisch mit der des in der Norm "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3" definierten FB "RALRM".

---

**Aufruf von RALRM**

"RALRM" können Sie in drei verschiedenen Betriebsarten (Parameter MODE) aufrufen, die in der folgenden Tabelle erläutert werden.

| MODE | RALRM ...  |
|------|--|
| 0    | ... zeigt die Alarm auslösende Komponente im Ausgangsparameter ID an und beschreibt den Ausgangsparameter NEW mit TRUE.  |
| 1    | ... beschreibt sämtliche Ausgangsparameter unabhängig von der Alarm auslösenden Komponente.  |
| 2    | ... prüft, ob die im Eingangsparameter F_ID angegebene Komponente den Alarm ausgelöst hat. <ul style="list-style-type: none"><li>• Falls nein, erhält NEW den Wert FALSE</li><li>• Falls ja, erhält NEW den Wert TRUE, und alle anderen Ausgangsparameter werden beschrieben</li></ul> |

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RALRM":

| Parameter              | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|------------------------|-------------|----------|------------------------------|---|
| MODE                   | Input       | INT      | E, A, M, D, L oder Konstante | Betriebsart   |
| F_ID                   | Input       | HW_IO    | D, L oder Konstante          | Hardware-Kennung der Baugruppe oder des Moduls<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der Komponente oder des Interface in der Hardware-Konfiguration hinterlegt. |
| MLEN                   | Input       | UINT     | E, A, M, D, L oder Konstante | Maximale Länge der zu empfangenen Alarminformation in Bytes.<br>Bei MLEN = 0 werden alle am Parameter AINFO spezifizierten Daten gelesen.   |
| NEW                    | Output      | BOOL     | E, A, M, D, L                | Ein neuer Alarm wurde empfangen.  |
| STATUS<br>(Seite 2402) | Output      | DWORD    | E, A, M, D, L                | Fehlercode  |
| ID                     | Output      | HW_IO    | E, A, M, D, L                | Hardware-Kennung der Baugruppe oder des Moduls, von der ein Alarm empfangen wurde.  |
| LEN                    | Output      | UINT     | E, A, M, D, L                | Länge der empfangenen Alarminformation  |
| TINFO<br>(Seite 2406)  | InOut       | VARIANT  | E, A, M, D, L                | Zielbereich für OB-Startinformation und Verwaltungsinformation  |
| AINFO<br>(Seite 2418)  | InOut       | VARIANT  | E, A, M, D, L                | Zielbereich für Kopfinformation und Alarmzusatzinformation<br>Für AINFO sollten Sie eine Länge von mindestens MLEN Bytes vorsehen.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Hinweis

Falls Sie den Zielbereich (Seite 2424) TINFO bzw. AINFO zu kurz wählen, kann RALRM nicht die gesamte Information eintragen.

## Parameter STATUS

### Beschreibung

Der Ausgangsparameter STATUS enthält Fehlerinformationen. Wird er als ARRAY[1...4] of BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

| Feldelement | Name         | Bedeutung   |
|-------------|--------------|---|
| STATUS[1]   | Function_Num | <ul style="list-style-type: none"> <li>B#16#00, falls kein Fehler</li> <li>Funktionskennung aus DPV1-PDU:<br/>Bei einem Fehler wird B#16#80 ausgegeben (bei einem Fehler beim Lesen eines Datensatzes B#16#DE und beim Schreiben eines Datensatzes B#16#DF).<br/>Falls kein DPV1-Protokollelement benutzt wird, wird B#16#C0 ausgegeben.</li> </ul> |
| STATUS[2]   | Error Decode | Ort der Fehlerkennung   |
| STATUS[3]   | Error_Code_1 | Fehlerkennung   |
| STATUS[4]   | Error_Code_2 | herstellerspezifische Erweiterung der Fehlerkennung   |

### Feldelement STATUS[2]

STATUS[2] kann folgende Werte annehmen:

| Error Decode (B#16#...) | Quelle     | Bedeutung   |
|-------------------------|------------|---|
| 00 bis 7F               | CPU        | kein Fehler oder keine Warnung  |
| 80                      | DPV1       | Fehler nach IEC 61158-6   |
| 81 bis 8F               | CPU        | B#16#8x zeigt einen Fehler im x-ten Aufrufparameter der Anweisung an. |
| FE, FF                  | DP Profile | profilspezifischer Fehler   |

### Feldelement STATUS[3]

STATUS[3] kann folgende Werte annehmen:

| Error Decode (B#16#...) | Error_Code_1 (B#16#...) | Erläuterung laut DPV1 | Bedeutung  |
|-------------------------|-------------------------|-----------------------|--|
| 00                      | 00                      |                       | kein Fehler, keine Warnung   |
| 70                      | 00                      | reserved, reject      | Erstaufruf; keine Datensatzübertragung aktiv                               |
|                         | 01                      | reserved, reject      | Erstaufruf; Datensatzübertragung angestoßen                                |
|                         | 02                      | reserved, reject      | Zwischenaufruf; Datensatzübertragung ist bereits aktiv                     |
| 80                      | 90                      | reserved, pass        | logische Anfangsadresse ungültig   |
|                         | 92                      | reserved, pass        | unzulässiger Typ bei VARIANT-Pointer                                       |
|                         | 93                      | reserved, pass        | Die mittels ID bzw. F_ID adressierte DP-Komponente ist nicht konfiguriert. |



| Error Decode (B#16#...) | Error_Code_1 (B#16#...) | Erläuterung laut DPV1 | Bedeutung  |
|-------------------------|-------------------------|-----------------------|--|
|                         | 96                      |                       | Der "RALRM (Seite 2399)" kann weder die OB-Startinformation noch die Verwaltungsinformation noch die Kopfinformation noch die Alarmzusatzinformation liefern. Bei den OBs 4x, 55, 56, 57, 82 und 83 können Sie mit Hilfe der Anweisung "DPNRM_DG (Seite 2450)" das aktuelle Diagnosetelegramm des betroffenen DP-Slaves asynchron lesen (Adressinformation aus OB-Startinformation). |
|                         | A0                      | read error            | negative Quittung beim Lesen von der Baugruppe   |
|                         | A1                      | write error           | negative Quittung beim Schreiben zur Baugruppe   |
|                         | A2                      | module failure        | DP-Protokollfehler bei Layer 2 (z. B. Slaveausfall oder Busstörungen)  |
|                         | A3                      | reserved, pass        | <ul style="list-style-type: none"> <li>• PROFIBUS DP: DP-Protokollfehler bei Direct-Data-Link-Mapper oder User-Interface/User</li> <li>• PROFINET IO: allgemeiner CM-Fehler</li> </ul>   |
|                         | A4                      | reserved, pass        | Kommunikation am PBUS+ gestört   |
|                         | A5                      | reserved, pass        | –  |
|                         | A7                      | reserved, pass        | DP-Slave oder Baugruppe ist beschäftigt (temporärer Fehler)  |
|                         | A8                      | version conflict      | DP-Slave oder Baugruppe meldet nicht passende Versionen  |
|                         | A9                      | feature not supported | Funktion wird vom DP-Slave oder der Baugruppe nicht unterstützt  |
|                         | AA bis AF               | user specific         | DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler seiner Anwendung. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach.   |
|                         | B0                      | invalid index         | Baugruppe kennt den Datensatz nicht<br>Datensatznummer $\geq 256$ ist unzulässig   |
|                         | B1                      | write length error    | Die Längenangabe im Parameter RECORD ist falsch;<br>bei "RALRM (Seite 2399)": Längenfehler in AINFO (Seite 2418) ,<br>bei "RDREC (Seite 2392)" und "WRREC (Seite 2394)": Längenfehler in MLEN  |
|                         | B2                      | invalid slot          | Der projektierte Steckplatz ist nicht belegt.  |
|                         | B3                      | type conflict         | Ist-Baugruppentyp ungleich Soll-Baugruppentyp  |
|                         | B4                      | invalid area          | DP-Slave oder Baugruppe meldet einen Zugriff auf einen unzulässigen Bereich  |
|                         | B5                      | state conflict        | DP-Slave oder Baugruppe ist nicht bereit   |
|                         | B6                      | access denied         | DP-Slave oder Baugruppe verweigert den Zugriff   |
|                         | B7                      | invalid range         | DP-Slave oder Baugruppe meldet einen unzulässigen Bereich eines Parameters oder eines Werts  |
|                         | B8                      | invalid parameter     | DP-Slave oder Baugruppe meldet einen unzulässigen Parameter  |

| Error_Decode (B#16#...) | Error_Code_1 (B#16#...) | Erläuterung laut DPV1    | Bedeutung   |
|-------------------------|-------------------------|--------------------------|---|
|                         | B9                      | invalid type             | DP-Slave oder Baugruppe meldet einen unzulässigen Typ<br>Bei "RDREC (Seite 2392)": Puffer zu klein (Lesen von Teilmengen ist nicht möglich)<br>Bei "WRREC (Seite 2394)": Puffer zu klein (Schreiben von Teilmengen ist nicht möglich)   |
|                         | BA bis BF               | user specific            | DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler beim Zugriff. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach.  |
|                         | C0                      | read constrain conflict  | Bei "WRREC (Seite 2394)": Die Daten können nur im STOP-Zustand der CPU geschrieben werden. Hinweis: Damit ist ein Schreiben durch das Anwenderprogramm nicht möglich. Sie können die Daten nur online mit PG/PC schreiben.<br>Bei "RDREC (Seite 2392)": Die Baugruppe führt den Datensatz, aber es liegen entweder noch keine Daten vor oder die Daten können nur im STOP-Zustand der CPU gelesen werden. Hinweis: Falls Daten nur im STOP-Zustand der CPU gelesen werden können, ist eine Auswertung durch das Anwenderprogramm nicht möglich. In diesem Fall können Sie die Daten nur online mit PG/PC lesen. |
|                         | C1                      | write constrain conflict | Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.  |
|                         | C2                      | resource busy            | Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.   |
|                         | C3                      | resource unavailable     | Die benötigten Betriebsmittel sind momentan belegt.   |
|                         | C4                      |                          | Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden.<br>Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte Ihren Aufbau auf elektrische Störquellen.  |
|                         | C5                      |                          | DP-Slave oder Baugruppe nicht verfügbar.  |
|                         | C6                      |                          | Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs  |
|                         | C7                      |                          | Auftragsabbruch wegen Neustart (Warmstart) oder Kaltstart des DP-Masters  |
|                         | C8 bis CF               |                          | DP-Slave oder Baugruppe meldet einen herstellerspezifischen Fehler zu seinen Ressourcen. Bitte sehen Sie in der Dokumentation des Herstellers des DP-Slaves bzw. der Baugruppe nach.  |
|                         | Dx                      | user specific            | DP-Slave-spezifisch. Siehe Beschreibung des DP-Slaves.  |
| 81                      | 00 bis FF               |                          | Fehler im ersten Aufrufparameter (bei "RALRM (Seite 2399)": MODE)   |
|                         | 00                      |                          | Unzulässige Betriebsart   |

| Error Decode (B#16#...) | Error_Code_1 (B#16#...) | Erläuterung laut DPV1 | Bedeutung   |
|-------------------------|-------------------------|-----------------------|---|
| 82                      | 00 bis FF               |                       | Fehler im zweiten Aufrufparameter   |
| :                       | :                       |                       | :   |
| 88                      | 00 bis FF               |                       | Fehler im achten Aufrufparameter (bei "RALRM (Seite 2399)": TINFO (Seite 2406))     |
|                         | 01                      |                       | Syntaxkennung falsch  |
|                         | 23                      |                       | Überschreitung des Mengengerüsts oder Zielbereich zu klein                          |
|                         | 24                      |                       | Bereichskennung falsch  |
|                         | 32                      |                       | DB/DI-Nr. außerhalb des Anwenderbereichs  |
|                         | 3A                      |                       | DB/DI-Nr. ist NULL bei Bereichskennung DB/DI oder angegebener DB/DI nicht vorhanden |
|                         |                         |                       |   |
| 89                      | 00 bis FF               |                       | Fehler im neunten Aufrufparameter (bei "RALRM (Seite 2399)": AINFO (Seite 2418))    |
|                         | 01                      |                       | Syntaxkennung falsch  |
|                         | 23                      |                       | Überschreitung des Mengengerüsts oder Zielbereich zu klein                          |
|                         | 24                      |                       | Bereichskennung falsch  |
|                         | 32                      |                       | DB/DI-Nr. außerhalb des Anwenderbereichs  |
|                         | 3A                      |                       | DB/DI-Nr. ist NULL bei Bereichskennung DB/DI oder angegebener DB/DI nicht vorhanden |
|                         |                         |                       |   |
| 8A                      | 00 bis FF               |                       | Fehler im 10. Aufrufparameter   |
| :                       | :                       |                       | :   |
| 8F                      | 00 bis FF               |                       | Fehler im 15. Aufrufparameter   |
|                         |                         |                       |   |
| FE, FF                  | 00 bis FF               |                       | profilspezifischer Fehler   |

## Feldelement STATUS[4]

STATUS[4] wird bei DPV1-Fehlern vom DP-Master an die CPU und die Anweisung durchgereicht. Ohne DPV1-Fehler wird der Wert auf "0" gesetzt mit folgenden Ausnahmen bei "RDREC":

- STATUS[4] enthält die Länge des Zielbereichs aus RECORD, falls MLEN > Länge des Zielbereichs aus RECORD
- STATUS[4]=MLEN, falls die tatsächliche Datensatzlänge < MLEN < Länge des Zielbereichs aus RECORD
- STATUS[4]=0, falls STATUS[4] > 255 gesetzt werden müsste

Bei PROFINET IO enthält STATUS[4] den Wert "0".

## Parameter TINFO

### Datenstruktur des Zielbereichs TINFO

Die Datenstruktur des Zielbereichs TINFO enthält die Startinformationen des Organisationsbausteins, in dem "RALRM" aktuell aufgerufen wurde.

Der Zielbereich TINFO kann die Startinformation mit Standardzugriff oder optimiertem Zugriff enthalten. Das Format der Startinformationen im Zielbereich TINFO muss immer den Startinformationen des entsprechenden Organisationsbausteins entsprechen.

- Die Startinformationen eines OB mit Standardzugriff liegen in der Bausteinschnittstelle immer in den ersten 20 Byte des Abschnitts "Temp". Für diese verwenden Sie die Datenstruktur "TI\_Classic".
- Die Startinformationen eines OB mit optimiertem Zugriff werden im Abschnitt "Input" abgelegt. Für diese OBs verwenden Sie den eine für den OB-Typ spezifische Datenstruktur.

Durch die Umstellung des Bausteinzugriffs (standard/optimiert) ändert sich auch die Bausteinschnittstelle.

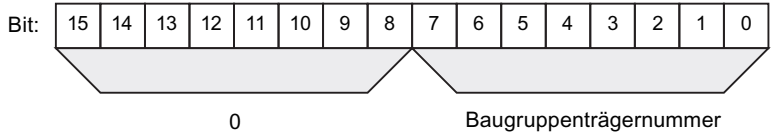
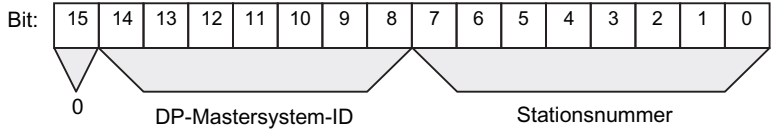
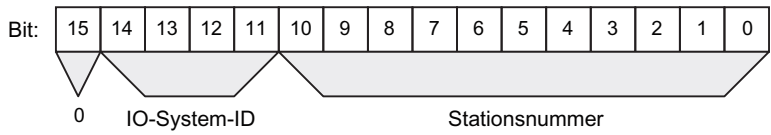
Die folgende Tabelle enthält eine Übersicht der Datenstrukturen, die in Abhängigkeit von dem Organisationsbaustein am Parameter TINFO verwendet werden.

| Name der Datenstruktur   | S7-1200 CPUs ab Version | S7-1500 CPUs ab Version | Datenstruktur verwendet für:  |
|--|-------------------------|-------------------------|---|
| Datenstruktur für Organisationsbausteine mit Standardzugriff             |                         |                         |   |
| TI_Classic   | -                       | V1                      | Organisationsbausteine ohne optimierten Bausteinzugriff.  |
| Datenstruktur für Organisationsbausteine mit optimiertem Bausteinzugriff |                         |                         |   |
| TI_ProgramCycle  | V2                      | V1                      | Zyklus-OB (Program cycle)   |
| TI_Startup   | V2                      | V1                      | Anlauf-OB (Startup)   |
| TI_Delay   | V2                      | V1                      | Verzögerungsalarm-OB (Time delay interrupt)   |
| TI_Cyclic  | V2                      | V1                      | Weckalarm-OB (Cyclic interrupt)   |
| TI_HWInterrupt   | V2                      | V1                      | Prozessalarm-OB (Hardware interrupt)  |
| TI_TimeError   | V2                      | V1                      | Zeitfehler-OB (Time error interrupt)  |
| TI_DiagnosticInterrupt   | V2                      | V1                      | Diagnosealarm-OB (Diagnostic error interrupt)   |
| TI_PlugPullModule  | V2                      | V1                      | Ziehen/Stecken-OB (Pull or plug of modules)   |
| TI_StationFailure  | V2                      | V1                      | Baugruppenträgerausfall-OB (Rack or station failure)  |
| TI_ProgIOAccessError   | V2                      | V1                      | <ul style="list-style-type: none"> <li>• Programmierfehler-OB (Programming error)</li> <li>• Peripheriezugriffsfehler-OB (IO access error)</li> </ul>                                 |
| TI_TimeOfDay   | V2                      | V1                      | Uhrzeitalarm-OB (Time of day)   |
| TI_SynchCycle  | -                       | V1                      | Taktsynchronalarm-OB (Synchronous Cycle)  |
| TI_Submodule   | V2                      | V1                      | <ul style="list-style-type: none"> <li>• Statusalarm-OB (Status)</li> <li>• Update-Alarm OB (Update)</li> <li>• OB für hersteller- bzw. profilspezifischen Alarm (Profile)</li> </ul> |

## Datenstruktur für Organisationsbausteine mit Standardzugriff

Die Datenstruktur TI\_Classic hat den folgenden Aufbau:

| Parameter   | Datentyp           | Byte      | Beschreibung  |
|---|--------------------|-----------|---|
| TI_Classic  |                    |           | Datenstruktur für Organisationsbausteine ohne optimierten Bausteinzugriff.  |
| Byte 0 bis 19: Startinformation des OB, in dem "RALRM" aufgerufen wurde.* |                    |           |   |
| EV_CLASS  | BYTE               | 0         | Ereignisklasse<br>Beispiel OB1:<br><ul style="list-style-type: none"> <li>• Bits 0-3: Kennung des Ereignis (1=kommendes Ereignis)</li> <li>• Bits 4-7: Ereignisklasse (1=Ereignisklasse 1)</li> </ul>   |
| EV_NUM  | BYTE               | 1         | Ereignisnummer (abhängig von OB-Typ)<br>Beispiel OB1 (SCAN_1):<br><ul style="list-style-type: none"> <li>• SCAN_1 = 1 bei Erstaufwurf</li> <li>• SCAN_1 = 3 bei allen weiteren Aufrufen</li> </ul>  |
| PRIORITY  | BYTE               | 2         | Prioritätsklasse  |
| NUM   | BYTE               | 3         | OB-Nummer   |
| TYP2_3  | BYTE               | 4         | Zusatzinformation   |
| TYP1  | BYTE               | 5         | Abhängig von dem verwendeten OB-Typ werden in den BYTES "TYP2_3" und "TYP1" unterschiedliche Informationen abgelegt.<br>Welche dies sind, ist in der Dokumentation der jeweiligen Organisationsbausteine beschrieben.<br>Beispiel (OB1):<br><ul style="list-style-type: none"> <li>• TYP2_3: OB1_RESERVED_1 (Reserviert)</li> <li>• TYP1: OB1_RESERVED_2 (Reserviert)</li> </ul>  |
| ZI1   | WORD               | 6 bis 7   | Zusatzinformation<br>Abhängig von dem verwendeten OB-Typ werden in "ZI1" unterschiedliche Informationen abgelegt.<br>Welche dies sind, ist in der Dokumentation der jeweiligen Organisationsbausteine beschrieben.<br>Beispiel (OB1):<br><ul style="list-style-type: none"> <li>• ZI1: OB1_PREV_CYCLE (Laufzeit des vorherigen Zyklus in ms)</li> </ul>   |
| ZI2_3   | DWORD              | 8 bis 11  | Zusatzinformation<br>Abhängig von dem verwendeten OB-Typ werden in "ZI2_3" unterschiedliche Informationen abgelegt.<br>Welche dies sind, ist in der Dokumentation der jeweiligen Organisationsbausteine beschrieben.<br>Beispiel (OB1):<br><ul style="list-style-type: none"> <li>• ZI2: OB1_MIN_CYCLE (minimale Zykluszeit (ms) seit dem letzten Anlauf)</li> <li>• ZI3: OB1_MAX_CYCLE (maximale Zykluszeit (ms) seit dem letzten Anlauf)</li> </ul> |
| OB_DATE_TIME  | DATE_AND_TIME (DT) | 12 bis 19 | Datum und Uhrzeit, zu denen der OB angefordert wurde.   |
| Byte 20 und 21: Adressinformationen                                       |                    |           |   |

| Parameter                              | Datentyp | Byte      | Beschreibung   |
|--|----------|-----------|--|
| address                                | WORD     | 20 und 21 | <p>Adressinformationen wie bei S7-300/400 CPUs:</p> <ul style="list-style-type: none"> <li>• Bei zentralem Aufbau die Baugruppenträgernummer (0-31):</li> </ul>  <ul style="list-style-type: none"> <li>• Bei dezentralem Aufbau mit PROFIBUS DP: <ul style="list-style-type: none"> <li>- DP-Mastersystem-ID (1-31)</li> <li>- Stationsnummer (0-127).</li> </ul> </li> </ul>  <ul style="list-style-type: none"> <li>• Bei dezentralem Aufbau mit PROFINET IO: <ul style="list-style-type: none"> <li>- die letzten beiden Stellen der PROFINET IO-System-ID (0-15), um die vollständige PROFINET IO-System-ID zu erhalten, müssen Sie 100 (dezimal) dazu addieren</li> <li>- Stationsnummer (0-2047).</li> </ul> </li> </ul>  |
| Byte 22 bis 31: Verwaltungsinformation |          |           |  |
| slv_prfl                               | BYTE     | 22        | <p>Slaveprofil wie bei S7-300/400 CPUs:</p> <ul style="list-style-type: none"> <li>• Bei zentralem Aufbau: 0 (Datensatz 0 oder Datensatz 1)</li> <li>• Bei dezentralem Aufbau: <ul style="list-style-type: none"> <li>- Bit 0 bis 3: Slavetyp <ul style="list-style-type: none"> <li>- 0000: DP (Struktur Datensatz 0)</li> <li>- 0001: DPS7 (Struktur Datensatz 0 oder Datensatz 1)</li> <li>- 0010: DPS7 V1 (Struktur Datensatz 0 oder Datensatz 1)</li> <li>- 0011: DPV1 (Struktur gemäß PROFIBUS DP Standard)</li> <li>- 0100 – 0111: reserviert</li> <li>- 1000: PROFINET IO (Struktur gemäß PROFINET IO Standard)</li> <li>- ab 1001: reserviert</li> </ul> </li> <li>- Bit 4 bis 7: Profiltyp (reserviert)</li> </ul> </li> </ul>   |

| Parameter   | Datentyp | Byte      | Beschreibung   |
|---|----------|-----------|--|
| intr_type   | BYTE     | 23        | <p>Alarminfotyp wie bei S7-300/400 CPUs:</p> <ul style="list-style-type: none"> <li>• Bei zentralem Aufbau: 0</li> <li>• Bei dezentralem Aufbau: <ul style="list-style-type: none"> <li>– Bit 0 bis 3: Alarminfotyp <ul style="list-style-type: none"> <li>- 0000: Alarm kommt von einer projektierten dezentralen Baugruppe</li> <li>- 0001: Alarm eines Nicht-DPV1-Slaves / Nicht-IO-Device oder eines nicht projektierten Steckplatzes</li> <li>- 0010: In der CPU erzeugter Alarm</li> <li>- ab 0011: reserviert</li> </ul> </li> <li>– Bit 4 bis 7: Strukturversion <ul style="list-style-type: none"> <li>- 0000: Initial</li> <li>- ab 0001: reserviert</li> </ul> </li> </ul> </li> </ul>  |
| flags1  | BYTE     | 24        | <p>Flags der PROFIBUS DP-Master-Anschaltung/PROFINET IO-Controller-Anschaltung wie bei S7-300/400 CPUs:</p> <ul style="list-style-type: none"> <li>• Bei zentralem Aufbau: 0</li> <li>• Bei dezentralem Aufbau: <ul style="list-style-type: none"> <li>– Bit 0 = 0: Alarm von einer integrierten Anschaltung (PROFINET IO oder PROFIBUS DP)</li> <li>– Bit 0 = 1: Alarm von einer externen Anschaltung (PROFINET IO oder PROFIBUS DP)</li> <li>– Bit 1 bis 7: reserviert</li> </ul> </li> </ul>  |
| flags2  | BYTE     | 25        | <p>Flags der PROFIBUS DP-Master-Anschaltung/PROFINET IO-Controller-Anschaltung wie bei S7-300/400 CPUs:</p> <ul style="list-style-type: none"> <li>• Bei zentralem Aufbau: 0</li> <li>• Bei dezentralem Aufbau mit PROFIBUS DP: <ul style="list-style-type: none"> <li>– Bit 0: EXT_DIAG_FLAG aus dem Diagnosetelegramm bzw. 0, falls dieses Bit beim Alarm nicht vorliegt. Das Bit ist 1, wenn der DP-Slave gestört ist.</li> <li>– Bit 1 bis 7: reserviert</li> </ul> </li> <li>• Bei dezentralem Aufbau mit PROFINET IO: <ul style="list-style-type: none"> <li>– Bit 0: ARDiagnosisstate bzw. 0, falls keine Information beim Alarm vorliegt. Das Bit ist 1, wenn das IO-Device gestört ist.</li> <li>– Bit 1 bis 7: reserviert</li> </ul> </li> </ul> |
| id  | UINT     | 26 und 27 | <p>Verwaltungsinformation</p> <ul style="list-style-type: none"> <li>• Bei zentralem Aufbau: 0</li> <li>• Bei dezentralem Aufbau mit PROFIBUS DP: PROFIBUS Identnummer als eindeutige Identifikation des PROFIBUS DP-Slave</li> <li>• Bei dezentralem Aufbau mit PROFINET IO: PROFINET IO-Device Identnummer als eindeutige Identifikation des PROFINET IO-Devices</li> </ul>  |
| manufacturer  | UINT     | 28 und 29 | Herstellerkennung (nur bei dezentralem Aufbau mit PROFINET IO).  |
| instance  | UINT     | 30 und 31 | Identnummer der Instanz (nur bei dezentralem Aufbau mit PROFINET IO).  |
| *Die Startinformationen sind abhängig von dem verwendeten OB. Die Startinformationen des jeweiligen OB-Typs entnehmen Sie der Schnittstelle oder der Dokumentation des OBs. |          |           |  |

### Datenstrukturen für Organisationsbausteine mit optimiertem Bausteinzugriff

Die Datenstrukturen für Organisationsbausteine mit optimiertem Bausteinzugriff haben den folgenden Aufbau:

- Byte 0 bis 3: Format der Startinformationen, Klasse und Nummer des aufgerufenen OBs (gleicher Aufbau für alle Datenstrukturen).
- Byte 4 bis 19: Optimierte Startinformation (Aufbau abhängig von dem OB-Typ). Die Daten in Byte 4 bis 19 entsprechen in Aufbau und Inhalt der entsprechenden OB-Schnittstelle.
- Byte 20 bis 31: Bei bestimmten OBs zusätzlich die Adress- und Verwaltungsinformation. Die Daten in Byte 20 bis 31 entsprechen den Daten der Bytes 20 bis 31 der Datenstruktur TI\_Classic.

In den folgenden Tabellen ist der Aufbau der Datenstrukturen beschrieben.

Tabelle 9-41 Zyklus-OB: Datenstruktur TI\_ProgramCycle

| Parameter       | Datentyp | Byte | Beschreibung  |
|-----------------|----------|------|---|
| TI_ProgramCycle |          |      | Datenstruktur für Zyklus-OB   |
| SI_Format       | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>           |
| OB_Class        | USINT    | 1    | OB-Klasse (=1)  |
| OB_Nr           | UINT     | 2    | OB-Nummer (1...32767).  |
| Initial_Call    | BOOL     | 4    | =TRUE im ersten Aufruf dieses OB bei: <ul style="list-style-type: none"> <li>• Übergang von STOP bzw. HALT nach RUN</li> <li>• Nach Neuladen</li> </ul> |
| Remanence       | BOOL     | 5    | =TRUE, falls remanente Daten verfügbar sind.  |

Tabelle 9-42 Anlauf-OB: Datenstruktur TI\_Startup

| Parameter     | Datentyp | Byte | Beschreibung  |
|---------------|----------|------|---|
| TI_Startup    |          |      | Datenstruktur für Anlauf-OB   |
| SI_Format     | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul> |
| OB_Class      | USINT    | 1    | OB-Klasse (=100)  |
| OB_Nr         | UINT     | 2    | OB-Nummer (1...32767).  |
| LostRetentive | BOOL     | 4    | =TRUE, falls die Inhalte remanenter Datenbereiche verloren gegangen sind.   |
| LostRTC       | BOOL     | 5    | =TRUE, falls die Uhrzeit der Echtzeituhr verloren gegangen ist.   |

Tabelle 9-43 Verzögerungsalarm-OB: Datenstruktur TI\_Delay

| Parameter | Datentyp | Byte | Beschreibung  |
|-----------|----------|------|---|
| TI_Delay  |          |      | Datenstruktur für Verzögerungsalarm-OB  |
| SI_Format | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul> |



| Parameter | Datentyp | Byte | Beschreibung  |
|-----------|----------|------|---|
| OB_Class  | USINT    | 1    | OB-Klasse (=20)   |
| OB_Nr     | UINT     | 2    | OB-Nummer (1...32767).  |
| Sign      | WORD     | 4    | Anwenderkennung: Eingangsparameter SIGN aus dem Aufruf der Anweisung "SRT_DINT (Seite 2530)". |

Tabelle 9-44 Weckalarm-OB: Datenstruktur TI\_Cyclic

| Parameter    | Datentyp | Byte | Beschreibung  |
|--------------|----------|------|---|
| TI_Cyclic    |          |      | Datenstruktur für Weckalarm-OB  |
| SI_Format    | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>           |
| OB_Class     | USINT    | 1    | OB-Klasse (=30)   |
| OB_Nr        | UINT     | 2    | OB-Nummer (1...32767).  |
| Initial_Call | BOOL     | 4    | =TRUE im ersten Aufruf dieses OB <ul style="list-style-type: none"> <li>• Beim Übergang von STOP bzw. HALT nach RUN</li> <li>• Nach Neuladen</li> </ul> |
| Event_Count  | INT      | 6    | Anzahl der verworfenen Startereignisse seit dem letzten Start dieses OBs.   |

Tabelle 9-45 Prozessalarm-OB: Datenstruktur TI\_HWInterrupt

| Parameter      | Datentyp | Byte | Beschreibung   |
|----------------|----------|------|--|
| TI_HWInterrupt |          |      | Datenstruktur für Prozessalarm-OB  |
| SI_Format      | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>                                |
| OB_Class       | USINT    | 1    | OB-Klasse (=40)  |
| OB_Nr          | UINT     | 2    | OB-Nummer (1...32767).   |
| LADDR          | HW_IO    | 4    | HW-Kennung des Moduls, das den Prozessalarm auslöst.   |
| USI            | WORD     | 6    | Kennung für zukünftige Erweiterungen (nicht anwenderrelevant).   |
| IChannel       | USINT    | 8    | Nummer des Kanals, der den Prozessalarm auslöst.   |
| EventType      | BYTE     | 9    | Kennung für den Ereignistyp, zu dem das Alarm auslösende Ereignis gehört (z. B. steigende Flanke).<br>Diese Kennung ist der Beschreibung des jeweiligen Moduls zu entnehmen. |
| address        | WORD     | 20   | Siehe Parameter "address" der Datenstruktur TI_Cassic.   |
| slv_prfl       | BYTE     | 22   | Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.  |
| intr_type      | BYTE     | 23   | Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.   |
| flags1         | BYTE     | 24   | Siehe Parameter "flags1" der Datenstruktur TI_Cassic.  |
| flags2         | BYTE     | 25   | Siehe Parameter "flags2" der Datenstruktur TI_Cassic.  |
| id             | UNIT     | 26   | Siehe Parameter "id" der Datenstruktur TI_Cassic.  |
| manufacturer   | UNIT     | 28   | Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.  |
| instance       | UNIT     | 30   | Siehe Parameter "instance" der Datenstruktur TI_Cassic.  |

Tabelle 9-46 Zeitfehler-OB: Datenstruktur TI\_TimeError

| Parameter    | Datentyp | Byte | Beschreibung   |
|--------------|----------|------|--|
| TI_TimeError |          |      | Datenstruktur für Zeitfehler-OB  |
| SI_Format    | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>  |
| OB_Class     | USINT    | 1    | OB-Klasse (=80)  |
| OB_Nr        | UINT     | 2    | OB-Nummer (1...32767).   |
| Csg_OBnr     | OB_ANY   | 4    | Nummer des OB, der bearbeitet wurde, als der Zeitfehler auftrat.   |
| Fault_ID     | BYTE     | 7    | Fehlercode. Mögliche Werte: <ul style="list-style-type: none"> <li>• B#16#01: Zykluszeit überschritten</li> <li>• B#16#02: Der angeforderte OB ist noch in Bearbeitung.</li> <li>• B#16#05: abgelaufener Uhrzeitalarm durch Uhrzeitsprung.</li> <li>• B#16#06: abgelaufener Uhrzeitalarm bei Wiedereintritt in RUN nach HALT.</li> <li>• B#16#07: Überlauf des OB-Anforderungspuffers für die aktuelle Prioritätsklasse.</li> <li>• B#16#08: Taktsynchronalarm-Zeitfehler.</li> <li>• B#16#09: Alarmverlust durch zu hohe Alarmlast</li> <li>• B#16#0B: Technologiesynchronalarm-Zeitfehler</li> </ul> |
| Csg_Prio     | UNIT     | 8    | Priorität des OB, der bearbeitet wurde, als der Zeitfehler auftrat.  |

Tabelle 9-47 Diagnosealarm-OB: Datenstruktur TI\_DiagnosticInterrupt

| Parameter              | Datentyp | Byte | Beschreibung   |
|------------------------|----------|------|--|
| TI_DiagnosticInterrupt |          |      | Datenstruktur für Diagnosealarm-OB   |
| SI_Format              | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>  |
| OB_Class               | USINT    | 1    | OB-Klasse (=82)  |
| OB_Nr                  | UINT     | 2    | OB-Nummer (1...32767).   |
| LADDR                  | HW_ANY   | 4    | Hardware-Kennung des Hardware-Objekts, das den Diagnosealarm ausgelöst hat.  |
| IO_State               | WORD     | 6    | Zustand des Hardware-Objekts: <ul style="list-style-type: none"> <li>• Bit 0: gut</li> <li>• Bit 1: deaktiviert</li> <li>• Bit 2: Wartungsbedarf</li> <li>• Bit 3: Wartungsanforderung</li> <li>• Bit 4: Fehler</li> <li>• Bit 5: nicht erreichbar</li> <li>• Bit 6: qualifiziert</li> <li>• Bit 7: nicht verfügbar</li> </ul> |
| Channel                | UINT     | 8    | Kanalnummer  |
| MultiError             | BOOL     | 10   | =TRUE, falls mehr als ein Fehler vorliegt.   |
| address                | WORD     | 20   | Siehe Parameter "address" der Datenstruktur TI_Cassic.   |
| slv_prfl               | BYTE     | 22   | Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.  |
| intr_type              | BYTE     | 23   | Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.   |

| Parameter    | Datentyp | Byte | Beschreibung  |
|--------------|----------|------|---|
| flags1       | BYTE     | 24   | Siehe Parameter "flags1" der Datenstruktur TI_Cassic.       |
| flags2       | BYTE     | 25   | Siehe Parameter "flags2" der Datenstruktur TI_Cassic.       |
| id           | UINT     | 26   | Siehe Parameter "id" der Datenstruktur TI_Cassic.           |
| manufacturer | UINT     | 28   | Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic. |
| instance     | UINT     | 30   | Siehe Parameter "instance" der Datenstruktur TI_Cassic.     |

Tabelle 9-48 Ziehen/Stecken-OB: Datenstruktur TI\_PlugPullModule

| Parameter         | Datentyp | Byte | Beschreibung   |
|-------------------|----------|------|--|
| TI_PlugPullModule |          |      | Datenstruktur für Ziehen/Stecken-OB  |
| SI_Format         | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>  |
| OB_Class          | USINT    | 1    | OB-Klasse (=83)  |
| OB_Nr             | UINT     | 2    | OB-Nummer (1...32767).   |
| LADDR             | HW_IO    | 4    | Hardware-Kennung des betroffenen Moduls bzw. Submoduls   |
| Event_Class       | BYTE     | 6    | <ul style="list-style-type: none"> <li>• B#16#38: (Sub-) Modul gesteckt</li> <li>• B#16#39: (Sub-) Modul gezogen bzw. nicht ansprechbar</li> </ul>   |
| Fault_ID          | BYTE     | 7    | <p>Fehlercode</p> <p>Über den Fehlercode wird ausgegeben, welches Ereignis den Start des Ziehen/Stecken-OB verursacht hat.</p> <ul style="list-style-type: none"> <li>• Bei Event_Class = B#16#38 - (Sub-) Modul gesteckt: <ul style="list-style-type: none"> <li>- B#16#54: PROFINET IO-Submodul gesteckt und entspricht parametrierem Submodul</li> <li>- B#16#55: PROFINET IO-Submodul gesteckt, entspricht aber nicht dem parametrierten Submodul</li> <li>- B#16#56: PROFINET IO-Submodul gesteckt, jedoch Fehler bei Baugruppenparametrierung</li> <li>- B#16#57: PROFINET IO-Submodul oder -Modul gesteckt, jedoch mit Störung oder Wartung</li> <li>- B#16#58: PROFINET IO-Submodul Zugriffsfehler beseitigt</li> </ul> </li> <li>• Bei Event_Class = B#16#39 - (Sub-) Modul gezogen bzw. nicht ansprechbar: <ul style="list-style-type: none"> <li>- B#16#51: PROFINET IO-Modul gezogen</li> <li>- B#16#54: PROFINET IO-Submodul gezogen</li> </ul> </li> </ul> |
| address           | WORD     | 20   | Siehe Parameter "address" der Datenstruktur TI_Cassic.   |
| slv_prfl          | BYTE     | 22   | Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.  |
| intr_type         | BYTE     | 23   | Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.   |
| flags1            | BYTE     | 24   | Siehe Parameter "flags1" der Datenstruktur TI_Cassic.  |
| flags2            | BYTE     | 25   | Siehe Parameter "flags2" der Datenstruktur TI_Cassic.  |
| id                | UINT     | 26   | Siehe Parameter "id" der Datenstruktur TI_Cassic.  |
| manufacturer      | UINT     | 28   | Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.  |
| instance          | UINT     | 30   | Siehe Parameter "instance" der Datenstruktur TI_Cassic.  |

Tabelle 9-49 Baugruppenträgerfehler-OB: Datenstruktur TI\_StationFailure

| Parameter                | Datentyp  | Byte | Beschreibung   |
|--------------------------|-----------|------|--|
| <b>TI_StationFailure</b> |           |      | Datenstruktur für Baugruppenträgerfehler-OB  |
| SI_Format                | USINT     | 0    | Format der Startinformationen:<br><ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>   |
| OB_Class                 | USINT     | 1    | OB-Klasse (=86)  |
| OB_Nr                    | UINT      | 2    | OB-Nummer (1...32767).   |
| LADDR                    | HW_Device | 4    | Hardware-Kennung des fehlerhaften Hardware-Objekts.  |
| Event_Class              | BYTE      | 6    | <ul style="list-style-type: none"> <li>• B#16#38: (Sub-) Modul gesteckt</li> <li>• B#16#39: (Sub-) Modul gezogen bzw. nicht ansprechbar</li> </ul>   |
| Fault_ID                 | BYTE      | 7    | <p>Fehlercode</p> <p>Über den Fehlercode wird ausgegeben, welches Ereignis den Start des Ziehen/ Stecken-OB verursacht hat.</p> <ul style="list-style-type: none"> <li>• Bei Event_Class = B#16#38 - (Sub-) Modul gesteckt: <ul style="list-style-type: none"> <li>- B#16#54: PROFINET IO-Submodul gesteckt und entspricht parametriertem Submodul</li> <li>- B#16#55: PROFINET IO-Submodul gesteckt, entspricht aber nicht dem parametrierten Submodul</li> <li>- B#16#56: PROFINET IO-Submodul gesteckt, jedoch Fehler bei Baugruppenparametrierung</li> <li>- B#16#57: PROFINET IO-Submodul oder -Modul gesteckt, jedoch mit Störung oder Wartung</li> <li>- B#16#58: PROFINET IO-Submodul Zugriffsfehler beseitigt</li> </ul> </li> <li>• Bei Event_Class = B#16#39 - (Sub-) Modul gezogen bzw. nicht ansprechbar: <ul style="list-style-type: none"> <li>- B#16#51: PROFINET IO-Modul gezogen</li> <li>- B#16#54: PROFINET IO-Submodul gezogen</li> </ul> </li> </ul> |
| address                  | WORD      | 20   | Siehe Parameter "address" der Datenstruktur TI_Cassic.   |
| slv_prfl                 | BYTE      | 22   | Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.  |
| intr_type                | BYTE      | 23   | Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.   |
| flags1                   | BYTE      | 24   | Siehe Parameter "flags1" der Datenstruktur TI_Cassic.  |
| flags2                   | BYTE      | 25   | Siehe Parameter "flags2" der Datenstruktur TI_Cassic.  |
| id                       | UINT      | 26   | Siehe Parameter "id" der Datenstruktur TI_Cassic.  |
| manufacturer             | UINT      | 28   | Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.  |
| instance                 | UINT      | 30   | Siehe Parameter "instance" der Datenstruktur TI_Cassic.  |

Tabelle 9-50 Programmierfehler-OB / Peripheriezugriffsfehler-OB: Datenstruktur TI\_ProgIOAccessError

| Parameter                   | Datentyp | Byte | Beschreibung*  |
|-----------------------------|----------|------|--|
| <b>TI_ProgIOAccessError</b> |          |      | Datenstruktur für Programmierfehler-OB und Peripheriezugriffsfehler-OB   |
| SI_Format                   | USINT    | 0    | Format der Startinformationen:<br><ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul> |

| Parameter | Datentyp | Byte | Beschreibung*   |
|-----------|----------|------|---|
| OB_Class  | USINT    | 1    | OB-Klasse: <ul style="list-style-type: none"> <li>• =121 für Programmierfehler-OB</li> <li>• =122 für Peripheriezugriffsfehler-OB</li> </ul>  |
| OB_Nr     | UINT     | 2    | OB-Nummer (1...32767).  |
| BlockNr   | UINT     | 4    | Nummer des Bausteins, in dem der Programmierfehler auftrat.   |
| Reaction  | USINT    | 6    | <ul style="list-style-type: none"> <li>• 0: Fehler ignorieren</li> <li>• 1: fehlerhaften Wert ersetzen</li> <li>• 2: Befehl überspringen</li> </ul>   |
| Fault_ID  | BYTE     | 7    | Fehlercode <ul style="list-style-type: none"> <li>• B#16#21: BCD-Konvertierungsfehler</li> <li>• B#16#22: Bereichslängenfehler beim Lesen</li> <li>• B#16#23: Bereichslängenfehler beim Schreiben</li> <li>• B#16#24: Bereichsfehler beim Lesen</li> <li>• B#16#25: Bereichsfehler beim Schreiben</li> <li>• B#16#26: Fehler bei Timer-Nr.</li> <li>• B#16#27: Fehler bei Zähler-Nr.</li> <li>• B#16#28: Lesender Zugriff auf ein Byte, Wort oder Doppelwort mit einem Pointer, dessen Bitadresse ungleich Null ist</li> <li>• B#16#29: Schreibender Zugriff auf ein Byte, Wort oder Doppelwort mit einem Pointer, dessen Bitadresse ungleich Null ist</li> <li>• B#16#30: Schreibender Zugriff auf einen schreibgeschützten Global-DB</li> <li>• B#16#31: Schreibender Zugriff auf einen schreibgeschützten Instanz-DB</li> <li>• B#16#32: DB-Nummernfehler beim Zugriff auf einen Global-DB</li> <li>• B#16#33: DB-Nummernfehler beim Zugriff auf einen Instanz-DB</li> <li>• B#16#34: Nummernfehler bei FC-Aufruf</li> <li>• B#16#35: Nummernfehler bei FB-Aufruf</li> <li>• B#16#42: Peripheriezugriffsfehler, lesend</li> <li>• B#16#43: Peripheriezugriffsfehler, schreibend</li> <li>• B#16#3A: Zugriff auf einen nicht geladenen DB; die DB-Nummer liegt im zulässigen Bereich</li> <li>• B#16#3C: Zugriff auf eine nicht geladene FC; die FC-Nummer liegt im zulässigen Bereich</li> <li>• B#16#3D: Zugriff auf eine nicht geladene Anweisung (SFC); die SFC-Nummer liegt im zulässigen Bereich</li> <li>• B#16#3E: Zugriff auf einen nicht geladenen FB; die FB-Nummer liegt im zulässigen Bereich</li> <li>• B#16#3F: Zugriff auf einen nicht vorhandenen SFB; die SFB-Nummer liegt im zulässigen Bereich</li> </ul> |
| BlockType | USINT    | 8    | Art des Bausteins, in dem der Fehler aufgetreten ist: <ul style="list-style-type: none"> <li>• OB: B#16#88</li> <li>• FC: B#16#8C</li> <li>• FB: B#16#8E</li> </ul>   |

| Parameter | Datentyp | Byte | Beschreibung*   |
|-----------|----------|------|---|
| Area      | USINT    | 9    | Speicherbereich, bei dem der fehlerhafte Zugriff auftrat: <ul style="list-style-type: none"> <li>• Lokaldaten: B#16#40 bis 4E, 86, 87, 8E, 8F, C0 bis CE</li> <li>• Prozessabbild der Eingänge: B#16#01</li> <li>• Prozessabbild der Ausgänge: B#16#02</li> <li>• Technologie-DB: B#16#04</li> <li>• E: B#16#81</li> <li>• A: B#16#82</li> <li>• M: B#16#83</li> <li>• DB: B#16#84, 85, 8A, 8B</li> </ul> |
| DBNr      | DB_ANY   | 10   | DB-Nr, falls Area = DB (Global-DB oder Instanz-DB)<br>Nur relevant bei Programmierfehler-OB.  |
| Csg_OBNr  | OB_ANY   | 12   | OB-Nummer: <ul style="list-style-type: none"> <li>• 121: Programmierfehler-OB</li> <li>• 122: Peripheriezugriffsfehler-OB</li> </ul>  |
| Csg_Prio  | USINT    | 14   | OB-Priorität  |
| Width     | USINT    | 15   | Art des Zugriffs, bei dem der Fehler auftrat: <ul style="list-style-type: none"> <li>• Bit: B#16#00</li> <li>• Byte: B#16#01</li> <li>• Word: B#16#02</li> <li>• DWord: B#16#03</li> <li>• LWord: B#16#04</li> </ul>  |

\* Abhängig davon, zu welchem OB (Peripheriezugriffsfehler-OB oder Programmierfehler-OB) Informationen ausgelesen werden, sind nur bestimmte Ausgabewerte möglich.

Tabelle 9-51 Uhrzeitalarm-OB: Datenstruktur TI\_TimeOfDay

| Parameter    | Datentyp | Byte | Beschreibung  |
|--------------|----------|------|---|
| TI_TimeOfDay |          |      | Datenstruktur für Uhrzeitalarm-OB   |
| SI_Format    | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>                           |
| OB_Class     | USINT    | 1    | OB-Klasse (=10)   |
| OB_Nr        | UINT     | 2    | OB-Nummer (1...32767).  |
| CaughtUp     | BOOL     | 4    | =TRUE, wenn ein OB-Aufruf wegen vorgestellter Uhrzeit nachgeholt wurde  |
| SecondTime   | BOOL     | 5    | =TRUE, wenn ein OB wegen zurückgestellter Uhrzeit ein zweites Mal aufgerufen wurde. Hinweis: SecondTime wird bei zurückgestellter Uhrzeit nur ein einziges Mal gesetzt. |

Tabelle 9-52 Taktsynchronalarm-OB: Datenstruktur TI\_SynchCycle

| Parameter     | Datentyp | Byte | Beschreibung  |
|---------------|----------|------|---|
| TI_SynchCycle |          |      | Datenstruktur für Taktsynchronalarm-OB  |
| SI_Format     | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul> |

| Parameter     | Datentyp | Byte | Beschreibung   |
|---------------|----------|------|--|
| OB_Class      | USINT    | 1    | OB-Klasse (=61)  |
| OB_Nr         | UINT     | 2    | OB-Nummer (1...32767).   |
| Initial_Call  | BOOL     | 4    | =TRUE im ersten Aufruf dieses OB bei: <ul style="list-style-type: none"> <li>• Übergang von STOP bzw. HALT nach RUN</li> <li>• Nach Neuladen</li> </ul>  |
| IO_System     | USINT    | 5    | Nummer des Alarm auslösenden Dezentralen Peripheriesystems   |
| Event_Count   | INT      | 6    | <ul style="list-style-type: none"> <li>• = n: Anzahl der verlorenen Takte</li> <li>• = -1: Es ist eine unbekannte Anzahl von Takten verlorengegangen (z. B. weil der Takt sich geändert hat).</li> </ul> |
| PIP_Input     | BOOL     | 10   | =TRUE: Das zugehörige Prozessabbild der Eingänge ist aktuell   |
| PIP_Output    | BOOL     | 11   | =TRUE: Das zugehörige Prozessabbild der Ausgänge wurde nach dem letzten Zyklus rechtzeitig an die Ausgänge übertragen  |
| SyncCycleTime | LTIME    | 16   | Berechnete Zykluszeit  |

Tabelle 9-53 Statusalarm-OB / Update-Alarm-OB / OB für hersteller- bzw. profilspezifischen Alarm: Datenstruktur TI\_Submodule

| Parameter    | Datentyp | Byte | Beschreibung  |
|--------------|----------|------|---|
| TI_Submodule |          |      | Datenstruktur für Statusalarm-OB, Update-Alarm-OB und OB für hersteller- bzw. profilspezifischen Alarm  |
| SI_Format    | USINT    | 0    | Format der Startinformationen: <ul style="list-style-type: none"> <li>• 16#FF: keine</li> <li>• 16#FE: Optimierte Startinformation</li> </ul>   |
| OB_Class     | USINT    | 1    | OB-Klasse <ul style="list-style-type: none"> <li>• =55 für Statusalarm-OB</li> <li>• =56 für Update-Alarm OB</li> <li>• =57 für hersteller- bzw. profilspezifischen Alarm-OB</li> </ul> |
| OB_Nr        | UINT     | 2    | OB-Nummer (1...32767).  |
| LADDR        | HW_IO    | 4    | Hardware-Adresse der Alarm auslösenden Komponente   |
| Slot         | UINT     | 6    | Steckplatz-Nr. der Alarm auslösenden Komponente   |
| Specifier    | WORD     | 8    | Alarm Specifier aus dem Alarm-Telegramm   |
| address      | WORD     | 20   | Siehe Parameter "address" der Datenstruktur TI_Cassic.  |
| slv_prfl     | BYTE     | 22   | Siehe Parameter "slv_prfl" der Datenstruktur TI_Cassic.   |
| intr_type    | BYTE     | 23   | Siehe Parameter "intr_type" der Datenstruktur TI_Cassic.  |
| flags1       | BYTE     | 24   | Siehe Parameter "flags1" der Datenstruktur TI_Cassic.   |
| flags2       | BYTE     | 25   | Siehe Parameter "flags2" der Datenstruktur TI_Cassic.   |
| id           | UINT     | 26   | Siehe Parameter "id" der Datenstruktur TI_Cassic.   |
| manufacturer | UINT     | 28   | Siehe Parameter "manufacturer" der Datenstruktur TI_Cassic.   |
| instance     | UINT     | 30   | Siehe Parameter "instance" der Datenstruktur TI_Cassic.   |

**Parameter AINFO**

**Datenstruktur des Zielbereichs AINFO bei Alarmen von PROFIBUS DP**

| Byte     | Bedeutung   |
|----------|---|
| 0 bis 3  | Kopfinformation, genaue Beschreibung s. u.          |
| 4 bis 63 | Alarmzusatzinformation: Daten zum jeweiligen Alarm: |
|          | dezentral: ARRAY[0] bis ARRAY[59]                   |

**Aufbau der Kopfinformation bei Alarmen von PROFIBUS DP**

| Byte   | Datentyp | Bedeutung   |
|--|----------|---|
| 0  | BYTE     | Länge der empfangenen Alarminformation in Bytes   |
|  |          | zentral: 4 bis 224  |
|  |          | dezentral: 4 bis 63   |
| 1  | BYTE     | zentral: reserviert   |
|  |          | dezentral: Kennung für den Alarmtyp   |
|  |          | 1: Diagnosealarm  |
|  |          | 2: Prozessalarm   |
|  |          | 3: Ziehen-Alarm   |
| 4: Stecken-Alarm   |          |   |
| 5: Statusalarm   |          |   |
| 6: Update-Alarm  |          |   |
| 31: Ausfall eines Erweiterungsgeräts, eines DP-Mastersystems oder einer DP-Station |          |   |
| 32 bis 126: herstellerspezifischer Alarm   |          |   |
| 2  | BYTE     | Steckplatznummer der Alarm auslösenden Komponente   |
| 3  | BYTE     | zentral: reserviert   |
|  |          | dezentral: Specifier  |
|  |          | Bits 0 und 1: 0: keine weitere Information;<br>1: kommendes Ereignis, Steckplatz gestört<br>2: gehendes Ereignis, Steckplatz nicht mehr gestört<br>3: gehendes Ereignis, Steckplatz weiterhin gestört |
|  |          | Bit 2: Add_Ack  |
|  |          | Bits 3 bis 7: Sequenznummer   |

**Datenstruktur des Zielbereichs AINFO bei Alarmen von PROFINET IO oder zentraler Peripherie**

| Byte        | Bedeutung  |
|-------------|--|
| 0 bis 25    | Kopfinformation, genaue Beschreibung s. u.   |
| 26 bis 1431 | Alarmzusatzinformation: Genormte Diagnosedaten zum jeweiligen Alarm:<br>ARRAY[0] bis ARRAY[1405]<br>Hinweis: Die Alarmzusatzinformation kann auch entfallen. |



### Aufbau der Kopfinformation bei Alarmen von PROFINET IO oder zentraler Peripherie

| Byte      | Datentyp | Bedeutung  |
|-----------|----------|--|
| 0 und 1   | WORD     | <ul style="list-style-type: none"> <li>• Bits 0 bis 7: Bausteintyp</li> <li>• Bits 8 bis 15: reserviert</li> </ul>   |
| 2 und 3   | WORD     | Bausteinlänge  |
| 4 und 5   | WORD     | Version: <ul style="list-style-type: none"> <li>• Bits 0 bis 7: low byte</li> <li>• Bits 8 bis 15: high byte</li> </ul>  |
| 6 und 7   | WORD     | Kennung für den Alarmtyp: <ul style="list-style-type: none"> <li>• 1: Diagnosealarm (kommend)</li> <li>• 2: Prozessalarm</li> <li>• 3: Ziehen-Alarm</li> <li>• 4: Stecken-Alarm</li> <li>• 5: Status-Alarm</li> <li>• 6: Update-Alarm</li> <li>• 7: Redundanz-Alarm</li> <li>• 8: Vom Supervisor gesteuert (Controlled_by_Supervisor)</li> <li>• 9: Vom Supervisor freigegeben (Released_by_Supervisor)</li> <li>• 10: Nicht die projektierte Baugruppe gesteckt</li> <li>• 11: Wiederkehr des Submoduls</li> <li>• 12: Diagnosealarm (gehend)</li> <li>• 13: Querverkehr-Verbindungsmeldung</li> <li>• 14: Nachbarschaftsänderungsmeldung</li> <li>• 15: Taktsynchronisationsmeldung (busseitig)</li> <li>• 16: Taktsynchronisationsmeldung (geräteseitig)</li> <li>• 17: Netzwerkkomponentenmeldung</li> <li>• 18: Uhrzeitsynchronisationsmeldung (busseitig)</li> <li>• 19 bis 31: Reserviert</li> <li>• 32 bis 127: Herstellerspezifischer Alarm</li> <li>• 128 bis 65535: Reserviert</li> </ul> |
| 8 bis 11  | DWORD    | API (Application Process Identifier)   |
| 12 bis 13 | WORD     | Steckplatznummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535)   |
| 14 bis 15 | WORD     | Submodulsteckplatznummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535)   |
| 16 bis 19 | DWORD    | Modul-Identifikation; eindeutige Information der Alarmquelle   |

| Byte      | Datentyp | Bedeutung   |
|-----------|----------|---|
| 20 bis 23 | DWORD    | Submodul-Identifikation; eindeutige Information der Alarmquelle   |
| 24 bis 25 | WORD     | <p>Alarm Specifier:</p> <ul style="list-style-type: none"> <li>• Bits 0 bis 10: Sequenznummer (Wertebereich 0 bis 2047)</li> <li>• Bit 11: Kanaldiagnose:                             <ul style="list-style-type: none"> <li>0: keine Kanaldiagnose vorhanden</li> <li>1: Kanaldiagnose vorhanden</li> </ul> </li> <li>• Bit 12: Status der herstellerspezifischen Diagnose:                             <ul style="list-style-type: none"> <li>0: keine herstellerspezifische Statusinformation vorhanden</li> <li>1: Herstellerspezifische Statusinformation vorhanden</li> </ul> </li> <li>• Bit 13: Status der Diagnose beim Submodul:                             <ul style="list-style-type: none"> <li>0: keine Statusinformation vorhanden, alle Fehler wurden beseitigt</li> <li>1: Mindestens eine Kanaldiagnose und/oder Statusinformation vorhanden</li> </ul> </li> <li>• Bit 14: reserviert</li> <li>• Bit 15: Application Relationship Diagnosis State:                             <ul style="list-style-type: none"> <li>– 0: keine der innerhalb dieser AR projektierten Baugruppen meldet eine Diagnose</li> <li>– 1: Mindestens eine innerhalb dieser AR projektierten Baugruppen meldet eine Diagnose</li> </ul> </li> </ul> |

**Aufbau der Alarmzusatzinformation bei Alarmen von PROFINET IO oder zentraler Peripherie**

Die Alarmzusatzinformation bei PROFINET IO ist abhängig von der Formatkennung. Sie kann aus mehreren Datenblöcken mit gleicher oder unterschiedlicher Formatkennung bestehen. Folgende Formatkennungen sind möglich:

- W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose

| Byte    | Datentyp | Bedeutung   |
|---------|----------|---|
| 0 bis 1 | WORD     | Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten:<br>W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose |
| 2 bis n | BYTE     | Siehe Handbuch des Herstellers.   |

- W#16#8000: Kanaldiagnose  
Die Kanaldiagnose wird in Blöcken zu jeweils 6 Byte ausgegeben. Die Alarmzusatzinformation (ohne Formatkennung) wird nur für die gestörten Kanäle ausgegeben.

| Byte    | Datentyp | Bedeutung   |   |
|---------|----------|---|---|
| 0 bis 1 | WORD     | Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten:<br>W#16#8000: Kanaldiagnose  |   |
| 2 bis 3 | WORD     | Kanalnummer der Alarm auslösenden Komponente (Wertebereich 0 bis 65535): <ul style="list-style-type: none"> <li>• W#16#0000 bis W#16#7FFF: Kanalnummer des Submoduls/Moduls</li> <li>• W#16#8000: ist der Stellvertreter für das gesamte Submodul</li> <li>• W#16#8001 bis W#16#FFFF: reserviert</li> </ul> |   |
| 4       | BYTE     | Bits 0 bis 2:   | Reserviert  |
|         |          | Bits 3 bis 4:   | Fehlerart: <ul style="list-style-type: none"> <li>• 0: reserviert</li> <li>• 1: kommender Fehler</li> <li>• 2: gehender Fehler</li> <li>• 3: gehender Fehler, weitere Fehler vorhanden</li> </ul> |
|         |          | Bits 5 bis 7:   | Kanalart: <ul style="list-style-type: none"> <li>• 0: reserviert</li> <li>• 1: Eingabekanal</li> <li>• 2: Ausgabekanal</li> <li>• 3: Ein-/Ausgabekanal</li> </ul>                                 |

| Byte    | Datentyp | Bedeutung  |
|---------|----------|--|
| 5       | BYTE     | Datenformat: <ul style="list-style-type: none"> <li>• B#16#00: freies Datenformat</li> <li>• B#16#01: Bit</li> <li>• B#16#02: 2 Bit</li> <li>• B#16#03: 4 Bit</li> <li>• B#16#04: Byte</li> <li>• B#16#05: Wort</li> <li>• B#16#06: Doppelwort</li> <li>• B#16#07: 2 Doppelworte</li> <li>• B#16#08 bis B#16#FF: reserviert</li> </ul>   |
| 6 bis 7 | WORD     | Fehlertyp: <ul style="list-style-type: none"> <li>• W#16#0000: reserviert</li> <li>• W#16#0001: Kurzschluss</li> <li>• W#16#0002: Unterspannung</li> <li>• W#16#0003: Überspannung</li> <li>• W#16#0004: Überlast</li> <li>• W#16#0005: Übertemperatur</li> <li>• W#16#0006: Leitungsbruch</li> <li>• W#16#0007: Oberer Grenzwert überschritten</li> <li>• W#16#0008: Unterer Grenzwert überschritten</li> <li>• W#16#0009: Fehler</li> <li>• W#16#000A bis W#16#000F: reserviert</li> <li>• W#16#0010 bis W#16#001F: herstellerspezifisch</li> <li>• W#16#0020 bis W#16#00FF: reserviert</li> <li>• W#16#0100 bis w#16#7FFF: herstellerspezifisch</li> <li>• W#16#8000: Gerätediagnose liegt vor</li> <li>• W#16#8001 bis W#16#FFFF: reserviert</li> </ul> Nicht jeder Kanal unterstützt jeden Fehlertyp. Details finden Sie in der Beschreibung der Diagnosedaten des entsprechenden Geräts. |

**Hinweis**

Der Teil "Kanalnummer" bis "Fehlertyp" kann 0 bis n mal auftreten.

- W#16#8001: MULTIPLE (unterschiedliche Arten von Diagnoseinformationen werden übertragen). Die Alarmzusatzinformation wird in diesem Fall in Blockstruktur mit variabler Länge übertragen.

| Byte      | Datentyp | Bedeutung  |
|-----------|----------|--|
| 0 bis 1   | WORD     | Formatkennung für den Aufbau der als Alarmzusatzinformation folgenden Daten:<br>W#16#8001: Herstellerspezifische Diagnose und/oder Kanaldiagnose   |
| 2 bis 3   | WORD     | Bausteintyp  |
| 4 bis 5   | WORD     | Bausteinlänge  |
| 6         | BYTE     | Version: high byte   |
| 7         | BYTE     | Version: low byte  |
| 8 bis 11  | DWORD    | API (nur falls das low byte von Version = 1)   |
| 12 bis 13 | WORD     | Steckplatznummer   |
| 14 bis 15 | WORD     | Substeckplatznummer  |
| 16 bis 17 | WORD     | Kanalnummer  |
| 18 bis 19 | WORD     | Kanaleigenschaften   |
| 20 bis 21 | WORD     | Formatkennung: <ul style="list-style-type: none"> <li>• W#16#0000 bis W#16#7FFF: Herstellerspezifische Diagnose</li> <li>• W#16#8000: Kanaldiagnose</li> <li>• W#16#8002: Erweiterte Kanaldiagnose</li> <li>• W#16#8003: Abgestufte erweiterte Kanaldiagnose</li> <li>• W#16#8004 bis W#16#80FF: reserviert</li> </ul> |
| 22 bis n  | BYTE     | Daten abhängig von der Formatkennung   |

#### Hinweis

Der Teil ab "Bausteintyp" kann 1 bis n mal auftreten.

- W#16#8002: Erweiterte Kanaldiagnose

| Byte      | Bedeutung                     |
|-----------|-------------------------------|
| 0 bis 1   | Formatkennung W#16#8002       |
| 2 bis 3   | Kanalnummer                   |
| 4 bis 5   | Kanaleigenschaften            |
| 6 bis 7   | Fehlertyp                     |
| 8 bis 9   | Zusätzlicher Fehlerwert       |
| 10 bis 13 | Zusätzliche Fehlerinformation |

- W#16#8003: Abgestufte erweiterte Kanaldiagnose

| Byte      | Bedeutung                     |
|-----------|-------------------------------|
| 0 bis 1   | Formatkennung W#16#8003       |
| 2 bis 3   | Kanalnummer                   |
| 4 bis 5   | Kanaleigenschaften            |
| 6 bis 7   | Fehlertyp                     |
| 8 bis 9   | Zusätzlicher Fehlerwert       |
| 10 bis 13 | Zusätzliche Fehlerinformation |
| 14 bis 17 | Qualified Channel Qualifier   |

- W#16#8100: Wartungsinformation

| Byte      | Bedeutung               |
|-----------|-------------------------|
| 0 bis 1   | Formatkennung W#16#8100 |
| 2 bis 3   | Bausteintyp             |
| 4 bis 5   | Bausteinlänge           |
| 6 bis 7   | Bausteinversion         |
| 8 bis 9   | Reserviert              |
| 10 bis 13 | Wartungszustand         |

**Hinweis**

Genauere Informationen zum Aufbau der Alarmzusatzinformation entnehmen Sie dem *Programmierhandbuch SIMATIC PROFINET IO Von PROFIBUS DP nach PROFINET IO* und dem aktuellen Stand der Norm IEC 61158-6-10-1.

**Zielbereich TINFO und AINFO**

**Zielbereich TINFO und AINFO**

Abhängig von dem jeweiligen OB, in dem "RALRM (Seite 2399)" aufgerufen wird, werden die Zielbereiche TINFO und AINFO nur teilweise beschrieben. Welche Information jeweils eingetragen wird, entnehmen Sie der folgenden Tabelle.

| Alarmtyp                     | OB-Klasse | TINFO OB-Startinformation | TINFO Verwaltungsinformation | AINFO Kopfinformation | AINFO Alarmzusatzinformation |  |
|------------------------------|-----------|---------------------------|------------------------------|-----------------------|------------------------------|--|
|                              |           |                           |                              |                       | zentral:                     | dezentral:   |
| Prozessalarm                 | 4x        | ja                        | ja                           | ja                    | zentral:                     | nein   |
|                              |           |                           |                              |                       | dezentral:                   | wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert |
| Statusalarm                  | 55        | ja                        | ja                           | ja                    | ja                           | ja   |
| Update-Alarm                 | 56        | ja                        | ja                           | ja                    | ja                           | ja   |
| Herstellerspezifischer Alarm | 57        | ja                        | ja                           | ja                    | ja                           | ja   |

| Alarmtyp   | OB-Klasse | TINFO OB-Startinformation | TINFO Verwaltungsinformation | AINFO Kopfinformation | AINFO Alarmzusatzinformation |  |
|--|-----------|---------------------------|------------------------------|-----------------------|------------------------------|--|
| Peripherie-Redundanzfehler                                   | 70        | ja                        | ja                           | nein                  | nein                         | nein   |
| Diagnosealarm  | 82        | ja                        | ja                           | ja                    | zentral:                     | Datensatz 1  |
|  |           |                           |                              |                       | dezentral:                   | wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert |
| Ziehen-/Stecken-Alarm  | 83        | ja                        | ja                           | ja                    | zentral:                     | Nein   |
|  |           |                           |                              |                       | dezentral:                   | wie vom PROFIBUS DP-Slave/PROFINET IO-Device geliefert |
| Sonderform des Ziehen-Alarms:<br>Vom Supervisor gesteuert    | 83        | ja                        | ja                           | ja                    | Nur PROFINET IO              |  |
| Sonderform des Stecken-Alarms:<br>Vom Supervisor freigegeben | 83        | ja                        | ja                           | ja                    | Nur PROFINET IO              |  |
| Nicht projektierte Baugruppe gesteckt                        | 83        | ja                        | ja                           | ja                    | Nur PROFINET IO              |  |
| Baugruppen-trägerausfall/<br>Stationsausfall                 | 86        | ja                        | ja                           | nein                  | nein                         |  |
| ... alle anderen OBs   |           | ja                        | nein                         | nein                  | nein                         |  |

## D\_ACT\_DP: DP-Slaves deaktivieren / aktivieren

### Beschreibung

Mit der Anweisung "D\_ACT\_DP" können Sie projektierte DP-Slaves/PROFINET IO-Devices gezielt deaktivieren und wieder aktivieren. Darüber hinaus können Sie für jeden eingesetzten DP-Slave bzw. für jedes eingesetzte PROFINET IO-Device ermitteln, ob diese Komponente momentan aktiviert oder deaktiviert ist.

Falls Sie mit Hilfe von der Anweisung einen Netzübergang vom Typ IE/PB Link PN IO deaktivieren, fallen alle daran angeschlossenen PROFIBUS DP-Slaves aus. Diese Ausfälle werden gemeldet.

Die Anweisung ist nicht anwendbar auf PROFIBUS PA-Feldgeräte, die über DP/PA Link an ein DP-Mastersystem angeschlossen sind.

---

#### **Hinweis**

So lange ein oder mehrere "D\_ACT\_DP"-Aufträge aktiv sind, können Sie keine geänderte Konfiguration vom PG in die CPU laden (im Rahmen von CiR).

Während des Ladens einer geänderten Konfiguration vom PG in die CPU im laufenden Betrieb (CiR) weist die CPU die Aktivierung eines "D\_ACT\_DP"-Auftrags ab.

Für die Bearbeitung eines Deaktivier- bzw. Aktivierauftrags sind mehrere Durchläufe durch den Zykluskontrollpunkt notwendig. Sie können daher nicht in einer programmierten Schleife auf das Ende eines solchen Auftrags warten.

---

#### **Arbeitsweise**

"D\_ACT\_DP" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "D\_ACT\_DP" mit REQ=1 aufrufen.

Über die Ausgangsparameter RET\_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

#### **Anwendungsbereich**

Wenn Sie in einer CPU DP-Slaves/PROFINET IO-Devices konfigurieren, die real nicht vorhanden sind oder aktuell nicht benötigt werden, greift die CPU dennoch regelmäßig auf diese DP-Slaves/PROFINET IO Devices zu. Nach deren Deaktivierung unterbleiben weitere CPU-Zugriffe. Bei PROFIBUS DP kann dadurch der schnellstmögliche DP-Buszyklus erreicht werden, und die entsprechenden Fehlerereignisse treten nicht mehr auf.

#### **Beispiele**

Im Serienmaschinenbau ist aus Sicht des Maschinenbauers eine Vielzahl von Maschinenoptionen möglich. Jede ausgelieferte Maschine beinhaltet jedoch nur eine individuelle Kombination ausgewählter Optionen.

Sämtliche möglichen Maschinenoptionen sind vom Hersteller als DP-Slaves/PROFINET IO Devices projektiert, um ein gemeinsames Anwenderprogramm über alle möglichen Optionen erstellen und pflegen zu können. Mit "D\_ACT\_DP" können Sie im Anlauf der Maschine alle nicht vorhandenen DP-Slaves/PROFINET IO Devices deaktivieren.

Eine ähnliche Situation besteht bei Werkzeugmaschinen, die viele Werkzeuge vorhalten, aber zu einem Zeitpunkt nur wenige einsetzen. Hier werden die Werkzeuge als DP-Slaves/PROFINET IO Devices realisiert. Das Anwenderprogramm aktiviert mit Hilfe von "D\_ACT\_DP" die aktuell benötigten und deaktiviert die erst später wieder einzusetzenden Werkzeuge.

#### **Identifikation eines Auftrags**

Falls Sie einen Deaktivierungs- bzw. Aktivierungsauftrag angestoßen haben und Sie "D\_ACT\_DP" erneut aufrufen, bevor dieser beendet wurde, hängt das weitere Verhalten der



Anweisung entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt. Stimmt der Eingangsparameter LADDR überein, so gilt der Aufruf als Folgeaufruf.

### Deaktivieren von DP-Slaves/PROFINET IO-Devices

Wenn Sie einen DP-Slave bzw. ein PROFINET IO Device mit "D\_ACT\_DP" deaktivieren, werden dadurch dessen Prozessausgänge auf die projektierten Ersatzwerte bzw. auf 0 gesetzt (sicherer Zustand). Der zugehörige DP-Master/PROFINET IO Controller spricht diese Komponente im Weiteren nicht mehr an. Deaktivierte DP-Slaves/PROFINET IO-Devices werden an den Fehler-LEDs des DP-Masters/PROFINET IO-Controllers oder der CPU nicht als gestört oder fehlend gekennzeichnet.

Das Prozessabbild der Eingänge von deaktivierten DP-Slaves/PROFINET IO-Devices wird mit 0 aktualisiert, d. h. es wird wie bei ausgefallenen DP-Slaves/PROFINET IO-Devices behandelt.

Falls Sie in Ihrem Programm mittels Direktzugriff auf die Nutzdaten eines zuvor deaktivierten DP-Slaves/PROFINET IO-Devices zugreifen, wird der Peripheriezugriffsfehler-OB aufgerufen und das zugehörige Starterereignis in den Diagnosepuffer eingetragen. Falls Sie per Anweisung (z. B. "RD\_REC (Seite 2430)") auf einen deaktivierten DP-Slave bzw. ein deaktiviertes PROFINET IO-Device zugreifen, erhalten Sie in RET\_VAL dieselbe Fehlerinformation wie bei einem nicht verfügbaren DP-Slave/PROFINET IO-Device.

Das Deaktivieren eines DP-Slaves/PROFINET IO-Devices verursacht keinen Start des Programmablauffehler-OB, auch wenn dessen Ein- bzw. Ausgänge zum systemseitig zu aktualisierenden Prozessabbild gehören. Es erfolgt auch kein Eintrag in den Diagnosepuffer.

Fällt eine DP-Station/PNIO-Station aus, nachdem Sie sie mit "D\_ACT\_DP" deaktiviert haben, wird der Ausfall vom Betriebssystem nicht erkannt.

Bei PROFIBUS DP gilt: Falls Sie DP-Slaves deaktivieren wollen, die als Sender am Querverkehr beteiligt sind, wird empfohlen, zuerst die Empfänger (Mithörer) zu deaktivieren, die mithören, welche Eingangsdaten der Sender seinem DP-Master schickt. Erst im Anschluss daran deaktivieren Sie den Sender.

### Aktivieren von DP-Slaves/PROFINET IO-Devices

Wenn Sie einen DP-Slave bzw. ein PROFINET IO-Device mit "D\_ACT\_DP" wieder aktivieren, wird diese Komponente vom zugehörigen DP-Master/PROFINET IO-Controller konfiguriert und parametrierung (wie bei der Wiederkehr einer ausgefallenen DP-Station/PROFINET IO-Station). Die Aktivierung ist abgeschlossen, wenn die Komponente Nutzdaten transferieren kann.

Das Aktivieren eines DP-Slaves/PROFINET IO-Devices verursacht keinen Start des Programmablauffehler-OBs, auch wenn dessen Ein- bzw. Ausgänge zum systemseitig zu aktualisierenden Prozessabbild gehören. Es erfolgt auch kein Eintrag in den Diagnosepuffer.

Wenn Sie versuchen, einen DP-Slave/PROFINET IO-Device mit "D\_ACT\_DP" zu aktivieren, der nicht erreichbar ist (z.B. weil dieser physikalisch vom Bus getrennt wurde), liefert die Anweisung nach Ablauf der konfigurierten Parametrierungszeit für dezentrale Peripherie den Fehlercode W#16#80A7. Der DP-Slave / das PROFINET IO-Device ist aktiviert und die Nichterreichbarkeit des aktivierten DP-Slaves / PROFINET IO-Device führt zur entsprechenden Anzeige in der Systemdiagnose.

Ist der DP-Slave/PROFINET IO-Device danach wieder erreichbar, resultiert dies in dem normalen Systemverhalten (z.B. in einem Aufruf des hierfür projektierten OBs).

**Hinweis**

Das Aktivieren eines DP-Slaves/PROFINET IO-Devices kann geraume Zeit dauern. Falls Sie einen laufenden Aktivierungsauftrag abbrechen wollen, starten Sie "D\_ACT\_DP" mit dem gleichen Wert für LADDR und MODE = 2. Sie wiederholen den Aufruf von "D\_ACT\_DP" mit MODE = 2 so lange, bis der erfolgreiche Abbruch des Aktivierungsauftrags mit RET\_VAL = 0 angezeigt wird.

Falls Sie DP-Slaves aktivieren wollen, die am Querverkehr beteiligt sind, wird empfohlen, zuerst die Sender und anschließend die Empfänger (Mithörer) zu aktivieren.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "D\_ACT\_DP":

| Parameter | Deklaration | Datentyp  | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------|------------------------------|--|
| REQ       | Input       | BOOL      | E, A, M, D, L                | Pegelgetriggelter Steuerparameter<br>REQ=1: Aktivieren bzw. Deaktivieren durchführen   |
| MODE      | Input       | USINT     | E, A, M, D, L oder Konstante | Auftragskennung<br>Mögliche Werte: <ul style="list-style-type: none"> <li>• 0: Auskunft einholen, ob die angesprochene Komponente aktiviert oder deaktiviert ist (Ausgabe über Parameter RET_VAL)</li> <li>• 1: DP-Slave/PROFINET IO-Device aktivieren</li> <li>• 2: DP-Slave/PROFINET IO-Device deaktivieren</li> </ul> |
| LADDR     | Input       | HW_DEVICE | E, A, M, D, L oder Konstante | Hardware-Kennung des DP-Slaves (HW_DPSlave) /PROFINET IO-Devices (HW_Device)<br><br>Die Nummer kann in der Netzsicht aus den Eigenschaften DP-Slaves/PROFINET IO-Devices oder dem Register "Systemkonstanten" der Standard-Variablentabelle übernommen werden.   |
| RET_VAL   | Return      | INT       | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.   |
| BUSY      | Output      | BOOL      | E, A, M, D, L                | Aktivkennung: <ul style="list-style-type: none"> <li>• BUSY=1: Der Auftrag ist noch aktiv.</li> <li>• BUSY=0: Der Auftrag wurde beendet.</li> </ul>  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...) | Erläuterung   |
|---------------------------|---|
| 0000                      | Der Auftrag wurde fehlerfrei durchgeführt.  |
| 0001                      | DP-Slave/PROFINET IO-Device ist aktiviert (Dieser Fehlercode ist nur möglich bei MODE = 0.)   |
| 0002                      | DP-Slave/PROFINET IO-Device ist deaktiviert (Dieser Fehlercode ist nur möglich bei MODE = 0.)   |
| 7000                      | Erstaufwurf mit REQ=0. Der über LADDR festgelegte Auftrag ist nicht aktiv; BUSY hat den Wert "0".   |
| 7001                      | Erstaufwurf mit REQ=1. Der über LADDR festgelegte Auftrag wurde angestoßen; BUSY hat den Wert "1".  |
| 7002                      | Zwischenaufwurf (REQ irrelevant). Der aktivierte Auftrag ist noch in Bearbeitung; BUSY hat den Wert "1".  |
| 8090                      | <ul style="list-style-type: none"> <li>Sie haben keine Baugruppe mit der in LADDR angegebenen Adresse projiziert.</li> <li>Sie betreiben Ihre CPU als I-Slave und haben in LADDR eine Adresse dieses I-Slaves angegeben.</li> </ul>   |
| 8092                      | Der laufende Deaktivierungsvorgang eines DP-Slaves/PROFINET IO-Devices (MODE=2) kann nicht durch seine Aktivierung (MODE=1) abgebrochen werden. Aktivieren Sie die Komponente zu einem späteren Zeitpunkt.  |
| 8093                      | Zu der in LADDR angegebenen Adresse gehört kein DP-Slave/PROFINET IO-Device der aktiviert oder deaktiviert werden kann.   |
| 8094                      | Sie haben versucht, ein Gerät zu aktivieren, das ein möglicher Partner eines Werkzeugwechsellports ist. Zu diesem Zeitpunkt ist an diesem Werkzeugwechsellport aber bereits ein anderes Gerät aktiviert. Das aktivierte Gerät bleibt aktiviert.   |
| 80A0                      | Fehler bei der Kommunikation zwischen der CPU und dem IO-Controller.  |
| 80A1                      | Die angesprochene Komponente konnte nicht parametrierbar werden. (Dieser Fehlercode ist nur möglich bei MODE = 1.)<br>Hinweis: Diese Fehlerinformation liefert "D_ACT_DP" nur dann, wenn während der Parametrierung des aktivierten Slaves/Devices diese Komponente wieder ausfällt. Wenn nur die Parametrierung einer einzelnen Baugruppe nicht erfolgreich war, liefert "D_ACT_DP" die Fehlerinformation W#16#0000.                                   |
| 80A2                      | Der angesprochene DP-Slave gibt keine Rückmeldung (Diese Fehlerinformation gibt es nicht bei PROFINET IO-Devices. Der Vorgang wird bei PROFINET nicht zeitüberwacht.)   |
| 80A3                      | Der betroffene DP-Master/PROFINET IO-Controller unterstützt diese Funktion nicht.   |
| 80A4                      | Die CPU unterstützt diese Funktion bei externen DP-Mastern/PROFINET IO-Controllern nicht.   |
| 80A6                      | Steckplatzfehler im DP-Slave/PROFINET IO-Device; es kann nicht auf alle Nutzdaten zugegriffen werden (Dieser Fehlercode ist nur möglich bei MODE=1).<br>Hinweis: Diese Fehlerinformation liefert "D_ACT_DP" nur dann, wenn nach der Parametrierung der aktivierten Komponente und vor dem Ende von "D_ACT_DP" die Komponente wieder ausfällt. Wenn nur eine einzelne Baugruppe nicht verfügbar ist, liefert "D_ACT_DP" die Fehlerinformation W#16#0000. |
| 80A7                      | Aktivierung eines nicht erreichbaren Geräts.  |
| 80AA                      | Aktivierung mit Fehlern im DP-Slave/PROFINET IO-Device: Unterschiede in der Konfiguration   |
| 80AB                      | Aktivierung mit Fehlern im DP-Slave/PROFINET IO-Device: Parametrierfehler   |
| 80AC                      | Aktivierung mit Fehlern im DP-Slave/PROFINET IO-Device: Wartungsbedarf erforderlich   |
| 80C1                      | "D_ACT_DP" wurde gestartet und wird mit einer anderen Adresse fortgesetzt (Dieser Fehlercode ist möglich bei MODE=1 und bei MODE=2).  |
| 80C3                      | <ul style="list-style-type: none"> <li>Temporärer Ressourcenfehler: Die CPU bearbeitet momentan das mögliche Maximum an Aktivierungs-/Deaktivierungsaufträgen. (Dieser Fehlercode ist nur möglich bei MODE = 1 und MODE = 2.)</li> <li>Die CPU erhält gerade eine geänderte Konfiguration. Das Deaktivieren und Aktivieren von DP-Slaves/PROFINET IO-Devices ist daher momentan nicht möglich.</li> </ul>   |
| 80C5                      | DP: Vom Anwender noch nicht abgeholte Aufträge werden durch Wiederanlauf verworfen.   |

| Fehlercode*<br>(W#16#...)   | Erläuterung   |
|---|---|
| 80C6  | PROFINET: Vom Anwender noch nicht abgeholte Aufträge werden durch Wiederanlauf verworfen. |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)                             |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Weitere**

**RD\_REC: Datensatz von Peripherie lesen**

**Beschreibung**

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM von der adressierten Baugruppe. Sie starten den Lesevorgang, indem Sie beim Aufruf den Eingangsparameter REQ mit "1" belegen. Falls der Lesevorgang sofort ausgeführt werden konnte, liefert die Anweisung am Ausgangsparameter BUSY den Wert "0" zurück. Falls BUSY den Wert "1" hat, ist der Lesevorgang noch nicht abgeschlossen.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620). Der gelesene Datensatz wird nach fehlerfreier Datenübertragung in den durch RECORD aufgespannten Zielbereich eingetragen.

---

**Hinweis**

Falls Sie einen Datensatz, dessen Nummer größer als eins ist, aus einer FM oder einem CP auslesen, die Sie vor Februar 1997 erworben haben (im Folgenden als "alte Baugruppen" bezeichnet), dann verhält sich "RD\_REC" anders als bei einer neuen Baugruppe. Dieser Sonderfall wird im Abschnitt "Verwendung alter S7-300-FMs und -CPs bei DS-Nr. >1" (siehe unten) behandelt.

Wenn ein DPV1-Slave über GSD-Datei projiziert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Masters auf "S7-kompatibel" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit "RD\_REC" von den E/A-Baugruppen gelesen werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz (projizierter Steckplatz + 3).

Abhilfe: Schnittstelle des DP-Masters auf "DPV1" umstellen.

---

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_REC":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung  |
|-----------|-------------|--------------|------------------------------|---|
| REQ       | Input       | BOOL         | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Lesen  |
| LADDR     | Input       | HW_IO (WORD) | E, A, M, D, L oder Konstante | Hardware-Kennung der Baugruppe.   |
| RECNUM    | Input       | BYTE         | E, A, M, D, L oder Konstante | Datensatznummer (zulässige Werte: 0 bis 240)  |
| RET_VAL   | Return      | INT          | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. Zusätzlich: Länge des tatsächlich übertragenen Datensatzes in Bytes (mögliche Werte: +1 bis +240), falls der Zielbereich größer ist als der übertragene Datensatz und bei der Übertragung kein Fehler auftrat.   |
| BUSY      | Output      | BOOL         | E, A, M, D, L                | BUSY = 1: Der Lesevorgang ist noch nicht beendet.   |
| RECORD    | Output      | ANY          | E, A, M, D, L                | Zielbereich für den gelesenen Datensatz. Sie müssen bei asynchroner Bearbeitung von "RD_REC" darauf achten, dass die Aktualparameter von RECORD bei allen Aufrufen dieselbe Längenangabe haben. Es ist nur der Datentyp BYTE zulässig.<br>Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPU's unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RECORD

### Hinweis

Wenn Sie sicherstellen wollen, dass immer der gesamte Datensatz gelesen wird, wählen Sie einen Zielbereich mit der Länge 241 Bytes. Bei einer fehlerfreien Datenübertragung steht dann in RET\_VAL die tatsächliche Datensatzlänge.

### Verwendung alter S7-300-FMs und -CPs bei DS-Nr. > 1

Falls Sie mit der Anweisung "RD\_REC" aus einer alten S7-300-FM oder einem alten S7-300-CP einen Datensatz auslesen wollen, dessen Nummer größer als eins ist, müssen Sie Folgendes beachten:

- Falls der Zielbereich größer als die tatsächliche Länge des gewünschten Datensatzes ist, werden keine Daten in RECORD eingetragen. RET\_VAL wird mit W#16#80B1 beschrieben.
- Falls der Zielbereich kleiner als die tatsächliche Länge des gewünschten Datensatzes ist, liest die CPU so viele Bytes ab Datensatzanfang, wie in der Längenangabe von RECORD angegeben ist, und trägt sie in RECORD ein. RET\_VAL wird mit "0" beschrieben.
- Falls die Längenangabe von RECORD gleich der tatsächlichen Länge des gewünschten Datensatzes ist, liest die CPU den gewünschten Datensatz und trägt ihn in RECORD ein, RET\_VAL wird mit "0" beschrieben.

### Parameter RET\_VAL

- Trat während der Bearbeitung der Funktion ein Fehler auf, enthält der Rückgabewert einen Fehlercode.
- Falls bei der Übertragung kein Fehler auftrat, enthält RET\_VAL
  - 0, falls der gesamte Zielbereich mit Daten aus dem selektierten Datensatz gefüllt wurde (Der Datensatz kann aber unvollständig sein.).
  - die Länge des tatsächlich übertragenen Datensatzes in Bytes (mögliche Werte: +1 bis +240), falls der Zielbereich größer ist als der übertragene Datensatz.

---

#### Hinweis

Falls der allgemeine Fehler W#16#8745 auftritt, zeigt dies nur an, dass beim Schreibvorgang auf das Prozessabbild der Zugriff auf mindestens ein Byte nicht möglich war. Der Datensatz wurde ordnungsgemäß von der Baugruppe gelesen und in den E/A-Speicherbereich geschrieben.

---

Bei den "echten" Fehlerinformationen (Fehlercodes W#16#8xyz) der folgenden Tabelle sind zwei Fälle zu unterscheiden:

- Temporäre Fehler (Fehlercodes W#16#80A2 bis 80A3, 80Cx):  
Bei dieser Fehlerart ist es möglich, dass sich der Fehler ohne Ihr Zutun behebt, d. h. es ist sinnvoll, dass Sie die Anweisung erneut aufrufen (ggf. mehrfach).  
Beispiel für einen temporären Fehler: Benötigte Betriebsmittel sind momentan belegt (W#16#80C3).
- Permanente Fehler (Fehlercodes W#16#809x, 80A0, 80A1, 80Bx):  
Bei dieser Fehlerart kann sich der Fehler nicht ohne Ihr Zutun beheben. Ein erneuter Aufruf der Anweisung ist erst sinnvoll, wenn Sie den Fehler beseitigt haben. Beispiel für einen permanenten Fehler: Falsche Längenangabe in RECORD (W#16#80B1).

#### Hinweis

Wenn Sie mit "WR\_REC (Seite 2435)" Datensätze zu einem DPV1-Slave übertragen oder mit RD\_REC Datensätze von einem DPV1-Slave lesen und dieser im DPV1-Mode arbeitet, bewertet der DP-Master die vom Slave erhaltene Fehlerinformation wie folgt:

Liegt die Fehlerinformation in den Bereichen W#16#8000 bis W#16#80FF oder W#16#F000 bis W#16#FFFF, reicht der DP-Master die Fehlerinformation an die Anweisung weiter. Liegt sie außerhalb, gibt er den Wert W#16#80A2 an die Anweisung und suspendiert den Slave.

Zur Beschreibung der von DPV1-Slaves stammenden Fehlerinformationen siehe STATUS[3] Parameter STATUS (Seite 2402).

#### Parameter RET\_VAL für WR\_REC und RD\_REC

| Fehlercode*<br>(W#16#...) | Erläuterung   | Einschränkung                 |
|---------------------------|---|-------------------------------|
| 0000                      | kein Fehler   | -                             |
| 7000                      | Erstaufwurf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.   | -                             |
| 7001                      | Erstaufwurf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert 1.  | Dezentrale Peripherie         |
| 7002                      | Zwischenaufwurf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert 1.  | Dezentrale Peripherie         |
| 8090                      | Angegebene logische Basisadresse ungültig: Es ist keine Zuordnung im SDB1/SDB2x vorhanden, oder es ist keine Basisadresse.  | -                             |
| 8092                      | In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben.   | -                             |
| 8093                      | Für die über LADDR und IOID ausgewählte Baugruppe ist diese Anweisung nicht zulässig. (Zulässig sind S7-300-Baugruppen bei S7-300, S7-400-Baugruppen bei S7-400, S7-DP-Baugruppen bei S7-300 und S7-400.) | -                             |
| 80A0                      | Negative Quittung beim Lesen von Baugruppe: Baugruppe während des Lesevorgangs gezogen oder Baugruppe defekt  | nur bei "RD_REC"              |
| 80A1                      | Negative Quittung beim Schreiben zur Baugruppe: Baugruppe während des Schreibvorgangs gezogen oder Baugruppe defekt   | nur bei "WR_REC (Seite 2435)" |
| 80A2                      | <ul style="list-style-type: none"> <li>• DP-Protokollfehler bei Layer2 (z. B. Slaveausfall oder Busstörungen)</li> <li>• Bei ET200S ist im DPV0-Mode kein Datensatz-Lesen möglich.</li> </ul>             | Dezentrale Peripherie         |

9.7 Referenzen

| Fehlercode*<br>(W#16#...)   | Erläuterung  | Einschränkung   |
|---|--|---|
| 80A3  | DP-Protokollfehler bei User-Interface/User   | Dezentrale Peripherie   |
| 80A4  | Kommunikation am PBUS+ gestört   | -   |
| 80B0  | <ul style="list-style-type: none"> <li>Anweisung für Baugruppentyp nicht möglich</li> <li>Baugruppe kennt den Datensatz nicht.</li> <li>Datensatznummer 241 ist unzulässig.</li> <li>Bei "WR_REC (Seite 2435)" sind die Datensätze 0 und 1 nicht erlaubt.</li> </ul>   | -   |
| 80B1  | Die Längenangabe in Parameter RECORD ist falsch.   | <ul style="list-style-type: none"> <li>Bei "WR_REC (Seite 2435)": Länge falsch</li> <li>bei "RD_REC" (nur möglich bei Verwendung alter S7-300-FMs und S7-300-CPs):Angabe &gt; DS-Länge</li> <li>bei DPNRM_DG: Angabe &lt; DS-Länge</li> </ul> |
| 80B2  | Der projektierte Steckplatz ist nicht belegt.  | -   |
| 80B3  | Ist-Baugruppentyp ungleich Soll-Baugruppentyp im SDB1  | -   |
| 80B7  | DP-Slave oder Baugruppe meldet einen unzulässigen Bereich eines Parameters oder eines Wertes.  | nur bei "RD_REC"  |
| 80C0  | <p>Bei "WR_REC (Seite 2435)": Die Daten können nur im STOP-Zustand der CPU geschrieben werden. Hinweis: Damit ist ein Schreiben durch das Anwenderprogramm nicht möglich. Sie können die Daten nur online mit PG/PC schreiben.</p> <p>Bei "RD_REC": Die Baugruppe führt den Datensatz, aber es liegen entweder noch keine Daten vor oder die Daten können nur im STOP-Zustand der CPU gelesen werden. Hinweis: Falls Daten nur im STOP-Zustand der CPU gelesen werden können, ist eine Auswertung durch das Anwenderprogramm nicht möglich. In diesem Fall können Sie die Daten nur online mit PG/PC lesen.</p> <p>Bei "DPNRM_DG (Seite 2450)": Es liegen keine Diagnosedaten vor.</p> | Bei "WR_REC (Seite 2435)" oder "RD_REC" oder "DPNRM_DG (Seite 2450)"  |
| 80C1  | Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.   | -   |
| 80C2  | Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.  | -   |
| 80C3  | Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.   | -   |
| 80C4  | Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte Ihren Aufbau auf elektrische Störquellen.  | -   |
| 80C5  | Dezentrale Peripherie nicht verfügbar.   | Dezentrale Peripherie   |
| 80C6  | Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund)   | Dezentrale Peripherie   |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  | -   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |   |



## WR\_REC: Datensatz in Peripherie schreiben

### Beschreibung

Mit der Anweisung "WR\_REC" übertragen Sie den Datensatz RECORD zu der adressierten Baugruppe.

Sie starten den Schreibvorgang, indem Sie beim Aufruf den Eingangsparameter REQ mit "1" belegen. Falls der Schreibvorgang sofort ausgeführt werden konnte, liefert die Anweisung am Ausgangsparameter BUSY den Wert "0" zurück. Falls BUSY den Wert "1" hat, ist der Schreibvorgang noch nicht abgeschlossen.

### Hinweis

Wenn ein DPV1-Slave über GSD-Datei projektiert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Masters auf "S7-kompatibel" eingestellt ist, dürfen im Anwenderprogramm keine Datensätze mit "WR\_REC" zu den E/A-Baugruppen geschrieben werden. Der DP-Master adressiert in diesem Fall den falschen Steckplatz (projektiertes Steckplatz + 3).

Abhilfe: Schnittstelle des DP-Masters auf "DPV1" umstellen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR\_REC":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-----------|-------------|--------------|------------------------------|--|
| REQ       | Input       | BOOL         | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Schreiben   |
| LADDR     | Input       | HW_IO (WORD) | E, A, M, D, L oder Konstante | Hardware-Kennung der Baugruppe.  |
| RECNUM    | Input       | BYTE         | E, A, M, D, L oder Konstante | Datensatznummer  |
| RECORD    | Input       | ANY          | E, A, M, D, L                | Datensatz. Es ist nur der Datentyp BYTE zulässig.  |
| RET_VAL   | Return      | INT          | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. |
| BUSY      | Output      | BOOL         | E, A, M, D, L                | BUSY = 1: Der Schreibvorgang ist noch nicht beendet.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RECORD

Die zu übertragenden Daten werden aus dem Parameter RECORD beim ersten Aufruf gelesen. Falls die Übertragung des Datensatzes länger als einen Aufruf dauert, so ist bei den Folgeaufrufen der Anweisung (zum gleichen Auftrag) der Inhalt des Parameters RECORD nicht mehr relevant.

## Parameter RET\_VAL

Siehe auch: RD\_REC: Datensatz von Peripherie lesen (Seite 2430)

---

### Hinweis

Falls der allgemeine Fehler W#16#8544 auftritt, zeigt dies nur an, dass der Zugriff auf mindestens ein Byte des den Datensatz enthaltenden E/A-Speicherbereichs gesperrt war. Die Datenübertragung wurde fortgesetzt.

---

## DPRD\_DAT: Konsistente Daten eines DP-Normslaves lesen

### Beschreibung

Mit der Anweisung "DPRD\_DAT" lesen Sie konsistente Daten eines DP-Normslaves/PROFINET IO-Devices aus.

Sie benötigen "DPRD\_DAT", weil Sie mit den Ladebefehlen, die auf die Peripherie bzw. auf das Prozessabbild der Eingänge zugreifen, maximal vier Bytes zusammenhängend auslesen können. Sie können konsistente Daten ggf. auch über das Prozessabbild der Eingänge einlesen. Ob Ihre CPU diese Funktionalität beherrscht, entnehmen Sie der zugehörigen Dokumentation. Weitere Informationen zu konsistenten Daten eines DP-Normslaves/PROFINET IO-Devices finden Sie in dem Abschnitt "Datenkonsistenz (Seite 2913)".

Diese Anweisung "DPRD\_DAT" kann bei Bedarf auch für einen Datenbereich ab 1 Byte verwendet werden. Die Maximallänge der Daten entnehmen Sie der Dokumentation Ihrer CPU (z. B. 64 Byte bei einer S7-1214).

- Über den Parameter LADDR wählen Sie den DP-Normslave/PROFINET IO-Device aus. Bei einem Zugriffsfehler wird der Fehlercode W#16#8090 gemeldet.
- Über den Parameter RECORD definieren Sie den Zielbereich der gelesenen Daten:
  - Der Zielbereich muss mindestens genauso lang sein wie die Eingänge der selektierten Baugruppe. Es werden nur die Eingänge übertragen, die übrigen Bytes werden nicht berücksichtigt. Bei einem DP-Normslave mit modularem Aufbau bzw. mit mehreren DP-Kennungen können Sie mit einem "DPRD\_DAT" -Aufruf jeweils nur auf die Daten einer Baugruppe der projektierten Hardware-Kennung zugreifen. Wird der Zielbereich zu klein gewählt, wird am Parameter RET\_VAL der Fehlercode 80B1 ausgegeben.
  - Als Datentyp können alle Bitfolgen und alle Ganzzahlen verwendet werden. Eine Verwendung dieser Datentypen in einer Datenstruktur vom Typ ARRAY ist ebenfalls zulässig. Der Datentyp STRING wird nicht unterstützt.
- Falls bei der Datenübertragung kein Fehler auftrat, werden die gelesenen Daten in den am Parameter RECORD definierten Zielbereich eingetragen.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPRD\_DAT":

| Parameter | Deklaration | Datentyp               | Speicherbereich           | Beschreibung   |
|-----------|-------------|------------------------|---------------------------|--|
| LADDR     | Input       | HW_IO                  | E, A, M, L oder Konstante | Hardware-Kennung des DP-Normslaves/ PROFINET IO-Devices, aus dem gelesen werden soll.                  |
| RET_VAL   | Return      | DINT, INT, LREAL, REAL | E, A, M, D, L             | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. |
| RECORD    | Output      | VARIANT                | E, A, M, D, L             | Zielbereich für die gelesenen Nutzdaten.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode* (W#16#...)   | Erläuterung  |
|--|--|
| 0000   | Es ist kein Fehler aufgetreten.  |
| 8090   | <ul style="list-style-type: none"> <li>Für die angegebene HW-Kennung haben sie keine Baugruppe projiziert, oder</li> <li>Sie haben die Einschränkung über die Länge der konsistenten Daten nicht beachtet, oder</li> <li>Sie haben als Adresse am Parameter LADDR keine HW-Kennung angegeben.</li> </ul> |
| 8092   | Am Parameter RECORD wurde ein anderer Datentyp als (Array of) Bitfolge oder Ganzzahl angegeben.  |
| 8093   | Für die unter LADDR angegebene HW-Kennung existiert keine DP-Baugruppe / kein PROFINET IO-Device, von der/dem Sie konsistente Daten lesen können. Dieser Fehlercode tritt auch auf, wenn die über LADDR adressierte Baugruppe nicht über Eingänge verfügt.   |
| 80A0   | Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.  |
| 80B1   | Die Länge des angegebenen Zielbereichs am Parameter RECORD ist kürzer als die projizierte Nutzdatenlänge.  |
| 80C0   | Die Daten wurden noch nicht gelesen.   |
| allgemeine Fehlerinformation   | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch


Parameter STATUS (Seite 2402)

## DPWR\_DAT: Konsistente Daten eines DP-Normslaves schreiben

### Beschreibung

Mit der Anweisung "DPWR\_DAT" übertragen Sie die Daten am Parameter RECORD konsistent zum adressierten DP-Normslave/PROFINET IO-Device und ggf. ins Prozessabbild (nämlich dann, wenn Sie den betroffenen Adressbereich des DP-Normslaves als Konsistenzbereich in einem Prozessabbild projiziert haben).

Sie benötigen "DPWR\_DAT", weil Sie mit den Transferbefehlen, die auf die Peripherie bzw. auf das Prozessabbild der Ausgänge zugreifen, maximal vier Bytes zusammenhängend schreiben können. Sie können konsistente Daten ggf. auch über das Prozessabbild der Ausgänge schreiben. Ob Ihre CPU diese Funktionalität beherrscht, entnehmen Sie der zugehörigen Dokumentation. Nutzen Sie beim Schreiben konsistenter Daten nicht beide Möglichkeiten gleichzeitig: Nutzen Sie entweder "DPWR\_DAT" oder das Schreiben über das Prozessabbild der Ausgänge. Weitere Informationen zu konsistenten Daten eines DP-Normslaves/PROFINET IO-Devices finden Sie in dem Abschnitt "Datenkonsistenz (Seite 2913)". Bei einem DP-Normslave mit modularem Aufbau können Sie nur auf eine Baugruppe des DP-Slaves zugreifen.

|   |                 |
|---|-----------------|
|    | <b>VORSICHT</b> |
| <b>Peripheriezugriff</b>  |                 |
| Vermeiden Sie bei Verwendung von "DPWR_DAT" Zugriffe auf Peripheriebereiche, denen Teilprozessabbilder mit OB6x-Anbindung (Taktsynchronalarml) zugeordnet sind. |                 |

Die Anweisung "DPRD\_DAT" kann bei Bedarf auch für einen Datenbereich ab 1 Byte verwendet werden. Die Maximallänge der Daten entnehmen Sie der Dokumentation Ihrer CPU (z. B. 64 Byte bei einer S7-1214).

- Über den Parameter LADDR wählen Sie den DP-Normslave/PROFINET IO-Device aus. Bei einem Zugriffsfehler auf die adressierte Baugruppe wird der Fehlercode 8090 ausgegeben.
- Über den Parameter RECORD definieren Sie den Quellbereich der zu schreibenden Daten:
  - Der Quellbereich muss mindestens genauso lang sein wie die Ausgänge der selektierten Baugruppe. Es werden nur die Ausgänge übertragen, die übrigen Bytes werden nicht berücksichtigt. Ist der Quellbereich am Parameter RECORD länger als die Ausgänge der projizierten Baugruppe, werden nur die Daten bis zur maximalen Länge der Ausgänge übertragen. Ist der Quellbereich am Parameter RECORD kürzer als die Ausgänge der projizierten Baugruppe, wird der Fehlercode 80B1 ausgegeben.
  - Als Datentyp können alle Bitfolgen und alle Ganzzahlen verwendet werden. Eine Verwendung dieser Datentypen in einer Datenstruktur vom Typ ARRAY ist ebenfalls zulässig. Der Datentyp STRING wird nicht unterstützt.

Die Datenübertragung erfolgt synchron, d. h. nach Beendigung der Anweisung ist der Schreibvorgang abgeschlossen.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPWR\_DAT":

| Parameter | Deklaration | Datentyp | Speicherbereich           | Beschreibung   |
|-----------|-------------|----------|---------------------------|--|
| LADDR     | Input       | HW_IO    | E, A, M, L oder Konstante | Hardware-Kennung des DP-Normslaves/PROFINET IO-Devices auf dessen PAA-Bereich geschrieben werden soll. |
| RECORD    | Input       | VARIANT  | E, A, M, D, L             | Quellbereich für die zu schreibenden Nutzdaten.  |
| RET_VAL   | Return      | INT      | E, A, M, D, L             | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode* (W#16#...)         | Erläuterung   |
|--------------------------------|---|
| 0000                           | Es ist kein Fehler aufgetreten.   |
| 8090                           | <ul style="list-style-type: none"> <li>Für die angegebene HW-Kennung haben sie keine Baugruppe projektiert, oder</li> <li>Sie haben die Einschränkung über die Länge der konsistenten Daten nicht beachtet, oder</li> <li>Sie haben keine HW-Kennung am Parameter LADDR angegeben.</li> </ul> |
| 8092                           | Am Parameter RECORD wurde ein anderer Datentyp als (Array of) Bitfolge oder Ganzzahl angegeben.   |
| 8093                           | Für die unter LADDR angegebene HW-Kennung existiert keine DP-Baugruppe/kein PROFINET IO-Device, auf die/das Sie konsistente Daten schreiben können. Dieser Fehlercode tritt auch auf, wenn der über LADDR adressierte DP-Normslave/PROFINET IO-Device nicht über Ausgänge verfügt.            |
| 80A1                           | Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt  |
| 80B1                           | Die Länge des angegebenen Quellbereichs am Parameter RECORD ist kürzer als die Ausgänge des projektierten DP-Normslave/PROFINET IO-Device.  |
| 80C1                           | Die Daten des vorangegangenen Schreibauftrags sind von dem DP-Normslave/PROFINET IO-Device noch nicht bearbeitet.   |
| allgemeine Fehlerinformationen | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## Siehe auch

Parameter STATUS (Seite 2402)

## iDevice / iSlave

### RCVREC: Datensatz empfangen

#### Beschreibung

Ein I-Device kann vom übergeordneten Controller einen Datensatz empfangen. Der Empfang erfolgt im Anwendungsprogramm mit der Anweisung "RCVREC" (receive record).

Die Anweisung hat folgende Betriebsarten:

- Überprüfung, ob dem I-Device eine Anforderung für einen Datensatzempfang vorliegt.
- Bereitstellen des Datensatzes an den Ausgangsparametern.
- Verschicken einer Antwort an den übergeordneten Controller.

Welche dieser Betriebsarten die Anweisung ausführt, legen Sie über den Eingangsparameter MODE fest (siehe unten).

Das I-Device muss sich im Betriebszustand RUN oder im Betriebszustand ANLAUF befinden.

Mit MLEN geben Sie vor, wie viele Bytes Sie maximal empfangen möchten. Den Zielbereich RECORD sollten Sie daher mindestens MLEN Bytes lang wählen.

Wenn ein Datensatz empfangen wurde (MODE=1 oder MODE=2), wird im Ausgangsparameter NEW angezeigt, dass der Datensatz in RECORD gespeichert wurde. Achten Sie dabei auf eine ausreichende Länge von RECORD. Der Ausgangsparameter LEN enthält die tatsächliche Länge des empfangenen Datensatzes in Bytes.

Für die positive Antwort an den übergeordneten Controller beschreiben Sie CODE1 und CODE2 mit Null. Wenn der empfangene Datensatz abgelehnt werden soll, geben Sie in CODE1 den Error Code 1 und in CODE2 den Error Code 2 der negativen Antwort an den übergeordneten Controller an.

---

#### Hinweis

Wenn das I-Device eine Anforderung für einen Datensatzempfang erhalten hat, müssen Sie die Zustellung dieser Anforderung innerhalb einer bestimmten Zeitdauer erkannt haben. Nach dem Erkennen müssen Sie dem übergeordneten Controller innerhalb derselben Zeitdauer eine Antwort geschickt haben. Andernfalls tritt beim I-Device ein Timeout-Fehler auf, der zur Folge hat, dass das Betriebssystem des I-Device eine negative Antwort an den übergeordneten Controller schickt. Den Wert für die Zeitdauer entnehmen Sie den Technischen Daten Ihrer CPU.

---

Nach Auftreten eines Fehlers enthält der Ausgangsparameter STATUS die Fehlerinformation.

## Betriebsarten

Die Betriebsart der Anweisung "RCVREC" legen Sie über den Eingangsparameter MODE fest. Dieser wird in der folgenden Tabelle erläutert.

| MODE | Bedeutung   |
|------|---|
| 0    | Überprüfen, ob eine Anforderung für einen Datensatzempfang vorliegt<br>Falls auf dem I-Device ein Datensatz vom übergeordneten Controller vorliegt, beschreibt die Anweisung nur die Ausgangsparameter NEW, SLOT, SUBSLOT, INDEX und LEN. Wenn Sie die Anweisung mehrmals hintereinander mit MODE=0 aufrufen, beziehen sich die Ausgangsparameter stets auf ein und dieselbe Anforderung. |
| 1    | Empfangen eines Datensatzes für irgendeinen Subslot des I-Device<br>Falls auf dem I-Device ein Datensatz vom übergeordneten Controller für irgendeinen Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter und überträgt den Datensatz in den Parameter RECORD.   |
| 2    | Empfangen eines Datensatzes für einen bestimmten Subslot des I-Device<br>Falls auf dem I-Device ein Datensatz vom übergeordneten Controller für einen bestimmten Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter und überträgt den Datensatz in den Parameter RECORD.   |
| 3    | Positive Antwort an den übergeordneten Controller schicken<br>Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz zu empfangen, akzeptiert den vorliegenden Datensatz und schickt eine positive Quittung an den übergeordneten Controller.  |
| 4    | Negative Antwort an den übergeordneten Controller schicken<br>Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz zu empfangen, lehnt den vorliegenden Datensatz ab und schickt eine negative Quittung an den übergeordneten Controller. Den Grund für die Ablehnung geben Sie in den Eingangsparametern CODE1 und CODE2 an.                          |

### Hinweis

Nach dem Eintreffen eines Datensatzes (NEW=1) müssen Sie "RCVREC" noch zweimal aufrufen, um seine vollständige Abarbeitung zu gewährleisten. Dabei müssen Sie die folgende Reihenfolge einhalten:

- Erster Aufruf mit MODE=1 oder MODE=2
- Zweiter Aufruf mit MODE=3 oder MODE=4

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCVREC":

| Parameter | Deklaration | Datentyp                    | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------------------------|------------------------------|---|
| MODE      | Input       | INT                         | E, A, M, D, L oder Konstante | Betriebsart   |
| F_ID      | Input       | HW_SUBMO<br>DULE<br>(DWORD) | E, A, M, D, L oder Konstante | Subslot im Transferbereich des I-Device für den zu erhaltenden Datensatz (nur relevant für MODE=2). Das high word ist stets auf Null zu setzen. |
| MLEN      | Input       | INT                         | E, A, M, D, L oder Konstante | Maximale Länge des zu empfangenden Datensatzes in Bytes.  |

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung  |
|-----------|-------------|--------------|------------------------------|---|
| CODE1     | Input       | BYTE         | E, A, M, D, L oder Konstante | Null (bei MODE=3) bzw. Error Code 1 (bei MODE=4)  |
| CODE2     | Input       | BYTE         | E, A, M, D, L oder Konstante | Null (bei MODE=3) bzw. Error Code 2 (bei MODE=4)  |
| NEW       | Output      | BOOL         | E, A, M, D, L                | <ul style="list-style-type: none"> <li>• MODE=0: Neuer Datensatz wurde empfangen</li> <li>• MODE=1 oder 2: Datensatz wurde in RECORD übertragen</li> </ul>  |
| STATUS    | Output      | DWORD        | E, A, M, D, L                | Fehlerinformation   |
| SLOT      | Output      | HW_SUBMODULE | E, A, M, D, L                | identisch zu F_ID   |
| SUBSLOT   | Output      | HW_SUBMODULE | E, A, M, D, L                | identisch zu F_ID   |
| INDEX     | Output      | UINT         | E, A, M, D, L                | Nummer des empfangenen Datensatzes  |
| LEN       | Output      | UINT         | E, A, M, D, L                | Länge des empfangenen Datensatzes   |
| RECORD    | InOut       | VARIANT      | E, A, M, D, L                | Zielbereich für den empfangenen Datensatz. Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPU's immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPU's unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Abschnitt: Parameter STATUS (Seite 2402)

### PRVREC: Datensatz bereitstellen

#### Beschreibung

Ein I-Device kann vom übergeordneten Controller eine Anforderung erhalten, einen Datensatz bereitzustellen. Die Bereitstellung erfolgt im Anwenderprogramm mit der Anweisung "PRVREC" (provide record).

Die Anweisung hat folgende Betriebsarten:

- Überprüfung, ob dem I-Device eine Anforderung für eine Datensatzbereitstellung vorliegt.
- Übertragung des angeforderten Datensatz an den übergeordneten Controller.
- Schicken einer Antwort an den übergeordneten Controller.

Welche dieser Betriebsarten die Anweisung ausführt, legen Sie über den Eingangsparameter MODE fest (siehe unten).



Das I-Device muss sich im Betriebszustand RUN oder im Betriebszustand ANLAUF befinden.

Mit LEN geben Sie vor, wie viele Bytes der zu versendende Datensatz maximal haben soll. Den Zielbereich RECORD sollten Sie daher mindestens LEN Bytes lang wählen.

Wenn eine Anforderung für eine Datensatzbereitstellung vorliegt (MODE=0), wird der Ausgangsparameter NEW auf TRUE gesetzt.

Wenn die Anforderung für eine Datensatzbereitstellung akzeptiert wird, beschreiben Sie für die positive Antwort an den übergeordneten Controller RECORD mit dem angeforderten Datensatz und CODE1 und CODE2 mit Null. Wenn die Anforderung für eine Datensatzbereitstellung abgelehnt werden soll, geben Sie in CODE1 den Error Code 1 und in CODE2 den Error Code 2 der negativen Antwort an den übergeordneten Controller an.

---

### Hinweis

Wenn das I-Device eine Anforderung für eine Datensatzbereitstellung erhalten hat, müssen Sie die Zustellung dieser Anforderung innerhalb einer bestimmten Zeitdauer erkannt haben. Nach dem Erkennen müssen Sie dem übergeordneten Controller innerhalb derselben Zeitdauer eine Antwort geschickt haben. Andernfalls tritt beim I-Device ein Timeout-Fehler auf, der zur Folge hat, dass das Betriebssystem des I-Device eine negative Antwort an den übergeordneten Controller schickt. Den Wert für die Zeitdauer entnehmen Sie den Technischen Daten Ihrer CPU.

---

Nach Auftreten eines Fehlers enthält der Ausgangsparameter STATUS die Fehlerinformation.

## Betriebsarten

Die Betriebsart der Anweisung "PRVREC" legen Sie über den Eingangsparameter MODE fest. Dieser wird in der folgenden Tabelle erläutert.

| MODE | Bedeutung  |
|------|--|
| 0    | Überprüfen, ob eine Anforderung für eine Datensatzbereitstellung vorliegt<br>Falls auf dem I-Device eine Anforderung vom übergeordneten Controller für eine Datensatzbereitstellung vorliegt, beschreibt die Anweisung nur die Ausgangsparameter NEW, SLOT, SUBSLOT, INDEX und RLEN. Wenn Sie die Anweisung mehrmals hintereinander mit MODE=0 aufrufen, beziehen sich die Ausgangsparameter stets auf ein und dieselbe Anforderung. |
| 1    | Empfangen einer Anforderung für eine Datensatzbereitstellung für irgendeinen Subslot des I-Device<br>Falls auf dem I-Device eine solche Anforderung vom übergeordneten Controller für irgendeinen Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter.   |
| 2    | Empfangen einer Anforderung für eine Datensatzbereitstellung für einen bestimmten Subslot des I-Device<br>Falls auf dem I-Device eine solche Anforderung vom übergeordneten Controller für einen bestimmten Subslot des I-Device vorliegt, beschreibt die Anweisung die Ausgangsparameter.   |
| 3    | Datensatz bereitstellen und positive Antwort an den übergeordneten Controller schicken<br>Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz bereitzustellen, stellt den angeforderten Datensatz in RECORD zur Verfügung und schickt eine positive Quittung an den übergeordneten Controller.   |
| 4    | Negative Antwort an den übergeordneten Controller schicken<br>Die Anweisung überprüft die Anforderung des übergeordneten Controllers, einen Datensatz bereitzustellen, lehnt diese Anforderung ab und schickt eine negative Quittung an den übergeordneten Controller. Den Grund für die Ablehnung geben Sie in den Eingangsparametern CODE1 und CODE2 an.   |

**Hinweis**

Nach dem Eintreffen einer Anforderung (NEW=1) müssen Sie die Anweisung noch zweimal aufrufen, um seine vollständige Abarbeitung zu gewährleisten. Dabei müssen Sie die folgende Reihenfolge einhalten:

- Erster Aufruf mit MODE=1 oder MODE=2
- Zweiter Aufruf mit MODE=3 oder MODE=4

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "PRVREC":

| Parameter | Deklaration | Datentyp             | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------------------|------------------------------|---|
| MODE      | Input       | INT                  | E, A, M, D, L oder Konstante | Betriebsart   |
| F_ID      | Input       | HW_SUBMODULE (DWORD) | E, A, M, D, L oder Konstante | Subslot im Transferbereich des I-Device für den zu versendenden Datensatz (nur relevant für MODE=2). Das high word ist stets auf Null zu setzen.  |
| CODE1     | Input       | BYTE                 | E, A, M, D, L oder Konstante | Null (bei MODE=3) bzw. Error Code 1 (bei MODE=4)  |
| CODE2     | Input       | BYTE                 | E, A, M, D, L oder Konstante | Null (bei MODE=3) bzw. Error Code 2 (bei MODE=4)  |
| LEN       | Input       | UINT                 | E, A, M, D, L oder Konstante | Maximale Länge des zu versendenden Datensatzes in Bytes   |
| NEW       | Output      | BOOL                 | E, A, M, D, L                | Neuer Datensatz wurde vom übergeordneten Controller angefordert.  |
| STATUS    | Output      | DWORD                | E, A, M, D, L                | Fehlerinformation   |
| SLOT      | Output      | HW_SUBMODULE         | E, A, M, D, L                | identisch zu F_ID   |
| SUBSLOT   | Output      | HW_SUBMODULE         | E, A, M, D, L                | identisch zu F_ID   |
| INDEX     | Output      | UINT                 | E, A, M, D, L                | Nummer des zu versendenden Datensatzes  |
| RLEN      | Output      | UINT                 | E, A, M, D, L                | Länge des zu versendenden Datensatzes   |
| RECORD    | InOut       | VARIANT              | E, A, M, D, L                | Bereitgestellter Datensatz<br>Hinweis: Beachten Sie, dass der Parameter RECORD bei S7-300-CPUs immer die vollständige Angabe der DB-Parameter erfordert (Bsp.: P#DB13.DBX0.0 Byte 100). Das Weglassen einer expliziten DB-Nr. ist für S7-300-CPU's unzulässig und führt zu einer Fehlermeldung im Anwenderprogramm. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

Zur Interpretation des Parameters STATUS siehe Abschnitt: Parameter STATUS (Seite 2402)

## PROFIBUS

### DPSYC\_FR: DP-Slaves synchronisieren / Eingänge einfrieren

#### Beschreibung

Mit der Anweisung können Sie eine oder mehrere Gruppen von DP-Slaves synchronisieren.

Dazu schicken Sie eines der folgenden Steuerkommandos oder eine Kombination davon an die betroffenen Gruppen:

- SYNC (gleichzeitiges Ausgeben und Einfrieren von Ausgangszuständen bei den DP-Slaves)
- UNSYNC (hebt das Steuerkommando SYNC wieder auf)
- FREEZE (Einfrieren von Eingangszuständen bei den DP-Slaves und Einlesen der eingefrorenen Eingänge)
- UNFREEZE (hebt das Steuerkommando FREEZE wieder auf)

Bevor Sie die oben genannten Steuerkommandos abschicken, müssen Sie die DP-Slaves per Projektierung in Gruppen eingeteilt haben. Sie müssen wissen, welcher DP-Slave welcher Gruppe mit welcher Nummer zugeordnet ist und welche Eigenschaften hinsichtlich SYNC-/FREEZE-Verhalten die einzelnen Gruppen haben.

---

#### Hinweis

Beachten Sie, dass die Steuerkommandos SYNC und FREEZE auch bei einem Neustart/Kaltstart ihre Gültigkeit behalten.

Beachten Sie außerdem, dass zu einem Zeitpunkt nur ein SYNC-/UNSYNC-Auftrag bzw. nur ein FREEZE/UNFREEZE-Auftrag angestoßen sein darf.

---

#### Arbeitsweise

"DPSYC\_FR" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "DPSYC\_FR" mit REQ=1 aufrufen.

Über die Ausgangsparameter RET\_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

#### Identifikation eines Auftrags

Falls Sie einen SYNC-/FREEZE-Auftrag angestoßen haben und Sie "DPSYC\_FR" erneut aufrufen, bevor dieser beendet wurde, dann hängt das weitere Verhalten der Anweisung

entscheidend davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt. Stimmen die Eingangsparameter LADDR, GROUP und MODE überein, so gilt der Aufruf als Folgeaufruf.

### Ausgänge von DP-Baugruppen beschreiben

Das Beschreiben der Ausgänge von DP-Baugruppen wird angestoßen durch

- Transfer-Befehle auf DP-Peripherie,
- das Schreiben des Prozessabbilds der Ausgänge in die Baugruppen (durch das Betriebssystem am Ende des OB 1 oder durch Aufruf der Anweisung "UPDAT\_PO (Seite 2387)"),
- Aufruf der Anweisung "DPWR\_DAT (Seite 2438)".

Im Normalfall überträgt der DP-Master die Ausgangsdaten zyklisch (im Zyklus des Busses PROFIBUS DP) an die Ausgänge der DP-Slaves.

Wenn Sie bestimmte Ausgangsdaten, die über mehrere Slaves verteilt sein können, exakt gleichzeitig auf die Ausgänge an den Prozess ausgeben wollen, schicken Sie mit Hilfe der Anweisung "DPSYC\_FR" das Steuerkommando SYNC an den zugehörigen DP-Master.

### Was bewirkt SYNC?

Mit dem Steuerkommando SYNC werden die DP-Slaves der genannten Gruppen in den Sync-Modus geschaltet, d. h. der DP-Master überträgt die aktuellen Ausgangsdaten und veranlasst die betroffenen DP-Slaves, die Ausgänge einzufrieren. Bei den folgenden Ausgabetelegrammen speichern die DP-Slaves die Ausgangsdaten in einem internen Puffer; der Zustand der Ausgänge bleibt unverändert.

Nach jedem Steuerkommando SYNC legen die DP-Slaves der selektierten Gruppen die Ausgangsdaten ihres internen Puffers auf die Ausgänge an den Prozess.

Die Ausgänge werden erst dann wieder zyklisch aktualisiert, wenn Sie mit Hilfe der Anweisung "DPSYC\_FR" das Steuerkommando UNSYNC absetzen.

---

#### Hinweis

Falls sich DP-Slaves der ausgewählten Gruppe(n) zum Zeitpunkt des abgeschickten Steuerkommandos nicht am Netz befinden oder ausgefallen sind, werden diese auch nicht in den SYNC-Modus geschaltet. Es erfolgt keine Mitteilung dieses Sachverhalts über den Rückgabewert der Anweisung.

---

### Eingänge von DP-Baugruppen lesen

Die Eingangsdaten von DP-Baugruppen werden gelesen

- mit Ladebefehlen auf DP-Peripherie,
- bei der Aktualisierung des Prozessabbilds der Eingänge (durch das Betriebssystem am Anfang des OB 1 oder durch Aufruf der Anweisung "UPDAT\_PI (Seite 2385)"),
- durch Aufruf der Anweisung "DPRD\_DAT (Seite 2436)".

Im Normalfall erhält der DP-Master diese Eingangsdaten zyklisch (im Zyklus des Busses PROFIBUS DP) von seinen DP-Slaves und stellt sie der CPU zur Verfügung.

Wenn Sie bestimmte Eingangsdaten, die über mehrere Slaves verteilt sein können, exakt gleichzeitig vom Prozess einlesen wollen, schicken Sie mit Hilfe der Anweisung "DPSYC\_FR" das Steuerkommando FREEZE an den zugehörigen DP-Master.

### Was bewirkt FREEZE?

Mit dem Steuerkommando FREEZE werden die betroffenen DP-Slaves in den Freeze-Modus geschaltet, d. h. der DP-Master veranlasst die betroffenen DP-Slaves, den aktuellen Zustand der Eingänge einzufrieren. Anschließend überträgt er die eingefrorenen Daten in den Eingangsbereich der CPU.

Nach jedem Steuerkommando FREEZE frieren die DP-Slaves den Zustand der Eingänge erneut ein.

Der DP-Master erhält erst dann wieder zyklisch den aktuellen Zustand der Eingänge, wenn Sie mit Hilfe der Anweisung "DPSYC\_FR" das Steuerkommando UNFREEZE absetzen.

---

#### Hinweis

Falls sich DP-Slaves der ausgewählten Gruppe(n) zum Zeitpunkt des abgeschickten Steuerkommandos nicht am Netz befinden oder ausgefallen sind, werden diese auch nicht in den Freeze-Modus geschaltet. Es erfolgt keine Mitteilung dieses Sachverhalts über den Rückgabewert der Anweisung.

---

### Datenkonsistenz

Aufgrund der asynchronen Arbeitsweise und der Unterbrechbarkeit durch höhere Prioritätsklassen sollten Sie bei Einsatz der Anweisung "DPSYC\_FR" darauf achten, dass die Prozessabbilder konsistent zu den tatsächlichen Ein- und Ausgängen der Peripherie sind.

Dies ist gewährleistet, wenn Sie eine der im Folgenden genannten Konsistenzregeln beachten:

- Definieren Sie für die "SYNC-Ausgänge" und die "FREEZE-Eingänge" geeignete Teilprozessabbilder (nur möglich bei S7-400). Rufen Sie die Anweisung "UPDAT\_PO (Seite 2387)" unmittelbar vor dem jeweiligen Erstaufruf eines SYNC-Auftrags auf. Rufen Sie die Anweisung "UPDAT\_PI (Seite 2385)" unmittelbar nach dem jeweiligen Letztaufruf eines FREEZE-Auftrags auf.
- Alternativ dazu: Verwenden Sie für Ausgänge, die von einem SYNC-Auftrag betroffen sind, und für Eingänge, die von einem FREEZE-Auftrag betroffen sind, nur direkte Peripheriezugriffe. Sie dürfen bei einem laufenden SYNC-Auftrag die betroffenen Ausgänge nicht beschreiben und bei einem laufenden FREEZE-Auftrag die betroffenen Eingänge nicht einlesen.

### Einsatz von DPWR\_DAT und DPRD\_DAT

Falls Sie die Anweisung "DPWR\_DAT (Seite 2438)" einsetzen, muss diese Anweisung abgeschlossen sein, bevor Sie für die zugehörigen Ausgänge einen SYNC-Auftrag abschicken.

Falls Sie die Anweisung "DPRD\_DAT (Seite 2436)" einsetzen, muss diese Anweisung abgeschlossen sein, bevor Sie für die zugehörigen Eingänge einen FREEZE-Auftrag abschicken.

### Anlauf und "DPSYC\_FR"

Das Absetzen der Steuerkommandos SYNC und FREEZE in den Anlauf-OBs liegt in alleiniger Verantwortung des Anwenders.

Falls die Ausgänge einer oder mehrerer Gruppen bereits beim Start des Anwenderprogramms im SYNC-Modus arbeiten sollen, müssen Sie im Anlauf diese Ausgänge initialisieren und die Anweisung "DPSYC\_FR" mit dem Steuerkommando SYNC vollständig abarbeiten.

Falls die Eingänge einer oder mehrerer Gruppen bereits beim Start des Anwenderprogramms im FREEZE-Modus arbeiten sollen, müssen Sie im Anlauf für diese Eingänge die Anweisung "DPSYC\_FR" mit dem Steuerkommando FREEZE vollständig abarbeiten.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPSYC\_FR":

| Parameter | Deklaration | Datentyp           | Speicherbereich              | Beschreibung  |
|-----------|-------------|--------------------|------------------------------|---|
| REQ       | Input       | BOOL               | E, A, M, D, L                | Pegelgetriggelter Steuerparameter REQ=1: Anstoß des SYNC-/FREEZE-Auftrags   |
| LADDR     | Input       | HW_DPMASTER (WORD) | E, A, M, D, L oder Konstante | Hardware-Kennung der DP-Master-Schnittstelle<br>Die Nummer kann in der Netzsicht aus den Eigenschaften der DP-Master-Schnittstelle oder dem Register "Systemkonstanten" der Standard-Variablen-tabelle übernommen werden. |
| GROUP     | Input       | BYTE               | E, A, M, D, L oder Konstante | Gruppenauswahl<br>Bit 0 = 1: Gruppe 1 ausgewählt<br>Bit 1 = 1: Gruppe 2 ausgewählt<br>:<br>Bit 7 = 1: Gruppe 8 ausgewählt<br>Pro Auftrag können Sie mehrere Gruppen auswählen. Der Wert B#16#0 ist nicht zugelassen.      |

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| MODE      | Input       | BYTE     | E, A, M, D, L oder Konstante | <p>Auftragskennung (Codierung gemäß EN 50 170 Volume 2, PROFIBUS)</p> <p>Bit 0: reserviert (Wert 0)</p> <p>Bit 1: reserviert (Wert 0)</p> <p>Bit 2:</p> <ul style="list-style-type: none"> <li>• = 1: UNFREEZE wird ausgeführt</li> <li>• = 0: keine Bedeutung</li> </ul> <p>Bit 3:</p> <ul style="list-style-type: none"> <li>• = 1: FREEZE wird ausgeführt</li> <li>• = 0: keine Bedeutung</li> </ul> <p>Bit 4:</p> <ul style="list-style-type: none"> <li>• = 1: UNSYNC wird ausgeführt</li> <li>• = 0: keine Bedeutung</li> </ul> <p>Bit 5:</p> <ul style="list-style-type: none"> <li>• = 1: SYNC wird ausgeführt</li> <li>• = 0: keine Bedeutung</li> </ul> <p>Bit 6: reserviert (Wert 0)</p> <p>Bit 7: reserviert (Wert 0)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <li>• bei genau einer Kennung pro Auftrag: <ul style="list-style-type: none"> <li>– B#16#04 (UNFREEZE)</li> <li>– B#16#08 (FREEZE)</li> <li>– B#16#10 (UNSYNC)</li> <li>– B#16#20 (SYNC)</li> </ul> </li> <li>• bei mehreren Kennungen pro Auftrag: <ul style="list-style-type: none"> <li>– B#16#14 (UNSYNC, UNFREEZE)</li> <li>– B#16#18 (UNSYNC, FREEZE)</li> <li>– B#16#24 (SYNC, UNFREEZE)</li> <li>– B#16#28 (SYNC, FREEZE)</li> </ul> </li> </ul> |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | <p>Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.</p> <p>Sie müssen RET_VAL nach jedem Bausteindurchlauf auswerten.</p>  |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | <p>BUSY=1:</p> <p>Der SYNC-/FREEZE-Auftrag ist noch nicht beendet.</p>   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

**Hinweis**

Falls Sie auf DPV1-Slaves zugreifen, können Fehlerinformationen dieser Slaves vom DP-Master an die Anweisung weitergereicht werden. Zur Beschreibung dieser Fehlerinformationen siehe STATUS[3], Parameter STATUS (Seite 2402).

| Fehlercode*<br>(W#16#...)  | Erläuterung   |
|--|---|
| 0000   | Der Auftrag wurde fehlerfrei durchgeführt.  |
| 7000   | Erstaufruf mit REQ=0. Der über LADDR, GROUP und MODE festgelegte Auftrag ist nicht aktiv; BUSY hat den Wert 0.  |
| 7001   | Erstaufruf mit REQ=1. Der über LADDR, GROUP und MODE festgelegte Auftrag wurde angestoßen; BUSY hat den Wert 1.   |
| 7002   | Zwischenaufruf (REQ irrelevant). Der aktivierte SYNC-/FREEZE-Auftrag läuft noch; BUSY hat den Wert 1.   |
| 8090   | Die über LADDR ausgewählte Baugruppe ist kein DP-Master.  |
| 8093   | Für die über LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig (Projektierung oder Version des DP-Masters).  |
| 8094   | Parameter GROUP falsch  |
| 8095   | Parameter MODE falsch   |
| 80B0   | Die über GROUP ausgewählte Gruppe ist nicht projektiert.  |
| 80B1   | Die über GROUP ausgewählte Gruppe ist dieser CPU nicht zugeordnet.  |
| 80B2   | Der über MODE spezifizierte SYNC-Auftrag ist auf der über GROUP ausgewählten Gruppe nicht zulässig.   |
| 80B3   | Der über MODE spezifizierte FREEZE-Auftrag ist auf der über GROUP ausgewählten Gruppe nicht zulässig.   |
| 80C2   | Temporärer Ressourcenmangel des DP-Masters: Der DP-Master bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.   |
| 80C3   | Dieser SYNC-/UNSYNC-Auftrag ist momentan nicht aktivierbar, da zu einem Zeitpunkt nur ein SYNC-/UNSYNC-Auftrag angestoßen sein darf. Bitte prüfen Sie Ihr Anwenderprogramm.         |
| 80C4   | Dieser FREEZE-/UNFREEZE-Auftrag ist momentan nicht aktivierbar, da zu einem Zeitpunkt nur ein FREEZE-/UNFREEZE-Auftrag angestoßen sein darf. Bitte prüfen Sie Ihr Anwenderprogramm. |
| 80C5   | Kurzschluss unmittelbar an der DP-Schnittstelle   |
| 80C6   | Auftragsabbruch wegen Peripherieabwurf durch die CPU  |
| 80C7   | Auftragsabbruch wegen Neustart (Warmstart) oder Kaltstart des DP-Masters  |
| allgemeine Fehlerinformation   | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**DPNRM\_DG: Diagnosedaten eines DP-Slaves lesen**

**Beschreibung**

Mit der Anweisung "DPNRM\_DG" lesen Sie die aktuellen Diagnosedaten eines DP-Slaves in der Form, wie sie durch EN 50 170 Volume 2, PROFIBUS festgelegt sind.



Den prinzipiellen Aufbau der Slave-Diagnose entnehmen Sie folgender Tabelle, weitergehende Informationen den Handbüchern zu den DP-Slaves.

| Byte  | Bedeutung                         |
|-------|-----------------------------------|
| 0     | Stationsstatus 1                  |
| 1     | Stationsstatus 2                  |
| 2     | Stationsstatus 3                  |
| 3     | Master-Stationsnummer             |
| 4     | Herstellerkennung (high byte)     |
| 5     | Herstellerkennung (low byte)      |
| 6 ... | Weitere slavespezifische Diagnose |

Die gelesenen Daten werden nach fehlerfreier Datenübertragung in den durch RECORD aufgespannten Zielbereich eingetragen. Sie starten den Lesevorgang, indem Sie beim Aufruf der Anweisung "DPNRM\_DG" den Eingangsparameter REQ mit "1" belegen.

### Arbeitsweise

Der Lesevorgang wird asynchron ausgeführt, d. h. er kann sich über mehrere Aufrufe erstrecken. Über die Ausgangsparameter RET\_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DPNRM\_DG":

| Parameter | Deklaration | Datentyp               | Speicherbereich              | Beschreibung   |
|-----------|-------------|------------------------|------------------------------|--|
| REQ       | Input       | BOOL                   | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Lesen   |
| LADDR     | Input       | HW_DPSLAVE (WORD)      | D, L oder Konstante          | Projektierte Diagnoseadresse des DP-Slaves<br>Hinweis: Adresse muss hexadezimal angegeben werden. Z. B. Diagnoseadresse 1022 bedeutet: LADDR:=W#16#3FE.  |
| RET_VAL   | Return      | DINT, INT, LREAL, REAL | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. Falls kein Fehler auftrat, steht in RET_VAL die Länge der tatsächlich übertragenen Daten. |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| RECORD    | Output      | VARIANT  | E, A, M, D, L   | Zielbereich für die gelesenen Diagnosedaten. Es ist nur der Datentyp BYTE zulässig. Die Mindestlänge des zu lesenden Datensatzes bzw. des Zielbereichs beträgt 6. Die Maximallänge des zu lesenden Datensatzes beträgt 240; bei Normslaves, bei denen die Anzahl der Normdiagnosedaten größer als 240 Bytes ist und maximal 244 Bytes beträgt, werden die ersten 240 Bytes in den Zielbereich übertragen und das entsprechende Overflow-Bit in den Daten gesetzt. |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | BUSY = 1:<br>Der Lesevorgang ist noch nicht beendet.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

Informationen zur Datentypkonvertierung in den einzelnen Programmiersprachen finden Sie unter "Auto-Hotspot".

### Parameter RECORD

Die CPU bewertet die tatsächliche Länge der gelesenen Diagnosedaten:

Falls die Längenangabe von RECORD

- kleiner als die Anzahl der gelieferten Daten ist, werden die Daten verworfen, und in RET\_VAL wird die zugehörige Fehlerinformation eingetragen.
- größer oder gleich der Anzahl der gelieferten Daten ist, werden die Daten in den Zielbereich übernommen, und in RET\_VAL wird die tatsächliche Länge als positiver Wert eingetragen.

---

#### Hinweis

Sie müssen darauf achten, dass die Aktualparameter von RECORD bei allen Aufrufen, die zu einem Auftrag gehören, übereinstimmen.

Ein Auftrag ist eindeutig festgelegt durch den Eingangsparameter LADDR.

---

### Normslaves mit mehr als 240 Bytes Diagnosedaten

Bei Normslaves, bei denen die Anzahl der Normdiagnosedaten zwischen 241 und 244 Bytes liegt, müssen Sie folgendes beachten:

Falls die Längenangabe von RECORD

- kleiner als 240 Bytes ist, werden die Daten verworfen, und in RET\_VAL wird die zugehörige Fehlerinformation eingetragen.
- größer oder gleich 240 Bytes ist, werden die ersten 240 Bytes der Normdiagnosedaten in den Zielbereich übertragen und das entsprechende Overflow-Bit in den Daten gesetzt.

## Parameter RET\_VAL

- Trat während der Bearbeitung der Funktion ein Fehler auf, enthält der Rückgabewert einen Fehlercode,
- Falls bei der Übertragung kein Fehler auftrat, enthält RET\_VAL die Länge der gelesenen Daten in Bytes als positive Zahl.

### Hinweis

Die Anzahl der gelesenen Daten ist bei einem DP-Slave von seinem Diagnosezustand abhängig.

Zur Auswertung der Fehlerinformationen des Parameters RET\_VAL siehe folgende Tabelle.

| Fehlercode (W#16#...) | Erläuterung  | Einschränkung         |
|-----------------------|--|-----------------------|
| 7000                  | Erstaufruf mit REQ = 0: keine Datenübertragung aktiv; BUSY hat den Wert "0".   | -                     |
| 7001                  | Erstaufruf mit REQ = 1: Datenübertragung angestoßen; BUSY hat den Wert "1".  | Dezentrale Peripherie |
| 7002                  | Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".  | Dezentrale Peripherie |
| 8090                  | Angegebene logische Basisadresse ungültig: Es ist keine Zuordnung im SDB1/SDB2x vorhanden, oder es ist keine Basisadresse.   | -                     |
| 8093                  | Für die über LADDR und IOID ausgewählte Baugruppe ist diese Anweisung nicht zulässig.  | -                     |
| 80A2                  | <ul style="list-style-type: none"> <li>• DP-Protokollfehler bei Layer2 (z. B. Slaveausfall oder Busstörungen)</li> <li>• Bei ET200S ist im DPV0-Mode kein Datensatz-Lesen möglich.</li> </ul>  | Dezentrale Peripherie |
| 80A3                  | DP-Protokollfehler bei User-Interface/User   | Dezentrale Peripherie |
| 80A4                  | Kommunikation am PBUS+ gestört   | Dezentrale Peripherie |
| 80B0                  | <ul style="list-style-type: none"> <li>• Anweisung für Baugruppentyp nicht möglich</li> <li>• Baugruppe kennt den Datensatz nicht.</li> <li>• Datensatznummer 241 ist unzulässig.</li> <li>• Bei "WR_REC (Seite 2394)" sind die Datensätze 0 und 1 nicht erlaubt.</li> </ul> | -                     |
| 80B1                  | Die Längenangabe in Parameter RECORD ist falsch.   | Angabe < DS-Länge     |
| 80B2                  | Der projektierte Steckplatz ist nicht belegt.  | -                     |
| 80B3                  | Ist-Baugruppentyp ungleich Soll-Baugruppentyp im SDB1  | -                     |
| 80C0                  | Es liegen keine Diagnosedaten vor.   | -                     |
| 80C1                  | Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.   | -                     |
| 80C2                  | Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.  | -                     |
| 80C3                  | Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.   | -                     |
| 80C4                  | Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte Ihren Aufbau auf elektrische Störquellen.  | -                     |
| 80C5                  | Dezentrale Peripherie nicht verfügbar.   | Dezentrale Peripherie |

| Fehlercode (W#16#...)   | Erläuterung  | Einschränkung         |
|---|--|-----------------------|
| 80C6  | Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund) | Dezentrale Peripherie |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  | -                     |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |                       |

## DP\_TOPOL: Topologie für DP-Mastersystem ermitteln

### Beschreibung

Mit der Anweisung stoßen Sie die Topologieermittlung für ein ausgewähltes DP-Mastersystem an. Mit dem Aufruf von der Anweisung werden alle Diagnose-Repeater an einem DP-Mastersystem angesprochen.

---

#### Hinweis

Die Topologieermittlung kann zu einem Zeitpunkt nur für ein DP-Mastersystem durchgeführt werden.

---

Die Topologieermittlung ist die Voraussetzung für die detaillierte Anzeige des Fehlerorts bei auftretenden Leitungsfehlern. Führen Sie nach dem Aufbau und nach jeder Änderung des physikalischen Aufbaus eines DP-Mastersystems die Topologieermittlung mit der Anweisung "DP\_TOPOL" erneut durch.

Änderungen des physikalischen Aufbaus sind:

- Änderung von Leitungslängen
- Hinzufügen oder Entfernen von Teilnehmern oder Komponenten mit Repeaterfunktion
- Ändern von Teilnehmeradressen

Wird von einem Diagnose-Repeater ein Fehler gemeldet, beschreibt "DP\_TOPOL" die Ausgänge DPR und DPRI für die Dauer eines "DP\_TOPOL"-Durchlaufs. Werden von mehreren Diagnose-Repeatern des ausgewählten DP-Mastersystems Fehler gemeldet, schreibt "DP\_TOPOL" in DPR und DPRI Informationen zum ersten fehlermeldenden Diagnose-Repeater. Die vollständige Diagnoseinformation können Sie mit der Anweisung "DPNRM\_DG (Seite 2450)" oder am PG auslesen. Falls kein Diagnose-Repeater einen Fehler meldet, haben die Ausgänge DPR und DPRI den Wert NULL.

Wenn Sie nach dem Auftreten eines Fehlers erneut eine Topologieermittlung anstoßen wollen, müssen Sie "DP\_TOPOL" zunächst zurücksetzen. Dies geschieht, indem Sie "DP\_TOPOL" mit REQ=0 und R=1 aufrufen.

### Arbeitsweise

"DP\_TOPOL" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Ermittlung der Bustopologie, indem Sie "DP\_TOPOL"

mit REQ=1 aufrufen. Falls Sie den Vorgang abbrechen möchten, rufen Sie "DP\_TOPOL" mit R=1 auf.

Über die Ausgangsparameter RET\_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

---

#### Hinweis

Die Topologieermittlung kann mehrere Minuten dauern.

---

### Identifikation eines Auftrags

Der Eingangsparameter DP\_ID legt einen Auftrag eindeutig fest.

Falls Sie "DP\_TOPOL" aufgerufen haben und Sie diese Anweisung erneut aufrufen, bevor die Topologieermittlung abgeschlossen ist, hängt das weitere Verhalten der Anweisung davon ab, ob es sich beim erneuten Aufruf um denselben Auftrag handelt: Stimmt der Parameter DP\_ID mit einem noch nicht abgeschlossenen Auftrag überein, so wird der Aufruf als Folgeaufruf interpretiert, und in RET\_VAL wird der Wert W#16#7002 eingetragen. Handelt es sich hingegen um einen weiteren Auftrag, weist die CPU diesen ab.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DP\_TOPOL":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------|------------------------------|--|
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | REQ=1: Topologieermittlung anstoßen  |
| R         | Input       | BOOL            | E, A, M, D, L oder Konstante | R=1: Abbruch der Topologieermittlung   |
| DP_ID     | Input       | HW_IOSYST<br>EM | E, A, M, D, L oder Konstante | DP-Mastersystem-ID desjenigen DP-Mastersystems, dessen Topologie ermittelt werden soll                 |
| RET_VAL   | Return      | INT             | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. |
| BUSY      | Output      | BOOL            | E, A, M, D, L                | BUSY=1: Topologieermittlung ist noch nicht abgeschlossen.  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| DPR       | Output      | BYTE     | E, A, M, D, L   | PROFIBUS-Adresse des Fehler meldenden Diagnose-Repeater  |
| DPRI      | Output      | BYTE     | E, A, M, D, L   | Messegment des Fehler meldenden Diagnose-Repeater:<br><ul style="list-style-type: none"> <li>• Bit 0 = 1: Temporäre Störungen auf Segment DP2</li> <li>• Bit 1 = 1: Dauerhafte Störungen auf Segment DP2</li> <li>• Bit 4 = 1: Temporäre Störungen auf Segment DP3</li> <li>• Bit 5 = 1: Dauerhafte Störungen auf Segment DP3</li> </ul> |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

Bei den "echten" Fehlerinformationen (Fehlercodes W#16#8xyz) der folgenden Tabelle sind zwei Fälle zu unterscheiden:

- Temporäre Fehler (Fehlercodes W#16#80A2 bis 80A4, 80C3, 80C5):  
Bei dieser Fehlerart ist es möglich, dass sich der Fehler ohne Ihr Zutun behebt, d. h. es ist sinnvoll, dass Sie "DP\_TOPOL" erneut aufrufen (ggf. mehrfach).  
Beispiel für einen temporären Fehler: Benötigte Betriebsmittel sind momentan belegt (W#16#80C3).
- Permanente Fehler (Fehlercodes W#16#8082, 80B0, 80B2):  
Bei dieser Fehlerart kann sich der Fehler nicht ohne Ihr Zutun beheben. Ein erneuter Aufruf von "DP\_TOPOL" ist erst sinnvoll, wenn Sie den Fehler beseitigt haben. Beispiel für einen permanenten Fehler: DP-Master / CPU unterstützt diesen Dienst nicht. (W#16#80B0).

| Fehlercode* (W#16#...) | Erläuterung  |
|------------------------|--|
| 0000                   | Der Auftrag wurde fehlerfrei durchgeführt.   |
| 7000                   | Erstaufwurf mit REQ=0. Es wird keine Topologieermittlung angestoßen. BUSY hat den Wert "0".  |
| 7001                   | Erstaufwurf mit REQ=1. Der Auftrag, die Topologieermittlung durchzuführen, wurde gestellt. BUSY hat den Wert "1".                              |
| 7002                   | Zwischenaufwurf (REQ irrelevant): Die Topologieermittlung ist noch nicht abgeschlossen. BUSY hat den Wert "1".                                 |
| 7010                   | Sie haben versucht, die Topologieermittlung abbrechen. Es gibt aber keinen laufenden Auftrag mit der angegebenen DP_ID. BUSY hat den Wert "0". |
| 7011                   | Erstaufwurf mit R=1. Der Abbruch der Topologieermittlung wurde angestoßen. BUSY hat den Wert "1".  |
| 7012                   | Zwischenaufwurf: Der Abbruch der Topologieermittlung ist noch nicht abgeschlossen. BUSY hat den Wert "1".                                      |
| 7013                   | Letztaufwurf: Die Topologieermittlung wurde abgebrochen. BUSY hat den Wert "0".  |
| 8082                   | Es ist kein DP-Mastersystem mit der angegebenen DP_ID projektiert.   |
| 80A2                   | Fehler bei Topologieermittlung; genauere Informationen entnehmen Sie den Ausgangsparametern DPR und DPRI.                                      |

| Fehlercode*<br>(W#16#...)   | Erläuterung   |
|---|---|
| 80A3  | Fehler bei Topologieermittlung: Überwachungszeit ist abgelaufen (Timeout).  |
| 80A4  | Kommunikation am PBUS+ gestört  |
| 80B0  | DP-Master / CPU unterstützt diesen Dienst nicht.  |
| 80B2  | Fehler bei Topologieermittlung: Am ausgewählten DP-Mastersystem wurde kein Diagnose-Repeater erkannt.   |
| 80C3  | Benötigte Betriebsmittel sind momentan belegt. Mögliche Ursache: Sie haben eine zweite Topologieermittlung angestoßen (nur eine Topologieermittlung ist zu einem Zeitpunkt zulässig). |
| 80C5  | DP-Mastersystem ist momentan nicht verfügbar.   |
| allgemeine Fehlerinformationen  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## ASi

### ASI\_CTRL: Verhalten des ASi-Masters steuern

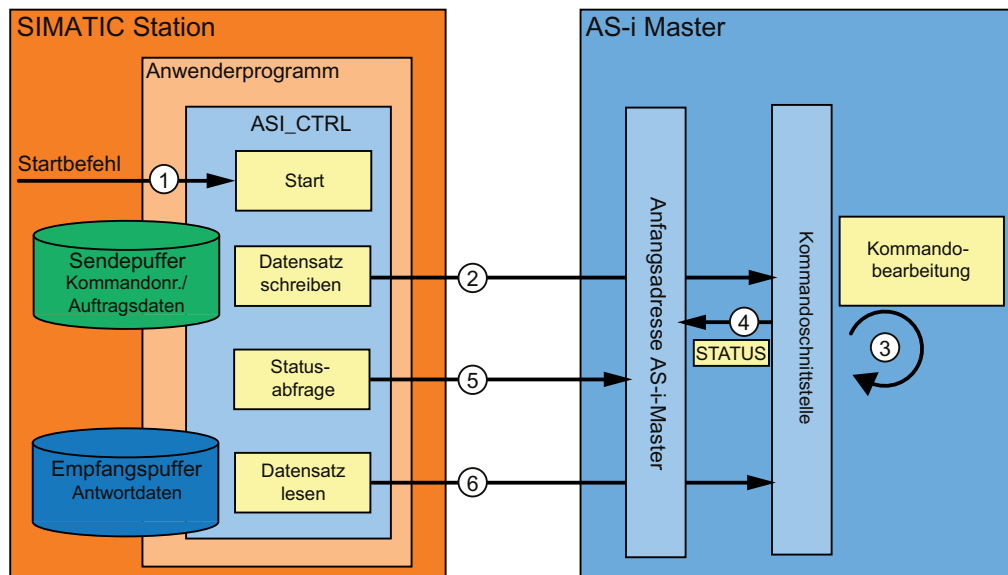
#### Beschreibung ASI\_CTRL

#### Beschreibung

Mit der Anweisung "ASI\_CTRL" kann das Verhalten des AS-i-Masters über das Anwenderprogramm der PLC gesteuert werden. Die Anweisung bearbeitet das Kommandoprotokoll selbstständig und ermöglicht auch die Parametrierungen an SIMATIC AS-i-Mastern sowie das Auslesen von Informationsdaten. Die Möglichkeiten und die Arbeitsweise der Kommandoschnittstelle sind im Handbuch des AS-i-Masters beschrieben.

Unterstützt werden sowohl zentral steckende AS-i-Master als auch dezentrale AS-i-Master über PROFIBUS DP. Auch Kombinationen mit PROFINET IO (Beispiel: IE/PB Link PN IO) sind möglich.

Die folgende Abbildung zeigt schematisch die Funktionalität der Anweisung "ASI\_CTRL":



- ① Start der Bearbeitung am Parameter REQ.
- ② Das Programm setzt das gewünschte Kommando über die Anweisung "RDREC" an den AS-i-Master ab.
- ③ Der AS-i-Master bearbeitet das Kommando.
- ④ Der aktuelle Status des AS-i-Master wird im Eingangsbereich der Binärdaten (logische Basisadresse) abgelegt.
- ⑤ Die Anweisung "ASI\_CTRL" fragt zyklisch die 4 Statusbits ab und wertet diese aus.
- ⑥ Ist die Kommandobearbeitung beendet, wird der Kommandoauftrag über "RDREC" abgeschlossen. Im Datenfeld von "RDREC" befinden sich je nach Kommando die Antwortdaten des Kommandos oder weitere Statusinformationen.

### Unterschiede im Kommandoaufruf bei IE/ AS-i Link und DP/AS-i Links

Für den Kommandoaustausch einer Steuerung mit einem AS-i Master gibt es wesentliche Unterschiede in der Vorgehensweise.

- Beim **IE/AS-i Link** (PROFINET) wird die Datensatzschnittstelle verwendet. Die verschiedenen Kommandos werden entweder durch "Datensatz schreiben" (Anweisung "WRREC") oder durch "Datensatz lesen" ("Anweisung "RDREC") von verschiedenen Datensatznummern aufgerufen.
- **DP/AS-i Links** (PROFIBUS) verwenden die Kommandoschnittstelle. Alle Kommandos werden durch ein paarweises "Datensatz schreiben" ("Anweisung "WRREC") und "Datensatz lesen" ("Anweisung "RDREC") von Datensatznummer 2 aufgerufen. Der Typ des Kommandos wird durch den Dateninhalt beim Schreibauftrag definiert.



## Änderungen gegenüber der Anweisung "ASi\_3422"

Die Anweisung "ASI\_CTRL" ist eine Überarbeitung der Anweisung "ASi\_3422" (S7-300/400) und weist Verbesserungen hinsichtlich Funktionalität und Kompatibilität auf. Im Detail wurden folgende Änderungen durchgeführt:

- Zum Schreiben und Lesen von Diagnosedatensätze wurden die Anweisungen "WR\_REC (Seite 2435)" und "RD\_REC (Seite 2430)" durch die Anweisungen "RDREC (Seite 2392)" und "WRREC (Seite 2394)" ersetzt. Sie sind in der Funktion identisch, unterstützen jedoch die Datenübertragung über PROFINET IO.
- Der Bausteintyp der Anweisung wurde von einer Funktion (FC) in einen Funktionsbaustein (FB) gewandelt. "ASI\_CTRL" verfügt über einen Instanzdatenbaustein und ist multiinstanzfähig.
- Die Benennung der Formalparameter von "ASI\_CTRL" ist konform zu den SIMATIC Systembausteinen. Der Eingangsparameter STARTUP entfällt. Die Definition des Parameters STATUS wurde angelehnt an die Anweisungen "RDREC (Seite 2392)" und "WRREC (Seite 2394)". Zusätzlich wurden die Statuskennungen für den Parameter DONE und den neu eingeführten Parameter BUSY angepasst.

## Arbeitsweise der Anweisung "ASI\_CTRL"

Die Anweisung "ASI\_CTRL" ist ein asynchron arbeitender Funktionsbaustein, das bedeutet die Bearbeitung erstreckt sich über mehrere Aufrufe.

- Gestartet wird ein Auftrag mit REQ = TRUE.
- Über die Ausgangsparameter BUSY und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt.
- Während einer Auftragsbearbeitung ist der Parameter BUSY gesetzt. Bei Erstauftrag enthält STATUS dabei den Wert 00700100<sub>H</sub>. Bei allen Folgeaufrufen, welche zu diesem Auftrag gehören, den Wert 00700200<sub>H</sub>. Mit Abschluss des Auftrages wird das Ergebnis an den Parametern DONE bzw. ERROR ausgegeben.
  - Ist kein Fehler aufgetreten, wird DONE gesetzt. Bei Aufträgen mit Antwortdaten vom AS-i-Master werden diese im angegebenen Empfangspuffer zur Verfügung gestellt. In diesem Fall wird im Parameter STATUS zusätzlich die Menge der gelieferten Daten in Bytes angezeigt. Bei Aufträgen ohne Antwortdaten wird in STATUS der Wert 00000000<sub>H</sub> eingetragen.
  - Tritt während der Auftragsbearbeitung ein Fehler auf, wird ERROR gesetzt. In diesem Fall ist der Inhalt des Empfangspuffers ungültig. Zur näheren Beschreibung des aufgetretenen Fehlers wird im Parameter STATUS ein Fehlercode eingetragen.

## Anzahl Kommandoaufrufe

Wenn Sie die Anweisung "ASI\_CTRL" zur Kommandobeauftragung nutzen, dürfen Sie nicht gleichzeitig weitere Kommandos über "RDREC (Seite 2392)" und "WRREC (Seite 2394)" an denselben AS-i-Master absetzen. Dies gilt auch für Mehrfachaufrufe der Anweisung an denselben AS-i-Master.

Die Anweisung "ASI\_CTRL" ist nicht unterbrechbar ablauffähig. Aufrufe dürfen daher nicht in Programmablaufebenen programmiert werden, die sich gegenseitig unterbrechen (z.B. durch Aufruf in OB 1 und in OB 35).

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "ASI\_CTRL":

| Parameter | Deklaration | Datentyp | Speicherbereich          | Beschreibung  |
|-----------|-------------|----------|--------------------------|---|
| REQ       | Input       | BOOL     | E, A, M, D, L, Konstante | Mit REQ = TRUE wird ein neuer Auftrag gestartet, falls nicht bereits ein Auftrag in Bearbeitung ist. Es findet keine Flankenauswertung statt.   |
| LADDR     | Input       | WORD     | E, A, M, D, L, Konstante | Anfangsadresse des AS-i-Masters im S7-Adressraum (logische Basisadresse). Die Anfangsadresse wird beim Projektieren des Masters in der Hardware-Konfiguration festgelegt.   |
| SD        | Input       | VARIANT  | E, A, M, D, L            | Sendepuffer<br>Der Parameter verweist auf einen Speicherbereich, in dem das Kommando zu spezifizieren ist (siehe "ASi-Kommandos (Seite 2461)").<br>Beispiel: P#DB101.DBX 0.0 BYTE 223   |
| RD        | Input       | VARIANT  | E, A, M, D, L            | Empfangspuffer<br>Dieser Puffer ist nur bei Kommandos relevant, die Antwortdaten liefern. Der Parameter verweist auf einen Speicherbereich, in dem eine Kommandoantwort abgelegt wird (siehe "ASi-Kommandos (Seite 2461)").<br>Beispiel: P#DB102.DBX 224.0 BYTE 221 |
| DONE      | Output      | BOOL     | A, M, D, L               | DONE = TRUE: Auftrag ohne Fehler beendet.   |
| BUSY      | Output      | BOOL     | A, M, D, L               | BUSY = TRUE: Auftrag in Bearbeitung.  |
| ERROR     | Output      | BOOL     | A, M, D, L               | ERROR = TRUE: Auftrag mit Fehler abgebrochen.   |
| STATUS    | Output      | DWORD    | M, D                     | Auftragsstatus / Fehlercode<br>Siehe Beschreibung "Parameter STATUS".   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Hinweis**

**Parameter LADDR, SD und RD**

Die Parameter LADDR, SD und RD dürfen während einer Auftragsbearbeitung in keinem Bausteindurchlauf verändert werden, sondern müssen konstant bleiben.

## Parameter STATUS

Die folgende Tabelle enthält die möglichen Anzeigen von STATUS in Abhängigkeit von DONE und ERROR.

| DONE | ERROR | STATUS                | Bedeutung   |
|------|-------|-----------------------|---|
| 0    | 0     | 00700000 <sub>H</sub> | Erstaufruf mit REQ = FALSE; keine Auftrag aktiv.  |
| 0    | 0     | 00700100 <sub>H</sub> | Erstaufruf mit REQ = TRUE; Auftrag angestoßen.  |
| 0    | 0     | 00700200 <sub>H</sub> | Folgeaufruf (REQ irrelevant); Auftrag wird derzeit noch bearbeitet.                                       |
| 1    | 0     | 00000000 <sub>H</sub> | Auftrag ohne Fehler abgeschlossen. Keine Antwortdaten.  |
| 1    | 0     | 0000xx00 <sub>H</sub> | Auftrag ohne Fehler abgeschlossen. Anzahl xx-Bytes Antwortdaten.  |
| 0    | 1     | C0818400 <sub>H</sub> | Datentyp des Formaloperanden RD unzulässig.   |
| 0    | 1     | C0818500 <sub>H</sub> | Kommunikationsfehler mit AS-i-Master (falsche Adresse am Parameter LADDR projiziert).                     |
| 0    | 1     | C0838100 <sub>H</sub> | Falsche AS-i-Slave-Adresse.   |
| 0    | 1     | C0838200 <sub>H</sub> | Der AS-i-Slave ist nicht aktiviert (nicht in LAS).  |
| 0    | 1     | C0838300 <sub>H</sub> | Fehler am AS-Interface (der Parameter SD ist eventuell zu klein eingestellt).                             |
| 0    | 1     | C0838400 <sub>H</sub> | Das Kommando ist im aktuellen Zustand des AS-i-Masters nicht zulässig.                                    |
| 0    | 1     | C0838500 <sub>H</sub> | Ein AS-i-Slave mit Adresse "0" existiert.   |
| 0    | 1     | C0838600 <sub>H</sub> | Der AS-i-Slave hat unzulässige Konfigurationsdaten (E/A bzw. ID-Codes).                                   |
| 0    | 1     | C083A100 <sub>H</sub> | Der angesprochene AS-i-Slave wurde am AS-Interface nicht gefunden.  |
| 0    | 1     | C083A200 <sub>H</sub> | Ein AS-i-Slave mit Adresse "0" existiert.   |
| 0    | 1     | C083A300 <sub>H</sub> | Ein AS-i-Slave mit der neuen Adresse ist bereits am AS-Interface vorhanden.                               |
| 0    | 1     | C083A400 <sub>H</sub> | Die AS-i-Slave-Adresse lässt sich nicht löschen.  |
| 0    | 1     | C083A500 <sub>H</sub> | Die AS-i-Slave-Adresse lässt sich nicht setzen.   |
| 0    | 1     | C083A600 <sub>H</sub> | Die AS-i-Slave-Adresse lässt sich nicht permanent speichern.  |
| 0    | 1     | C083A700 <sub>H</sub> | Fehler beim Lesen des Extended ID1-Codes.   |
| 0    | 1     | C083A800 <sub>H</sub> | Die Zieladresse ist nicht plausibel (z. B. wird eine B-Slave-Adresse für einen Standard-Slave verwendet). |
| 0    | 1     | C083B100 <sub>H</sub> | Es ist ein Längenfehler beim Stringtransfer aufgetreten.  |
| 0    | 1     | C083B200 <sub>H</sub> | Es ist ein Protokollfehler beim Stringtransfer aufgetreten.   |
| 0    | 1     | C083F800 <sub>H</sub> | Die Auftragsnummer oder der Auftragsparameter sind unbekannt.   |
| 0    | 1     | C083F900 <sub>H</sub> | Der AS-i-Master hat einen EEPROM-Fehler festgestellt.   |

## ASi-Kommandos

### Beschreibung

Durch die Kommandoschnittstelle ist die Möglichkeit gegeben, Parametrier- und Informationsdaten zwischen Steuerung und AS-i-Master auszutauschen.

Mit diesen Kommandos

- wird die komplette Funktionalität des Masterprofiles M4 der AS-i-Masterspezifikation zur Verfügung gestellt.
- kann der AS-i-Master komplett von der Steuerung aus projiziert werden.

**Hinweis**

**Unterstützte AS-i-Kommandos**

Welche AS-i Kommandos unterstützt werden sowie eine detaillierte Beschreibung finden Sie im Handbuch des jeweiligen AS-i-Masters.

**Allgemeine Struktur des Sendepuffers**

Die allgemeine Struktur des Sendepuffers für Kommandos und Auftragsdaten zeigt die folgende Tabelle. Der Bereich für die Kommandonummer muss immer belegt sein. Die Anzahl Bytes für die Auftragsdaten sind in der Beschreibung des jeweiligen Kommandos enthalten (siehe Dokumentation des AS-i-Masters). "q" entspricht dabei der Anfangsadresse des Sendepuffers.

| Byte    | Bit 7            | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---------|------------------|-------|-------|-------|-------|-------|-------|-------|
|         | <b>Bedeutung</b> |       |       |       |       |       |       |       |
| q + 0   | Kommandonummer   |       |       |       |       |       |       |       |
| q + 1   | Auftragsdaten    |       |       |       |       |       |       |       |
| q + 2   | Auftragsdaten    |       |       |       |       |       |       |       |
| q + ... | Auftragsdaten    |       |       |       |       |       |       |       |

**Allgemeine Struktur des Empfangspuffers**

Die allgemeine Struktur des Empfangspuffers für die Antwortdaten des Kommandos zeigt folgende Tabelle. Die Anzahl Bytes für die Antwortdaten ist vom jeweiligen Kommando abhängig. Einige Kommandos liefern keine Antwortdaten und benötigen somit lediglich die Angabe eines virtuellen Empfangspuffers, der nicht mit Daten befüllt wird. "n" entspricht dabei der Anfangsadresse des Empfangspuffers.

| Byte    | Bit 7                 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---------|-----------------------|-------|-------|-------|-------|-------|-------|-------|
|         | <b>Bedeutung</b>      |       |       |       |       |       |       |       |
| n + 0   | Kommandonummer (Echo) |       |       |       |       |       |       |       |
| n + 1   | Antwortdaten          |       |       |       |       |       |       |       |
| n + 2   | Antwortdaten          |       |       |       |       |       |       |       |
| n + ... | Antwortdaten          |       |       |       |       |       |       |       |

**ACHTUNG****Speicherbereiche können überschrieben werden**

Ist der Empfangspuffer der Anweisung "ASI\_CTRL" zu kurz, werden eventuell angrenzende Speicherbereiche überschrieben. Die Längenangabe im ANY-Zeiger des Parameters RD beim Aufruf der Anweisung "ASI\_CTRL" ist irrelevant. Die benötigte Länge des Empfangspuffers ist der Beschreibung des Kommandos zu entnehmen.

Für die Kommandonummern 39<sub>H</sub>, 41<sub>H</sub>, 42<sub>H</sub>, 43<sub>H</sub> und 44<sub>H</sub> gilt:

Der Empfangspuffer muss eine Länge von 221 Byte haben (Byte 0 bis 220), auch wenn das Kommando weniger Daten zurückliefert. Je nach Kommando werden die im Empfangspuffer enthaltenen höchsten Bytes vom AS-i-Master eventuell mit Nullwerten überschrieben.

**AS-i-Kommandos**

Die folgende Tabelle zeigt eine Auswahl möglicher AS-i-Kommandos.

| Name   | Parameter                       | Rückgabe                            | Codierung       |
|--|---------------------------------|-------------------------------------|-----------------|
| Parameterwert_projektieren<br>(Set_Permanent_Parameter)              | Slave-Adresse, Parameter        |                                     | 00 <sub>H</sub> |
| Projektierten_Parameterwert_lesen<br>(Get_Permanent_Parameter)       | Slave-Adresse                   | Parameter                           | 01 <sub>H</sub> |
| Parameterwert_schreiben<br>(Write_Parameter)                         | Slave-Adresse, Parameter        | Parameterecho                       | 02 <sub>H</sub> |
| Parameterwert_lesen<br>(Read_Parameter)                              | Slave-Adresse                   | Parameterwert                       | 03 <sub>H</sub> |
| Ist_Parameterwerte_projektieren<br>(Store_Actual_Parameters)         |                                 |                                     | 04 <sub>H</sub> |
| Konfigurationsdaten_projektieren                                     | Slave-Adresse,<br>Konfiguration |                                     | 25 <sub>H</sub> |
| Projektierte_Konfigurationsdaten_lesen                               | Slave-Adresse                   | projektierte<br>Konfigurationsdaten | 26 <sub>H</sub> |
| Ist_Konfigurationsdaten_projektieren<br>(Store_Actual_Configuration) |                                 |                                     | 07 <sub>H</sub> |
| Ist-Konfigurationsdaten_lesen  | Slave-Adresse                   | Ist-Konfigurationsdaten             | 28 <sub>H</sub> |
| LPS_projektieren   | LPS                             |                                     | 29 <sub>H</sub> |
| Offlinemodus_setzen  | Mode                            |                                     | 0A <sub>H</sub> |
| Autoprogrammieren_wählen   | Mode                            |                                     | 0B <sub>H</sub> |
| Betriebsmodus_setzen   | Mode                            |                                     | 0C <sub>H</sub> |
| AS-iSlave-Adresse_ändern<br>(Change_AS-iSlave_Address)               | Adresse1, Adresse2              |                                     | 0D <sub>H</sub> |
| AS-iSlavestatus_lesen  | Slave-Adresse                   | Fehlerrecord des AS-<br>iSlaves     | 0F <sub>H</sub> |
| Listen_und_Flags_lesen   |                                 | LDS, LAS, LPS, Flags                | 30 <sub>H</sub> |

| Name   | Parameter                       | Rückgabe  | Codierung       |
|--|---------------------------------|---|-----------------|
| Gesamtkonfiguration_lesen                      |                                 | Ist-Konfigurationsdaten, Aktuelle Parameter, LAS, Flags | 39 <sub>H</sub> |
| Gesamtkonfiguration_projektieren               | Gesamtkonfiguration             |   | 3A <sub>H</sub> |
| Parameterliste_schreiben                       | Parameter Liste                 |   | 3C <sub>H</sub> |
| Parameterecholiste_lesen                       |                                 | Parameterecho-Liste                                     | 33 <sub>H</sub> |
| CTT2-Request_schreiben                         | Slave Adresse<br>CTT2-String    | CTT2-String   | 44 <sub>H</sub> |
| Versionskennung_lesen                          |                                 | Versions-String   | 14 <sub>H</sub> |
| AS-iSlave-ID_lesen                             | Slave-Adresse                   | ID-Code   | 17 <sub>H</sub> |
| AS-iSlave-Extended-ID1_lesen                   | Slave-Adresse                   | Extended ID1-Code                                       | 37 <sub>H</sub> |
| AS-iSlave-Extended-ID1_schreiben               | Extended ID1-Code               |   | 3F <sub>H</sub> |
| AS-iSlave-Extended-ID2_lesen                   | Slave-Adresse                   | Extended ID2-Code                                       | 38 <sub>H</sub> |
| AS-iSlave-EA_lesen                             | Slave-Adresse                   | E/A-Konfiguration                                       | 18 <sub>H</sub> |
| Peripheriefehlerliste_lesen                    |                                 | LPF   | 3E <sub>H</sub> |
| AS-i-Slave_Parameter-String_schreiben          | Slave-Adresse, Parameter-String |   | 40 <sub>H</sub> |
| AS-iSlave_Parameter-String_lesen               | Slave-Adresse                   | Parameter-String  | 41 <sub>H</sub> |
| AS-iSlave_ID-String_lesen                      | Slave-Adresse                   | ID-String   | 42 <sub>H</sub> |
| AS-iSlave_Diagnose-String_lesen                | Slave-Adresse                   | Diagnose-String   | 43 <sub>H</sub> |
| AS-iStrang_Fehlerzähler_lesen                  |                                 |   | 4A <sub>H</sub> |
| AS-iStrang_Fehlerzähler_lesen_und_löschen      |                                 |   | 4B <sub>H</sub> |
| AS-iSlave_Fehlerzähler_lesen                   | Slave-Adresse                   |   | 4C <sub>H</sub> |
| AS-iSlave_Fehlerzähler_lesen_und_löschen       | Slave-Adresse                   |   | 4D <sub>H</sub> |
| Zusätzliches Kommando für den DP/ AS-i F-Link: |                                 |   |                 |
| AS-i-Status/Diag_der_F-Slaves                  |                                 | Status / Diagnose aller ASIsafe-Salves                  | 51 <sub>H</sub> |

**Hinweis**

**Erneute Initialisierung der Kommandoschnittstelle des AS-i-Masters**

Ein weiteres Kommando, welches in der Tabelle nicht erwähnt wird, ist das Kommando 77<sub>H</sub>. Mit diesem Aufruf wird die Kommandoschnittstelle des AS-i-Masters neu initialisiert. Falls der angegebene AS-i-Master zu diesem Zeitpunkt mit einem Kommando beschäftigt sein sollte, wird dieses abgebrochen.

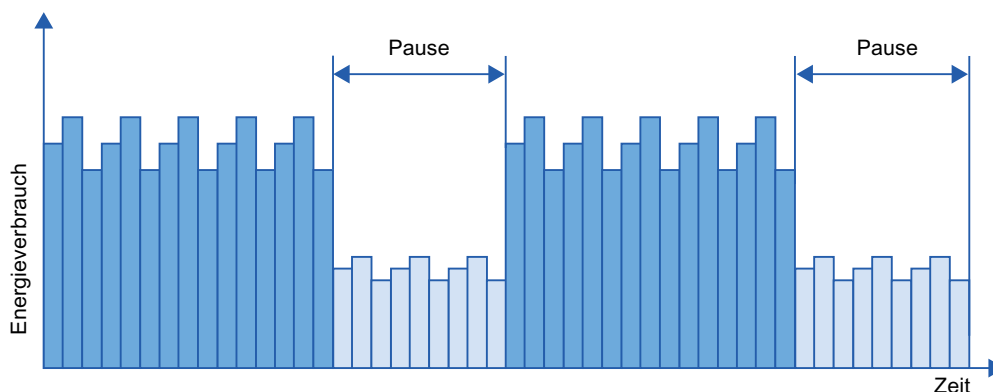
Ab Version V2.1.20 des DP/AS-i LINK Advanced wird zusätzlich das Kommando 0E<sub>H</sub> angeboten. Mittels dieses Aufrufes kann die Funktion Erdschlussüberwachung eines Strangs freigegeben oder gesperrt werden.

### 9.7.3.5 PROFlenergy

#### Beschreibung PROFlenergy

#### PROFlenergy

PROFlenergy ist ein hersteller- und geräteunabhängiges Profil für das Energiemanagement mit PROFINET. Mit PROFlenergy lassen sich Geräte koordiniert und zentral abschalten, um während Produktionspausen oder ungeplanten Unterbrechungen den Stromverbrauch zu reduzieren.



Die Abschaltung der PROFINET-Geräte bzw. der Powermodule erfolgt über spezielle Kommandos im Anwenderprogramm des PROFINET IO-Controllers. Es wird keine zusätzliche Hardware benötigt. Die PROFlenergy-Kommandos werden direkt von den PROFINET-Geräten interpretiert.

#### PROFlenergy-Controller (PE-Controller)

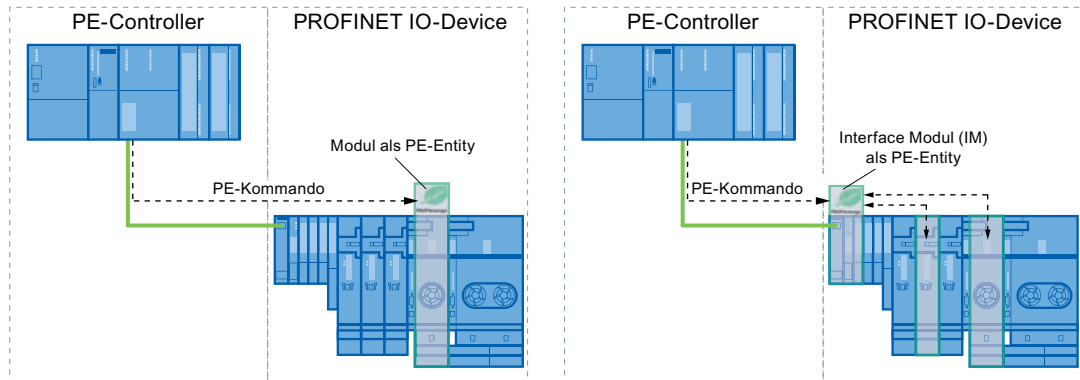
Der PE-Controller ist eine PLC, die bei unterlagerten Geräten den Ruhezustand aktiviert bzw. deaktiviert. Die Abschaltung bzw. Reaktivierung von einzelnen Produktionskomponenten oder ganzen Produktionslinien erfolgt dabei über das Anwenderprogramm. Über entsprechende Anweisungen (Funktionsbausteine) werden Kommandos (z.B. "Start\_Pause" oder "End\_Pause") an das unterlagerte Gerät abgesetzt. Das Senden der Kommandos erfolgt über das PROFINET-Kommunikationsprotokoll.

#### PROFlenergy-Entity (PE-Entity)

Die PE-Entity empfängt die PROFlenergy-Kommandos des PE-Controllers und setzt die Kommandos entsprechend um (z.B. durch Rücksenden eines Messwerts oder Aktivierung eines Energiesparmodus). Die Umsetzung der PE-Entity in einem PROFlenergy-fähigem Gerät ist geräte- und herstellerepezifisch.

Die PE-Entity kann z. B. umgesetzt werden:

- Innerhalb des Proxys eines Moduls: Die PE-Kommandos sind jeweils gültig für das adressierte Modul und ggf. unterlagerte Module.
- Innerhalb des Proxys einer Baugruppe: Die PE-Kommandos sind jeweils gültig für verschiedene Module innerhalb der Baugruppe.



- Innerhalb eines Moduls ohne Proxy-Funktion innerhalb des Netzwerks: Die PE-Kommandos sind in diesem Fall nur für das jeweilige Modul gültig.

## PROFenergy-Anweisungen

- Anweisungen für IO-Controller
  - Mit der Anweisung "PE\_START\_END (Seite 2467)" wird auf einfachste Art und Weise der Ruhezustand der PROFINET-Geräte aktiviert oder deaktiviert (PROFenergy-Kommandos "Start\_Pause" und "End\_Pause"). Das geschieht über eine positive und negative Signalfanke in der Anweisung.
  - Mit der Anweisung "PE\_CMD (Seite 2471)" lassen sich alle PROFenergy-Kommandos einschließlich "Start\_Pause" und "End\_Pause" übertragen. Darüber hinaus kann mit weiteren Kommandos z. B. der aktuelle Status der PROFINET-Geräte oder das Verhalten während der Pausen abgefragt werden.
  - Mit der Anweisung "PE\_DS3\_Write\_ET200S (Seite 2476)" werden die Einstellungen für das Schaltverhalten von bis zu 8 Steckplätzen der ET 200S festgelegt. Die Anweisung gehört nicht zu den PROFenergy-Anweisungen, ergänzt aber die PROFenergy-Funktionen für eine ET 200S.
- Anweisung für iDevices
 

Mit der Anweisung "PE\_I\_DEV (Seite 2489)" lässt sich PROFenergy auch auf iDevices realisieren. Die Anweisung empfängt PROFenergy-Kommandos auf dem iDevice und gibt sie an das Anwenderprogramm zur Bearbeitung weiter. Nach der Bearbeitung des Kommandos durch das Anwenderprogramm ruft es die Anweisung "PE\_I\_DEV (Seite 2489)" erneut auf, um die Quittung an den IO-Controller zu senden. Für diese Antworten steht ihnen für jedes Kommando ein entsprechender Hilfsbaustein zur Verfügung, der die Anweisung "PE\_I\_DEV (Seite 2489)" mit den Antwortdaten versorgt.

## PROFenergy-Kommandos (PE-Kommandos)

PE-Kommandos werden von dem PE-Controller an die PE-Entity gesendet. Das PE-Kommando kann entweder ein Steuerungskommando sein, um eine PE-Entity in einen



Energiesparmodus zu schalten oder ein Kommando zum Auslesen eines Status oder Messwerts:

- PE-Kommandos zur Steuerung  
PROFenergy unterstützt zwei Steuerungskommandos, die entweder über die Anweisung "PE\_Start\_End (Seite 2467)" oder die Anweisung "PE\_CMD (Seite 2471)" ausgeführt werden können:
  - Start\_Pause: Starten eines geeigneten Energiesparmodus (PE Energy-saving mode)
  - End\_Pause: Beenden des Energiesparmodus (Wechsel in den Modus PE\_ready\_to\_operate)
- PE-Kommandos zum Auslesen eines Status oder Messwerts  
Über die folgenden Statuskommandos können mit der Anweisung "PE\_CMD (Seite 2471)" bestimmte Zustandsinformationen durch die Steuerung ausgelesen werden:
  - PE\_Identify: Auslesen, welche PE-Kommandos von der PE-Entity unterstützt werden.
  - PEM\_Status: Auslesen des momentan aktiven Modus einer PE-Entity (z. B. PE\_ready\_to\_operate).
  - Query\_Modes: Ausgabe einer Übersicht zu allen unterstützten Energiesparmodi, inklusive der Zeit- und Energieinformationen
  - Query\_Measurement: Ausgabe der Messwerte einer PE-Entity

## Applikationsbeispiele

Beispiele zur Anwendung der PROFenergy-Anweisungen stehen über den Industry Online Support unter dem Beitrag "PROFenergy - Energie sparen mit SIMATIC S7 (<http://support.automation.siemens.com/WW/view/de/41986454>)" zur Verfügung.

## Siehe auch

Service und Support (<http://support.automation.siemens.com/>)

## IO-Controller

### PE\_START\_END: Energiesparmodus starten und beenden

#### Beschreibung

Mit der Anweisung "PE\_START\_END" wird für die angegebene PE-Entity (z.B. die ET 200S) die Energiesparpause gestartet und beendet.

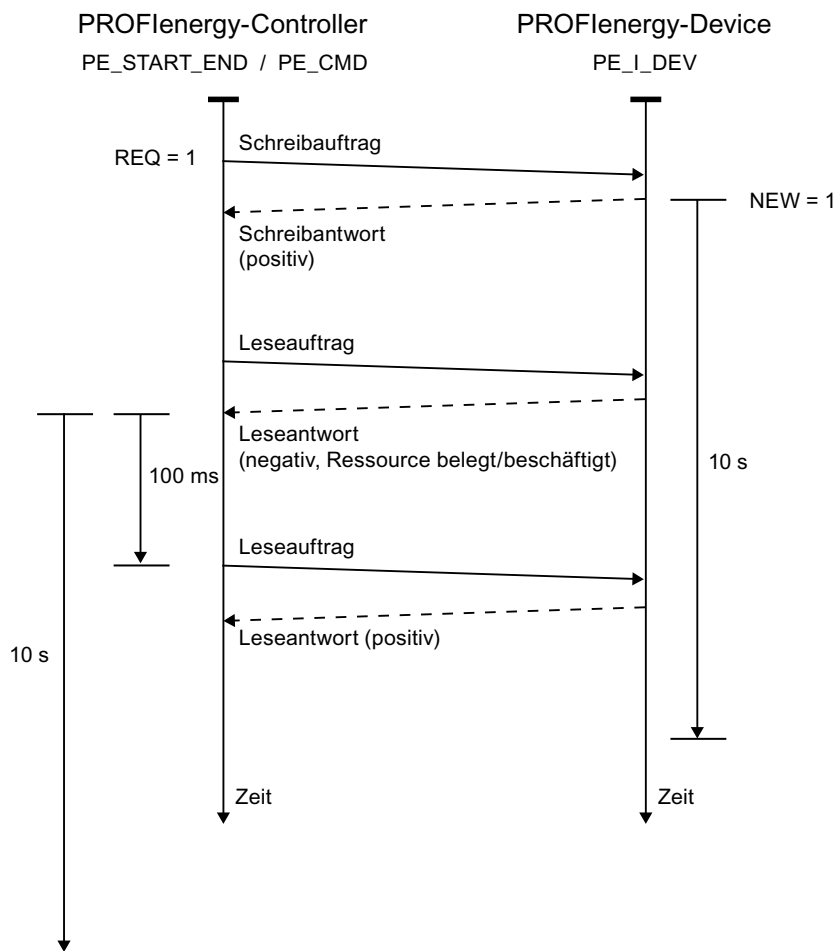
Die Anweisung "PE\_START\_END" wird im PE-Controller verwendet, vorzugsweise wenn an den zugeordneten PE-Devices nur Feldgeräte angeschlossen sind, bei denen keine Energiedaten ausgelesen werden müssen. Zum Lesen der Energiedaten kann alternativ die Anweisung "PE\_CMD (Seite 2471)" verwendet werden.

Die Energiesparmodi werden im Anwenderprogramm des PE-Controllers projiziert. Der tatsächlich eingenommene Energiesparmodus wird nach Ausführung von "PE\_START\_END" von der PE-Entity zurückgemeldet und am Parameter PE\_MODE\_ID ausgegeben.

### Schreib- und Leseaufträge der Anweisung "PE\_START\_END"

Die Anweisung "PE\_START\_END" versendet intern mittels "WRREC (Seite 2394)" einen PROFIenergy-Befehl als Schreibauftrag an die PE-Entity. Anschließend wartet "PE\_START\_END" auf die Quittung von der PE-Entity. Dazu wird alle 100 Millisekunden der Quittungsdatensatz mit der Anweisung "RDREC (Seite 2392)" gelesen. Solange keine Quittung der PE-Entity eingetroffen ist, wird der Leseauftrag für 10 Sekunden im Abstand von jeweils 100 Millisekunden wiederholt. Die Antwortdaten der PE-Entity werden ebenfalls mit der Anweisung "RDREC (Seite 2392)" gelesen.

Die folgende Abbildung zeigt das Ablaufdiagramm der Schreib- und Leseaufträge:



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE\_START\_END":

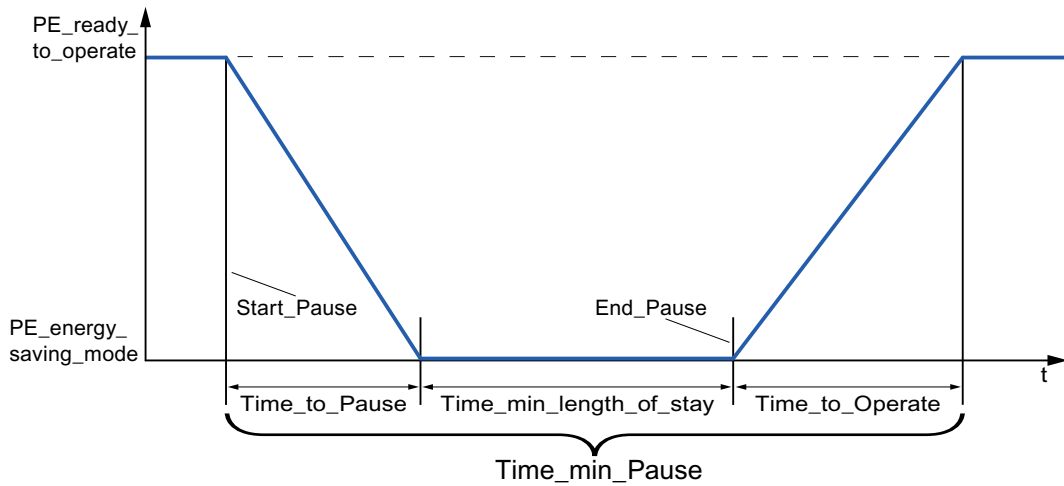
| Parameter  | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|------------|-------------|--------------|------------------------------|--|
| START      | Input       | BOOL         | E, A, M, D, L oder Konstante | Senden des PE-Kommandos "Start_Pause" an die PE-Entity mit der Adresse am Parameter ID.  |
| END        | Input       | BOOL         | E, A, M, D, L oder Konstante | Senden des PE-Kommandos "End_Pause" an die PE-Entity mit der Adresse am Parameter ID.  |
| ID         | Input       | HW_SUBMODULE | E, A, M, D, L oder Konstante | Adresse der PE-Entity (z.B. ET 200S). Die Adresse kann aus der Hardware-Konfiguration übernommen werden.                                     |
| PAUSE_TIME | Input       | TIME         | E, A, M, D, L oder Konstante | Geplante Pausendauer. <ul style="list-style-type: none"> <li>• Bereich: T#1MS bis T#24D20H31M23S647MS</li> <li>• Startwert: T#0MS</li> </ul> |
| VALID      | Output      | BOOL         | E, A, M, D, L                | PE-Kommando erfolgreich abgesetzt.   |
| BUSY       | Output      | BOOL         | E, A, M, D, L                | Bearbeitung des PE-Kommandos läuft noch.   |
| ERROR      | Output      | BOOL         | E, A, M, D, L                | Bei der Bearbeitung trat ein Fehler auf. Die Fehlermeldung wird am Parameter STATUS ausgegeben.  |
| STATUS     | Output      | DWORD        | E, A, M, D, L                | Bausteinstatus / Fehlernummer (siehe "Parameter STATUS").  |
| PE_MODE_ID | Output      | BYTE         | E, A, M, D, L                | Identifikationsnummer des Energiesparmodus (Energiesparlevel, der während der Pausendauer eingenommen wird).                                 |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter PAUSE\_TIME

Mit dem Parameter PAUSE\_TIME wird der PE-Entity die Dauer der Energiesparpause vorgegeben. In der PE-Entity wird geprüft, ob die vorgegebene Pausendauer ausreichend groß ist und umgesetzt werden kann. Die Mindest-Pausendauer (Time\_min\_Pause) muss dabei größer sein, als die Summe der Zeiten, die das Gerät zum Wechsel in den

Energiesparmodus (Time\_to\_Pause) und zum Wechsel zurück in den Betriebsmodus (Time\_to\_Operate) benötigt.



Bei der ET 200S erfolgt eine Prüfung, ob die geplante Pausendauer größer oder gleich der minimalen Pausendauer (PM-E\_Pause\_Min) ist, die auf der ET 200S hinterlegt ist. Diese beträgt fest 10 Sekunden. Wird eine kleinere Pause verwendet, bleiben die Powermodule (PM-E) der ET 200S eingeschaltet.

Es erfolgt kein automatisches Wiedereinschalten nach Ablauf der Pausenzeit, die Baugruppe bleibt bis zu dem "END"-Befehl im Zustand AUS. Damit wird ein unkoordiniertes Wiedereinschalten, was zu ungewollten Lastspitzen führen kann, verhindert.

### Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wird er als ARRAY[1...4] of BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

| Feldelement | Name         | Bedeutung   |
|-------------|--------------|---|
| STATUS[1]   | Function_Num | Fehlerursache <ul style="list-style-type: none"> <li>• B#16#00: Kein Fehler</li> <li>• B#16#DE: Fehler beim Lesen des Datensatzes</li> <li>• B#16#DF: Fehler beim Schreiben des Datensatzes</li> <li>• B#16#C0: Fehlermeldung durch die Anweisung oder durch die intern verwendeten Kommunikationsanweisungen "RDREC (Seite 2392)" und "WRREC (Seite 2394)".</li> </ul> |
| STATUS[2]   | Error Decode | Ort der Fehlerkennung <ul style="list-style-type: none"> <li>• 80: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch</li> <li>• FE: DP/PNIO-Profil - PROFInergy-spezifischer Fehler</li> </ul>  |

| Feldelement | Name         | Bedeutung   |
|-------------|--------------|---|
| STATUS[3]   | Error_Code_1 | Fehlerkennung <ul style="list-style-type: none"> <li>• Bei Error Decode = <b>80</b>:               <ul style="list-style-type: none"> <li>– <b>80</b>: Gleichzeitig eine steigende Flanke an den Eingangsparametern START und END.</li> <li>– <b>81</b>: Längenkonflikt bei den Parametern CMD_PARAM und CMD_PARAM_LEN.</li> <li>– <b>82-8F</b>: weitere Fehlermeldungen (reserviert)</li> </ul> </li> <li>• Bei Error Decode = <b>FE</b>:               <ul style="list-style-type: none"> <li>– <b>01</b>: "Service Request ID" ungültig</li> <li>– <b>02</b>: Falsche "Request_Reference"</li> <li>– <b>03</b>: "Modifier" ungültig</li> <li>– <b>04</b>: "Data Structure Identifier RQ" ungültig</li> <li>– <b>05</b>: "Data Structure Identifier RS" ungültig</li> <li>– <b>06</b>: "PE energy-saving modes" werden nicht unterstützt</li> <li>– <b>07</b>: "Response" ist zu lange (die maximal übertragbare Länge wurde überschritten)</li> <li>– <b>08</b>: "Count" ungültig</li> <li>– <b>50</b>: Es steht kein passender "energy mode" zur Verfügung.</li> <li>– <b>51</b>: Angegebener Zeitwert wird nicht unterstützt.</li> <li>– <b>52</b>: "PE_Mode_ID" unzulässig</li> </ul> </li> </ul> |
| STATUS[4]   | Error_Code_2 | herstellerspezifische Erweiterung der Fehlerkennung   |

**Hinweis****Fehlermeldungen der Anweisungen RDREC und WRREC**

Die Anweisung "PE\_START\_END" verwendet zur Kommunikation die Anweisungen "WRREC (Seite 2394)" und "RDREC (Seite 2392)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "WRREC (Seite 2394)" und "RDREC (Seite 2392)" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 2402).

**Siehe auch**

Beschreibung PROFIenergy (Seite 2465)

**PE\_CMD: Energiesparmodus starten und beenden / Zustandsinformationen auslesen****Beschreibung**

Die Anweisung "PE\_CMD" wird im PE-Controller verwendet und löst in der PE-Entity eine Energiesparpause aus oder beendet eine Pause. Zusätzlich können mit "PE\_CMD" weitere Informationen und Energiemesswerte aus einer PE-Entity ausgelesen werden.

Die Anweisung kann vorzugsweise bei PE-Controllern eingesetzt werden, an deren zugeordneten PE-Devices Feldgeräte angeschlossen sind, von denen Energiemesswerte

ausgelesen werden sollen. Ist dies nicht der Fall, kann zum starten und beenden der Pausen auch die Anweisung "PE\_START\_END (Seite 2467)" verwendet werden.

### Übertragung der PROFenergy-Kommandos (PE-Kommandos)

Die Anweisung "PE\_CMD" überträgt ein PROFenergy-Kommando an eine PE-Entity.

Die Anweisung kann auch eingesetzt werden, wenn das PROFenergy-Profil zukünftig um weitere Kommandos erweitert wird. Die Kommandos, die nach dem aktuellen PROFenergy-Profil verwendet werden können, sind in der Beschreibung der Parameter CMD und CMD\_MODIFIER aufgelistet (siehe Tabelle "Parameter CMD und CMD\_MODIFIER").

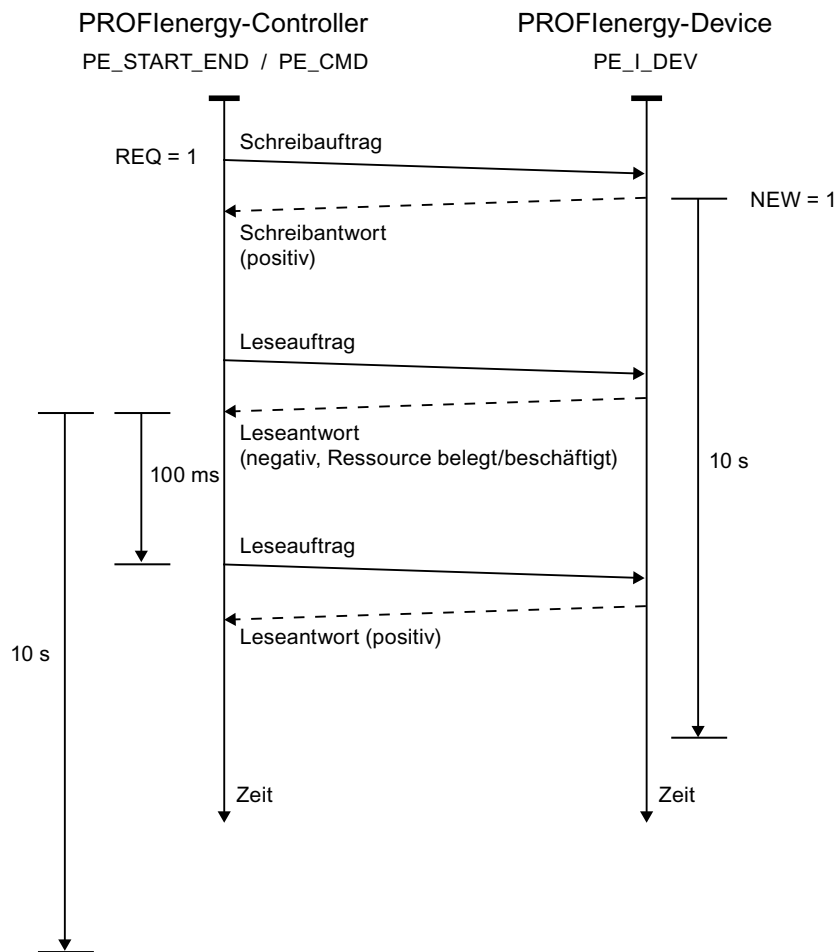
- Den einzelnen PE-Kommandos, die mit der Anweisung an die PE-Entity übertragen werden, sind definierte "Service\_Request\_IDs" zugeordnet. Die Service\_Request\_IDs 01...05 und 16 werden am Parameter CMD zugewiesen.
- Über den Parameter CMD\_MODIFIER werden die beiden PE-Kommandos 04 (Query\_Modes) und 16 (Query\_Measurement) näher spezifiziert.
- Für einzelne PE-Kommandos werden am Parameter CMD\_PARA zusätzliche Werte übergeben (siehe Beschreibung der einzelnen PE-Kommandos). Der Parameter CMD\_PARA\_LEN legt die Länge der Daten am Parameter CMD\_PARA fest.

Die Kommandos werden ohne Plausibilitätstest übertragen. Die Antwortdaten der PE-Entity werden in dem durch VARIANT-Zeiger adressierten Datenbereich RESPONSE\_DATA abgelegt (die Inhalte der Antworttelegramme sind in den Beschreibungen des jeweiligen PE-Kommandos enthalten).

### Schreib- und Leseaufträge der Anweisung "PE\_CMD"

Die Anweisung "PE\_CMD" versendet mittels "WRREC (Seite 2394)" einen PROFenergy-Befehl als Schreibauftrag an die PE-Entity. Anschließend wartet "PE\_CMD" auf die Quittung der PE-Entity. Dazu wird alle 100 Millisekunden der Quittungsdatensatz mit der Anweisung "RDREC (Seite 2392)" gelesen. Solange keine Quittung der PE-Entity eingetroffen ist, wird der Leseauftrag für 10 Sekunden im Abstand von jeweils 100 Millisekunden wiederholt. Die Antwortdaten der PE-Entity werden ebenfalls mit der Anweisung "RDREC (Seite 2392)" gelesen.

Die folgende Abbildung zeigt das Ablaufdiagramm der Schreib- und Leseaufträge:



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE\_CMD":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-----------|-------------|--------------|------------------------------|--|
| REQ       | Input       | BOOL         | E, A, M, D, L oder Konstante | Startet die Übertragung des PE-Kommandos bei positiver Flanke.   |
| ID        | Input       | HW_SUBMODULE | E, A, M, D, L oder Konstante | Adresse der PE-Entity<br>Die Adresse kann bei einem PROFINET-IO Device aus der Hardware-Konfiguration übernommen werden.   |
| CMD       | Input       | BYTE         | E, A, M, D, L oder Konstante | Service-Request-ID des PROFINET-Kommandos entsprechend dem PROFINET-Profil (siehe "Parameter CMD und CMD_MODIFIER").<br>Nach Erweiterungen des PROFINET-Profiles sind weitere Service-Request-IDs möglich. |

| Parameter     | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|---------------|-------------|----------|------------------------------|---|
| CMD_MODIFIER  | Input       | BYTE     | E, A, M, D, L oder Konstante | PROFenergy-Unterkommando (nur bei CMD=3 oder CMD=16, siehe "Parameter CMD und CMD_MODIFIER")<br>Nach Erweiterungen des PROFenergy-Profiles sind weitere Unterkommandos möglich.   |
| CMD_PARA      | Input       | VARIANT  | E, A, M, D, L                | Parameter für die PE-Kommandos: <ul style="list-style-type: none"> <li>• Get mode: PE_mode_ID</li> <li>• Get measurement values: List of Measurement_Ids</li> </ul> Es wird der komplette Service Data Request eingetragen. |
| CMD_PARA_LEN  | Input       | INT      | E, A, M, D, L                | Tatsächliche Länge der Parameter zu dem Kommando (<= Länge in CMD_PARA, wird von der Anweisung geprüft).  |
| RESPONSE_DATA | InOut       | VARIANT  | E, A, M, D, L                | PROFenergy-Information<br>Je nach Kommando gesamtes Antwort-Telegramm einschließlich Block-Header.<br>Hinweis: Wenn der Puffer zu klein ist, werden nur die Anzahl Bytes eingetragen, wie im VARIANT-Zeiger angegeben.      |
| VALID         | Output      | BOOL     | E, A, M, D, L                | Kommando erfolgreich abgesetzt.   |
| BUSY          | Output      | BOOL     | E, A, M, D, L                | Kommandobearbeitung läuft noch.   |
| ERROR         | Output      | BOOL     | E, A, M, D, L                | Bei der Bearbeitung trat ein Fehler auf.  |
| STATUS        | Output      | DWORD    | E, A, M, D, L                | Bausteinstatus / Fehlernummer (siehe "Parameter STATUS"):   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter CMD und CMD\_MODIFIER

| CMD | CMD_MODIFIER | PROFenergy-Kommando                                 | Beschreibung   |
|-----|--------------|---|--|
| 01  | 0            | Start_Pause (Seite 2479)                            | Starten des Energiesparmodus oder Wechsel in einen anderen Energiesparmodus. |
| 02  | 0            | End_Pause (Seite 2479)                              | Beenden des Energiesparmodus.  |
| 03  | 1            | Query_Modes - List energy saving Modes (Seite 2480) | Ausgeben der unterstützten Energiesparmodi.                                  |
|     | 2            | Query_Modes - Get Mode (Seite 2481)                 | Attribute zu dem momentan aktivierten Energiesparmodus ausgeben.             |
| 04  | 0            | PEM_Status (Seite 2483)                             | Status des Energiesparmodus abfragen.  |
| 05  | 0            | PE_Identify (Seite 2485)                            | Anzahl und die Beschreibung der unterstützten PE-Kommandos auslesen.         |



| CMD | CMD_MODIFIER | PROFenergy-Kommando   | Beschreibung  |
|-----|--------------|---|---|
| 16  | 1            | Query_Measurement -<br>Get_Measurement_List<br>(Seite 2486)   | Auflistung der durch die PE-Entity unterstützten Messwerte. |
|     | 2            | Query_Measurement -<br>Get_Measurement_Values<br>(Seite 2488) | Ausgabe der Messwerte der PE-Entity.                        |

## Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wird er als ARRAY[1...4] of BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

| Feldelement | Name         | Bedeutung   |
|-------------|--------------|---|
| STATUS[1]   | Function_Num | Fehlerursache <ul style="list-style-type: none"> <li>• B#16#00: Kein Fehler</li> <li>• B#16#DE: Fehler beim Lesen des Datensatzes</li> <li>• B#16#DF: Fehler beim Schreiben des Datensatzes</li> <li>• B#16#C0: Fehlermeldung der intern verwendeten Kommunikationsanweisungen "RDREC (Seite 2392)" und "WRREC (Seite 2394)".</li> </ul>  |
| STATUS[2]   | Error Decode | Ort der Fehlerkennung <ul style="list-style-type: none"> <li>• <b>80</b>: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch</li> <li>• <b>FE</b>: DP/PNIO-Profil - PROFenergy-spezifischer Fehler</li> </ul>  |
| STATUS[3]   | Error_Code_1 | Fehlerkennung <ul style="list-style-type: none"> <li>• Bei Error Decode = <b>80</b>: <ul style="list-style-type: none"> <li>– <b>81</b>: Längenkonflikt bei den Parametern CMD_PARA und CMD_PARA_LEN bzw. maximale Datensatzlänge (4095 Byte) überschritten.</li> <li>– <b>82-8F</b>: weitere Fehlermeldungen (reserviert)</li> </ul> </li> <li>• Bei Error Decode = <b>FE</b>: <ul style="list-style-type: none"> <li>– <b>01</b>: "Service Request ID" ungültig</li> <li>– <b>02</b>: Falsche "Request_Reference"</li> <li>– <b>03</b>: "Modifier" ungültig</li> <li>– <b>04</b>: "Data Structure Identifier RQ" ungültig</li> <li>– <b>05</b>: "Data Structure Identifier RS" ungültig</li> <li>– <b>06</b>: "PE energy-saving modes" werden nicht unterstützt</li> <li>– <b>07</b>: "Response" ist zu lange (die maximal übertragbare Länge wurde überschritten)</li> <li>– <b>08</b>: "Count" ungültig</li> <li>– <b>50</b>: Es steht kein passender Energiesparmodus (energy mode) zur Verfügung.</li> <li>– <b>51</b>: Angegebener Zeitwert wird nicht unterstützt.</li> <li>– <b>52</b>: "PE_Mode_ID" ungültig</li> </ul> </li> </ul> |
| STATUS[4]   | Error_Code_2 | herstellerspezifische Erweiterung der Fehlerkennung   |

**Hinweis**

**Fehlermeldungen der Anweisungen RDREC und WRREC**

Die Anweisungen "PE\_CMD" verwenden zur Kommunikation die Anweisungen "WRREC (Seite 2394)" und "RDREC (Seite 2392)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "WRREC (Seite 2394)" und "RDREC (Seite 2392)" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 2402).

**Siehe auch**

Beschreibung PROFlenergy (Seite 2465)

**PE\_DS3\_Write\_ET200S: Schaltverhalten der Powermodule einstellen**

**Beschreibung**

Die Anweisung "PE\_DS3\_Write\_ET200S" sendet grundsätzliche Einstellungen zum Schaltverhalten der Powermodule an die ET 200S. Mit der Anweisung "PE\_DS3\_Write\_ET200S" kann das Schaltverhalten für bis zu 8 Slots der ET 200S (z.B. für Powermodule) festgelegt werden.

**Hinweis**

Diese Anweisung ist nicht Bestandteil des PROFlenergy-Profiles sondern ergänzt SIMATIC-spezifische Funktionen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "PE\_DS3\_Write\_ET200S":

| Parameter | Deklaration | Datentyp         | Speicherbereich              | Beschreibung  |
|-----------|-------------|------------------|------------------------------|---|
| ENABLE    | Input       | BOOL             | E, A, M, D, L oder Konstante | Eine positive Flanke stößt die Übertragung des Datensatzes an. Der Datensatz muss nach Spannung AUS/EIN erneut übertragen werden. |
| ID        | Input       | HW_SUBMODU<br>LE | E, A, M, D, L oder Konstante | Adresse der ET 200S<br>Die Adresse kann aus der Hardware-Konfiguration übernommen werden.   |
| SLOT_NO_X | Input       | INT              | E, A, M, D, L oder Konstante | Steckplatznummer des X. schaltbaren Powermoduls.  |

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| FUNC_X    | Input       | INT      | E, A, M, D, L oder Konstante | Funktion der Baugruppe auf diesem Steckplatz. Über den Parameter FUNC_X legen Sie das Schaltverhaltens des PM-E (Powermodul der ET 200S) fest: <ul style="list-style-type: none"> <li>• FALSE: <ul style="list-style-type: none"> <li>- Bei "PAUSE_START": <ul style="list-style-type: none"> <li>-Kein Einfluss auf PM-E</li> <li>-PM-E bleibt eingeschaltet</li> </ul> </li> <li>- Bei "PAUSE_STOP": <ul style="list-style-type: none"> <li>-Schaltet PM_E wieder ein</li> </ul> </li> </ul> </li> <li>• TRUE: <ul style="list-style-type: none"> <li>- Bei "PAUSE_START": <ul style="list-style-type: none"> <li>-Schaltet PM_E aus</li> </ul> </li> <li>- Bei "PAUSE_STOP": <ul style="list-style-type: none"> <li>-Schaltet PM-E wieder ein</li> </ul> </li> </ul> </li> </ul> |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | Übertragung noch nicht abgeschlossen.   |
| DONE      | Output      | BOOL     | E, A, M, D, L                | Übertragung ohne Fehler beendet.  |
| ERROR     | Output      | BOOL     | E, A, M, D, L                | Übertragung mit Fehler beendet.   |
| STATUS    | Output      | DWORD    | E, A, M, D, L                | Fehlernummer (siehe Parameter STATUS der Anweisung "PE_Start_End (Seite 2467)")   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Siehe auch

Beschreibung PROFIenergy (Seite 2465)

## PROFIenergy-Kommandos

### Aufbau der Antworttelegramme

#### Aufbau des Antworttelegramms entsprechend dem PROFIenergy-Profil

Die folgende Tabelle zeigt den grundsätzlichen Aufbau des Antworttelegramms gemäß des PROFIenergy-Profiles. Das Antworttelegramm besteht aus einem allgemeinen Teil (Header) und einem spezifischen Teil (Service Data Response). Der Inhalt des spezifischen Teils des Antworttelegramms ist jeweils in der Beschreibung des entsprechenden PROFIenergy-Kommandos enthalten.

| Blockdefinition | Attribute        | Wert               | Datentyp | Beschreibung   |
|-----------------|------------------|--------------------|----------|--|
| BlockHeader     | BlockType        | 801 <sub>hex</sub> | WORD     |  |
|                 | BlockLength      |                    | WORD     | Anzahl der Bytes ohne Berücksichtigung der Felder BlockType und BlockLength. |
|                 | BlockVersionHigh | 1 <sub>hex</sub>   | BYTE     |  |
|                 | BlockVersionLow  | 0 <sub>hex</sub>   | BYTE     |  |

| Blockdefinition         | Attribute                    | Wert                                   | Datentyp | Beschreibung   |
|-------------------------|------------------------------|--|----------|--|
| Response Header         | Service_Request_ID           | 1 <sub>hex</sub> bis FF <sub>hex</sub> | BYTE     | ID des durchgeführten PE-Kommandos. Die ID des durch die PE-Entity bearbeiteten PE-Kommandos wird im Antworttelegramm zurückgesendet: <ul style="list-style-type: none"> <li>• 01: Start_Pause</li> <li>• 02: End_Pause</li> <li>• 03: Query_Modes</li> <li>• 04: PEM_Status</li> <li>• 05: PE_Identify</li> <li>• 06 bis 09: reserviert</li> <li>• 16: Query_Measurement</li> <li>• 11 bis CF: reserviert</li> <li>• D0 bis FF: herstellerspezifisch</li> </ul> |
|                         | Request_Reference            | 1 <sub>hex</sub> bis FF <sub>hex</sub> | BYTE     | Eindeutige Nummer zur Identifikation des Anfrage/Antwort-Paars (wird vom Server in der Antwort zurückgeliefert).   |
| Service Header Response | Status                       | 1 <sub>hex</sub> bis FF <sub>hex</sub> | BYTE     | Information, ob das PE-Kommando ausgeführt wurde: <ul style="list-style-type: none"> <li>• 00: reserviert</li> <li>• 01: abgeschlossen</li> <li>• 02: mit Fehler abgeschlossen</li> <li>• 03: Daten unvollständig</li> <li>• 04 bis CF: reserviert</li> <li>• D0 bis FF: Abhängig von der Service_Request_ID</li> </ul>  |
|                         | Data_Structure_Identifier_RS | 1 <sub>hex</sub> bis FF <sub>hex</sub> | BYTE     | <ul style="list-style-type: none"> <li>• 00: reserviert</li> <li>• 01 bis FF: Datenstruktur abhängig von der Service_Request_ID</li> <li>• 0xFF - Fehler</li> </ul>  |
| Service Data Response   |                              |  |          | Abhängig von der Service-Request-ID: <ul style="list-style-type: none"> <li>• Zu den Service-Request-IDs, siehe Parameter CMD und CMD_MODIFIER der Anweisung "PE_CMD (Seite 2471)".</li> <li>• Die spezifischen Inhalte des Antworttelegramms sind jeweils in der Beschreibung des PE-Kommandos enthalten (siehe z. B. Kommando "Start_Pause (Seite 2479)").</li> </ul>  |

## PE-Kommando "Start\_Pause"

### Beschreibung

Mit dem PE-Kommando "Start\_Pause" starten Sie den Energiesparmodus. Das Kommando Start\_Pause kann verwendet werden um:

- Die PE-Entity vom Status "Betriebsbereit" (PE\_ready\_to\_operate) in einen Energiesparmodus (PE\_energy\_saving\_mode) zu schalten.
- Die PE-Entity von einem Energiesparmodus in einen anderen zu schalten. Der Energieverbrauch kann durch das Umschalten des Energiesparmodus steigen oder sinken.

### Aufruf des PE-Kommandos "Start\_Pause"

Der Aufruf des Kommandos "Start\_Pause" mit der Anweisung "PE\_CMD (Seite 2471)" erfolgt mit den folgenden Parametern:

| Parameter    | Wert    | Beschreibung  |
|--------------|---------|---|
| CMD          | 1       | Aufruf des PE-Kommandos "Start_Pause".                                      |
| CMD_MODIFIER | 0       | Bei Kommando "Start_Pause" keine weitere Spezifikation des Kommandoaufrufs. |
| CMD_PARA_LEN | 4       | Länge des Parameters CMD_PARA von 4 Byte.                                   |
| CMD_PARA     | VARIANT | VARIANT-Zeiger auf Wert für "Pause_Time" (TIME).                            |

### Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE\_DATA (siehe Anweisung "PE\_CMD (Seite 2471)") referenzierten Datenbaustein geschrieben:

| Attribut   | Wert      | Datentyp | Beschreibung                               |
|------------|-----------|----------|--|
| PE_Mode_ID | 1 bis 255 | BYTE     | Identifikationsnummer des Energiesparmodus |
| Reserviert | 0         | BYTE     | -  |

## PE-Kommando "End\_Pause"

### Beschreibung

Mit dem PE-Kommando "End\_Pause" beenden Sie den Energiesparmodus der PE-Entity.

### Aufruf des PE-Kommandos "End\_Pause"

Der Aufruf des Kommandos "End\_Pause" mit der Anweisung "PE\_CMD (Seite 2471)" erfolgt mit den folgenden Parametern:

| Parameter    | Wert       | Beschreibung  |
|--------------|------------|---|
| CMD          | 2          | Aufruf des PE-Kommandos "End_Pause".                                      |
| CMD_MODIFIER | 0          | Bei Kommando "End_Pause" keine weitere Spezifikation des Kommandoaufrufs. |
| CMD_PARA_LEN | 0          | Länge des Parameters CMD_PARA von 0 Byte.                                 |
| CMD_PARA     | irrelevant | -   |

### Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE\_DATA (siehe "PE\_CMD (Seite 2471)") referenzierten Datenbaustein geschrieben:

| Attribut        | Wert | Datentyp | Beschreibung  |
|-----------------|------|----------|---|
| Time_to_operate | -    | DWORD    | Erwartete Zeit, um in den Modus "PE_ready_to_operate" umzuschalten. |

### PE-Kommando "Query\_modes" - "List\_Energy\_Saving\_Modes"

#### Beschreibung

Mit dem PE-Kommando "Query\_modes" und dem Unterkommando (Modifier) "List\_Energy\_Saving\_Modes" geben Sie alle durch die PE-Entity unterstützten Energiesparmodi (PE\_Mode\_ID) aus.

Das Abfrageergebnis wird in Form des Antworttelegramms in den über den Parameter RESPONSE\_DATA referenzierten Datenbaustein geschrieben.

### Aufruf des PE-Kommandos "Query\_modes" - "List\_Energy\_Saving\_Modes"

Der Aufruf des Kommandos "List\_Energy\_Saving\_Modes" mit der Anweisung "PE\_CMD (Seite 2471)" erfolgt mit den folgenden Parametern:

| Parameter    | Wert       | Beschreibung   |
|--------------|------------|--|
| CMD          | 3          | Aufruf des PE-Kommandos "Query_modes".   |
| CMD_MODIFIER | 1          | Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "List_Energy_Saving_Modes" zur Ausgabe der Anzahl und Typen der unterstützten Energiesparmodi. |
| CMD_PARA_LEN | 0          | Länge des Parameters CMD_PARA von 0 Byte.  |
| CMD_PARA     | irrelevant | -  |

## Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE\_DATA (siehe "PE\_CMD (Seite 2471)") referenzierten Datenbaustein geschrieben:

| Attribut              | Wert | Datentyp            | Beschreibung   |
|-----------------------|------|---------------------|--|
| Number_of_PE_Mode_IDs | 1    | BYTE                | Die Anzahl der Energiesparmodi.  |
| PE_Mode_IDs           | -    | Array [...] of BYTE | Array mit den IDs der unterstützten Energiesparmodi. Die Bedeutung der einzelnen IDs ist abhängig von der PE-Entity. |

## PE-Kommando "Query\_modes" - "Get\_Mode"

### Beschreibung

Mit dem PE-Kommando "Query\_modes" und dem Unterkommando (Modifier) "Get\_Mode" geben Sie die Attribute zu dem momentan aktivierten Energiesparmodus aus.

## Aufruf des PE-Kommandos "Query\_modes" - "Get\_Mode"

Der Aufruf des Kommandos mit dem mit der Anweisung "PE\_CMD" erfolgt mit den folgenden Parametern:

| Parameter    | Wert    | Beschreibung  |
|--------------|---------|---|
| CMD          | 3       | Aufruf des PE-Kommandos "Query_modes"   |
| CMD_MODIFIER | 2       | Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "Get_Mode" zur Ausgabe des Status des momentan aktivierten Modus. |
| CMD_PARA_LEN | 1       | Länge des Parameters CMD_PARA von 1 Byte.   |
| CMD_PARA     | VARIANT | VARIANT -Zeiger auf den Wert für PE_MODE_ID.  |

### Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE\_DATA (siehe "PE\_CMD (Seite 2471)") referenzierten Datenbaustein geschrieben:

| Attribut                             | Wert   | Datentyp | Beschreibung   |
|--------------------------------------|--|----------|--|
| PE_Mode_ID                           | <ul style="list-style-type: none"> <li>• 0<br/>Modus "PE_power_off"</li> <li>• 1...254<br/>Energiesparmodus der PE-Entity (herstellerspezifisch)</li> <li>• 255<br/>Modus "PE_ready_to_operate"</li> </ul>   | BYTE     | ID des momentan aktiven Energiesparmodus.  |
| PE_Mode_Attributes                   | Bit 0: <ul style="list-style-type: none"> <li>• = 0: Nur statische Energieverbrauchs- und Zeitwerte verfügbar.</li> <li>• = 1: Dynamische Energieverbrauchs- und Zeitwerte verfügbar.</li> </ul> Bit 1 bis 7: <ul style="list-style-type: none"> <li>• Reserviert</li> </ul> | BYTE     |  |
| Time_min_Pause <sup>1</sup>          | Zeitdifferenz ohne Datum   | DWORD    | Minimale Pausenzeit für den PE-Modus. Die minimale Pausenzeit ist die Summe aus den Werten der folgenden Attribute: <ul style="list-style-type: none"> <li>• Time_to_Pause</li> <li>• Time_to_operate</li> <li>• Time_min_length_of_stay</li> </ul> Siehe Beschreibung "Parameter PAUSE_TIME" der Anweisung "PE_START_END: Energiesparmodus starten und beenden (Seite 2467)". |
| Time_to_Pause <sup>1</sup>           | Zeitdifferenz ohne Datum   | DWORD    | Ausschaltzeit: Zeitdauer vom Aufruf des Energiesparmodus bis zum Start des Energiesparmodus (Übergangsdauer von PE_ready_to_operate zu PE_energy_saving_mode). Die Ausschaltzeit ist abhängig von der PE-Entity.   |
| Time_to_operate <sup>1</sup>         | Zeitdifferenz ohne Datum   | DWORD    | Einschaltzeit: Zeitdauer des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate). Die Zeitdauer wird bei der Ausgabe durch die PE-Entity dynamisch berechnet.  |
| Time_min_length_of_stay <sup>1</sup> |  | DWORD    | Mindestdauer, in der der Energiesparmodus in der PE-Entity aktiv ist.  |



| Attribut                                   | Wert | Datentyp | Beschreibung   |
|--|------|----------|--|
| Time_max_length_of_stay <sup>1</sup>       |      | DWORD    | Maximale Dauer, in der der Energiesparmodus in der PE-Entity aktiv ist.  |
| Mode_Power_Consumption <sup>2</sup>        |      | REAL     | Stromverbrauch der PE-Entity im aktivierten Energiesparmodus.<br>Einheit: kW   |
| Energy_Consumption_to_pause <sup>2</sup>   |      | REAL     | Energieverbrauch der PE-Entity während des Übergangs vom Bereitschaftsmodus (PE_ready_to_operate) zum Energiesparmodus (PE_energy_saving_mode)<br>Einheit: kWh |
| Energy_Consumption_to_operate <sup>2</sup> |      | REAL     | Energieverbrauch der PE-Entity während des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate)<br>Einheit: kWh |

<sup>1</sup> Ist die Zeitdauer unendlich, wird 0xFFFFFFFF ausgegeben. Ist die Zeitdauer null, wird "0" ausgegeben.

<sup>2</sup> Sind die Daten zu Energie- und Stromverbrauch durch die PE-Entity nicht definiert, wird als Wert "0,0" ausgegeben.

## PE-Kommando "PEM\_Status"

### Beschreibung

Mit dem PE-Kommando "PEM\_Status" fragen Sie den Status eines momentan aktiven Energiesparmodus einer PE-Entity ab.

### Aufruf des PE-Kommandos "PEM\_Status"

Der Aufruf des Kommandos "PEM\_Status" mit der Anweisung "PE\_CMD" erfolgt mit den folgenden Parametern:

| Parameter    | Wert       | Beschreibung   |
|--------------|------------|--|
| CMD          | 4          | Aufruf des PE-Kommandos "PEM_Status".                                      |
| CMD_MODIFIER | 0          | Bei Kommando "PEM_Status" keine weitere Spezifikation des Kommandoaufrufs. |
| CMD_PARA_LEN | 0          | Länge des Parameters CMD_PARA von 0 Byte.                                  |
| CMD_PARA     | irrelevant | -  |

### Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE\_DATA (siehe "PE\_CMD (Seite 2471)") referenzierten Datenbaustein geschrieben:

| Attribut                          | Wert   | Datentyp | Beschreibung   |
|-----------------------------------|--|----------|--|
| PE_Mode_ID_Source                 | <ul style="list-style-type: none"> <li>• 0<br/>Modus "PE_power_off"</li> <li>• 1 bis 254<br/>Energiesparmodus der PE-Entity (herstellerspezifisch)</li> <li>• 255<br/>Modus "PE_ready_to_operate"</li> </ul> | BYTE     | Modus, in dem sich die PE-Entity vor dem Absetzen eines PE-Kommandos befindet.   |
| PE_Mode_ID_Destination            | <ul style="list-style-type: none"> <li>• 0<br/>Modus "PE_power_off"</li> <li>• 1 bis 254<br/>Energiesparmodus der PE-Entity (herstellerspezifisch)</li> <li>• 255<br/>Modus "PE_ready_to_operate"</li> </ul> | BYTE     | Modus, in dem sich die PE-Entity nach der Durchführung eines PE-Kommandos befindet.  |
| Time_to_operate                   | Zeitdifferenz ohne Datum.  | DWORD    | Einschaltzeit: Zeitdauer des Übergangs vom Energiesparmodus (PE_energy_saving_mode) zum Bereitschaftsmodus (PE_ready_to_operate).<br>Die Zeitdauer wird bei der Ausgabe durch die PE-Entity dynamisch berechnet. |
| Remaining_time_to_destination     | Zeitdifferenz ohne Datum.  | DWORD    | Verbleibende Zeit zur Umschaltung in den anderen Modus.  |
| Mode_Power_Consumption            |  | REAL     | Stromverbrauch der PE-Entity im aktivierten Energiesparmodus.<br>Einheit: kW   |
| Energy_Consumption_to_Destination |  | REAL     | Energieverbrauch für momentane PE-Transition<br>Einheit: kWh   |
| Energy_Consumption_to_operate     |  | REAL     | Energieverbrauch der PE-Entity während des Übergangs vom Energiesparmodus (PE_energy-saving mode) zum Bereitschaftsmodus (PE_ready_to_operate)<br>Einheit: kWh   |

## PE-Kommando "PE\_Identify"

### Beschreibung

Mit dem PE-Kommando "PE\_Identify" lesen sie die Anzahl und die Beschreibung der durch die PE-Entity unterstützten PE-Kommandos aus. Welche und wie viele Kommandos unterstützt werden, ist abhängig von der PE-Entity. Da PE\_Identify selbst ein PE-Kommando ist, werden bei einer positiven Antwort mindestens drei unterstützte PE-Kommandos ausgegeben: Start\_Pause, End\_Pause und PE\_Identify.

### Aufruf des PE-Kommandos "PE\_Identify"

Der Aufruf des Kommandos "PE\_Identify" mit der Anweisung "PE\_CMD (Seite 2471)" erfolgt mit den folgenden Parametern:

| Parameter    | Wert       | Beschreibung  |
|--------------|------------|---|
| CMD          | 5          | Aufruf des Kommandos "PE_Identify".   |
| CMD_MODIFIER | 0          | Bei Kommando "PE_Identify" keine weitere Spezifikation des Kommandoaufrufs. |
| CMD_PARA_LEN | 0          | Länge des Parameters CMD_PARA von 0 Byte.                                   |
| CMD_PARA     | irrelevant | -   |

### Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE\_DATA (siehe "PE\_CMD (Seite 2471)") referenzierten Datenbaustein geschrieben:

| Attribut           | Wert | Datentyp | Beschreibung   |
|--------------------|------|----------|--|
| Count <sup>1</sup> | 6    | BYTE     | Anzahl der unterstützten PROFlenergy-Kommandos         |
| Start_Pause        | 1    | BYTE     | Erstes unterstütztes PE-Kommando (Service_Request_ID)  |
| End_Pause          | 2    | BYTE     | ...  |
| Query_Modes        | 3    | BYTE     | ...  |
| PEM_Status         | 4    | BYTE     | ...  |
| PE_Identify        | 5    | BYTE     | ...  |
| Query_Measurement  | 16   | BYTE     | Letztes unterstütztes PE-Kommando (Service_Request_ID) |

<sup>1</sup> Die Anzahl der unterstützten Kommandos ist herstellerspezifisch und hängt von der verwendeten PE-Entity ab. Die angegebenen Werte sind ein Beispiel für ein Antworttelegramm, wenn alle 6 PE-Kommandos unterstützt werden.

## PE-Kommando "Query\_Measurement" - "Get\_Measurement\_list"

### Beschreibung

Mit dem PE-Kommando "Query\_Measurement" und dem Unterkommando (Modifier) "Get\_measurement\_list" führen Sie eine Abfrage durch, welche Messwerte durch die PE-Entity unterstützt werden. Die unterstützten Messwerte werden als Liste in dem über den Parameter RESPONSE\_DATA referenzierten Datenbaustein hinterlegt.

### Aufruf des PE-Kommandos "Query\_Measurement" - "Get\_Measurement\_list"

Der Aufruf des Kommandos mit der Anweisung "PE\_CMD (Seite 2471)" erfolgt mit den folgenden Parametern:

| Parameter    | Wert       | Beschreibung  |
|--------------|------------|---|
| CMD          | 16         | Aufruf des Kommandos "Query_Measurement"  |
| CMD_MODIFIER | 1          | Spezifikation des Kommandoaufrufs: Auswahl des Unterkommandos "Get_Measurement_List" zur Ausgabe einer Liste der unterstützten Messwerte. |
| CMD_PARA_LEN | 0          | Länge des Parameters CMD_PARA von 0 Byte.   |
| CMD_PARA     | irrelevant | -   |

### Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der PE-Entity werden in den am Parameter RESPONSE\_DATA (siehe "PE\_CMD (Seite 2471)") referenzierten Datenbaustein geschrieben:

| Attribut        | Wert | Datentyp | Beschreibung   |
|-----------------|------|----------|--|
| Count           | -    | BYTE     | Anzahl der Measurement-IDs   |
| reserved        | -    | BYTE     |  |
| ...             |      |          |  |
| Measurement_ID  | -    | WORD     | Erste unterstützte Measurement_ID. Die Measurement_ID ist herstellerspezifisch. Weitere Informationen entnehmen Sie dem Handbuch der jeweiligen PE-Entity.   |
| Accuracy_Domain | -    | BYTE     | Siehe Tabelle "Genauigkeitsbereiche".  |
| Accuracy_Class  | -    | BYTE     | Siehe Tabellen "Genauigkeitsklassen".  |
| Range           | -    | REAL     | Spezifiziert den Skalen-Endwert für den Messwert (nur bei Genauigkeitsbereich 1). Das Attribut Range verwendet die gleiche Einheit wie über das Attribut Measurement_ID definiert (für jede Measurement_ID wird nur eine Einheit verwendet). |
| ...             |      |          |  |
| Measurement_ID  | -    | WORD     | Letzte unterstützte Measurement_ID   |
| Accuracy_Domain | -    | BYTE     | Siehe Tabelle "Genauigkeitsbereiche".  |

| Attribut       | Wert | Datentyp | Beschreibung  |
|----------------|------|----------|---|
| Accuracy_Class | -    | BYTE     | Siehe Tabellen "Genauigkeitsklassen".   |
| Range          | -    | REAL     | Spezifiziert den Skalenendwert für den Messwert (nur bei Genauigkeitsbereich 1). Das Attribut Range verwendet die gleiche Einheit wie über das Attribut Measurement_ID definiert (für jede Measurement_ID wird nur eine Einheit verwendet). |

## Genauigkeitsbereiche

| Genauigkeitsbereich | Beschreibung  |
|---------------------|---|
| 0                   | Reserviert  |
| 1                   | Die Genauigkeitsabweichung wird in Prozent des Skalenendwerts angegeben. Der Prozentsatz der möglichen Abweichung ist in Genauigkeitsklassen unterteilt (siehe Tabelle: Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2).  |
| 2                   | Die Genauigkeitsabweichung wird in Prozent des aktuellen Messwerts angegeben. Der Prozentsatz der möglichen Abweichung ist in Genauigkeitsklassen unterteilt (siehe Tabelle: Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2).   |
| 3                   | Die Messgenauigkeit erfolgt nach der Norm IEC 61557-12.<br>Die Funktions-Leistungsklassen für Geräte zur Leistungs-Messung und Überwachung (PMD) ohne externe Sensoren und System-Leistungsklassen für PMD mit externen Sensoren sind in der entsprechend der Tabelle "Genauigkeitsklassen des Genauigkeitsbereichs 3" codiert. |
| 4                   | Die Angabe der Genauigkeit erfolgt entsprechend der Norm EN 50470-3, Kapitel 8 (siehe auch Tabelle: Genauigkeitsklassen des Genauigkeitsbereichs 4).  |

## Genauigkeitsklassen

Tabelle 9-54 Genauigkeitsklassen der Genauigkeitsbereiche 1 und 2

| Genauigkeitsklasse | 0          | 1     | 2     | 3     | 4    | 5    | 6    | 7  | 8    |
|--------------------|------------|-------|-------|-------|------|------|------|----|------|
| Bedeutung          | reserviert | 0,01% | 0,02% | 0,05% | 0,1% | 0,2% | 0,5% | 1% | 1,5% |

| Genauigkeitsklasse | 9  | 10   | 11 | 12 | 13  | 14  | 15   | >15             |
|--------------------|----|------|----|----|-----|-----|------|-----------------|
| Bedeutung          | 2% | 2,5% | 3% | 5% | 10% | 20% | >20% | Nicht definiert |

Tabelle 9-55 Genauigkeitsklassen des Genauigkeitsbereichs 3

| Genauigkeitsklasse | 0          | 1    | 2    | 3   | 4   | 5   | 6 | 7   | 8 |
|--------------------|------------|------|------|-----|-----|-----|---|-----|---|
| Bedeutung          | reserviert | 0,02 | 0,05 | 0,1 | 0,2 | 0,5 | 1 | 1,5 | 2 |

| Genauigkeitsklasse | 9   | 10 | 11 | 12 | 13 | 14  | >13             |
|--------------------|-----|----|----|----|----|-----|-----------------|
| Bedeutung          | 2,5 | 3  | 5  | 10 | 20 | 20% | Nicht definiert |

Tabelle 9-56 Genauigkeitsklassen des Genauigkeitsbereichs 4

| Genauigkeitsklasse | 0          | 1   | 2   | 3   | 4   | 5   | 6   | >7              |
|--------------------|------------|-----|-----|-----|-----|-----|-----|-----------------|
| Bedeutung          | reserviert | 0,5 | 1,0 | 1,5 | 2,0 | 2,5 | 3,0 | Nicht definiert |

## PE-Kommando "Query\_Measurement" - "Get\_Measurement\_values"

### Beschreibung

Mit dem PE-Kommando "Query\_Measurement" und dem Unterkommando (Modifier) "Get\_measurement\_values" geben Sie die von der PE-Entity unterstützten Messwerte aus. Die Messwerte werden als Liste in dem über den Parameter RESPONSE\_DATA referenzierten Datenbaustein hinterlegt.

### Aufruf des PE-Kommandos "Query\_Measurement" - "Get\_Measurement\_values"

Der Aufruf des Kommandos mit der Anweisung "PE\_CMD (Seite 2471)" erfolgt mit den folgenden Parametern:

| Parameter    | Wert    | Beschreibung  |
|--------------|---------|---|
| CMD          | 16      | Aufruf des Kommandos "Query_Measurement"  |
| CMD_MODIFIER | 2       | Spezifikation des Kommandoaufrufs: Auswahl des Kommandos "Get_Measurement_Values" zur Ausgabe einer Liste der unterstützten Messwerte.                                    |
| CMD_PARA_LEN | 0       | Abhängig von der Anzahl der Messwerte. Die Länge des Parameters ergibt sich aus dem Attribut count und der Summe der Längen der Attribute für die übertragenen Messwerte. |
| CMD_PARA     | VARIANT | VARIANT-Zeiger auf Datenstruktur mit Auflistung der abzufragenden Messwerte (siehe "Parameter CMD_PARA").   |

### Parameter CMD\_PARA

Die Struktur, die über den VARIANT-Zeiger am Parameter CMD\_PARA angegeben wird, muss den folgenden Aufbau haben:

| Attribut       | Wert | Datentyp | Beschreibung                           |
|----------------|------|----------|--|
| Count          | -    | BYTE     | Anzahl der Messwerte (Measurement-IDs) |
| reserved       | 0    | BYTE     | Nicht verwendet                        |
| Measurement_ID | -    | WORD     | Erster abgefragter Messwert            |
| ...            |      |          |  |
| Measurement_ID | -    | WORD     | Letzter abgefragter Messwert           |

## Antworttelegramm (Service Data Response)

Die folgenden Daten des Antworttelegramms der der PE-Entity werden in den am Parameter RESPONSE\_DATA (siehe "PE\_CMD (Seite 2471)") referenzierten Datenbaustein geschrieben:

| Attribut                      | Wert             | Datentyp | Beschreibung   |
|-------------------------------|------------------|----------|--|
| Count <sup>1</sup>            | -                | BYTE     | Anzahl der Messwerte (Measurement-IDs)   |
| reserved                      | 0                | BYTE     | Nicht verwendet  |
|                               |                  |          |  |
| Length_of_Structure           | 2 bis 65535      | WORD     | Länge der Struktur in Bytes.   |
| Measurement_Data_Structure_ID | 1 = simple value | BYTE     | Definiert die folgende Struktur.   |
| Measurement_ID                | 0 bis 65535      | WORD     | ID des unterstützten Messwertes.   |
| Status_of_Measurement_Value   | 1 bis 3          | BYTE     | Status des Messwerts: <ul style="list-style-type: none"> <li>• 1: Gültig</li> <li>• 2: Nicht unterstützt</li> <li>• 3: Nicht gültig</li> </ul> |
| Transmission_Data_Type        | -                | REAL     |  |
| End_of_demand                 | -                | TOD      | Optionale Zeitstempelung mit dem Datentyp TimeOfDay.   |
| ...                           |                  |          |  |
| Length_of_Structure           | -                | WORD     | Länge der Struktur in Bytes.   |
| Measurement_Data_Structure_ID | -                | BYTE     | Definiert die folgende Struktur.   |
| Measurement_ID                | -                | WORD     | ID des unterstützten Messwertes.   |
| Status_of_Measurement_Value   | -                | BYTE     | Status des Messwerts: <ul style="list-style-type: none"> <li>• 1: Gültig</li> <li>• 2: Nicht unterstützt</li> <li>• 3: Nicht gültig</li> </ul> |
| Transmission_Data_Type        | -                | REAL     |  |
| End_of_demand                 | -                | TOD      | Optionale Zeitstempelung mit dem Datentyp TimeOfDay.   |

<sup>1</sup> Wenn die Datenlänge der abgefragten Messwerte die Größe der PDU (Protocol Data Unit) der Protokollschicht übersteigt, werden die Daten unvollständig übertragen und nur die unterstützten Messwerte ausgegeben.

## iDevice / iSlave

### PE\_I\_DEV: PROFlenergy-Kommandos im iDevice steuern

#### Beschreibung

Die Anweisung "PE\_I\_DEV" wird zur Abwicklung des PROFlenergy-Profiles im intelligenten IO-Device (iDevice) verwendet. Was in einem normalen PROFlenergy-fähigen IO-Device, z. B.

der ET 200S, von der Firmware erledigt wird, übernehmen im iDevice die Anweisung "PE\_I\_DEV" und die zugehörigen Hilfsbausteine:

- Die Anweisung "PE\_I\_DEV" wird zyklisch vom Anwenderprogramm des iDevice aufgerufen und empfängt alle PROFIenergy-Kommandos.
- Die Generierung der PROFIenergy-Antwort erfolgt durch Parametrierung eines Hilfsbausteins. Die Reaktion in der Pause ist frei programmierbar. Die Antwortdaten müssen innerhalb von 10 Sekunden bereitgestellt werden, da sonst am Parameter STATUS der Anweisung im IO-Controller der "Stateconflict 0x80B5" auftritt.

Zur Verwendung der Anweisung sind keine expliziten Kenntnisse des PROFIenergy-Standards erforderlich.

### PROFIenergy-Hilfsbausteine (PE-Hilfsbausteine)

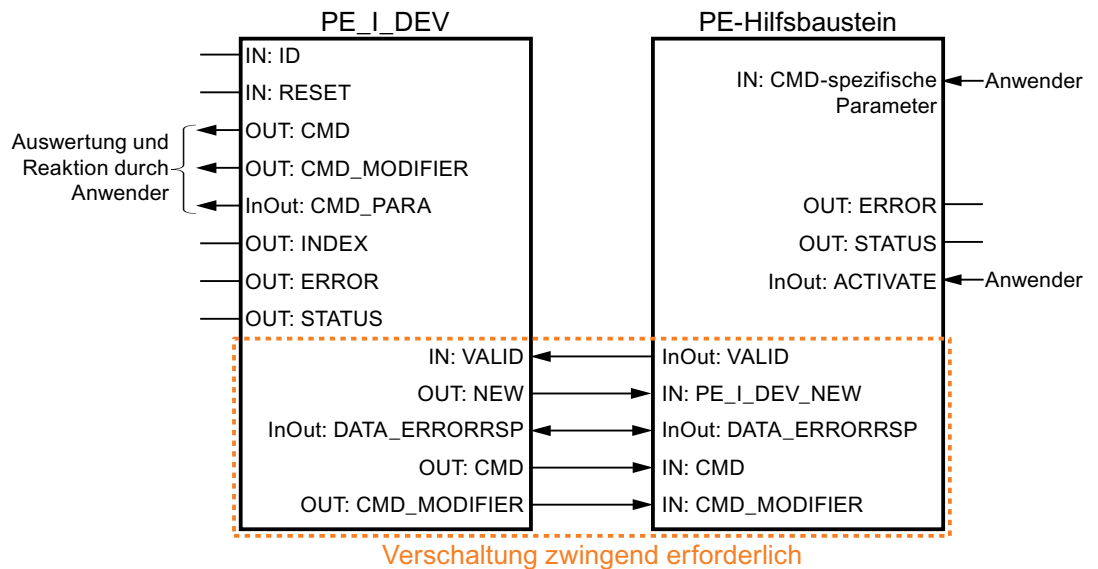
Über die PE-Hilfsbausteine generieren Sie das Antworttelegramm. Hierzu geben Sie die Antwortdaten (im Klartext) an den Eingangsparametern des jeweiligen Bausteins an.

- Für jedes PROFIenergy-Kommando gibt es einen entsprechenden Hilfsbaustein für eine positive Antwort:
  - PE-Kommando "Start\_Pause": PE\_Start\_RSP (Seite 2495)
  - PE-Kommando "End\_Pause": PE\_End\_RSP (Seite 2496)
  - PE-Kommando "Query\_modes" - "List\_Energy\_Saving\_Modes": PE\_List\_Modes\_RSP (Seite 2497)
  - PE-Kommando "Query\_modes" - "Get\_Mode": PE\_Get\_Mode\_RSP (Seite 2499)
  - PE-Kommando "PEM\_Status": PE\_PEM\_Status\_RSP (Seite 2501)
  - PE-Kommando "PE\_Identify": PE\_Identify\_RSP (Seite 2502)
  - PE-Kommando "Query\_Measurement" - "Get\_Measurement\_list": PE\_Measurement\_List\_RSP (Seite 2504)
  - PE-Kommando "Query\_Measurement" - "Get\_Measurement\_values": PE\_Measurement\_Value\_RSP (Seite 2505)
- Unabhängig vom PROFIenergy-Kommando gibt es zusätzlich einen gemeinsamen Hilfsbaustein für eine negative Antwort (siehe PE\_Error\_RSP (Seite 2493)).



### Verschaltung der Hilfsbausteine

Die Anweisung "PE\_I\_DEV" und die Hilfsbausteine sind aufeinander abgestimmt. Die Parameter werden teilweise einfach verschaltet. Die folgende Grafik zeigt, welche der Parameter verschaltet werden müssen.



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PE\_I\_DEV":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| RESET     | Input       | BOOL     | E, A, M, D, L oder Konstante | Rücksetzen der Anweisung.   |
| ID        | Input       | DWORD    | E, A, M, D, L                | Adresse des iDevice<br>Die Adresse kann aus der Hardware-Konfiguration übernommen werden.   |
| VALID     | Input       | BOOL     | E, A, M, D, L                | Die Antwortdaten für den PROFienergy-Controller stehen bereit und können gesendet werden.   |
| CMD_PARA  | Output      | VARIANT  | E, A, M, D, L                | Parameter für: <ul style="list-style-type: none"> <li>• Get mode: PE_mode_ID</li> <li>• Get measurement values: Liste der Measurement_IDs (Liste der IDs der zu lesenden Variablen; es können entweder eine einzelne Variable oder auch mehrere Variablen auf einmal gelesen werden.)</li> </ul> Maximale Länge: 234 Byte |

| Parameter         | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-------------------|-------------|----------|------------------------------|--|
| DATA_ERRO<br>RRSP | InOut       | VARIANT  | E, A, M, D, L                | Zeiger auf den Datenbereich, der die Quittungsdaten für den PROFIenergy-Controller enthält.<br><br>Dieser muss mit dem Zeiger übereinstimmen, der auch bei den Hilfsbausteinen verwendet wird.   |
| INDEX             | Output      | INT      | E, A, M, D, L oder Konstante | Datensatznummer des PROFIenergy Records (0x80A0 fest)  |
| CMD               | Output      | INT      | E, A, M, D, L oder Konstante | Service-Request-ID des PROFIenergy-Kommandos entsprechend dem PROFIenergy-Profil (siehe "Parameter CMD und CMD_MODIFIER").<br><br>Nach Erweiterungen des PROFIenergy-Profiles sind weitere PE- Kommandos (Service-Request-IDs) möglich.  |
| CMD_<br>MODIFIER  | Output      | INT      | E, A, M, D, L oder Konstante | PROFIenergy-Unterkommando:<br><ul style="list-style-type: none"> <li>• Nur bei CMD=3 oder CMD=16, siehe "Parameter CMD und CMD_MODIFIER".</li> <li>• Bei allen anderen Kommandos: "0".</li> </ul> Nach Erweiterungen des PROFIenergy-Profiles sind weitere Unterkommandos möglich. |
| NEW               | Output      | BOOL     | E, A, M, D, L                | Neue Daten vom PROFIenergy-Controller verfügbar.   |
| ERROR             | Output      | BOOL     | E, A, M, D, L                | Befehl mit Fehler beendet.   |
| STATUS            | Output      | DWORD    | E, A, M, D, L                | Fehlerinformation (siehe "Parameter STATUS").  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter CMD und CMD\_MODIFIER

| CMD | CMD_<br>MODIFIER | PROFIenergy-Kommando                       | Beschreibung   |
|-----|------------------|--|--|
| 01  | 0                | Start_Pause                                | Starten des Energiesparmodus oder Wechsel in einen anderen Energiesparmodus. |
| 02  | 0                | End_Pause                                  | Beenden des Energiesparmodus.  |
| 03  | 1                | Query_Modes - List energy saving Modes     | Ausgeben der unterstützten Energiesparmodi.                                  |
|     | 2                | Query_Modes - Get Mode                     | Attribute zu dem momentan aktivierten Energiesparmodus ausgeben.             |
| 04  | 0                | PEM_Status                                 | Status des Energiesparmodus abfragen.  |
| 05  | 0                | PE_Identify                                | Anzahl und die Beschreibung der unterstützten PE-Kommandos auslesen.         |
| 16  | 1                | Query_Measurement - Get_Measurement_List   | Auflistung der durch die der PE-Entity unterstützten Messwerte.              |
|     | 2                | Query_Measurement - Get_Measurement_Values | Ausgabe der Messwerte der der PE-Entity.                                     |

## Parameter STATUS

Am Ausgangsparameter STATUS werden Fehlerinformationen ausgegeben. Wird er als ARRAY[1...4] of BYTE interpretiert, hat die Fehlerinformation folgende Struktur:

| Feldelement | Name         | Bedeutung  |
|-------------|--------------|--|
| STATUS[1]   | Function_Num | Fehlerursache <ul style="list-style-type: none"> <li>• B#16#00: Kein Fehler</li> <li>• B#16#DE: Fehler beim Lesen des Datensatzes</li> <li>• B#16#DF: Fehler beim Schreiben des Datensatzes</li> <li>• B#16#C0: Fehlermeldung durch die Anweisung "PE_I_DEV" oder der intern verwendeten Kommunikationsanweisungen "RDREC (Seite 2392)" und "WRREC (Seite 2394)".</li> </ul> |
| STATUS[2]   | Error Decode | Ort der Fehlerkennung <ul style="list-style-type: none"> <li>• 80: DPV1-Fehler nach Norm IEC 61158-6 oder anweisungsspezifisch</li> </ul>  |
| STATUS[3]   | Error_Code_1 | Fehlerkennung (bei Error Decode = 80): <ul style="list-style-type: none"> <li>• B1: Write length error (Fehler bei der Schreiblänge oder unzureichende Längenangabe über den Datentyp VARIANT).</li> </ul>   |
| STATUS[4]   | Error_Code_2 | Bei PROFINET-Fehler: Ausgabe der Fehlermeldung des IO-Controllers. Ist kein PROFINET-Fehler aufgetreten ist der Wert in STATUS[4] = "0".   |

### Hinweis

#### Fehlermeldungen der Anweisungen RDREC und WRREC

Die Anweisung "PE\_I\_DEV" verwendet zur Kommunikation die Anweisungen "WRREC (Seite 2394)" und "RDREC (Seite 2392)". Fehlermeldungen dieser Anweisungen werden entsprechend in den Feldelementen STATUS[1] bis STATUS[4] ausgegeben.

Zur Bedeutung der Fehlercodes der Anweisungen "WRREC (Seite 2394)" und "RDREC (Seite 2392)" siehe Beschreibung des entsprechenden Parameters STATUS (Seite 2402).

### Siehe auch

Beschreibung PROFIenergy (Seite 2465)

### Hilfsbausteine der Anweisung PE\_I\_DEV

#### PE\_Error\_RSP: Negative Antwort auf Kommando generieren

### Beschreibung

Der Hilfsbaustein "PE\_Error\_RSP" (Response with failure) generiert eine negative Antwort, falls das angeforderte Kommando generell oder temporär nicht unterstützt wird. Die Generierung der Antwort erfolgt unabhängig vom angeforderten Kommando.

**Parameter**

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_Error\_RSP":

| Parameter      | Deklaration | Datentyp | Anfangswert | Beschreibung   |
|----------------|-------------|----------|-------------|--|
| PE_I_DEV_NEW   | Input       | BOOL     | 0           | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.  |
| ERROR_CODE     | Input       | BYTE     | 0           | Fehlernummer   |
| ACTIVATE       | InOut       | BOOL     | 0           | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID          | InOut       | BOOL     | 1           | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.<br>Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFenergy-Controller stehen bereit und gesendet werden können.  |
| DATA_ERROR_RSP | InOut       | VARIANT  | 0           | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERROR_RSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFenergy Telegramm.<br>Mindestlänge: 244 Byte  |
| ERROR          | Output      | BOOL     | 0           | <ul style="list-style-type: none"> <li>• "0": Kein Fehler</li> <li>• "1": Fehler aufgetreten.</li> </ul>   |
| STATUS         | Output      | WORD     | 0           | <ul style="list-style-type: none"> <li>• "0": Kein Fehler</li> <li>• "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul>  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**PE\_Start\_RSP: Antwort auf Kommando zum Pausenstart generieren****Beschreibung**

Der Hilfsbaustein "PE\_Start\_RSP" (Start Pause) generiert die Antwort auf das PE-Kommando Start\_Pause (Seite 2479). Die Anweisung gibt zurück, welchen Energiesparzustand das Gerät einnimmt (Parameter PE\_MODE\_ID).

Wenn Sie bei verschiedenen langen Pausen unterschiedlich reagieren, können Sie das in der Rückmeldung über den eingenommenen Energiesparzustand zurückmelden (PE\_Mode\_ID = 1 für eine kurze Pause, PE\_Mode\_ID = 2 für eine längere Pause usw.).

**Parameter**

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_Start\_RSP":

| Parameter    | Deklaration | Datentyp | Anfangswert | Beschreibung  |
|--------------|-------------|----------|-------------|---|
| PE_I_DEV_NEW | Input       | BOOL     | 0           | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.   |
| CMD          | Input       | INT      | 0           | Service-Request-ID des PROFlenergy-Kommandos<br>Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.  |
| PE_MODE_ID   | Input       | BYTE     | 0           | PE-Modus, den der Prozess einnimmt  |
| ACTIVATE     | InOut       | BOOL     | 0           | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt.<br>Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID        | InOut       | BOOL     | 1           | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.<br>Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFlenergy-Controller stehen bereit und gesendet werden können.  |

| Parameter     | Deklaration | Datentyp | Anfangswert | Beschreibung   |
|---------------|-------------|----------|-------------|--|
| DATA_ERRORRSP | InOut       | VARIANT  | 0           | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERRORRSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFenergy Telegramm.<br>Mindestlänge: 244 Byte |
| ERROR         | Output      | BOOL     | 0           | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"1": Fehler aufgetreten.</li> </ul>   |
| STATUS        | Output      | WORD     | 0           | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul>  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### PE\_End\_RSP: Antwort auf Kommando zum Pausenende generieren

#### Beschreibung

Der Hilfsbaustein "PE\_End\_RSP" generiert die Antwort auf das PE-Kommando End\_Pause (Seite 2479). Als Antwort wird die benötigte Zeitdauer zurückgegeben, um von dem aktuellen Betriebszustand in den Betriebszustand "Ready\_To\_Operate" zu wechseln.

#### Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_End\_RSP":

| Parameter       | Deklaration | Datentyp | Anfangswert | Beschreibung  |
|-----------------|-------------|----------|-------------|---|
| PE_I_DEV_NEW    | Input       | BOOL     | 0           | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht. |
| CMD             | Input       | INT      | 0           | Service-Request-ID des PROFenergy-Kommandos<br>Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.                                 |
| Time_to_Operate | Input       | DWORD    | 0           | Zeitdauer, um vom aktuellen Betriebszustand nach "Ready_To_Operate" zu wechseln.  |

| Parameter      | Deklaration | Datentyp | Anfangswert | Beschreibung  |
|----------------|-------------|----------|-------------|---|
| ACTIVATE       | InOut       | BOOL     | 0           | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt.<br>Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID          | InOut       | BOOL     | 1           | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.<br>Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFenergy-Controller stehen bereit und gesendet werden können.   |
| DATA_ERROR_RSP | InOut       | VARIANT  | 0           | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERROR_RSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFenergy Telegramm.<br>Mindestlänge: 244 Byte   |
| ERROR          | Output      | BOOL     | 0           | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"1": Fehler aufgetreten.</li> </ul>  |
| STATUS         | Output      | WORD     | 0           | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul>   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## PE\_List\_Modes\_RSP: Abgefragte Engiesparmodi als Antwort generieren

### Beschreibung

Der Hilfsbaustein "PE\_List\_Modes\_RSP" generiert die Antwort auf das PE-Kommando List\_Energy\_Saving\_Modes (Seite 2480). Die generierte Antwort enthält die Anzahl und die IDs der unterstützten Energiesparzustände.

**Parameter**

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_List\_Modes\_RSP":

| Parameter             | Deklaration | Datentyp | Anfangswert | Beschreibung   |
|-----------------------|-------------|----------|-------------|--|
| PE_I_DEV_NEW          | Input       | BOOL     | 0           | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.  |
| CMD                   | Input       | INT      | 0           | Service-Request-ID des PROFIenergy-Kommandos<br>Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.   |
| CMD_MODIFIER          | Input       | INT      | 0           | PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.   |
| Number_of_PE_Mode_IDs | Input       | BYTE     | 0           | Anzahl der unterstützten Energiesparzustände. Zulässige Werte: 1 bis 254   |
| PE_MODE_ID            | Input       | VARIANT  | 0           | Zeigt auf den Bereich, in dem die IDs der unterstützten Energiesparzustände (PE_Mode_ID) hinterlegt sind. Zulässiger Bereich: 1 bis 254.   |
| ACTIVATE              | InOut       | BOOL     | 0           | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangsparameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt.<br>Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID                 | InOut       | BOOL     | 1           | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.<br>Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.   |
| DATA_ERROR_RSP        | InOut       | VARIANT  | 0           | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERROR_RSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm. Mindestlänge: 244 Byte  |



| Parameter | Deklaration | Datentyp | Anfangswert | Beschreibung  |
|-----------|-------------|----------|-------------|---|
| ERROR     | Output      | BOOL     | 0           | <ul style="list-style-type: none"> <li>• "0": Kein Fehler</li> <li>• "1": Fehler aufgetreten.</li> </ul>  |
| STATUS    | Output      | WORD     | 0           | <ul style="list-style-type: none"> <li>• "0": Kein Fehler</li> <li>• "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul> |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## PE\_Get\_Mode\_RSP: Abgefragte Energiedaten als Antwort generieren

### Beschreibung

Der Hilfsbaustein "PE\_Get\_Mode\_RSP" generiert die Antwort auf das Kommando Get\_Mode (Seite 2481). Die Antwort enthält die Zeiten und Leistungs- bzw. Energiedaten eines einzelnen Energiesparzustands.

### Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_Get\_Mode\_RSP":

| Parameter      | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|----------------|-------------|----------|-----------------|---|
| PE_I_DEV_NEW   | Input       | BOOL     | 0               | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.   |
| CMD            | Input       | INT      | 0               | Service-Request-ID des PROFlenergy-Kommandos<br>Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.  |
| CMD_MODIFIER   | Input       | INT      | 0               | PROFlenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.  |
| PE_Mode_ID     | Input       | BYTE     | E, A, M, D, L   | ID des momentan aktiven Energiesparmodus.   |
| Time_min_Pause | Input       | DWORD    | E, A, M, D, L   | Mindest-Pausendauer für diesen PE-Energiesparmodus.<br>Diese bildet die Summe der drei Parameter: <ul style="list-style-type: none"> <li>• Time_to_Pause</li> <li>• Time_to_operate</li> <li>• Time_min_length_of_stay</li> </ul> |

| Parameter                | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|--------------------------|-------------|----------|-----------------|--|
| Time_to_Pause            | Input       | DWORD    | E, A, M, D, L   | Zeitdauer von der Flanke am Parameter START (siehe "PE_I_DEV (Seite 2489)") bis zum Erreichen des angeforderten PE-Energiesparmodus.   |
| Time_to Operate          | Input       | DWORD    | E, A, M, D, L   | Maximale Einschaltdauer bis "PE_ready_to_operate". Der Parameter "Time_to_operate" kann direkt verwendet werden für entsprechende Berechnungen. Der Wert kann entweder ein statischer Maximalwert sein oder durch die PE-Entity dynamisch berechnet werden.  |
| Time_min_Lenght_of_stay  | Input       | DWORD    | E, A, M, D, L   | Minimale Zeitdauer, in welcher die PE-Entity in diesem PE-Modus verbleiben muss.   |
| Time_max_Lenght_of_stay  | Input       | DWORD    | E, A, M, D, L   | Maximale Zeitdauer, in welcher die PE-Entity in diesem PE-Modus verbleiben kann.   |
| Mode_Power_Consumption   | Input       | DWORD    | E, A, M, D, L   | Energieverbrauch im aktuellen PE-Modus in [kW].  |
| Energy_Consum_to_Pause   | Input       | DWORD    | E, A, M, D, L   | Energieverbrauch von "PE_ready_to_operate" bis zum aktuellen PE -Modus in [kWh].   |
| Energy_Consum_to_operate | Input       | DWORD    | E, A, M, D, L   | Energieverbrauch vom aktuellen PE-Modus bis "PE_ready_to_operate" in [kWh].  |
| ACTIVATE                 | InOut       | BOOL     | 0               | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt. Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID                    | InOut       | BOOL     | 1               | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFenergy-Controller stehen bereit und gesendet werden können.   |
| DATA_ERROR_RSP           | InOut       | VARIANT  | 0               | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERROR_RSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFenergy Telegramm. Mindestlänge: 244 Byte   |
| ERROR                    | Output      | BOOL     | 0               | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"1": Fehler aufgetreten.</li> </ul>   |
| STATUS                   | Output      | WORD     | 0               | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul>  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## PE\_PEM\_Status\_RSP: PEM-Status als Antwort generieren

### Beschreibung

Der Hilfsbaustein "PE\_PEM\_Status\_RSP" generiert die Antwort auf das Kommando PEM\_Status (Seite 2483).

### Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_PEM\_Status\_RSP":

| Parameter                                      | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|--|-------------|----------|-----------------|---|
| PE_I_DEV_NEW                                   | Input       | BOOL     | 0               | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.   |
| CMD  | Input       | INT      | 0               | Service-Request-ID des PROFlenergy-Kommandos<br>Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.  |
| PE_MODE_ID_Source                              | Input       | BYTE     | E, A, M, D, L   | Source und Destination für PEM_STATUS.<br>Werte:<br><ul style="list-style-type: none"> <li>• 0x00: PE_POWER_OFF</li> <li>• 0x01 – 0xFE: Herstellerspezifisch</li> <li>• 0xFF: PE_READY_TO_OPERATE</li> </ul>                                      |
| PE_MODE_ID_Destination                         | Input       | BYTE     | E, A, M, D, L   |   |
| Time_to_Operate <sup>1</sup>                   | Input       | DWORD    | E, A, M, D, L   | Maximale Einschaltdauer bis "PE_ready_to_operate".<br>"Time_to_operate" kann direkt verwendet werden für entsprechende Berechnungen. Der Wert kann entweder ein statischer Maximal-Wert sein oder durch die PE-Entity dynamisch berechnet werden. |
| Remaining_time_to_destination <sup>1</sup>     | Input       | DWORD    | E, A, M, D, L   | Optional: Verbleibende Zeit bis zum angeforderten PE-Modus. Dynamischer Wert oder statischer Maximal-Wert   |
| Mode_Power_Consumption <sup>2</sup>            | Input       | DWORD    | E, A, M, D, L   | Energieverbrauch im aktuellen PE-Modus in [kW].   |
| Energy_Consumption_to_Destination <sup>2</sup> | Input       | DWORD    | E, A, M, D, L   | Energieverbrauch bis zum angeforderten PE-Modus in [kWh].   |
| Energy_Consumption_to_operate <sup>2</sup>     | Input       | DWORD    | E, A, M, D, L   | Energieverbrauch vom aktuellen PE-Modus bis "PE_ready_to_operate" in [kWh].   |

| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|--|-------------|----------|-----------------|--|
| ACTIVATE   | InOut       | BOOL     | 0               | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangsparameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt.<br>Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID  | InOut       | BOOL     | 1               | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.<br>Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.   |
| DATA_ERROR_RSP   | InOut       | VARIANT  | 0               | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERROR_RSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm.<br>Mindestlänge: 244 Byte   |
| ERROR  | Output      | BOOL     | 0               | <ul style="list-style-type: none"> <li>• "0": Kein Fehler</li> <li>• "1": Fehler aufgetreten.</li> </ul>   |
| STATUS   | Output      | WORD     | 0               | <ul style="list-style-type: none"> <li>• "0": Kein Fehler</li> <li>• "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul>  |
| <p><sup>1</sup> Wenn die Zeitdauer unbegrenzt ist, kann der max. Wert "0xFFFFFFFF" angegeben werden. Wenn die Zeitdauer "Null" ist, kann "0x00" verwendet werden.</p> <p><sup>2</sup> Wenn ein Energieverbrauchswert nicht definiert ist, kann "0,0" angegeben werden.</p> |             |          |                 |  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## PE\_Identify\_RSP: Unterstützte PROFIenergy-Kommandos als Antwort generieren

### Beschreibung

Der Hilfsbaustein "PE\_Identify\_RSP" generiert die Antwort auf das Kommando PE\_Identify (Seite 2485). In der Antwort auf das Kommando geben Sie an, welche PROFIenergy-Kommandos unterstützt werden. Beachten Sie, dass PE\_IDENTIFY selbst ein PE-Kommando ist und in der Antwort entsprechend mit angegeben wird.

## Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_Identify\_RSP":

| Parameter         | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-------------------|-------------|----------|-----------------|---|
| PE_I_DEV_NEW      | Input       | BOOL     | 0               | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.   |
| CMD               | Input       | INT      | 0               | Service-Request-ID des PROFIenergy-Kommandos<br>Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.  |
| Start_Pause       | Input       | BOOL     | E, A, M, D, L   | Jeweils ein Parameter für jedes der entsprechenden PROFIenergy-Kommandos:<br><ul style="list-style-type: none"> <li>• 0: PE-Kommando wird nicht unterstützt</li> <li>• 1: PE-Kommando wird unterstützt</li> </ul>   |
| End_Pause         | Input       | BOOL     | E, A, M, D, L   |   |
| Query_Modes       | Input       | BOOL     | E, A, M, D, L   |   |
| PEM_Status        | Input       | BOOL     | E, A, M, D, L   |   |
| PE_Identify       | Input       | BOOL     | E, A, M, D, L   |   |
| Query_Measurement | Input       | BOOL     | E, A, M, D, L   |   |
| ACTIVATE          | InOut       | BOOL     | 0               | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt.<br>Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID             | InOut       | BOOL     | 1               | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.<br>Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.  |
| DATA_ERROR_RSP    | InOut       | VARIANT  | 0               | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERROR_RSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm.<br>Mindestlänge: 244 Byte  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| ERROR     | Output      | BOOL     | 0               | <ul style="list-style-type: none"> <li>• "0": Kein Fehler</li> <li>• "1": Fehler aufgetreten.</li> </ul>  |
| STATUS    | Output      | WORD     | 0               | <ul style="list-style-type: none"> <li>• "0": Kein Fehler</li> <li>• "0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul> |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### PE\_Measurement\_List\_RSP: Liste der unterstützten Messwerte als Antwort generieren

#### Beschreibung

Der Hilfsbaustein "PE\_Measurement\_List\_RSP" generiert die Antwort auf das Kommando Get\_measurement\_list (Seite 2486). In der Antwort geben Sie an, welche Messwerte (Measurement-IDs) unterstützt werden.

#### Parameter

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_Measurement\_List\_RSP":

| Parameter        | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|------------------|-------------|----------|-----------------|--|
| PE_I_DEV_NEW     | Input       | BOOL     | 0               | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.  |
| CMD              | Input       | INT      | 0               | Service-Request-ID des PROFIenergy-Kommandos<br>Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.                                 |
| CMD_MODIFIER     | Input       | INT      | 0               | PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.                   |
| Count            | Input       | BYTE     | E, A, M, D, L   | Anzahl der unterstützten Messwerte (Measurement-IDs)   |
| Measurement_List | Input       | VARIANT  | D               | Zeiger auf das Array mit den unterstützten Measurement_IDs.<br>Zum Aufbau des Arrays gemäß PROFIenergy Profil siehe: PE-Kommando "Query_Measurement" - "Get_Measurement_list" (Seite 2486) |

| Parameter      | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|----------------|-------------|----------|-----------------|---|
| ACTIVATE       | InOut       | BOOL     | 0               | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangs-Parameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt.<br>Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID          | InOut       | BOOL     | 1               | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.<br>Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFenergy-Controller stehen bereit und gesendet werden können.   |
| DATA_ERROR_RSP | InOut       | VARIANT  | 0               | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERROR_RSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFenergy Telegramm.<br>Mindestlänge: 244 Byte   |
| ERROR          | Output      | BOOL     | 0               | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"1": Fehler aufgetreten.</li> </ul>  |
| STATUS         | Output      | WORD     | 0               | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul>   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## PE\_Measurement\_Value\_RSP: Angeforderte Messwerte als Antwort generieren

### Beschreibung

Der Hilfsbaustein "PE\_Measurement\_Value\_RSP" generiert die Antwort auf das Kommando Get\_measurement\_values (Seite 2488). In der Antwort geben Sie die Werte der angeforderten Messwerte zurück.

**Parameter**

Die folgende Tabelle zeigt die Parameter des Hilfsbausteins "PE\_Measurement\_Value\_RSP":

| Parameter          | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|--------------------|-------------|----------|-----------------|--|
| PE_I_DEV_NEW       | Input       | BOOL     | 0               | Der Parameter muss mit dem Ausgangsparameter NEW der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden. Der Hilfsbaustein wird nur bearbeitet, wenn am Parameter der Wert "1" ansteht.  |
| CMD                | Input       | INT      | 0               | Service-Request-ID des PROFIenergy-Kommandos<br>Der Parameter muss mit dem Ausgangsparameter CMD der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.   |
| CMD_MODIFIER       | Input       | INT      | 0               | PROFIenergy-Unterkommando (Auswertung nur bei CMD=3 oder CMD=16). Der Parameter muss mit dem Ausgangsparameter CMD_MODIFIER der Anweisung "PE_I_DEV" verschaltet werden.   |
| Count              | Input       | BYTE     | E, A, M, D, L   | Anzahl der Messwerte (Measurement_Values).   |
| Measurement_Values | Input       | VARIANT  | D               | Zeiger auf das Array mit den Messwerten (Measurement_IDs).<br>Zum Aufbau des Arrays gemäß PROFIenergy Profil, siehe PE-Kommando "Query_Measurement" - "Get_Measurement_values" (Seite 2488)  |
| ACTIVATE           | InOut       | BOOL     | 0               | Bei einer positiven Flanke an dem Eingang ACTIVATE kopiert die Anweisung die Eingangsparameter in den Datenbereich DATA_ERROR_RSP. Der Parameter wird danach von der Anweisung zurückgesetzt.<br>Der Parameter muss innerhalb von 10 Sekunden gesetzt werden, nachdem am Parameter NEW der Anweisung "PE_I_DEV (Seite 2489)" eine positive Flanke erkannt wurde. |
| VALID              | InOut       | BOOL     | 1               | Der Parameter muss mit dem Eingang VALID der Anweisung "PE_I_DEV (Seite 2489)" verschaltet werden.<br>Der Parameter wird vom Hilfsbaustein gesetzt, wenn die Antwortdaten für den PROFIenergy-Controller stehen bereit und gesendet werden können.   |
| DATA_ERROR_RSP     | InOut       | VARIANT  | 0               | Zeiger auf dem Datenbereich, in dem die Antwortdaten abgelegt werden. Der Parameter ist identisch mit dem Zeiger bei DATA_ERROR_RSP der Anweisung "PE_I_DEV (Seite 2489)". Der adressierte Datenbereich enthält das gesamte PROFIenergy Telegramm.<br>Mindestlänge: 244 Byte   |



| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| ERROR     | Output      | BOOL     | 0               | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"1": Fehler aufgetreten.</li> </ul>  |
| STATUS    | Output      | WORD     | 0               | <ul style="list-style-type: none"> <li>"0": Kein Fehler</li> <li>"0x80B1": Fehler bei der VARIANT-Angabe, z. B. falscher Bereich</li> </ul> |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### 9.7.3.6 Baugruppenparametrierung

#### Datensätze schreiben und lesen

##### Prinzip

Es gibt Baugruppen, die über einen Systemdatenbereich verfügen, auf den Sie von Ihrem Programm aus nur schreibend zugreifen können. Dieser Bereich enthält Datensätze mit den Nummern 0 bis maximal 240, wobei nicht jede Baugruppe über alle Datensätze verfügt (siehe folgende Tabelle).

Darüber hinaus können Baugruppen auch einen Systemdatenbereich besitzen, auf den Sie von Ihrem Programm aus nur lesend zugreifen können. Dieser Bereich enthält Datensätze mit den Nummern 0 bis maximal 240, wobei nicht jede Baugruppe über alle Datensätze verfügt.

---

##### Hinweis

Es gibt Baugruppen, die über beide Systemdatenbereiche verfügen. Dabei handelt es sich um physikalisch unterschiedliche Bereiche, die lediglich die logische Aufteilung in Datensätze gemeinsam haben.

---

#### Nur beschreibbarer Systemdatenbereich

Die folgende Tabelle zeigt den Aufbau des nur beschreibbaren Systemdatenbereiches. Sie zeigt auf, wie groß die einzelnen Datensätze sein dürfen und mit welchen Anweisungen sie geschrieben werden können.

| Datensatz-Nummer | Inhalt        | Größe          | beschreibbar mit Anweisung                   |
|------------------|---------------|----------------|--|
| 0                | Parameter     | -              | WR_DPARM (Seite 2515)                        |
| 1                | Parameter     | -              | WR_DPARM (Seite 2515)                        |
| 2 bis 127        | Anwenderdaten | je ≤ 240 Bytes | WR_DPARM (Seite 2515)<br>WR_REC (Seite 2435) |
| 128 bis 240      | Parameter     | je ≤ 240 Bytes | WR_DPARM (Seite 2515)<br>WR_REC (Seite 2435) |

### Nur lesbarer Systemdatenbereich

Die folgende Tabelle zeigt den Aufbau des nur lesbaren Systemdatenbereiches. Sie zeigt auf, wie groß die einzelnen Datensätze sein dürfen und mit welchen Anweisungen sie gelesen werden können.

| Datensatz-Nummer | Inhalt   | Größe           | lesbar mit Anweisung |
|------------------|--|-----------------|----------------------|
| 0                | baugruppenspezifische Diagnosedaten (systemeinheitlich festgelegt) | 4 Bytes         | RD_REC (Seite 2430)  |
| 1                | kanalspezifische Diagnosedaten (incl. Datensatz 0)                 | 4 bis 220 Bytes | RD_REC (Seite 2430)  |
| 2 bis 127        | Anwenderdaten  | je ≤ 240 Bytes  | RD_REC (Seite 2430)  |
| 128 bis 240      | Diagnosedaten  | je ≤ 240 Bytes  | RD_REC (Seite 2430)  |

### Systemressourcen

Wenn Sie mehrere asynchron laufende Datensatzübertragungen kurz nacheinander anstoßen, so wird gewährleistet, dass alle Aufträge durchgeführt werden und keine gegenseitige Beeinflussung stattfindet.

Wird die Begrenzung der Systemressourcen erreicht, so erhalten Sie dies in RET\_VAL mitgeteilt. Der temporäre Fehlerfall kann durch eine Wiederholung des Auftrags behoben werden.

Die maximale Anzahl "gleichzeitig" aktiver Aufträge eines Anweisungstyps ist CPU-abhängig.

### RD\_DPAR: Datensatz einer Baugruppe lesen

#### Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer INDEX der adressierten Komponente aus den projektierten Systemdaten. Es kann sich dabei um eine zentral steckende Baugruppe oder eine dezentrale Komponente (PROFIBUS DP oder PROFINET IO) handeln.

Der Wert TRUE des Ausgangsparameters VALID zeigt an, dass der Datensatz erfolgreich in den Zielbereich RECORD übertragen wurde. In diesem Fall enthält der Ausgangsparameter LEN die Länge der gelesenen Daten in Bytes.

Falls bei der Datensatzübertragung ein Fehler auftrat, wird dies über den Ausgangsparameter ERROR angezeigt. Der Ausgangsparameter STATUS enthält in diesem Fall die Fehlerinformation.

## Arbeitsweise

Die Anweisung "RD\_DPAR" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten die Datensatzübertragung, indem Sie "RD\_DPAR" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und die Bytes 2 und 3 des Ausgangsparameters STATUS wird der Zustand des Auftrags angezeigt. Dabei entsprechen die Bytes 2 und 3 von STATUS dem Ausgangsparameter RET\_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

Die Datensatzübertragung ist abgeschlossen, wenn der Ausgangsparameter BUSY den Wert FALSE angenommen hat.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_DPAR":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Lesen   |
| LADDR     | Input       | HW_IO    | E, A, M, D, L oder Konstante | Identifikationsnummer der CPU oder des Interfaces.<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder des Interface in der Hardware-Konfiguration hinterlegt.   |
| INDEX     | Input       | INT      | E, A, M, D, L oder Konstante | Datensatznummer  |
| RECORD    | InOut       | VARIANT  | E, A, M, D, L                | Zielbereich für den gelesenen Datensatz  |
| VALID     | Output      | BOOL     | E, A, M, D, L                | Neuer Datensatz wurde empfangen und ist gültig   |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | BUSY = 1: Der Auftrag ist noch nicht beendet.  |
| ERROR     | Output      | BOOL     | E, A, M, D, L                | ERROR = 1: Beim Lesevorgang trat ein Fehler auf.   |
| STATUS    | Output      | DWORD    | E, A, M, D, L                | <ul style="list-style-type: none"> <li>Aufrufkennung (Bytes 2 und 3) bzw. Fehlercode</li> <li>Byte 1: B#16#00, falls kein Fehler. Andernfalls Funktionskennung aus DPV1-PDU: Im Fehlerfall bei Datensatz lesen B#16#DE, im Fehlerfall bei Datensatz schreiben B#16#DF. Falls kein DPV1-Protokollelement benutzt wird: B#16#C0.</li> <li>Byte 4: herstellerspezifische Erweiterung der Fehlerkennung</li> </ul> |
| LEN       | Output      | INT      | E, A, M, D, L                | Länge der gelesenen Datensatzinformation   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter STATUS**

| Fehlercode*<br>(W#16#...) | Erläuterung   | Einschränkung   |
|---------------------------|---|---|
| 0000                      | kein Fehler   | -   |
| 7000                      | Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.  | -   |
| 7001                      | Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".   | Dezentrale Peripherie                                     |
| 7002                      | Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".   | Dezentrale Peripherie                                     |
| 8090                      | Angegebene logische Basisadresse ungültig.  | -   |
| 8092                      | Am Parameter RECORD wurde ein anderer Datentyp als (Array of) Bitfolge oder Ganzzahl angegeben.   | -   |
| 8093                      | Für die über LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig.  | -   |
| 80A1                      | Negative Quittung beim Senden des Datensatzes zur Baugruppe (Baugruppe während des Sendens gezogen oder Baugruppe defekt).  | -   |
| 80A2                      | DP-Protokollfehler bei Layer 2, evtl. Hardware-/Schnittstellenfehler im DP-Slave.   | Dezentrale Peripherie                                     |
| 80A3                      | DP-Protokollfehler bei User Interface/User.   | Dezentrale Peripherie                                     |
| 80A4                      | Kommunikation am K-Bus gestört.   | Fehler tritt auf zwischen CPU und externer DP-Anschaltung |
| 80B0                      | Anweisung für Baugruppentyp nicht möglich, oder die Baugruppe kennt den Datensatz nicht.  | -   |
| 80B1                      | Die Länge des zu übertragenden Datensatzes ist falsch.  | -   |
| 80B2                      | Der projektierte Steckplatz ist nicht belegt.   | -   |
| 80B3                      | Ist-Baugruppentyp ungleich Soll-Baugruppentyp.  | -   |
| 80C1                      | Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.  | -   |
| 80C2                      | Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.   | -   |
| 80C3                      | Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.  | -   |
| 80C4                      | Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte ihren Aufbau auf elektrische Störquellen. | -   |
| 80C5                      | Dezentrale Peripherie nicht verfügbar oder deaktiviert.   | Dezentrale Peripherie                                     |
| 80C6                      | Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund).   | Dezentrale Peripherie                                     |
| 80D0                      | Für die Baugruppe ist kein Eintrag vorhanden.   | -   |
| 80D1                      | Die Datensatznummer ist für die Baugruppe nicht projektiert (Die Datensatznummern >241 werden abgewiesen.).   | -   |
| 80D2                      | Die Baugruppe ist laut Baugruppenkennung nicht parametrierbar.  | -   |
| 80D5                      | Der Datensatz ist statisch.   | -   |

| Fehlercode*<br>(W#16#...)   | Erläuterung   | Einschränkung |
|---|---|---------------|
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832) | -             |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |               |

## Siehe auch

RD\_DPARM: Datensatz aus projektierten Systemdaten lesen (Seite 2513)

## RD\_DPARA: Datensatz einer Baugruppe asynchron lesen

### Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM einer ausgewählten Baugruppe aus den projektierten Systemdaten. Der gelesene Datensatz wird in den durch den Parameter RECORD definierten Zielbereich eingetragen.

### Arbeitsweise

Die Anweisung "RD\_DPARA" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie die Anweisung mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET\_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_DPARA":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Lesen   |
| LADDR     | Input       | HW_IO    | E, A, M, D, L oder Konstante | Identifikationsnummer der CPU oder des Interfaces.<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder des Interface in der Hardware-Konfiguration hinterlegt. |
| RECNUM    | Input       | BYTE     | E, A, M, D, L oder Konstante | Datensatznummer (zulässige Werte: 0 bis 240)   |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| RET_VAL   | Return      | INT      | E, A, M, D, L   | <p>Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.</p> <p>Falls bei der Übertragung kein Fehler auftrat, sind die folgenden Fälle zu unterscheiden:</p> <ul style="list-style-type: none"> <li>RET_VAL enthält die Länge des tatsächlich gelesenen Datensatzes in Bytes, falls der Zielbereich größer ist als der gelesene Datensatz.</li> <li>RET_VAL enthält "0", falls die Länge des gelesenen Datensatzes gleich der Länge des Zielbereichs ist.</li> </ul> |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | BUSY = 1: Der Auftrag ist noch nicht beendet.  |
| RECORD    | Output      | VARIANT  | E, A, M, D, L   | Zielbereich für den gelesenen Datensatz.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode*<br>(W#16#...) | Erläuterung   | Einschränkung         |
|---------------------------|---|-----------------------|
| 0000                      | kein Fehler   | -                     |
| 7000                      | Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.  | -                     |
| 7001                      | Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".   | Dezentrale Peripherie |
| 7002                      | Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".   | Dezentrale Peripherie |
| 8090                      | Angegebene logische Basisadresse ungültig.  | -                     |
| 8092                      | Am Parameter RECORD wurde ein anderer Datentyp als (Array of) Bitfolge oder Ganzzahl angegeben.   | -                     |
| 8093                      | Für die über LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig.  | -                     |
| 80A1                      | Negative Quittung beim Senden des Datensatzes zur Baugruppe (Baugruppe während des Sendens gezogen oder Baugruppe defekt).                            | -                     |
| 80A2                      | DP-Protokollfehler bei Layer 2, evtl. Hardware-/Schnittstellenfehler im DP-Slave.   | Dezentrale Peripherie |
| 80A3                      | DP-Protokollfehler bei User Interface/User.   | Dezentrale Peripherie |
| 80A4                      | Kommunikation am PBUS+ gestört  | -                     |
| 80B0                      | Anweisung für Baugruppentyp nicht möglich, oder die Baugruppe kennt den Datensatz nicht.  | -                     |
| 80B1                      | Die Länge des zu übertragenden Datensatzes ist falsch. Bei RD_DPARM (Seite 2513): Die Länge des durch RECORD aufgespannten Zielbereichs ist zu klein. | -                     |
| 80B2                      | Der projektierte Steckplatz ist nicht belegt.   | -                     |
| 80B3                      | Ist-Baugruppentyp ungleich Soll-Baugruppentyp.  | -                     |

| Fehlercode*<br>(W#16#...)   | Erläuterung   | Einschränkung         |
|---|---|-----------------------|
| 80C1  | Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.  | -                     |
| 80C2  | Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.   | -                     |
| 80C3  | Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.  |                       |
| 80C4  | Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte ihren Aufbau auf elektrische Störquellen. | -                     |
| 80C5  | Dezentrale Peripherie nicht verfügbar oder deaktiviert.   | Dezentrale Peripherie |
| 80C6  | Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund).   | Dezentrale Peripherie |
| 80D0  | Für die Baugruppe ist kein Eintrag vorhanden.   | -                     |
| 80D1  | Die Datensatznummer ist für die Baugruppe nicht projektiert (Die Datensatznummern >241 werden abgewiesen.).   | -                     |
| 80D2  | Die Baugruppe ist laut Baugruppenkennung nicht parametrierbar.  | -                     |
| 80D5  | Der Datensatz ist statisch.   | -                     |
| allgemeine Fehlerinformationen  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   | -                     |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |                       |

## RD\_DPARM: Datensatz aus projektierten Systemdaten lesen

### Beschreibung

Mit der Anweisung lesen Sie den Datensatz mit der Nummer RECNUM der adressierten Baugruppe aus den projektierten Systemdaten. Der gelesene Datensatz wird in den durch den Parameter RECORD aufgespannten Zielbereich eingetragen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_DPARM":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| IOID      | Input       | BYTE     | E, A, M, D, L oder Konstante | Kennung des Adressbereichs:<br><ul style="list-style-type: none"> <li>• B#16#54 = Peripherie Eingang (PE)</li> <li>• B#16#55 = Peripherie Ausgang (PA)</li> </ul> Handelt es sich um eine Mischbaugruppe, ist die Bereichskennung der niedrigeren Adresse anzugeben. Bei gleichen Adressen ist B#16#54 anzugeben. |
| LADDR     | Input       | HW_IO    | E, A, M, D, L oder Konstante | Identifikationsnummer der CPU oder des Interfaces<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder des Interface in der Hardware-Konfiguration hinterlegt.   |
| RECNUM    | Input       | BYTE     | E, A, M, D, L oder Konstante | Datensatznummer (zulässige Werte: 0 bis 240)  |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Länge des gelesenen Datensatzes in Bytes, falls der gelesene Datensatz in den Zielbereich passt und bei der Übertragung kein Fehler auftrat.<br><br>Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.  |
| RECORD    | Output      | VARIANT  | E, A, M, D, L                | Zielbereich für den gelesenen Datensatz.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode* (W#16#...) | Erläuterung   | Einschränkung         |
|------------------------|---|-----------------------|
| 0000                   | kein Fehler   | -                     |
| 7000                   | Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.                | -                     |
| 7001                   | Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".               | Dezentrale Peripherie |
| 7002                   | Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1". | Dezentrale Peripherie |
| 8090                   | Angegebene logische Basisadresse ungültig.  | -                     |
| 8092                   | In der VARIANT-Referenz am Parameter RECORD ist eine Typangabe ungleich BYTE angegeben. | -                     |
| 8093                   | Für die über LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig.            | -                     |



| Fehlercode*<br>(W#16#...)    | Erläuterung   | Einschränkung |
|------------------------------|---|---------------|
| 80B1                         | Die Länge des durch RECORD aufgespannten Zielbereichs ist zu klein.   | -             |
| 80C3                         | Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.  | -             |
| 80D0                         | Für die Baugruppe ist kein Eintrag vorhanden.   | -             |
| 80D1                         | Die Datensatznummer ist für die Baugruppe nicht projektiert (Die Datensatznummern >241 werden abgewiesen.). | -             |
| 80D2                         | Die Baugruppe ist laut Baugruppenkennung nicht parametrierbar.  | -             |
| 80D5                         | Der Datensatz ist statisch.   | -             |
| allgemeine Fehlerinformation | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   | -             |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## WR\_DPARM: Datensatz übertragen

### Beschreibung

Mit der Anweisung "WR\_DPARM" übertragen Sie den Datensatz mit der Nummer RECNUM aus den Projektierungsdaten zur adressierten Baugruppe. Dabei ist es ohne Bedeutung, ob der Datensatz statisch oder dynamisch ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WR\_DPARM":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| REQ       | Input       | BOOL     | E, A, M, D, L                | REQ = 1: Anforderung zu Schreiben   |
| LADDR     | Input       | HW_IO    | E, A, M, D, L oder Konstante | Identifikationsnummer der CPU oder des Interfaces.<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder des Interface in der Hardware-Konfiguration hinterlegt.<br>Die möglichen DB-Nummernbänder finden Sie im jeweiligen Handbuch der Baugruppe. |
| RECNUM    | Input       | BYTE     | E, A, M, D, L oder Konstante | Datensatznummer   |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode.  |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | BUSY = 1: Der Schreibvorgang ist noch nicht beendet.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Erläuterung   | Einschränkung         |
|---|---|-----------------------|
| 0000  | kein Fehler   | -                     |
| 7000  | Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert 0.  | -                     |
| 7001  | Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".   | Dezentrale Peripherie |
| 7002  | Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".   | Dezentrale Peripherie |
| 8090  | Angegebene logische Basisadresse ungültig.  | -                     |
| 8093  | Für die über LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig.  | -                     |
| 80A1  | Negative Quittung beim Senden des Datensatzes zur Baugruppe (Baugruppe während des Sendens gezogen oder Baugruppe defekt).  | -                     |
| 80A2  | DP-Protokollfehler bei Layer 2, evtl. Hardware-/Schnittstellenfehler im DP-Slave.   | Dezentrale Peripherie |
| 80A3  | DP-Protokollfehler bei User Interface/User.   | Dezentrale Peripherie |
| 80A4  | Kommunikation am PBUS+ gestört  | -                     |
| 80B0  | Anweisung für Baugruppentyp nicht möglich, oder die Baugruppe kennt den Datensatz nicht.  | -                     |
| 80B1  | Die Länge des zu übertragenden Datensatzes ist falsch.  | -                     |
| 80B2  | Der projektierte Steckplatz ist nicht belegt.   | -                     |
| 80B3  | Ist-Baugruppentyp ungleich Soll-Baugruppentyp.  | -                     |
| 80C1  | Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.  | -                     |
| 80C2  | Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.   | -                     |
| 80C3  | Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.  | -                     |
| 80C4  | Interner temporärer Fehler. Auftrag konnte nicht ausgeführt werden. Wiederholen Sie den Auftrag. Bei häufigem Auftreten dieses Fehlers überprüfen Sie bitte ihren Aufbau auf elektrische Störquellen. | -                     |
| 80C5  | Dezentrale Peripherie nicht verfügbar oder deaktiviert.   | Dezentrale Peripherie |
| 80C6  | Datensatzübertragung wurde abgebrochen wegen Prioritätsklassenabbruchs (Wiederanlauf oder Hintergrund).   | Dezentrale Peripherie |
| 80D0  | Für die Baugruppe ist kein Eintrag vorhanden.   | -                     |
| 80D1  | Die Datensatznummer ist für die Baugruppe nicht projektiert (Die Datensatznummern >241 werden abgewiesen.).   | -                     |
| 80D2  | Die Baugruppe ist laut Baugruppenkennung nicht parametrierbar.  | -                     |
| 80D5  | Der Datensatz ist statisch.   | -                     |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   | -                     |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |                       |

### 9.7.3.7 Alarmer

#### ATTACH: OB zu Alarmereignis zuordnen

##### Beschreibung

Mit der Anweisung "ATTACH" ordnen Sie einem Ereignis einen Organisationsbaustein (OB) zu.

Am Parameter OB\_NR geben Sie die symbolische oder numerische Bezeichnung des Organisationsbausteins an, der dem am Parameter EVENT angegebenen Ereignis zugeordnet wird.

Wenn das Ereignis am Parameter EVENT nach der fehlerfreien Ausführung der Anweisung "ATTACH" auftritt, wird der Organisationsbaustein am Parameter OB\_NR aufgerufen und dessen Programm ausgeführt.

Mit dem Parameter ADD legen Sie fest, ob bisherige Zuordnungen des Organisationsbausteins zu anderen Ereignissen aufgehoben oder beibehalten werden. Bei einem Wert "0" am Parameter ADD werden die vorhandenen Zuordnungen durch die aktuelle Zuordnung ersetzt.

##### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ATTACH":

| Parameter | Deklaration | Datentyp          | Speicherbereich              | Beschreibung  |
|-----------|-------------|-------------------|------------------------------|---|
| OB_NR     | Input       | OB_ATT (INT)      | E, A, M, D, L oder Konstante | Organisationsbaustein (Nummern bis 32767 werden unterstützt.)   |
| EVENT     | Input       | EVENT_ATT (DWORD) | D, L oder Konstante          | Ereignis, z. B. Anlegen eines ProcessEvents (Ausfall eines Hardware Moduls; Ablesen der HW-ID (16#C0xyzz) und am Baustein abfragen)   |
| ADD       | Input       | BOOL              | E, A, M, D, L oder Konstante | Auswirkungen auf die bisherigen Zuordnungen: <ul style="list-style-type: none"> <li>• ADD=0 (Vorbelegung): Dieses Ereignis ersetzt alle bisherigen Ereigniszusordnungen für diesen OB.</li> <li>• ADD=1: Dieses Ereignis wird den bisherigen Ereigniszusordnungen für diesen OB hinzugefügt.</li> </ul> |
| RET_VAL   | Return      | INT               | E, A, M, D, L                | Status der Anweisung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung                 |
|---|------------------------------|
| 0   | Kein Fehler                  |
| 8090  | OB ist nicht vorhanden       |
| 8091  | OB hat den falschen Typ      |
| 8093  | Ereignis ist nicht vorhanden |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                              |

## DETACH: OB vom Alarmereignis trennen

### Beschreibung

Mit der Anweisung machen Sie zur Laufzeit die bestehende Zuordnung eines Organisationsbausteins zu einem oder mehreren Ereignissen rückgängig.

- Wenn Sie ein einzelnes Ereignis ausgewählt haben, wird die Zuordnung des OB zu diesem Ereignis rückgängig gemacht. Alle anderen momentan bestehenden Zuordnungen bleiben erhalten. Ein einzelnes Ereignis können Sie über die Klappliste des Operandenplatzhalters am Parameter EVENT auswählen.
- Wenn Sie kein Ereignis ausgewählt haben, werden alle momentan bestehenden Zuordnungen des Organisationsbausteins zu Ereignissen rückgängig gemacht.

Am Parameter OB\_NR geben Sie die symbolische oder numerische Bezeichnung des Organisationsbausteins ein, dessen Zuordnung zu dem am Parameter EVENT angegebenen Ereignis rückgängig gemacht wird.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DETACH":

| Parameter | Deklaration | Datentyp          | Speicherbereich              | Beschreibung  |
|-----------|-------------|-------------------|------------------------------|---|
| OB_NR     | Input       | OB_ATT (INT)      | E, A, M, D, L oder Konstante | Organisationsbaustein (Nummern bis 32768 werden unterstützt.) |
| EVENT     | Input       | EVENT_ATT (DWORD) | D, L oder Konstante          | Ereignis  |
| RET_VAL   | Return      | INT               | E, A, M, D, L                | Status der Anweisung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung                        |
|---|-------------------------------------|
| 0   | Kein Fehler                         |
| 1   | Keine Zuordnung vorhanden (Warnung) |
| 8090  | OB ist nicht vorhanden              |
| 8091  | OB ist vom falschen Typ             |
| 8093  | Ereignis nicht vorhanden            |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                                     |

## Weckalarm

### SET\_CINT: Weckalarm-Parameter setzen

#### Beschreibung

Mit der Anweisung setzen Sie die Parameter für einen Weckalarm-OB. Der Startzeitpunkt für einen Weckalarm-OB ergibt sich aus dem jeweiligen Zeitintervall des OBs und der Phasenverschiebung.

- Der Zeitintervall eines OBs ist der Zeitabstand, in dem der OB regelmäßig aufgerufen wird. Beträgt das Zeitintervall z. B. 100µs wird der OB während der Programmausführung alle 100µs aufgerufen.
- Die Phasenverschiebung ist ein Zeitintervall, um das der Aufruf eines Weckalarm-OBs verschoben wird. Die Phasenverschiebung können Sie verwenden, um niederpriorie Organisationsbausteine in einem genauen Zeitraster zu bearbeiten.

Existiert der OB nicht oder wird der verwendete Zeitintervall nicht unterstützt, wird eine entsprechende Fehlermeldung am Parameter RET\_VAL ausgegeben.

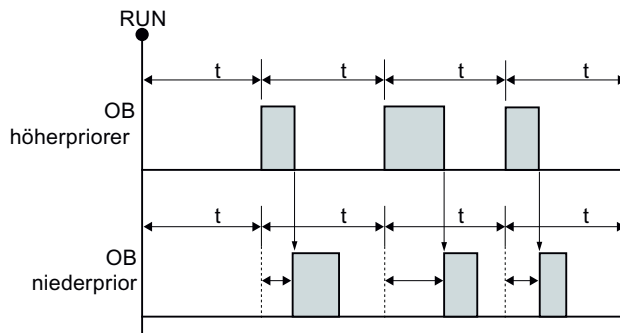
Ein Zeitintervall am Parameter CYCLE von "0" bedeutet, dass der OB nicht aufgerufen wird.

#### Arbeitsweise

Werden ein niederpriorer OB und ein höherpriorer OB im gleichen Zeitintervall aufgerufen, wird der niederpriorer OB erst aufgerufen, nachdem der höherpriorer OB abgearbeitet wurde.

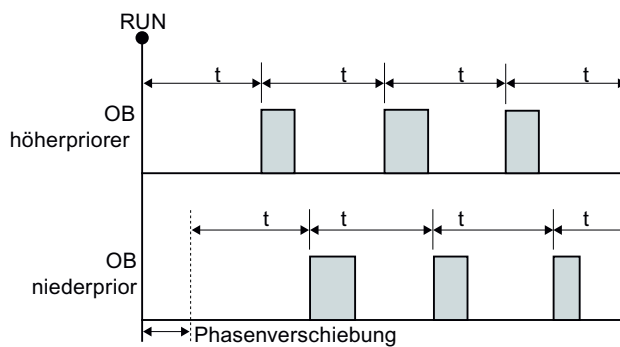
Der Aufrufzeitpunkt für den niederprioreren OB kann sich je nach Länge des Bearbeitungszeitraums des höherprioreren OBs verschieben.

OB-Aufruf ohne Phasenverschiebung



Wird für den niederprioreren OB eine Phasenverschiebung projektiert und ist die Phasenverschiebung größer als die jeweils aktuelle Bearbeitungszeit des höherprioreren OBs wird der Baustein in einem festen Zeitraster aufgerufen.

OB-Aufruf mit Phasenverschiebung



### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET\_CINT":

| Parameter | Deklaration | Datentyp  | Speicherbereich              | Beschreibung                   |
|-----------|-------------|-----------|------------------------------|--------------------------------|
| OB_NR     | Input       | OB_CYCLIC | E, A, M, D, L oder Konstante | OB-Nummer (<32768)             |
| CYCLE     | Input       | UDINT     | E, A, M, D, L oder Konstante | Zeitintervall in Microsekunden |
| PHASE     | Input       | UDINT     | E, A, M, D, L oder Konstante | Phasenverschiebung             |
| RET_VAL   | Return      | INT       | E, A, M, D, L                | Status der Anweisung           |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0   | Kein Fehler   |
| 8090  | OB ist nicht vorhanden oder OB ist vom falschen Typ |
| 8091  | Falscher Zeitintervall                              |
| 8092  | Falsche Phasenverschiebung                          |
| 80B2  | Kein Ereignis zu OB zugeordnet                      |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## QRY\_CINT: Weckalarm-Parameter abfragen

### Beschreibung

Über die Anweisung können Sie die aktuellen Parameter eines Weckalarm-OBs abfragen. Der Weckalarm-OB wird über den Parameter OB\_NR identifiziert.

Die Werte der abgefragten Parameter des Weckalarms entsprechen denen zum Zeitpunkt der Ausführung der Anweisung "QRY\_CINT".

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "QRY\_CINT":

| Parameter | Deklaration | Datentyp           | Speicherbereich                 | Beschreibung  |
|-----------|-------------|--------------------|---------------------------------|---|
| OB_NR     | Input       | OB_CYCLIC<br>(INT) | E, A, M, D, L oder<br>Konstante | OB-Nummer (<32768) oder symbolische Adressierung über den Namen des OBs (z.B. OB_MyOB)  |
| CYCLE     | Output      | UDINT              | E, A, M, D, L                   | Zeitintervall in Microsekunden  |
| PHASE     | Output      | UDINT              | E, A, M, D, L                   | Phasenverschiebung  |
| STATUS    | Output      | WORD               | E, A, M, D, L                   | Status des Weckalarms: <ul style="list-style-type: none"> <li>• Bit 0 bis Bit 4: siehe Parameter STATUS</li> <li>• Weitere Bits: immer "0"</li> </ul> |
| RET_VAL   | Return      | INT                | E, A, M, D, L                   | Status der Anweisung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Bit | Wert | Bedeutung                        |
|-----|------|----------------------------------|
| 0   | 0    | Die CPU befindet sich in RUN.    |
|     | 1    | Die CPU befindet sich im Anlauf. |

| Bit          | Wert | Bedeutung  |
|--------------|------|--|
| 1            | 0    | Der Weckalarm ist freigegeben.                     |
|              | 1    | Der Weckalarm ist verzögert.                       |
| 2            | 0    | Der Weckalarm ist nicht aktiviert oder abgelaufen. |
|              | 1    | Der Weckalarm ist aktiviert                        |
| 3            | 0    | -  |
|              | 1    | -  |
| 4            | 0    | Ein OB mit der angegebenen Nummer existiert nicht  |
|              | 1    | Ein OB mit der angegebenen Nummer existiert        |
| Weitere Bits |      | Immer "0"  |

### Parameter RET\_VAL

Tritt ein Fehler auf, wird am Parameter RET\_VAL der entsprechende Fehlercode angezeigt und der Parameter STATUS auf "0" gesetzt.

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 0   | Kein Fehler  |
| 8090  | OB ist nicht vorhanden oder OB ist vom falschen Typ. |
| 80B2  | Kein Ereignis zu OB zugeordnet                       |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Uhrzeitalarm

#### SET\_TINT: Uhrzeitalarm einstellen

#### Beschreibung

Mit der Anweisung können Sie Startdatum und -uhrzeit der Uhrzeitalarm-Organisationsbausteine einstellen. Bei der Startuhrzeit werden Ihre Sekunden- und Millisekundenangabe ignoriert und auf "0" gesetzt.



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET\_TINT":

| Parameter | Deklaration | Datentyp        | Speicherbereich                 | Beschreibung   |
|-----------|-------------|-----------------|---------------------------------|--|
| OB_NR     | Input       | OB_TOD<br>(INT) | E, A, M, D, L oder<br>Konstante | Nummer des OB, der zum Zeitpunkt SDT +<br>Vielfaches von PERIOD gestartet wird (OB 10<br>bis OB 17 und ab OB 123).   |
| SDT       | Input       | DT              | D, L                            | Startdatum und Startuhrzeit Bei der<br>Startuhrzeit werden Ihre Sekunden- und<br>Millisekundenangabe ignoriert und auf 0<br>gesetzt.<br>Falls Sie einen monatlichen Start eines<br>Uhrzeitalarm-OB vorgeben wollen, sind beim<br>Startdatum nur die Tage 1, 2, ... 28 möglich.   |
| PERIOD    | Input       | WORD            | E, A, M, D, L oder<br>Konstante | Periode vom Ausgangspunkt SDT an:<br><ul style="list-style-type: none"> <li>• W#16#0000 = Einmal</li> <li>• W#16#0201 = Minütlich</li> <li>• W#16#0401 = Stündlich</li> <li>• W#16#1001 = Täglich</li> <li>• W#16#1201 = Wöchentlich</li> <li>• W#16#1401 = Monatlich</li> <li>• W#16#1801 = Jährlich</li> <li>• W#16#2001 = Monatsende</li> </ul> |
| RET_VAL   | Return      | INT             | E, A, M, D, L                   | Tritt während der Bearbeitung der Anweisung<br>ein Fehler auf, enthält der Aktualparameter<br>von RET_VAL einen Fehlercode.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0000  | Es ist kein Fehler aufgetreten.   |
| 8090  | Fehlerhafter Parameter OB_NR  |
| 8091  | Fehlerhafter Parameter SDT  |
| 8092  | Fehlerhafter Parameter PERIOD   |
| 80A1  | Der eingestellte Startzeitpunkt liegt in der Vergangenheit. (Dieser Fehlercode tritt nur bei PERIOD = W#16#0000 auf.) |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## SET\_TINTL: Uhrzeitalarm einstellen

### Beschreibung

Mit der Anweisung "SET\_TINTL" können Sie Startdatum und -uhrzeit der Uhrzeitalarm-Organisationsbausteine aus dem Anwenderprogramm heraus einstellen, ohne dass in der Hardware-Konfiguration Einstellungen vorgenommen wurden.

Das Startdatum und die Startuhrzeit geben Sie über den Parameter SDT vor. Die Periode, in der die Anweisung wiederholt aufgerufen werden soll (z. B. täglich, einmal pro Woche) können Sie über den Parameter PERIOD vorgeben. Wenn Sie für den Wiederholungszeitraum "monatlich" einstellen, dürfen Sie beim Startdatum nur einen Tag zwischen dem 1. und 28. angeben. Die Tage 29 bis 31 dürfen nicht parametrisiert werden, da z. B. im Februar kein Prozessalarm aufgerufen würde. Falls Sie am Ende eines jeden Monats den Uhrzeitalarm auslösen möchten, nutzen Sie die Funktion "Monatsende".

Über den Parameter ACTIVATE geben Sie vor, ob die vorgenommenen Einstellungen für den Organisationsbaustein direkt übernommen werden sollen (ACTIVATE = true) oder die Einstellungen erst nach dem Aufruf von "ACT\_TINT (Seite 2526)" für den Uhrzeitalarm-Organisationsbaustein übernommen werden sollen (ACTIVATE = false).

---

#### Hinweis

Verwenden Sie beim Aufruf der Uhrzeitalarm-Organisationsbausteine mit einer Startzeit innerhalb der zweiten Stunde beim Wechsel von Sommer- auf Winterzeit zusätzlich einen Verzögerungsalarm während der ersten Stunde der Zeitumstellung.

---

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SET\_TINTL":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung   |
|-----------|-------------|--------------|------------------------------|--|
| OB_NR     | Input       | OB_TOD (INT) | E, A, M, D, L oder Konstante | Nummer des OB, der zum Zeitpunkt SDT + Vielfaches von PERIOD gestartet wird (OB 10 bis OB 17 und ab OB 123).                     |
| SDT       | Input       | DTL          | D, L                         | Startdatum und Startuhrzeit: Bei der Startuhrzeit werden Ihre Sekunden- und Millisekundenangaben ignoriert und auf "0" gesetzt.  |
| LOCAL     | Input       | BOOL         | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>LOCAL = true: Lokalzeit verwenden</li> <li>LOCAL = false: Systemzeit verwenden</li> </ul> |

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| PERIOD    | Input       | WORD     | E, A, M, D, L oder Konstante | Periode vom Ausgangspunkt SDT an: <ul style="list-style-type: none"> <li>• W#16#0000 = Einmal</li> <li>• W#16#0201 = Minütlich</li> <li>• W#16#0401 = Stündlich</li> <li>• W#16#1001 = Täglich</li> <li>• W#16#1201 = Wöchentlich</li> <li>• W#16#1401 = Monatlich</li> <li>• W#16#1801 = Jährlich</li> <li>• W#16#2001 = Monatsende</li> </ul> |
| ACTIVATE  | Input       | BOOL     | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• ACTIVATE = true: Anweisung ausführen</li> <li>• ACTIVATE = false: Anweisung nur bei Aufruf von "ACT_TINT (Seite 2526)" ausführen</li> </ul>  |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Beschreibung  |
|---|---|
| 0000  | Es ist kein Fehler aufgetreten.   |
| 8090  | Fehlerhafter Parameter OB_NR  |
| 8091  | Fehlerhafter Parameter SDT  |
| 8092  | Fehlerhafter Parameter PERIOD   |
| 80A1  | Der eingestellte Startzeitpunkt liegt in der Vergangenheit. (Dieser Fehlercode tritt nur bei PERIOD = W#16#0000 auf.) |
| allgemeine Fehlerinformationen  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   |
| * Die Fehlercodes im Programmmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## CAN\_TINT: Uhrzeitalarm stornieren

### Beschreibung

Mit der Anweisung "CAN\_TINT" können Sie Startdatum und -uhrzeit eines angegebenen Uhrzeitalarm-Organisationsbausteins löschen. Der Uhrzeitalarm wird dadurch deaktiviert und der Organisationsbaustein wird nicht mehr aufgerufen.

Möchten Sie den Uhrzeitalarm wieder nutzen, müssen Sie zuerst den Startzeitpunkt neu stellen (Anweisung "SET\_TINTL (Seite 2524)") und dann den Uhrzeitalarm aktivieren (Anweisung "ACT\_TINT (Seite 2526)").

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "CAN\_TINT":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung  |
|-----------|-------------|--------------|------------------------------|---|
| OB_NR     | Input       | OB_TOD (INT) | E, A, M, D, L oder Konstante | Nummer des OBs, dessen Startdatum und -uhrzeit gelöscht werden soll.  |
| RET_VAL   | Return      | INT          | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode* (W#16#...)  | Beschreibung   |
|---|--|
| 0000  | Es ist kein Fehler aufgetreten.  |
| 8090  | Fehlerhafter Parameter OB_NR   |
| 80A0  | Kein Startdatum/-uhrzeit festgelegt für den betreffenden Uhrzeitalarm-OB |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)            |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**ACT\_TINT: Uhrzeitalarm aktivieren**

**Beschreibung**

Mit der Anweisung können Sie einen Uhrzeitalarm-Organisationsbaustein aktivieren.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ACT\_TINT":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung  |
|-----------|-------------|--------------|------------------------------|---|
| OB_NR     | Input       | OB_TOD (INT) | E, A, M, D, L oder Konstante | Nummer des OBs, der aktiviert werden soll.  |
| RET_VAL   | Return      | INT          | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Beschreibung  |
|---|---|
| 0000  | Es ist kein Fehler aufgetreten.   |
| 8090  | Fehlerhafter Parameter OB_NR  |
| 80A0  | Startdatum und -uhrzeit sind nicht gestellt für den betreffenden Uhrzeitalarm-OB.               |
| 80A1  | Die aktivierte Zeit liegt in der Vergangenheit; Fehler tritt nur auf bei einmaliger Ausführung. |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)                                   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## QRY\_TINT: Status des Uhrzeitalarms abfragen

### Beschreibung

Mit der Anweisung können Sie sich den Status eines Uhrzeitalarm-Organisationsbausteins am Ausgangsparameter STATUS anzeigen lassen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "QRY\_TINT":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung  |
|-----------|-------------|--------------|------------------------------|---|
| OB_NR     | Input       | OB_TOD (INT) | E, A, M, D, L oder Konstante | Nummer des OB, dessen Status abgefragt wird (OB 10 bis OB 17 und ab OB 123).  |
| RET_VAL   | Return      | INT          | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode. |
| STATUS    | Output      | WORD         | E, A, M, D, L                | Status des Uhrzeitalarms; siehe folgende Tabelle.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

Tritt ein Fehler auf (siehe Parameter RET\_VAL), wird am Parameter STATUS "0" ausgegeben.

| Bit    | Wert | Bedeutung  |
|--------|------|--|
| 0      | 0    | In Run.  |
|        | 1    | Während des Anlaufs.   |
| 1      | 0    | Der Uhrzeitalarm ist freigegeben.  |
|        | 1    | Der Uhrzeitalarm ist gesperrt.   |
| 2      | 0    | Der Uhrzeitalarm ist nicht aktiviert oder abgelaufen.                        |
|        | 1    | Der Uhrzeitalarm ist aktiviert.  |
| 4      | 0    | Ein OB mit einer OB-Nummer wie am Parameter OB_NR angegeben existiert nicht. |
|        | 1    | Ein OB mit einer OB-Nummer wie am Parameter OB_NR angegeben ist vorhanden.   |
| 6      | 0    | Grundlage für den Uhrzeitalarm ist die Basiszeit                             |
|        | 1    | Grundlage für den Uhrzeitalarm ist die Ortszeit                              |
| Andere |      | Immer "0"  |

### Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Beschreibung  |
|---|---|
| 0000  | Es ist kein Fehler aufgetreten.                               |
| 8090  | Fehlerhafter Parameter OB_NR                                  |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832) |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## Verzögerungsalarm

### Verwendung von Verzögerungsalarmen

#### Definition

Nachdem Sie die Anweisung "SRT\_DINT (Seite 2530)" aufgerufen haben, wird vom Betriebssystem nach Ablauf der parametrisierten Verzögerungszeit ein Alarm generiert, d. h. der parametrisierte Verzögerungsalarm-OB wird aufgerufen.

#### Voraussetzungen für Aufruf

Damit ein Verzögerungsalarm vom Betriebssystem aufgerufen werden kann, müssen folgende Voraussetzungen erfüllt sein:

- Der Verzögerungsalarm-OB muss gestartet sein durch die Anweisung "SRT\_DINT (Seite 2530)".
- Der Verzögerungsalarm-OB darf nicht per Projektierung abgewählt sein.
- Der Verzögerungsalarm-OB muss in der CPU vorhanden sein.

#### Zweck der Anweisungen "SRT\_DINT", "CAN\_DINT" und "QRY\_DINT"

Die Anweisungen verwenden Sie, um Verzögerungsalarme

- zu starten ("SRT\_DINT (Seite 2530)")
- zu stornieren ("CAN\_DINT (Seite 2531)")
- abzufragen ("QRY\_DINT (Seite 2532)").

#### Auswirkungen auf den Verzögerungsalarm

Die folgende Tabelle zeigt, welche Situationen welche Auswirkungen auf den Verzögerungsalarm haben.

| Wenn ...  | und ...  | dann ...  |
|---|--|---|
| ein Verzögerungsalarm gestartet wird (Aufruf von "SRT_DINT (Seite 2530)") | der Verzögerungsalarm bereits gestartet ist,                               | wird die Verzögerungszeit überschrieben; der Verzögerungsalarm wird erneut gestartet.   |
|   | der Verzögerungsalarm-OB nicht vorhanden ist zum Zeitpunkt seines Aufrufs, | erzeugt das Betriebssystem einen Prioritätsklassenfehler (Aufruf des OB 85; falls OB 85 nicht vorhanden ist, geht die CPU in den Betriebszustand STOP). |

| Wenn ...                            | und ...  | dann ...   |
|-------------------------------------|--|--|
|                                     | der Start in einem Anlauf-OB erfolgte und die Verzögerungszeit abgelaufen ist, bevor die CPU im RUN ist, | verzögert sich der Aufruf des Verzögerungsalarm-OBs, bis die CPU im RUN ist.   |
| die Verzögerungszeit abgelaufen ist | ein zuvor gestarteter Verzögerungsalarm-OB noch in Bearbeitung ist,                                      | erzeugt das Betriebssystem einen Zeitfehler (Aufruf des OB 80; falls der OB 80 nicht vorhanden ist, geht die CPU in den Betriebszustand STOP). |

### Verhalten bei Neustart (Warmstart) und Kaltstart

Bei Neustart (Warmstart) und Kaltstart werden alle im Anwenderprogramm über Anweisungen eingestellte Verzögerungsalarm-Einstellungen gelöscht.

### Start in einem Anlauf-OB

Ein Verzögerungsalarm kann in einem Anlauf-OB gestartet werden. Für den Aufruf des Verzögerungsalarm-OBs müssen zwei Bedingungen erfüllt sein:

- Die Verzögerungszeit ist abgelaufen.
- Die CPU ist im Betriebszustand RUN.

Wenn die Verzögerungszeit abgelaufen ist und die CPU sich noch nicht im Betriebszustand RUN befindet, dann verzögert sich der Aufruf des Verzögerungsalarm-OB, bis die CPU in RUN ist. Der Verzögerungsalarm-OB wird dann noch vor der ersten Anweisung im OB Main[OB 1] aufgerufen.

### SRT\_DINT: Verzögerungsalarm starten

#### Beschreibung

Mit dieser Anweisung "SRT\_DINT" starten Sie einen Verzögerungsalarm, der nach dem Ablauf der am Parameter DTIME angegebenen Verzögerungszeit einen Verzögerungsalarm-OB aufruft. Die Verzögerungszeit wird gestartet, wenn am Freigabeeingang EN eine negative Flanke erzeugt wird. Während die Verzögerungszeit abläuft, muss der Freigabeeingang EN den Signalzustand "0" führen. Wenn der Ablauf der Verzögerungszeit unterbrochen wird, wird der OB, der am Parameter OB\_NR parametrisiert wurde, nicht ausgeführt.

#### Genauigkeit

Die Zeit zwischen Aufruf der Anweisung "SRT\_DINT" und dem Start des Verzögerungsalarm-OBs ist maximal eine Millisekunde größer als die parametrisierte Verzögerungszeit, sofern keine Unterbrechungsereignisse den Aufruf verzögern.



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SRT\_DINT":

| Parameter | Deklaration | Datentyp       | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------------|------------------------------|---|
| OB_NR     | Input       | OB_DELAY (INT) | E, A, M, D, L oder Konstante | Nummer des OBs, der nach einer Verzögerungszeit ausgeführt wird   |
| DTIME     | Input       | TIME           | E, A, M, D, L oder Konstante | Verzögerungszeit (1 bis 60000 ms)<br>Größere Zeiten können Sie realisieren, indem Sie z. B. in einem Verzögerungsalarm-OBs einen Zähler benutzen. |
| SIGN      | Input       | WORD           | E, A, M, D, L oder Konstante | Kennzeichen, das beim Aufruf des Verzögerungsalarm-OBs in der Startereignisinformation des OBs erscheint.   |
| RET_VAL   | Return      | INT            | E, A, M, D, L                | Status der Anweisung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0000  | Kein Fehler   |
| 8090  | Fehlerhafter Parameter OB_NR                                  |
| 8091  | Fehlerhafter Parameter DTIME                                  |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832) |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## CAN\_DINT: Verzögerungsalarm stornieren

### Beschreibung

Mit dieser Anweisung stornieren Sie einen gestarteten Verzögerungsalarm und dadurch auch den Aufruf des Verzögerungsalarm-OBs, der nach der parametrisierten Verzögerungszeit ausgeführt werden soll. Die Nummer des Organisationsbausteins, dessen Aufruf storniert wird, geben Sie am Parameter OB\_NR an.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "CAN\_DINT":

| Parameter | Deklaration | Datentyp       | Speicherbereich              | Beschreibung                                 |
|-----------|-------------|----------------|------------------------------|--|
| OB_NR     | Input       | OB_DELAY (INT) | E, A, M, D, L oder Konstante | Nummer des OBs, dessen Aufruf storniert wird |
| RET_VAL   | Return      | INT            | E, A, M, D, L                | Status der Anweisung                         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode*<br>(W#16#...) | Beschreibung                           |
|---------------------------|--|
| 0000                      | Kein Fehler                            |
| 8090                      | Fehlerhafter Parameter OB_NR           |
| 80A0                      | Verzögerungsalarm ist nicht gestartet. |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**QRY\_DINT: Zustand eines Verzögerungsalarms abfragen**

**Beschreibung**

Mit der Anweisung "QRY\_DINT" können Sie den Zustand eines Verzögerungsalarms abfragen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "QRY\_DINT":

| Parameter | Deklaration | Datentyp       | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------------|------------------------------|--|
| OB_NR     | Input       | OB_DELAY (INT) | E, A, M, D, L oder Konstante | Nummer des OBs, dessen Zustand abgefragt wird.   |
| RET_VAL   | Return      | INT            | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Aktualparameter von RET_VAL einen Fehlercode. Am Parameter STATUS wird als Wert "0" angezeigt. |
| STATUS    | Output      | WORD           | E, A, M, D, L                | Zustand des Verzögerungsalarms; siehe folgende Tabelle.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Bit          | Wert | Bedeutung  |
|--------------|------|--|
| 0            | 0    | Betriebssystem in RUN  |
|              | 1    | Betriebssystem im Anlauf   |
| 1            | 0    | Verzögerungsalarm ist vom Betriebssystem freigegeben.            |
|              | 1    | Verzögerungsalarm ist gesperrt.                                  |
| 2            | 0    | Verzögerungsalarm ist nicht aktiviert oder abgelaufen.           |
|              | 1    | Verzögerungsalarm ist aktiviert.                                 |
| 3            | -    | -  |
| 4            | 0    | Verzögerungsalarm-OB mit der angegebenen Nummer existiert nicht. |
|              | 1    | Verzögerungsalarm-OB mit der angegebenen Nummer vorhanden.       |
| Weitere Bits |      | Immer "0"  |

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0000  | Es ist kein Fehler aufgetreten.                               |
| 8090  | Fehlerhafte Angabe am Parameter OB_NR                         |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832) |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## Synchronfehlerereignisse

### Synchronfehlerereignisse maskieren

#### Einleitung

Synchronfehlerereignisse sind Programmier- und Zugriffsfehlerereignisse. Solche Fehlerereignisse treten auf durch die Programmierung mit falschen Operandenbereichen, -nummern oder falschen Adressen. Diese Fehlerereignisse **maskieren** heißt,

- maskierte Fehlerereignisse lösen keinen Fehler-OB aus und führen nicht zu einer programmierten Ersatzreaktion.
- von den maskierten Fehlern "merkt sich" die CPU die Fehler, die aufgetreten sind, in einem Ereignisstatusregister.

Die Maskierung erfolgt durch Aufruf der Anweisung "MSK\_FLT (Seite 2540)".

Fehlerereignisse **demaskieren** heißt, eine vorher vorgenommene Maskierung wieder rückgängig zu machen und das zugehörige Bit im Ereignisstatusregister der aktuellen Prioritätsklasse zu löschen. Die Maskierung wird aufgehoben

- durch einen Aufruf der Anweisung "DMSK\_FLT (Seite 2541)"
- wenn die aktuelle Prioritätsklasse beendet ist.

Tritt ein Fehlerereignis nach seiner Demaskierung auf, dann wird vom Betriebssystem der zugehörige Fehler-OB gestartet. Für die Reaktion auf Programmierfehler programmieren Sie den OB 121 und für die Reaktion auf Zugriffsfehler programmieren Sie den OB 122.

Maskierte und aufgetretene Fehlerereignisse können Sie mit der Anweisung "READ\_ERR (Seite 2541)" auslesen.

---

**Hinweis**

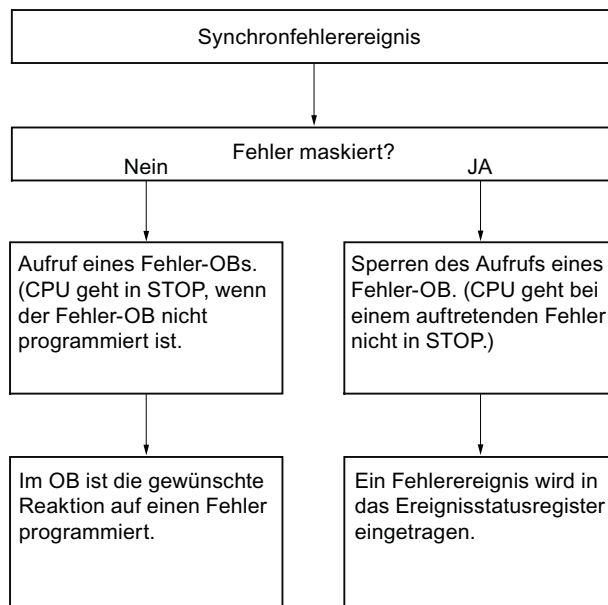
Unabhängig von einer Maskierung oder Demaskierung der Fehlerereignisse erfolgt bei S7-300 mit Ausnahme der CPU 318 ein Eintragen des Fehlerereignisses in den Diagnosepuffer, und die Sammelfehler-LED der CPU leuchtet.

---

**Fehlerbehandlung allgemein**

Wenn in einem Anwenderprogramm Programmier- und Zugriffsfehler auftreten, dann können Sie verschieden darauf reagieren:

- Sie können einen Fehler-OB programmieren und das Betriebssystem ruft diesen Fehler-OB auf, wenn ein entsprechendes Fehlerereignis auftritt.
- Sie können den Aufruf des Fehler-OB für jede Prioritätsklasse einzeln sperren. In diesem Fall geht die CPU nicht in STOP, wenn in der Prioritätsklasse ein entsprechender Fehler auftritt. Die CPU trägt die aufgetretenen Fehler in ein Ereignisstatusregister ein. Sie können an diesem Eintrag aber nicht erkennen, wann und wie oft dieser Fehler aufgetreten ist.



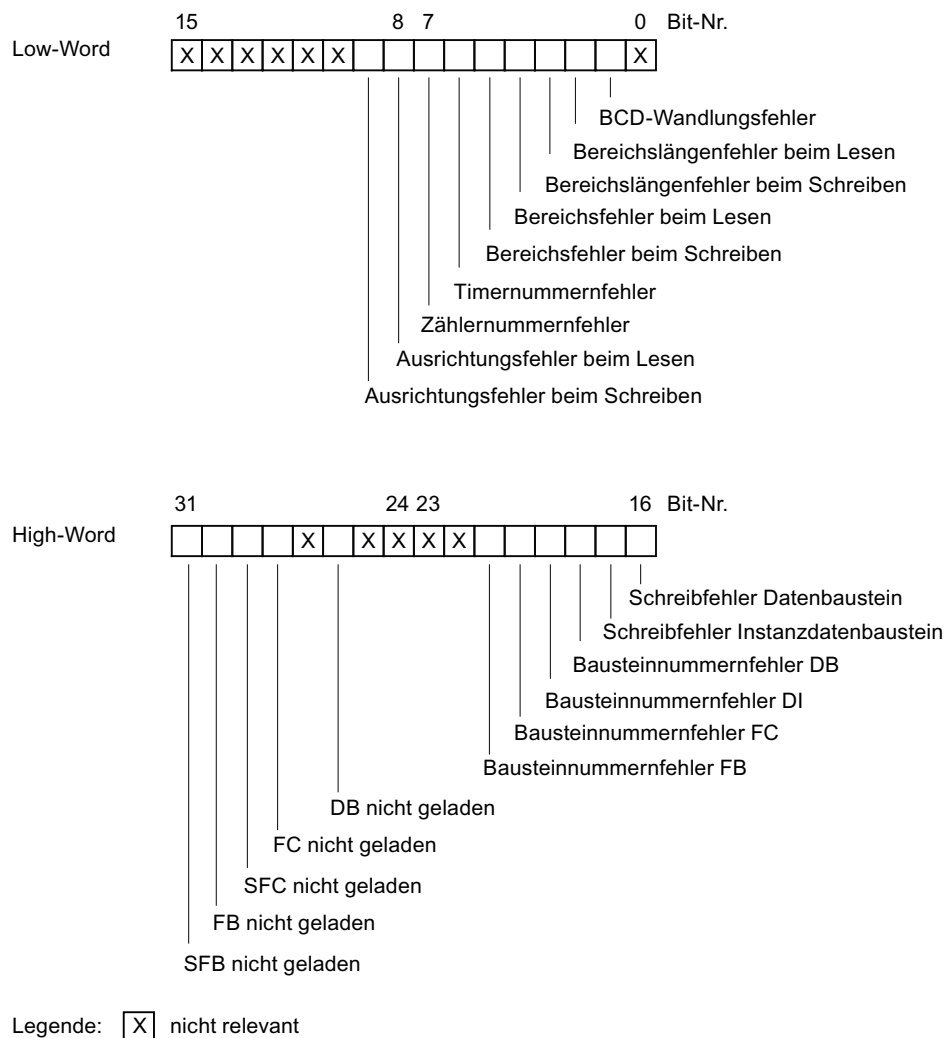
## Fehlermaske

Die Synchronfehlerereignisse sind einem bestimmten Bitmuster zugeordnet, der **Fehlermaske**. Diese Fehlermaske finden Sie in den Eingangs- und Ausgangsparametern der Anweisungen "MSK\_FLT (Seite 2540)", "DMSK\_FLT (Seite 2541)", "READ\_ERR (Seite 2541)" wieder.

Bei den Synchronfehlerereignissen wird zwischen Programmier- und Zugriffsfehlern unterschieden, die Sie in zwei Fehlermasken maskieren können. Die jeweiligen Fehlermasken finden Sie in den folgenden Bildern.

## Programmierfehlermaske

Das folgende Bild zeigt Ihnen das Bitmuster der Fehlermaske für Programmierfehler. Die Fehlermaske für Programmierfehler steht in den Parametern "PRGFLT\_..." (siehe im Folgenden "Programmierfehler, Low-Word" bzw. "Programmierfehler, High-Word").



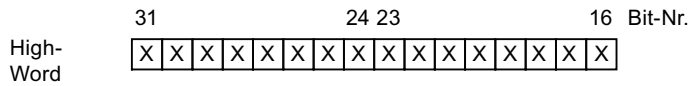
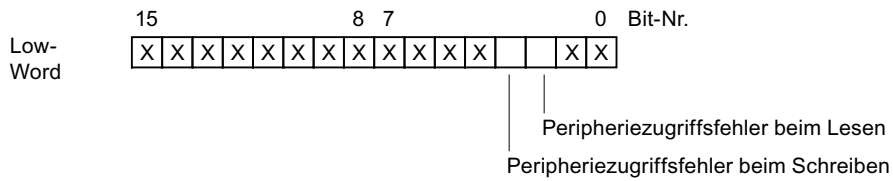
**Nicht relevante Bits**

Im Bild obigen bedeutet x für die ...

|                         |  |                                      |
|-------------------------|--|--------------------------------------|
| • ... Eingangsparameter | für "MSK_FLT (Seite 2540)", "DMSK_FLT (Seite 2541)", "READ_ERR (Seite 2541)" | = "0"                                |
| • ... Ausgangsparameter | für "MSK_FLT (Seite 2540)", "DMSK_FLT (Seite 2541)"                          | = "1" für S7-300<br>= "0" für S7-400 |
|                         | für "READ_ERR (Seite 2541)"  | = "0"                                |

**Zugriffsfehlermaske für alle CPUs**

Das folgende Bild zeigt Ihnen für alle CPUs das Bitmuster der Fehlermaske für Zugriffsfehler. Die Fehlermaske für Zugriffsfehler steht in den Parametern ACCFLT\_... Eine Erläuterung der Zugriffsfehler finden Sie in der Tabelle "Mögliche Fehlerursachen für alle CPUs 31x bis auf die CPU 318", bzw. Tabelle "Mögliche Fehlerursachen für alle CPUs 41x und CPU 318".

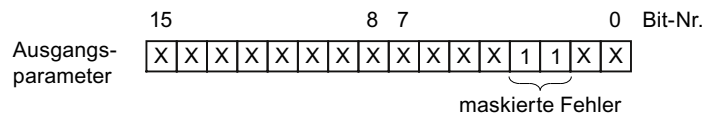
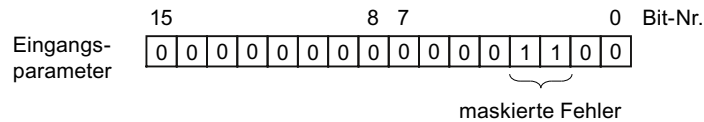


Legende: X nicht relevant

### Beispiel

Das folgende Bild zeigt Ihnen für alle CPUs, wie das Low-Word der Fehlermaske für Zugriffsfehler mit allen maskierten Fehlern aussieht.

- als Eingangsparameter für "MSK\_FLT (Seite 2540)"
- als Ausgangsparameter für "MSK\_FLT (Seite 2540)"



- Legende:
- X nicht relevant
  - 0 nicht maskiert
  - 1 maskiert

### Programmierfehler Low-Word

Die folgende Tabelle enthält die Fehler, die dem Low-Word der Fehlermaske für Programmierfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.

Mögliche Fehlerursachen für Programmierfehler, Low-Word

| Fehler                              | Ereignis-ID (W#16#...) | Fehler tritt auf, ...  |
|-------------------------------------|------------------------|--|
| BCD-Wandlungsfehler                 | 2521                   | ... wenn der zu wandelnde Wert keine BCD-Zahl ist (z. B. 5E8).   |
| Bereichslängenfehler beim Lesen     | 2522                   | ... wenn ein adressierter Operand nicht vollständig innerhalb des möglichen Operandenbereichs liegt.<br>Beispiel: MW 320 soll gelesen werden, obwohl der Merkerbereich nur 256 Byte groß ist.                  |
| Bereichslängenfehler beim Schreiben | 2523                   | ... wenn ein adressierter Operand nicht vollständig innerhalb des möglichen Operandenbereichs liegt.<br>Beispiel: Ein Wert soll auf MW 320 geschrieben werden, obwohl der Merkerbereich nur 256 Byte groß ist. |

| Fehler                            | Ereignis-ID (W#16#...) | Fehler tritt auf, ...  |
|-----------------------------------|------------------------|--|
| Bereichsfehler beim Lesen         | 2524                   | ... wenn bei indirekter, bereichsübergreifender Adressierung eine falsche Bereichskennung für den Operanden angegeben wird.<br>Beispiel:<br>Richtig: LAR1 P#E 12.0<br>L W[AR1, P#0.0]<br>Falsch: LAR1 P#12.0<br>L W[AR1, P#0.0] Bei dieser Operation wird der Bereichsfehler gemeldet. |
| Bereichsfehler beim Schreiben     | 2525                   | ... wenn bei indirekter, bereichsübergreifender Adressierung eine falsche Bereichskennung für den Operanden angegeben wird.<br>Beispiel:<br>Richtig: LAR1 P#E 12.0<br>T W[AR1, P#0.0]<br>Falsch: LAR1 P#12.0<br>T W[AR1, P#0.0] Bei dieser Operation wird der Bereichsfehler gemeldet. |
| Timernummernfehler                | 2526                   | ... wenn auf einen nicht vorhandenen Timer zugegriffen wird.<br>Beispiel:SI T [MW 0] mit MW 0 = 129; der Timer 129 soll gestartet werden, obwohl nur 128 Timer vorhanden sind.   |
| Zählernummernfehler               | 2527                   | ... wenn auf einen nicht vorhandenen Zähler zugegriffen wird.<br>Beispiel: ZV Z [MW 0] mit MW 0 = 600; es wird auf den Zähler 600 zugegriffen, obwohl nur 512 Zähler vorhanden sind (CPU 416-1).   |
| Ausrichtungsfehler beim Lesen     | 2528                   | ... wenn ein Byte-, Wort- oder Doppelwortoperand mit einer Bitadresse ≠ 0 adressiert wird.<br>Beispiel: Richtig: LAR1 P#M12.0<br>L B[AR1, P#0.0]<br>Falsch: LAR1 P#M12.4<br>L B[AR1, P#0.0]  |
| Ausrichtungsfehler beim Schreiben | 2529                   | ... wenn ein Byte-, Wort- oder Doppelwortoperand mit einer Bitadresse ≠ 0 adressiert wird.<br>Beispiel:<br>Richtig: LAR1 P#M12.0<br>T B[AR1, P#0.0]<br>Falsch: LAR1 P#M12.4<br>T B[AR1, P#0.0]   |

### Programmierfehler High-Word

Die folgende Tabelle enthält die Fehler, die dem High-Word der Fehlermaske für Programmierfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.



## Mögliche Fehlerursachen für Programmierfehler, High-Word

| Fehler                                | Ereignis-ID<br>(W#16#...) | Fehler tritt auf, ...   |
|---------------------------------------|---------------------------|---|
| Schreibfehler Datenbaustein           | 2530                      | ... wenn der Datenbaustein, auf den geschrieben wird, schreibgeschützt ist.                                     |
| Schreibfehler<br>Instanzdatenbaustein | 2531                      | ... wenn der Instanzdatenbaustein, auf den geschrieben wird, schreibgeschützt ist.                              |
| Bausteinnummernfehler DB              | 2532                      | ... wenn ein Datenbaustein aufgeschlagen wird, dessen Nummer größer als die größte zulässige Nummer ist.        |
| Bausteinnummernfehler DI              | 2533                      | ... wenn ein Instanzdatenbaustein aufgeschlagen wird, dessen Nummer größer als die größte zulässige Nummer ist. |
| Bausteinnummernfehler FC              | 2534                      | ... wenn eine Funktion aufgerufen wird, deren Nummer größer als die größte zulässige Nummer ist.                |
| Bausteinnummernfehler FB              | 2535                      | ... wenn ein Funktionsbaustein aufgerufen wird, dessen Nummer größer als die größte zulässige Nummer ist.       |
| DB nicht geladen                      | 253A                      | ... wenn der aufzuschlagende Datenbaustein nicht geladen ist.   |
| Anweisung nicht geladen               | 253C bis 253F             | ... wenn die aufzurufende Anweisung nicht geladen ist.  |

## Zugriffsfehler

Die folgende Tabelle enthält für alle CPUs die Fehler, die der Fehlermaske für Zugriffsfehler zugeordnet sind. Den jeweiligen Fehlern zugeordnet sind die möglichen Fehlerursachen.

| Fehler                                  | Ereignis-ID<br>(W#16#...) | Fehler tritt auf, ...   |
|---|---------------------------|---|
| Peripheriezugriffsfehler beim Lesen     | 2942                      | ... wenn der Adresse im Peripheriebereich keine Signalbaugruppe zugeordnet ist.<br>Oder<br>... wenn der Zugriff auf diesen Peripheriebereich nicht innerhalb der eingestellten Baugruppenüberwachungszeit quittiert wird (Quittungsverzug). |
| Peripheriezugriffsfehler beim Schreiben | 2943                      | ... wenn der Adresse im Peripheriebereich keine Signalbaugruppe zugeordnet ist.<br>oder<br>... wenn der Zugriff auf diesen Peripheriebereich nicht innerhalb der eingestellten Baugruppenüberwachungszeit quittiert wird (Quittungsverzug). |

## MSK\_FLT: Synchronfehlerereignisse maskieren

### Beschreibung

Die Anweisung nutzen Sie, um die Reaktion der CPU auf Synchronfehlerereignisse zu steuern. Dazu maskieren Sie die jeweiligen Synchronfehlerereignisse (Fehlermasken siehe Synchronfehlerereignisse maskieren (Seite 2533)). Mit dem Aufruf von "MSK\_FLT" maskieren Sie die Synchronfehlerereignisse in der aktuellen Prioritätsklasse.

Wenn Sie in den Eingangsparametern einzelne Bits der Synchronfehlermasken auf "1" setzen, dann behalten andere, vorher bereits gesetzte Bits ihren Wert "1". Sie erhalten neue Fehlermasken, die Sie über die Ausgangsparameter auslesen können. Die von Ihnen maskierten Synchronfehlerereignisse rufen keinen OB auf, sondern werden nur in einem Ereignisstatusregister eingetragen. Das Ereignisstatusregister können Sie mit der Anweisung "READ\_ERR (Seite 2541)" auslesen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MSK\_FLT":

| Parameter       | Deklaration | Datentyp | Speicherbereich              | Beschreibung                     |
|-----------------|-------------|----------|------------------------------|----------------------------------|
| PRGFLT_SET_MASK | Input       | DWORD    | E, A, M, D, L oder Konstante | Zu maskierende Programmierfehler |
| ACCFLT_SET_MASK | Input       | DWORD    | E, A, M, D, L oder Konstante | Zu maskierende Zugriffsfehler    |
| RET_VAL         | Return      | INT      | E, A, M, D, L                | Fehlerinformation                |
| PRGFLT_MASKED   | Output      | DWORD    | E, A, M, D, L                | Maskierte Programmierfehler      |
| ACCFLT_MASKED   | Output      | DWORD    | E, A, M, D, L                | Maskierte Zugriffsfehler         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Beschreibung  |
|---|---|
| 0000  | Keiner der Fehler war bereits maskiert.   |
| 0001  | Mindestens einer der Fehler war bereits maskiert, trotzdem werden die anderen Fehler maskiert.    |
| -   | allgemeine Fehlerinformation<br>Siehe auch: Fehler-ID lokal abfragen über GetErrorID (Seite 2304) |
| * Die Fehlercodes im Programmreditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET\_VAL (Seite 1622)

## DMSK\_FLT: Synchronfehlerereignisse demaskieren

### Beschreibung

Die Anweisung nutzen Sie, um die mit "MSK\_FLT (Seite 2540)" maskierten Fehlerereignisse zu demaskieren. Dazu müssen Sie in den Eingangsparametern die entsprechenden Bits der Fehlermasken auf "1" setzen. Mit dem Aufruf von "DMSK\_FLT" demaskieren Sie die entsprechenden Synchronfehlerereignisse der aktuellen Prioritätsklasse. Gleichzeitig werden die abgefragten Einträge im Ereignisstatusregister gelöscht. Die neuen Fehlermasken können Sie über die Ausgangsparameter auslesen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DMSK\_FLT":

| Parameter         | Deklaration | Datentyp | Speicherbereich              | Beschreibung                       |
|-------------------|-------------|----------|------------------------------|------------------------------------|
| PRGFLT_RESET_MASK | Input       | DWORD    | E, A, M, D, L oder Konstante | Zu demaskierende Programmierfehler |
| ACCFLT_RESET_MASK | Input       | DWORD    | E, A, M, D, L oder Konstante | Zu demaskierende Zugriffsfehler    |
| RET_VAL           | Return      | INT      | E, A, M, D, L                | Fehlerinformation                  |
| PRGFLT_MASKE D    | Output      | DWORD    | E, A, M, D, L                | Noch maskierte Programmierfehler   |
| ACCFLT_MASKE D    | Output      | DWORD    | E, A, M, D, L                | Noch maskierte Zugriffsfehler      |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Beschreibung   |
|---|--|
| 0000  | Alle angegebenen Fehler wurden demaskiert.   |
| 0001  | Mindestens einer der Fehler war nicht maskiert, trotzdem werden die anderen Fehler demaskiert. |
| -   | allgemeine Fehlerinformation<br>Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 2304)  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## READ\_ERR: Ereignisstatusregister auslesen

### Beschreibung

Mit der Anweisung lesen Sie das Ereignisstatusregister aus. Das Ereignisstatusregister entspricht in seinem Aufbau den Programmier- und Zugriffsfehlermasken, die Sie mit

"MSK\_FLT (Seite 2540)" und "DMSK\_FLT (Seite 2541)" als Eingangsparameter programmieren können.

In die Eingangsparameter tragen Sie ein, welche Synchronfehlerereignisse Sie aus dem Ereignisstatusregister abfragen wollen. Mit dem Aufruf von "READ\_ERR" lesen Sie die gewünschten Einträge aus dem Ereignisstatusregister aus und löschen gleichzeitig diese Einträge.

Aus dem Ereignisstatusregister entnehmen Sie die Information, welche der maskierten Synchronfehler in der aktuellen Prioritätsklasse mindestens einmal aufgetreten sind. Ein gesetztes Bit bedeutet, dass der zugehörige maskierte Synchronfehler mindestens einmal aufgetreten ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "READ\_ERR":

| Parameter    | Deklaration | Datentyp | Speicherbereich              | Beschreibung                   |
|--------------|-------------|----------|------------------------------|--------------------------------|
| PRGFLT_QUERY | Input       | DWORD    | E, A, M, D, L oder Konstante | Abfrage der Programmierfehler  |
| ACCFLT_QUERY | Input       | DWORD    | E, A, M, D, L oder Konstante | Abfrage der Zugriffsfehler     |
| RET_VAL      | Return      | INT      | E, A, M, D, L                | Fehlerinformation              |
| PRGFLT_CLR   | Output      | DWORD    | E, A, M, D, L                | Aufgetretene Programmierfehler |
| ACCFLT_CLR   | Output      | DWORD    | E, A, M, D, L                | Aufgetretene Zugriffsfehler    |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0000  | Alle abgefragten Fehler sind maskiert.  |
| 0001  | Mindestens einer der abgefragten Fehler ist nicht maskiert.                                       |
| -   | allgemeine Fehlerinformation<br>Siehe auch: Fehler-ID lokal abfragen über GetErrorID (Seite 2304) |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET\_VAL (Seite 1622)

## Asynchronfehlerereignis

### DIS\_IRT: Unterbrechungsereignis sperren

#### Beschreibung

Mit der Anweisung "DIS\_IRT" sperren Sie die Bearbeitung neuer Alarm- und Asynchronfehlerereignisse. Sperren heißt, dass das Betriebssystem der CPU bei einem Unterbrechungsereignis

- **weder** einen Alarm-OB bzw. einen Asynchronfehler-OB aufruft,
- **noch** die festgelegte Reaktion bei nicht programmiertem Alarm-OB bzw. Asynchronfehler-OB auslöst.

Die Sperre der Bearbeitung von Alarm- und Asynchronfehlerereignissen bleibt über alle Prioritätsklassen gültig. Die Sperre kann nur mit der Anweisung "EN\_IRT (Seite 2544)" aufgehoben werden bzw. auch bei Neustart (Warmstart) und bei Kaltstart.

Ob das Betriebssystem das aufgetretene Alarm- bzw. Asynchronfehlerereignis in den Diagnosepuffer einträgt, ist abhängig von Ihrer Wahl des Eingangsparameters MODE.

---

#### Hinweis

Beachten Sie, dass bei Programmierung der Anweisung "DIS\_IRT" alle eintreffenden Alarme verworfen werden!

---

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DIS\_IRT":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| MODE      | Input       | BYTE     | E, A, M, D, L oder Konstante | Gibt an, welche Alarm- und Asynchronfehlerereignisse gesperrt werden.                                  |
| OB_NR     | Input       | INT      | E, A, M, D, L oder Konstante | OB-Nummer  |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter MODE

| MODE<br>(B#16#...) | Bedeutung   |
|--------------------|---|
| 00                 | Alle neu auftretenden Alarm- und Asynchronfehlerereignisse werden gesperrt. (Nicht gesperrt werden Synchronfehlerereignisse.) Dem Parameter OB_NR weisen Sie den Wert "0" zu. Die Einträge in den Diagnosepuffer erfolgen weiterhin.  |
| 01                 | Alle neu auftretenden Ereignisse einer angegebenen Alarmklasse werden gesperrt. Die Alarmklasse kennzeichnen Sie wie folgt: <ul style="list-style-type: none"> <li>• Uhrzeitalarme: 10</li> <li>• Verzögerungsalarne: 20</li> <li>• Weckalarne: 30</li> <li>• Prozessalarne: 40</li> <li>• Alarne für DPV1: 50</li> <li>• Multicomputingalarm: 60</li> <li>• Redundanzfehleralarne: 70</li> <li>• Asynchrone Fehleralarne: 80</li> </ul> Die Einträge in den Diagnosepuffer erfolgen weiterhin. |
| 02                 | Alle neu auftretenden Ereignisse eines angegebenen Alarms werden gesperrt. Den Alarm kennzeichnen Sie durch die OB-Nummer. Die Einträge in den Diagnosepuffer erfolgen weiterhin.   |

### Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0000  | Es ist kein Fehler aufgetreten.                               |
| 8090  | Der Eingangsparameter OB_NR enthält einen unzulässigen Wert.  |
| 8091  | Der Eingangsparameter MODE enthält einen unzulässigen Wert.   |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832) |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### EN\_IRT: Unterbrechungsereignis freigeben

#### Beschreibung

Mit der Anweisung geben Sie die mit der Anweisung "DIS\_IRT (Seite 2543)" gesperrte Bearbeitung neuer Alarm- und Asynchronfehlerereignisse wieder frei. Freigeben heißt, dass das Betriebssystem der CPU bei einem Unterbrechungsereignis

- einen Alarm-OB bzw. einen Asynchronfehler-OB aufruft oder
- die festgelegte Reaktion bei nichtprogrammiertem Alarm-OB bzw. Asynchronfehler-OB auslöst.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EN\_IRT":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| MODE      | Input       | BYTE     | E, A, M, D, L oder Konstante | Gibt an, welche Alarm- und Asynchronfehlerereignisse freigegeben werden (siehe unten).                 |
| OB_NR     | Input       | INT      | E, A, M, D, L oder Konstante | OB-Nummer  |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Tritt während der Bearbeitung der Anweisung ein Fehler auf, enthält der Rückgabewert einen Fehlercode. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter MODE

| MODE | Bedeutung   |
|------|---|
| 0    | Alle neu auftretenden Alarm- und Asynchronfehlerereignisse werden freigegeben.  |
| 1    | Alle neu auftretenden Ereignisse einer angegebenen Alarmklasse werden freigegeben. Die Alarmklasse kennzeichnen Sie wie folgt: <ul style="list-style-type: none"> <li>• Uhrzeitalarme: 10</li> <li>• Verzögerungsalarme: 20</li> <li>• Weckalarme: 30</li> <li>• Prozessalarme: 40</li> <li>• Alarme für DPV1: 50</li> <li>• Multicomputingalarm: 60</li> <li>• Redundanzfehleralarml: 70</li> <li>• Asynchrone Fehleralarml: 80</li> </ul> |
| 2    | Alle neu auftretenden Ereignisse eines angegebenen Alarms werden freigegeben. Den Alarm kennzeichnen Sie durch die OB-Nummer.   |

## Parameter RET\_VAL

| Fehlercode* (W#16#...)       | Beschreibung  |
|------------------------------|---|
| 0000                         | Es ist kein Fehler aufgetreten.                               |
| 8090                         | Der Eingangsparameter OB_NR enthält einen unzulässigen Wert.  |
| 8091                         | Der Eingangsparameter MODE enthält einen unzulässigen Wert.   |
| allgemeine Fehlerinformation | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832) |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## DIS\_AIRT: Bearbeitung von höherprioren Alarm- und Asynchronfehlerereignissen verzögern

### Beschreibung

Mit "DIS\_AIRT" verzögern Sie die Bearbeitung von Alarm-OBs, deren Priorität höher als die Priorität des aktuellen Organisationsbausteins ist.

Sie können "DIS\_AIRT" in einem Organisationsbaustein mehrmals aufrufen. Die Aufrufe von "DIS\_AIRT" werden vom Betriebssystem gezählt. Bei jeder Ausführung von "DIS\_AIRT" wird die Bearbeitung weiter verzögert. Um eine Verzögerung aufzuheben, müssen Sie die Anweisung "EN\_AIRT (Seite 2546)" ausführen. Zum Aufheben aller Verzögerungen muss die Anzahl der Ausführungen von "EN\_AIRT (Seite 2546)" gleich der Anzahl der Aufrufe von "DIS\_AIRT" sein.

Die Anzahl der Verzögerungen können Sie am Parameter RET\_VAL der Anweisung "DIS\_AIRT" abfragen. Bei einem Wert "0" am Parameter RET\_VAL liegen keine Verzögerungen vor.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DIS\_AIRT":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung             |
|-----------|-------------|----------|-----------------|--------------------------|
| RET_VAL   | Return      | INT      | E, A, M, D, L   | Anzahl der Verzögerungen |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## EN\_AIRT: Bearbeitung von höherprioren Alarm- und Asynchronfehlerereignissen freigeben

### Beschreibung

Mit "EN\_AIRT" geben Sie die Bearbeitung von Organisationsbausteinen beim Auftreten von Alarmen frei, die durch die Anweisung "DIS\_AIRT (Seite 2546)" verzögert wurde.

Mit einer Ausführung von "EN\_AIRT" heben Sie eine Bearbeitungsverzögerung auf, die bei einem Aufruf von "DIS\_AIRT (Seite 2546)" vom Betriebssystem registriert wurde. Zum Aufheben aller Verzögerungen muss die Anzahl der Ausführungen von "EN\_AIRT" gleich der Anzahl der Aufrufe von "DIS\_AIRT (Seite 2546)" sein. Wenn Sie zum Beispiel "DIS\_AIRT (Seite 2546)" fünf Mal aufgerufen haben und dadurch die Bearbeitung auch fünf Mal verzögert haben, dann müssen Sie die Anweisung "EN\_AIRT" fünf Mal ausführen, um alle fünf Verzögerungen aufzuheben.

Am Parameter RET\_VAL der Anweisung "EN\_AIRT" können Sie die Anzahl der Alarmverzögerungen abfragen, die nach der Ausführung von "EN\_AIRT" noch nicht frei gegeben sind. Der Wert "0" am Parameter RET\_VAL bedeutet, dass alle durch "DIS\_AIRT (Seite 2546)" aktivierten Verzögerungen aufgehoben sind.



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "EN\_AIRT":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                                 |
|-----------|-------------|----------|-----------------|--|
| RET_VAL   | Return      | INT      | E, A, M, D, L   | Anzahl der noch programmierten Verzögerungen |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### 9.7.3.8 Meldungen

#### Program\_Alarm: Programmmeldung mit Begleitwerten erzeugen

##### Beschreibung

Die Anweisung "Program\_Alarm" überwacht ein Signal und generiert bei einem Signalwechsel am Parameter SIG eine Programmmeldung (zur Definition siehe auch: Auto-Hotspot). Bei einem Signalwechsel von 0 auf 1 wird eine kommende Meldung generiert und bei einem Signalwechsel von 1 auf 0 eine gehende Programmmeldung. Die Programmmeldung wird jeweils synchron zum Programmablauf ausgelöst.

An die Programmmeldung können Sie an den Parametern SD\_i ( $0 \leq i \leq 10$ ) bis zu zehn Begleitwerte anhängen. Die Begleitwerte werden zum Zeitpunkt des Signalwechsels am Parameter SIG erfasst und der Programmmeldung zugeordnet. Weitere Informationen zum Aufbau von Begleitwerten finden Sie in den folgenden Abschnitten: Auto-Hotspot, Auto-Hotspot, Auto-Hotspot.

Jede Meldung, sowohl kommende als auch gehende, werden mit einem Zeitstempel versehen:

- Standardmäßig wird hierzu die aktuelle Systemzeit der PLC beim Signalwechsel benutzt (Standardwert am Parameter TIMESTAMP).
- Wenn Sie abweichend hiervon einen Zeitstempel vorgeben wollen, können Sie diesen an den Parameter TIMESTAMP anlegen.  
Der Zeitwert muss hierbei immer in Systemzeit (d. h. UTC) angegeben werden, da diese Zeit bei der anlagenweiten Uhrzeitsynchronisation verwendet wird.
- Falls eine Meldung mit einer Lokalzeit gestempelt werden soll, so muss ein Konvertierungsbaustein vorgeschaltet werden, der die Lokalzeit auf die Systemzeit umrechnet. Nur dadurch ist garantiert, dass die Zeitstempel in der Meldungsanzeige richtig dargestellt werden.

Um wieder die aktuelle Systemzeit der CPU zu verwenden, stellen Sie den Parameter TIMESTAMP auf seinen Standardwert (LDT#1970-1-1-0:0:0.0).

#### Aufruf der Anweisung "Program\_Alarm"

Die Anweisung "Program\_Alarm" kann nur in einem Funktionsbaustein (FB) aufgerufen werden.

Nach dem Einfügen der Anweisung in den FB wird im Abschnitt "Static" der FB-Schnittstellen eine Multiinstanz vom Datentyp "Program\_Alarm" angelegt. Den Namen dieser Multiinstanz können Sie im aufgeblendeten Dialog frei wählen. Der Name dieser Multiinstanz ist gleichzeitig der Name der Programmmeldung.

Ergänzen Sie anschließend die Parameter der Anweisung "Program\_Alarm" entsprechend Ihren Anforderungen (siehe Tabelle "Parameter").

### Projektieren der Programmmeldung

Wenn sie den Namen der Programmmeldung im Abschnitt "Static" oder im Netzwerk des FBs anwählen, werden im Fenster "Eigenschaften" die Einstellungen der Programmmeldung angezeigt. Hier können Sie die Auswahl der Meldeklasse, Priorität, etc. vornehmen und den Meldetext editieren.

Die hier vorgenommenen Einstellungen können auch in der Projektnavigation bearbeitet werden. Öffnen Sie hierzu unter "PLC-Meldungen" das Register "Programmmeldungen". In der Tabelle "Meldungstypen" werden alle bereits angelegten Programmmeldungen angezeigt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Program\_Alarm":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| SIG       | Input       | BOOL     | E, A, M, D, L oder Konstante | Das zu überwachende Signal.  |
| TIMESTAMP | Input       | LDT      | M, D, L oder Konstante       | <p>Dieser Parameter wird dazu verwendet, einen Alarm mit einem Zeitstempel zu versehen, der z. B. von einem dezentral gestempelten Eingangssignal stammt. Der Zeitwert ist immer in Systemzeit (d. h. UTC) anzugeben, da diese Zeit bei der anlagenweiten Uhrzeitsynchronisation verwendet wird.</p> <ul style="list-style-type: none"> <li>• "Nicht verschaltet" bedeutet, dass die Systemzeit der CPU beim Signalwechsel als Zeitstempel für den Alarm verwendet wird (Default).</li> <li>• Die Übergabe einer Systemzeit bedeutet, dass diese beim Signalwechsel als Zeitstempel für den Alarm verwendet wird.</li> </ul> <p>Hinweis: Falls ein Alarm mit einer Lokalzeit gestempelt werden soll, so muss ein Konvertierungsbaustein vorgeschaltet werden, der die Lokalzeit auf die Systemzeit umrechnet. Nur dadurch ist garantiert, dass der Zeitstempel in der Alarmanzeige richtig dargestellt wird.</p> |

| Parameter | Deklaration | Datentyp   | Speicherbereich  | Beschreibung  |
|-----------|-------------|--|------------------|---|
| SD_i      | Input       | LREAL, REAL,<br>DINT, INT,<br>DWORD,<br>WORD, BYTE,<br>BOOL, STRING<br>oder CHAR | E, A, M, D, T, Z | i-ter Begleitwert ( $0 \leq i \leq 10$ )  |
| ERROR     | Input       | BOOL   | E, A, M, D, L    | Zustandsparameter ERROR<br>ERROR = TRUE bedeutet, dass bei der Bearbeitung ein Fehler aufgetreten ist. Die mögliche Fehlerursache wird am Parameter STATUS angezeigt. |
| STATUS    | Input       | WORD   | E, A, M, D, L    | Zustandsparameter STATUS<br>Anzeige der Fehlerinformation (siehe "Parameter ERROR und STATUS").   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter ERROR und STATUS

Die folgende Tabelle enthält alle spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

| ERROR | STATUS* | Erläuterung  |
|-------|---------|--|
| 0     | 0000    | Kein Fehler  |
| 0     | 7001    | Meldung wird generiert.  |
| 1     | 8001    | Ungültige statische Meldungsinformation.   |
| 1     | 8002    | Keine gültige statische Meldungsinformation.   |
| 1     | 8004    | Maximale Größe von 512 Bytes für die Meldung erreicht.   |
| 1     | 8005    | Kommende Meldung.  |
| 1     | 8007    | Gehende Meldung, der keine kommende Meldung vorausgegangen ist.  |
| 1     | 8087    | Statische Meldungen deaktiviert.   |
| 1     | 8089    | Länge der Meldung zu groß.   |
| 1     | 80Ax    | Ungültiger Wert am Parameter Nr. x.  |
| 1     | 80C1    | Die CPU kann derzeit keine Meldungen generieren, da Initialisierungsroutinen ablaufen (Das ist z. B. nach Download im RUN der Fall.). Versuchen Sie es später noch einmal. |
| 1     | 80C2    | Die maximale Anzahl von Meldungen pro Zeiteinheit wurde gesendet. Versuchen Sie es später noch einmal.   |
| 1     | 80C3    | Alle dynamischen Meldungs-Instanzen werden verwendet. Versuchen Sie es später noch einmal.   |
| 1     | 80C4    | Meldung wird ausgegeben und kann nicht überschrieben werden. Versuchen Sie es später noch einmal.  |

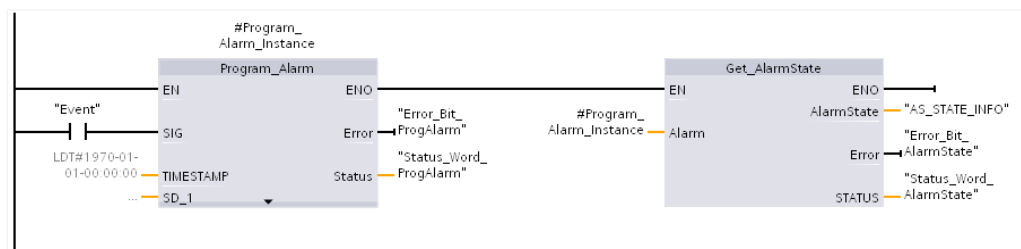
\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Get\_AlarmState: Meldungszustand ausgeben

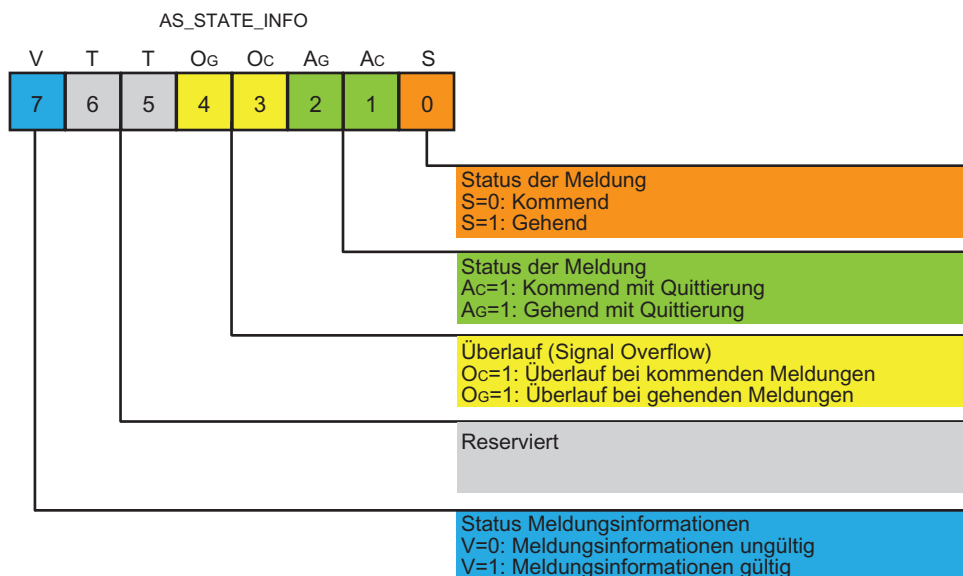
#### Beschreibung

Die Anweisung "Get\_AlarmState" gibt den Meldungszustand einer Programmmeldung aus. Die Ausgabe des Meldungszustands bezieht sich jeweils auf eine Programmmeldung, die über die Anweisung "Program\_Alarm (Seite 2547)" erstellt wurde.

Die Auswahl der Programmmeldung erfolgt über den Eingangsparameter Alarm. An dem Parameter Alarm geben Sie den Instanz-DB der Anweisung "Program\_Alarm" an.



Über den Ausgangsparameter AlarmState wird der Meldungszustand in einem Byte ausgegeben. Die Bedeutung der einzelnen Bits ist in der folgenden Abbildung dargestellt:



Weitere Informationen über die möglichen Meldezustände finden Sie in dem Abschnitt "Auto-Hotspot".

Über die Ausgangsparameter ERROR und STATUS wird der Bearbeitungszustand der Anweisung angezeigt.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Get\_AlarmState":

| Parameter  | Deklaration | Datentyp   | Speicherbereich | Beschreibung   |
|------------|-------------|------------|-----------------|--|
| Alarm      | Input       | ALARM_BASE | D               | Instanz der Anweisung "Program_Alarm (Seite 2547)".  |
| AlarmState | Output      | BYTE       | E, A, M, D, L   | Status der Meldung als Bitfeld.  |
| ERROR      | Output      | BOOL       | E, A, M, D, L   | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: kein Fehler</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS     | Output      | WORD       | E, A, M, D, L   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Fehlercode* (W#16#...) | Erläuterung  |
|------------------------|--|
| 8001                   | Ungültige statische Meldungsinanz.                 |
| 8002                   | ID der Meldung ungültig.                           |
| 8003                   | Keine aktiven Meldungen innerhalb der Meldeklasse. |

\* Die Fehlercodes im Programmreditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## Gen\_UsrMsg: Anwenderdiagnosemeldungen erzeugen

### Beschreibung

Mit der Anweisung "Gen\_UsrMsg" erzeugen Sie eine Anwenderdiagnosemeldung. Mit Anwenderdiagnosemeldungen können Sie einen Anwendereintrag in den Diagnosepuffer schreiben und eine entsprechende Meldung senden.

Die Erstellung des Eintrags in den Diganosepuffer und die Abfrage des Parameters Send erfolgt synchron. Das Übermitteln des Inforeports erfolgt asynchron.

Über die Parameter TextID, TextListID, AlarmDomain und AssocValues definieren Sie den Inhalt des Anwendereintrags in den Diagnosepuffer.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Gen\_UsrMsg":

| Parameter   | Deklaration | Datentyp             | Speicherbereich              | Beschreibung  |
|-------------|-------------|----------------------|------------------------------|---|
| Mode        | Input       | UInt                 | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• 1: Kommende Meldung</li> <li>• 2: Gehende Meldung</li> </ul>         |
| Send        | Input       | Bool                 | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• 0: Kein Inforeport senden</li> <li>• 1: Inforeport senden</li> </ul> |
| TextID      | Input       | UInt                 | E, A, M, D, L oder Konstante | ID des anwenderdefinierten Texts  |
| TextListID  | Input       | UInt                 | E, A, M, D, L oder Konstante | ID der anwenderdefinierten Textliste  |
| AlarmDomain | Input       | UInt                 | E, A, M, D, L oder Konstante | Verwendete Meldungs-Domäne  |
| Ret_Val     | Return      | Int                  | E, A, M, D, L                | Fehlercode der Anweisung  |
| AssocValues | InOut       | Array [0..7] of UInt | E, A, M, D, L                | Begleitwerte  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

Die folgende Tabelle enthält alle spezifischen Fehlerinformationen, die über den Parameter RET\_VAL ausgegeben werden können.

| Fehlercode* (W#16#...)  | Erläuterung  |
|---|--|
| 0000  | Kein Fehler  |
| 8080  | Wert am Parameter MODE nicht unterstützt.                |
| 80C1  | Ressourcenengpass aufgrund zu vieler paralleler Aufrufe. |
| * Die Fehlercodes im Programmmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### 9.7.3.9 Diagnose

#### RD\_SINFO: Startinformation des aktuellen OBs auslesen

##### Beschreibung

Mit der Anweisung "RD\_SINFO" lesen Sie die Startinformation

- des zuletzt aufgerufenen OBs, der noch nicht vollständig abgearbeitet wurde oder
- des zuletzt gestarteten Anlauf-OBs.

Beide Startinformationen enthalten keinen Zeitstempel. Erfolgt der Aufruf im OB 100, OB 101 oder OB 102, werden zwei identische Startinformationen zurückgeliefert.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_SINFO":

| Parameter   | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-------------|-------------|----------|-----------------|---|
| RET_VAL     | Return      | INT      | E, A, M, D, L   | Fehlerinformation                                   |
| TOP_SI      | Output      | VARIANT  | D, L            | Startinformation des aktuellen OBs                  |
| START_UP_SI | Output      | VARIANT  | D, L            | Startinformation des zuletzt gestarteten Anlauf-OBs |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## SDTs des Parameters TOP\_SI

Die folgende Tabelle zeigt die möglichen SDTs des Parameters TOP\_SI:

| Organisationsbausteine (OB) | Systemdatentypen (SDT) | Nummern der Systemdatentypen |
|-----------------------------|------------------------|------------------------------|
| Alle                        | SI_classic             | 592                          |
|                             | SI_none                | 593                          |
| ProgramCycleOB              | SI_ProgramCycle        | 594                          |
| TimeOfDayOB                 | SI_TimeOfDay           | 595                          |
| TimeDelayOB                 | SI_Delay               | 596                          |
| CyclicOB                    | SI_Cyclic              | 597                          |
| ProcessEventOB              | SI_HWInterrupt         | 598                          |
| ProfileEventOB              | SI_Submodule           | 601                          |
| StatusEventOB               |                        |                              |
| UpdateEventOB               |                        |                              |
| SynchronousCycleOB          | SI_SynchCycle          | 602                          |
| IOredundancyErrorOB         | SI_IORedundancyError   | 604                          |
| CPUREdundancyErrorOB        | SI_CPUREdundancyError  | 605                          |
| TimeErrorOB                 | SI_TimeError           | 606                          |
| DiagnosticErrorOB           | SI_DiagnosticInterrupt | 607                          |
| PullPlugEventOB             | SI_PlugPullModule      | 608                          |
| PeripheralAccessErrorOB     | SI_AccessError         | 609                          |
| RackStationFailureOB        | SI_StationFailure      | 610                          |
| ServoOB                     | SI_Servo               | 611                          |
| IpoOB                       | SI_Ipo                 | 612                          |
| StartupOB                   | SI_Startup             | 613                          |
| ProgrammingErrorOB          | SI_ProgIOAccessError   | 614                          |
| IOaccessErrorOB             |                        |                              |

### SDTs des Parameters START\_UP\_SI

Die folgende Tabelle zeigt die möglichen SDTs des Parameters START\_UP\_SI:

| Systemdatentypen (SDT) | Nummern der Systemdatentypen |
|------------------------|------------------------------|
| SI_classic             | 592                          |
| SI_none                | 593                          |
| SI_Startup             | 613                          |

### Strukturen

Die folgenden Tabellen zeigen die Bedeutung der Strukturelemente der einzelnen Strukturen:

Tabelle 9-57 Struktur SI\_classic

| Strukturelement | Datentyp | Beschreibung  |
|-----------------|----------|---|
| EV_CLASS        | BYTE     | <ul style="list-style-type: none"> <li>Bits 0 bis 3: Ereigniskennung</li> <li>Bits 4 bis 7: Ereignisklasse</li> </ul>                 |
| EV_NUM          | BYTE     | Ereignisnummer  |
| PRIORITY        | BYTE     | Nummer der Ablaufebene<br>(Bedeutung von B#16#FE: OB nicht vorhanden oder gesperrt oder im aktuellen Betriebszustand nicht startbar.) |
| NUM             | BYTE     | OB-Nummer   |
| TYP2_3          | BYTE     | Datenkennung 2_3:<br>kennzeichnet die in ZI2_3 eingetragene Information   |
| TYP1            | BYTE     | Datenkennung 1:<br>kennzeichnet die in ZI1 eingetragene Information   |
| ZI1             | WORD     | Zusatzinformation 1   |
| ZI2_3           | DWORD    | Zusatzinformation 2_3   |

Tabelle 9-58 Struktur SI\_none

| Strukturelement | Datentyp | Beschreibung  |
|-----------------|----------|---|
| SI_Format       | USINT    | <ul style="list-style-type: none"> <li>16#FF = Keine Angabe</li> <li>16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT    | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT     | OB-Nummer (1 ... 32767)   |

Tabelle 9-59 Struktur SI\_ProgramCycle

| Strukturelement | Datentyp   | Beschreibung  |
|-----------------|------------|---|
| SI_Format       | USINT      | <ul style="list-style-type: none"> <li>16#FF = Keine Angabe</li> <li>16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 1 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT       | OB-Nummer (1 ... 32767)   |



| Strukturelement | Datentyp | Beschreibung                     |
|-----------------|----------|----------------------------------|
| Initial_Call    | BOOL     | Für OB_Class = 1, 30, 52, 61, 65 |
| Remanence       | BOOL     | Für OB_Class = 1                 |

Tabelle 9-60 Struktur SI\_TimeOfDay

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 10 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| CaughtUp        | BOOL        | Für OB_Class = 10   |
| SecondTime      | BOOL        | Für OB_Class = 10   |

Tabelle 9-61 Struktur SI\_Delay

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 20 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| Sign            | WORD        | Für OB_Class = 20   |

Tabelle 9-62 Struktur SI\_Cyclic

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 30 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| Initial_Call    | BOOL        | Für OB_Class = 1, 30, 52, 61, 65  |
| Event_Count     | INT         | Für OB_Class = 30, 51, 52, 61, 65, 91, 92   |

Tabelle 9-63 Struktur SI\_HWInterrupt

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 40 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| LADDR           | HW_IO       | Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92   |
| USI             | WORD        | Für OB_Class = 40   |
| IChannel        | USINT       | Für OB_Class = 40   |
| EventType       | BYTE        | Für OB_Class = 40   |

Tabelle 9-64 Struktur SI\_Submodule

| Strukturelement | Datentyp | Beschreibung  |
|-----------------|----------|---|
| SI_Format       | USINT    | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT    | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT     | OB-Nummer (1 ... 32767)   |
| LADDR           | HW_IO    | Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92   |
| Slot            | UINT     | Für OB_Class = 55, 56, 57   |
| Specifier       | WORD     | Für OB_Class = 55, 56, 57   |

Tabelle 9-65 Struktur SI\_SynchCycle

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 61 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| Initial_Call    | BOOL        | Für OB_Class = 1, 30, 52, 61, 65  |
| PIP_Input       | BOOL        | Für OB_Class = 61, 91, 92   |
| PIP_Output      | BOOL        | Für OB_Class = 61, 91, 92   |
| IO_System       | USINT       | Für OB_Class = 61, 91, 92   |
| Event_Count     | INT         | Für OB_Class = 30, 51, 52, 61, 65, 91, 92   |
| SyncCycleTime   | LTIME       | Berechnete Zykluszeit   |

Tabelle 9-66 Struktur SI\_IORedundancyError

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 70 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| LADDR           | HW_ANY      | Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92   |
| Event_Class     | BYTE        | Für OB_Class = 70, 83, 85, 86   |
| Fault_ID        | BYTE        | Für OB_Class = 70, 80, 83, 85, 86   |

Tabelle 9-67 Struktur SI\_CPURedundancyError

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 72 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| Switch_Over     | BOOL        | Für OB_Class = 72   |

Tabelle 9-68 Struktur SI\_TimeError

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 80 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| Fault_ID        | BYTE        | Für OB_Class = 70, 80, 83, 85, 86   |
| Csg_OBnr        | OB_ANY      | Für OB_Class = 80   |
| Csg_Prio        | UINT        | Für OB_Class = 80   |

Tabelle 9-69 Struktur SI\_DiagnosticInterrupt

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 82 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| IO_State        | WORD        | Für OB_Class = 82   |
| LADDR           | HW_ANY      | Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92   |
| Channel         | UINT        | Für OB_Class = 82   |
| MultiError      | BOOL        | Für OB_Class = 82   |

Tabelle 9-70 Struktur SI\_PlugPullModule

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 83 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| LADDR           | HW_IO       | Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92   |
| Event_Class     | BYTE        | Für OB_Class = 70, 83, 85, 86   |
| Fault_ID        | BYTE        | Für OB_Class = 70, 80, 83, 85, 86   |

Tabelle 9-71 Struktur SI\_AccessError

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 85 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| LADDR           | HW_IO       | Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92   |
| Event_Class     | BYTE        | Für OB_Class = 70, 83, 85, 86   |
| Fault_ID        | BYTE        | Für OB_Class = 70, 80, 83, 85, 86   |
| IO_Addr         | UINT        | Für OB_Class = 85   |
| IO_LEN          | UINT        | Für OB_Class = 85   |

Tabelle 9-72 Struktur SI\_StationFailure

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 86 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| LADDR           | HW_IO       | Für OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92   |
| Event_Class     | BYTE        | Für OB_Class = 70, 83, 85, 86   |
| Fault_ID        | BYTE        | Für OB_Class = 70, 80, 83, 85, 86   |

Tabelle 9-73 Struktur SI\_Servo

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 91 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| Initial_Call    | BOOL        | Für OB_Class = 1, 30, 52, 61, 65  |
| PIP_Input       | BOOL        | Für OB_Class = 61, 91, 92   |
| PIP_Output      | BOOL        | Für OB_Class = 61, 91, 92   |
| IO_System       | USINT       | Für OB_Class = 61, 91, 92   |
| Event_Count     | INT         | Für OB_Class = 30, 51, 52, 61, 65, 91, 92   |
| Synchronous     | BOOL        |   |

Tabelle 9-74 Struktur SI\_Ipo

| Strukturelement | Datentyp    | Beschreibung  |
|-----------------|-------------|---|
| SI_Format       | USINT       | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 92 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT        | OB-Nummer (1 ... 32767)   |
| Initial_Call    | BOOL        | Für OB_Class = 1, 30, 52, 61, 65  |
| PIP_Input       | BOOL        | Für OB_Class = 61, 91, 92   |
| PIP_Output      | BOOL        | Für OB_Class = 61, 91, 92   |
| IO_System       | USINT       | Für OB_Class = 61, 91, 92   |
| Event_Count     | INT         | Für OB_Class = 30, 51, 52, 61, 65, 91, 92   |
| Reduction       | UINT        | Für OB_Class = 92   |

Tabelle 9-75 Struktur SI\_Startup

| Strukturelement | Datentyp     | Beschreibung  |
|-----------------|--------------|---|
| SI_Format       | USINT        | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT := 100 | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT         | OB-Nummer (1 ... 32767)   |

| Strukturelement | Datentyp | Beschreibung       |
|-----------------|----------|--------------------|
| LostRetentive   | BOOL     | Für OB_Class = 100 |
| LostRTC         | BOOL     | Für OB_Class = 100 |

Tabelle 9-76 Struktur SI\_ProgIOAccessError

| Strukturelement | Datentyp | Beschreibung  |
|-----------------|----------|---|
| SI_Format       | USINT    | <ul style="list-style-type: none"> <li>• 16#FF = Keine Angabe</li> <li>• 16#FE = Optimierte Startinformation</li> </ul> |
| OB_Class        | USINT    | OB-Klasse für "Keine Angabe" oder "Optimierte Startinformation"   |
| OB_Nr           | UINT     | OB-Nummer (1 ... 32767)   |
| BlockNr         | UINT     | Für OB_Class = 121, 122   |
| Reaction        | USINT    | Für OB_Class = 121, 122   |
| Fault_ID        | BYTE     | Für OB_Class = 121, 122   |
| BlockType       | USINT    | Für OB_Class = 121, 122   |
| Area            | USINT    | Für OB_Class = 121, 122   |
| DBNr            | DB_ANY   | Für OB_Class = 121, 122   |
| Csg_OBNr        | OB_ANY   | Für OB_Class = 121, 122   |
| Csg_Prio        | USINT    | Für OB_Class = 121, 122   |
| Width           | USINT    | Für OB_Class = 121, 122   |

### Hinweis

Die angegebenen Strukturelemente der Struktur SI\_classic entsprechen inhaltlich genau den temporären Variablen eines OBs, wenn dieser mit der Bausteineigenschaft "Standard" erstellt wurde.

Bitte beachten Sie jedoch, dass die temporären Variablen in den einzelnen OBs andere Namen und andere Datentypen haben können. Beachten Sie weiterhin, dass die Aufrufchnittstelle der OBs zusätzlich Datum und Uhrzeit, zu denen der OB angefordert wurde, enthält.

Die Bits 4 bis 7 des Strukturelements EV\_CLASS enthalten die Ereignisklasse. Hier sind folgende Werte möglich:

- 1: Startereignisse von Standard-OBs
- 2: Startereignisse von Synchronfehler-OBs
- 3: Startereignisse von Asynchronfehler-OBs

Das Strukturelement PRIORITY liefert die zum aktuellen OB gehörige Prioritätsklasse.

Neben diesen beiden Elementen ist noch NUM von Bedeutung. NUM enthält die Nummer des aktuellen OBs bzw. des zuletzt gestarteten Anlauf-OBs.

### Parameter RET\_VAL

Die folgende Tabelle zeigt die Bedeutung der Werte des Parameters RET\_VAL:

| Fehlercode * (W#16#...) | Erläuterung  |
|-------------------------|--|
| 8080                    | Startinformation des aktuellen OBs passt nicht zum angegebenen anwenderdefinierten Datentyp                  |
| 8081                    | Startinformation des aktuellen OBs passt nicht zum angegebenen Systemdatentyp                                |
| 8082                    | Startinformation des zuletzt gestarteten Anlauf-OBs passt nicht zum angegebenen anwenderdefinierten Datentyp |
| 8083                    | Startinformation des zuletzt gestarteten Anlauf-OBs passt nicht zum angegebenen Systemdatentyp               |

### Beispiel

Der zuletzt aufgerufene OB, der noch nicht vollständig abgearbeitet wurde, sei der OB 80, der zuletzt gestartete Anlauf-OB sei der OB 100.

Die folgende Tabelle zeigt die Zuordnung zwischen den Strukturelementen des Parameters TOP\_SI von der Anweisung "RD\_SINFO" und den zugehörigen lokalen Variablen des OB 80.

| TOP_SI Strukturelement | Datentyp | OB 80 - Zugehörige lokale Variable | Datentyp |
|------------------------|----------|------------------------------------|----------|
| EV_CLASS               | BYTE     | OB80_EV_CLASS                      | BYTE     |
| EV_NUM                 | BYTE     | OB80_FLT_ID                        | BYTE     |
| PRIORITY               | BYTE     | OB80_PRIORITY                      | BYTE     |
| NUM                    | BYTE     | OB80_OB_NUMBR                      | BYTE     |
| TYP2_3                 | BYTE     | OB80_RESERVED_1                    | BYTE     |
| TYP1                   | BYTE     | OB80_RESERVED_2                    | BYTE     |
| Z11                    | WORD     | OB80_ERROR_INFO                    | WORD     |
| Z12_3                  | DWORD    | OB80_ERR_EV_CLASS                  | BYTE     |
|                        |          | OB80_ERR_EV_NUM                    | BYTE     |
|                        |          | OB80_OB_PRIORITY                   | BYTE     |
|                        |          | OB80_OB_NUM                        | BYTE     |

Die folgende Tabelle zeigt die Zuordnung zwischen den Strukturelementen des Parameters START\_UP\_SI von der Anweisung "RD\_SINFO" und den zugehörigen lokalen Variablen des OB 100.

| START_UP_SI Strukturelement | Datentyp | OB 100 - Lokale Variable | Datentyp |
|-----------------------------|----------|--------------------------|----------|
| EV_CLASS                    | BYTE     | OB100_EV_CLASS           | BYTE     |
| EV_NUM                      | BYTE     | OB100_STRTUP             | BYTE     |
| PRIORITY                    | BYTE     | OB100_PRIORITY           | BYTE     |
| NUM                         | BYTE     | OB100_OB_NUMBR           | BYTE     |
| TYP2_3                      | BYTE     | OB100_RESERVED_1         | BYTE     |
| TYP1                        | BYTE     | OB100_RESERVED_2         | BYTE     |

| START_UP_SI<br>Strukturelement | Datentyp | OB 100 - Lokale Variable | Datentyp |
|--------------------------------|----------|--------------------------|----------|
| ZI1                            | WORD     | OB100_STOP               | WORD     |
| ZI2_3                          | DWORD    | OB100_STRT_INFO          | DWORD    |

## Siehe auch

Fehlerauswertung mit dem Ausgangsparameter RET\_VAL (Seite 1622)

## LED: LED-Status lesen

### Beschreibung

Mit der Anweisung "LED" können Sie den Status (z. B. "Ein" oder "Aus") einer bestimmten Baugruppen-LED auslesen.

- Über den Parameter LADDR adressieren Sie die CPU oder das Interface.
- Über den Parameter LED wählen Sie die LED der Baugruppe aus, deren aktueller Status über die Anweisung ausgelesen werden soll.
- Der Parameter RET\_VAL gibt bei Aufruf der Anweisung den Status der ausgewählten LED aus. Je nach ausgewählter LED können nur bestimmte Statusinformationen angezeigt werden, beispielsweise haben bestimmte LEDs nur eine Farbe. Die Information, welcher Status für eine bestimmte LED möglich ist, entnehmen sie der Hardware-Dokumentation der jeweiligen Baugruppe.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LED":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| LADDR     | Input       | HW_IO    | E, A, M, L oder Konstante    | Identifikationsnummer der CPU oder des Interfaces. Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder im Interface in der Hardware-Konfiguration hinterlegt.                                      |
| LED       | Input       | UINT     | E, A, M, D, L oder Konstante | Identifikationsnummer der LED: <ul style="list-style-type: none"> <li>• 1: STOP/RUN</li> <li>• 2: ERROR</li> <li>• 3: MAINT (Wartung)</li> <li>• 4: Redundant</li> <li>• 5: Link (grün)</li> <li>• 6: Rx/Tx (gelb)</li> </ul> |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Status der LED  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| RET_VAL | Beschreibung  |
|---------|---|
| 0 bis 9 | <p>LED Status:</p> <ul style="list-style-type: none"> <li>• 0 = LED existiert nicht</li> <li>• 1 = Permanent ausgeschaltet</li> <li>• 2 = Farbe 1 (z. B. bei LED STOP/RUN: grün) permanent ein</li> <li>• 3 = Farbe 2 (z. B. bei LED STOP/RUN: orange) permanent ein</li> <li>• 4 = Farbe 1 blinkt mit 2 Hz</li> <li>• 5 = Farbe 2 blinkt mit 2 Hz</li> <li>• 6 = Farbe 1 und 2 blinken im Wechsel mit 2 Hz</li> <li>• 7 = LED ist aktiv, Farbe 1</li> <li>• 8 = LED ist aktiv, Farbe 2</li> <li>• 9 = LED existiert, Statusinformation ist jedoch nicht verfügbar</li> </ul> |
| 8091    | Über den Parameter LADDR adressierte Baugruppe ist nicht vorhanden.   |
| 8092    | Über den Parameter LADDR wurde eine Baugruppe adressiert, das keine LEDs unterstützt  |
| 8093    | Die am Parameter LED angegebene Identifikationsnummer ist nicht definiert.  |
| 80Bx    | Die am Parameter LADDR angegebene CPU unterstützt die Anweisung "LED" nicht.  |

### Get\_IM\_Data: Identifikations- und Maintenedaten lesen

#### Beschreibung

Die Anweisung "Get\_IM\_Data" liest die Identifikations & Maintenance Daten (I&M-Daten) eines Geräts aus. Über den Parameter LADDR wählen Sie über die Hardware-Kennung das Gerät aus, dessen I&M-Daten Sie auslesen möchten.

Über den Parameter IM\_TYPE definieren Sie, welche Daten über die Anweisung ausgelesen werden sollen. Für das TIA-Portal V12 SP1 wird das Auslesen der I&M 0-Daten unterstützt (IM\_TYPE = 0). Die I&M 0-Daten sind die gerätespezifischen Basisinformationen eines Geräts und enthalten Informationen wie die Herstellerkennung, die Bestellnummer, die Seriennummer oder die Hardware- und Firmwareversion. Diese Informationen werden auch in der Ansicht "Online & Diagnose" eines Geräts angezeigt.

Die gelesenen I&M 0-Daten werden in den am Parameter DATA adressierten Bereich geschrieben. Zur Ablage der Daten können Sie eine Zeichenkette (STRING) oder eine Datenstruktur (ARRAY of CHAR/BYTE oder STRUCT) verwenden:

- Wenn Sie am Parameter DATA eine Zeichenkette (Datentyp STRING) adressieren, passt sich die Länge der Zeichenkette automatisch der Länge der gelesenen I&M-Daten an (maximal 254 Zeichen).
- Wenn Sie am Parameter DATA eine Datenstruktur vom Datentyp STRUCT erstellen, verwenden Sie einen Datenbaustein ohne optimierten Bausteinzugriff (siehe Kategorie "Attribute" der Bausteineigenschaften).
- Wenn Sie am Parameter DATA eine Datenstruktur (ARRAY of CHAR/BYTE oder STRUCT) adressieren, werden die gelesenen I&M-Daten in die einzelnen Komponenten des verwendeten Datentyps geschrieben. Ist die adressierte Datenstruktur länger als die gelesenen Daten, werden die übrigen Komponenten mit Nullen aufgefüllt.



- Für das Auslesen der I&M 0-Daten (Parameter IM\_TYPE = 0) kann auch der Struktur "luM0\_data" verwendet werden.
- Weitere Datentypen sind nicht zulässig. Wird am Parameter DATA ein anderer Datentyp als STRING, ARRAY of BYTE/CHAR oder STRUCT verwendet, wird am Parameter STATUS der Fehlercode 8093 ausgegeben.

Über die Ausgangsparameter BUSY, DONE, ERROR und die mittleren zwei Bytes des Ausgangsparameters STATUS wird der Bearbeitungszustand des Leseauftrags angezeigt.

### Definition: Identifikations- und Maintenance-Daten (I&M)

Identifikations- und Maintenance-Daten (I&M) sind in einer Baugruppe gespeicherte Informationen, welche das Überprüfen der Anlagenkonfiguration, das Auffinden von Hardware-Änderungen einer Anlage oder das Beheben von Fehlern erleichtern.

- Identifikationsdaten (I-Daten) sind gleichbleibende Informationen zu einem Gerät, auf die nur lesend zugegriffen werden kann.
- Maintenance-Daten (M-Daten) sind anlagenabhängige Informationen, wie z. B. Einbauort oder Einbaudatum. Maintenance-Daten werden während der Projektierung erstellt und auf die Baugruppe geschrieben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "Get\_IM\_Data":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| LADDR     | Input       | HW_IO    | E, A, M, D, L oder Konstante | Hardware-Kennung des Geräts. Die Nummer wird automatisch vergeben und ist in den Eigenschaften des Geräts in der Hardware-Konfiguration hinterlegt.                               |
| IM_TYPE   | Input       | UINT     | E, A, M, D, L oder Konstante | Identifikations- und Maintenance-Datennummer<br>Verschalten Sie diesen Parameter mit dem Wert "0".  |
| DATA      | InOut       | VARIANT  | E, A, M, D, L                | Bereich zur Ablage der gelesenen Identifikations- und Maintenance-Daten.  |
| DONE      | Output      | BOOL     | E, A, M, D, L                | Die Anweisung wurde erfolgreich ausgeführt. Die I&M-Daten wurden an den Parameter DATA übertragen.  |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Ausführung der Anweisung abgeschlossen.</li> <li>• 1: Ausführung der Anweisung noch nicht abgeschlossen.</li> </ul> |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter<br><ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige sollten Sie deshalb den Wert von STATUS in einen freien Datenbereich kopieren.   |

Weitere Informationen zu den Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter DATA

Zum Auslesen der I&M 0-Daten (Parameter IM\_TYPE = 0) kann zur Ablage der Daten auch die Struktur "IuM0\_data" verwendet werden:

| Parameter             | Datentyp   | Bytes | Beschreibung                               |
|-----------------------|------------|-------|--|
| Manufacturer_ID       | UINT       | 2     | Herstellerkennung (z. B. "42" für SIEMENS) |
| Order_ID              | CHAR[20]   | 20    | Bestellnummer                              |
| Serial_Number         | CHAR[16]   | 16    | Seriennummer                               |
| Hardware_Revision     | UNIT       | 2     | Hardware Ausgabestand                      |
| Software_Revision     | STRUCT     | 4     | Firmware Ausgabestand                      |
|                       | Type       | CHAR  | -  |
|                       | Functional | UNIT8 | -  |
|                       | Bugfix     | UNIT8 | -  |
|                       | Internal   | UINT8 | -  |
| Revision_Counter      | UINT       | 2     | Revisionszähler                            |
| Profile_ID            | UNIT       | 2     | Profil                                     |
| Profile_Specific_Type | UNIT       | 2     | Geräteklasse                               |
| IM_Version            | UNIT       | 2     | -  |
| I&M_Supported         | UNIT       | 2     | -  |

### Parameter STATUS

| Fehlercode* (W#16#...) | Beschreibung                 |
|------------------------|------------------------------|
| 0                      | Kein Fehler                  |
| 7000                   | Kein Auftrag in Bearbeitung. |

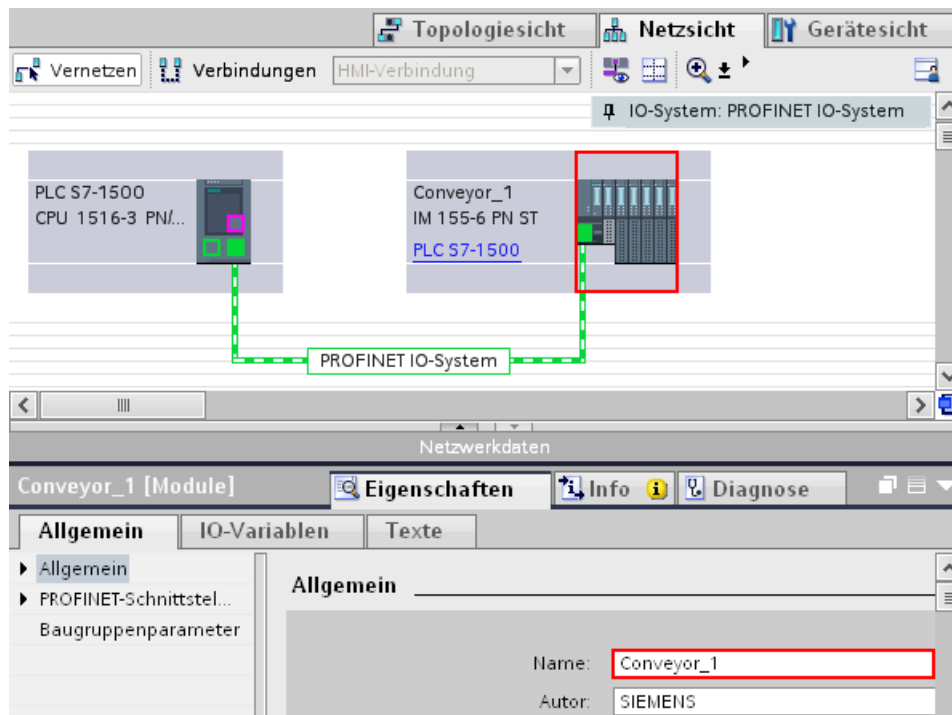
| Fehlercode*<br>(W#16#...) | Beschreibung  |
|---------------------------|---|
| 7001                      | Erster Aufruf der asynchronen Anweisung "Get_IM_Data". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).  |
| 7002                      | Weiterer Aufruf der asynchronen Anweisung "Get_IM_Data". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).  |
| 8091                      | Das am Parameter LADDR adressierte Gerät ist nicht vorhanden.   |
| 8092                      | LADDR adressiert ein Gerät, das die Ausgabe von I&M-Daten nicht unterstützt.  |
| 8093                      | Datentyp am Parameter DATA wird nicht unterstützt.  |
| 80B1                      | Die Anweisung "Get_IM_Data" wird von der verwendeten CPU nicht unterstützt.   |
| 80B2                      | Ungültiger Wert am Parameter IM_TYPE oder der gewählte IM_TYPE wird von der CPU oder dem adressierten Gerät nicht unterstützt.  |
| 8752                      | Der am Parameter DATA angegebene Speicherbereich ist zu klein, um alle I&M-Daten abzulegen. Die gelesenen I&M-Daten werden nur bis zur maximalen Länge des angegebenen Speicherbereichs abgelegt. |

\* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## GET\_NAME: Name einer Baugruppe auslesen

### Beschreibung

Die Anweisung "GET\_NAME" liest den Namen eines IO-Device aus. Der Name wird in der Netzsicht und in den Eigenschaften des IO-Device angezeigt:



Die Auswahl des IO-Device erfolgt über die Hardware-Kennung des PROFINET IO-Systems (am Parameter LADDR) und der Gerätenummer des IO-Device (Parameter STATION\_NR).

Nach erfolgreicher Ausführung der Anweisung wird der Name des IO-Device in den über den Parameter DATA adressierten Bereich geschrieben.

Die Länge des Namens wird am Parameter LEN ausgegeben. Ist der Name länger als der am Parameter DATA angegebene Bereich, wird dieser nur bis zur maximalen Länge des adressierten Bereichs geschrieben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GET\_NAME":

| Parameter  | Deklaration | Datentyp    | Speicherbereich              | Beschreibung  |
|------------|-------------|-------------|------------------------------|---|
| LADDR      | Input       | HW_IOSYSTEM | E, A, M, D, L oder Konstante | Hardware-Kennung des PROFINET-IO Systems. Die Nummer kann in der Netzsicht aus den Eigenschaften des PROFINET-IO Systems übernommen werden.   |
| STATION_NR | Input       | UINT        | E, A, M, D, L oder Konstante | Gerätenummer des IO-Device. Die Gerätenummer kann in der Netzsicht aus den Eigenschaften des IO-Device unter "Ethernet-Adressen" übernommen werden.   |
| DATA       | InOut       | VARIANT     | E, A, M, D, L                | Zeiger auf den Bereich, in den der Name der Baugruppe geschrieben wird.   |
| DONE       | Output      | BOOL        | E, A, M, D, L                | Die Anweisung wurde erfolgreich ausgeführt. Name der Baugruppe an den Bereich am Parameter DATA übertragen.   |
| BUSY       | Output      | BOOL        | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Ausführung der Anweisung abgeschlossen.</li> <li>• 1: Ausführung der Anweisung noch nicht abgeschlossen.</li> </ul>   |
| ERROR      | Output      | BOOL        | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| LEN        | Output      | DINT        | E, A, M, D, L                | Länge des Namens des IO-Device (Anzahl der Zeichen).  |
| STATUS     | Output      | WORD        | E, A, M, D, L                | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Fehlercode*<br>(W#16#...)   | Erläuterung  |
|---|--|
| 0   | Kein Fehler  |
| 7000  | Kein Auftrag in Bearbeitung  |
| 7001  | Erster Aufruf der asynchronen Anweisung "GET_NAME". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).  |
| 7002  | Weiterer Aufruf der asynchronen Anweisung "GET_NAME". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).  |
| 8090  | Der Eingangsparameter LADDR enthält einen unzulässigen Wert. Mögliche Ursachen: <ul style="list-style-type: none"> <li>Die angegebene Nummer ist keine Hardware-Kennung eines PROFINET-IO Systems.</li> <li>Sie haben keine Nummer am Parameter LADDR angegeben.</li> </ul>  |
| 8092  | Der Wert am Parameter LADDR adressiert kein PROFINET-IO System.  |
| 8093  | Datentyp am Parameter DATA wird durch die Anweisung nicht unterstützt.   |
| 8095  | Gerätenummer (Parameter STATION_NR) existiert nicht in dem ausgewählten PROFINET-IO System.  |
| 80B1  | Anweisung wird durch die verwendete CPU nicht unterstützt.   |
| 80C3  | Temporärer Ressourcenfehler: Die CPU bearbeitet momentan das mögliche Maximum an gleichzeitigen Bausteinaufrufen. Vor der Beendigung von mindestens einem Bausteinaufruf kann "GET_NAME" nicht ausgeführt werden.  |
| 8852  | Der an dem Parameter DATA angegebene Bereich ist zu kurz für den vollständigen Namen des IO-Device. Der Name wird bis zur maximal möglichen Länge geschrieben.<br><br>Um den vollständigen Namen auszulesen, verwenden Sie am Parameter DATA einen längeren Datenbereich. Der Bereich muss hierzu mindestens die Anzahl der Zeichen am Parameter LEN umfassen. |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## GetStationInfo: Informationen eines IO-Device auslesen

### Beschreibung

Mit der Anweisung "GetStationInfo" lesen Sie die Informationen eines PROFINET IO-Device aus. Mit der Anweisung können auch die Informationen eines IO-Device gelesen werden, welches sich in einem unterlagerten IO-System befindet (Anbindung über CP/CM).

Das IO-Device wird über die Hardware-Kennung der Station am Parameter LADDR adressiert. Die Hardware-Kennung wird in der Ansicht "Geräte & Netze" in den Eigenschaften der Station angezeigt.

Über den Parameter MODE wählen Sie aus, welche Informationen gelesen werden sollen. Die Anweisung unterstützt in TIA V12 SP1 das Auslesen der IP-Adressparameter nach IPv4 (MODE = 1).

Am Parameter DATA geben Sie den Datenbereich an, in den die gelesenen Adressdaten geschrieben werden sollen. Zur Ablage der IP-Adresse verwenden Sie die Struktur "IF\_CONF\_v4".

Über den Steuerparameter REQ aktivieren Sie das Auslesen der Adressdaten. Hierzu muss das IO-Device erreichbar sein.

Über die Ausgangsparameter BUSY, DONE, ERROR und den Ausgangsparameter STATUS wird der Bearbeitungszustand des Leseauftrags angezeigt.

**Hinweis**

**Adressieren Sie das IO-Device ausschließlich über die Hardware-Kennung der Station**

Die Station, das IO-Device und die PROFINET-Schnittstelle haben jeweils eine eigene Hardware-Kennung. Verwenden Sie für die Anweisung "GetStationInfo" ausschließlich die Hardware-Kennung der Station.

Wird über den Parameter LADDR z. B. eine PROFINET-Schnittstelle adressiert, werden die Adressdaten nicht gelesen und der Fehlercode 8092 ausgegeben.

Um die Adressdaten einer integrierten PROFINET-Schnittstelle oder eines CM/CP im zentralen Aufbau auszulesen, verwenden Sie die Anweisung "RDREC".

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "GetStationInfo":

| Parameter | Deklaration | Datentyp     | Speicherbereich                 | Beschreibung  |
|-----------|-------------|--------------|---------------------------------|---|
| REQ       | Input       | BOOL         | E, A, M, D, L                   | Steuerparameter Request<br>Aktiviert das Auslesen der Informationen bei REQ = "1".  |
| LADDR     | Input       | HW_DEVICE    | E, A, M, D, L<br>oder Konstante | Hardware-Kennung der Station des IO-Device<br>Die Nummer kann in der Netzsicht aus den Eigenschaften der Station oder dem Register "Systemkonstanten" der Standard-Variablen-tabelle übernommen werden. |
| DETAIL    | Input       | HW_SUBMODULE | E, A, M, D, L<br>oder Konstante | Der Parameter DETAIL wird nicht verwendet. Belassen Sie den Parameter unverschaltet.  |
| MODE      | Input       | UNIT         | E, A, M, D, L<br>oder Konstante | Auswahl der zu lesenden Adressdaten<br>Verschalten Sie den Parameter MODE mit "1" (Auslesen der Adressparameter nach IPv4).   |
| DATA      | InOut       | VARIANT      | D, L                            | Zeiger auf den Bereich, in den die Adressdaten des IO-Device geschrieben werden. Verwenden Sie bei MODE = 1 die Struktur "IF_CONF_v4".  |
| DONE      | Output      | BOOL         | E, A, M, D, L                   | Die Anweisung wurde erfolgreich ausgeführt. Die Adressdaten wurden an den Parameter DATA übertragen.  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| BUSY      | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter<br><ul style="list-style-type: none"> <li>• 0: Ausführung der Anweisung abgeschlossen.</li> <li>• 1: Ausführung der Anweisung noch nicht abgeschlossen.</li> </ul>   |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter<br><ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den STATUS in einen freien Datenbereich kopieren.   |

Weitere Informationen zu den Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter DATA

Verwenden Sie am Parameter DATA die Struktur "IF\_CONF\_v4" zur Ablage der Adressparameter nach IPv4.

| Byte    | Parameter        | Datentyp             | Startwert | Beschreibung   |
|---------|------------------|----------------------|-----------|--|
| 0 ... 1 | Id               | UINT                 | 30        | ID der Struktur "IF_CONF_v4".  |
| 2 ... 3 | Length           | UNIT                 | 18        | Länge der gelesenen Daten in BYTE.   |
| 4 ... 5 | Mode             | UNIT                 | 0         | Für die Anweisung "GetStationInfo" nicht relevant (wird auf "0" belassen).   |
| 6 ... 9 | InterfaceAddress | ARRAY [1..4] of BYTE | -         | IP-Adresse des IO-Device im Format IP_V4, z. B. für 192.168.3.10:<br><ul style="list-style-type: none"> <li>• addr[1] = 192</li> <li>• addr[2] = 168</li> <li>• addr[3] = 3</li> <li>• addr[4] = 10</li> </ul> |

| Byte         | Parameter     | Datentyp                   | Startwert | Beschreibung   |
|--------------|---------------|----------------------------|-----------|--|
| 10 ...<br>13 | SubnetMask    | ARRAY<br>[1..4] of<br>BYTE | -         | Subnetzmaske des IO-Device im Format IP_V4, z. B. für 255.255.255.0:<br><ul style="list-style-type: none"> <li>• addr[1] = 255</li> <li>• addr[2] = 255</li> <li>• addr[3] = 255</li> <li>• addr[4] = 0</li> </ul> |
| 14 ...<br>17 | DefaultRouter | ARRAY<br>[1..4] of<br>BYTE | -         | IP-Adresse des Routers im Format IP_V4, z. B. für 192.168.3.1:<br><ul style="list-style-type: none"> <li>• addr[1] = 192</li> <li>• addr[2] = 168</li> <li>• addr[3] = 3</li> <li>• addr[4] = 1</li> </ul>         |

### Parameter STATUS

| Fehlercode*<br>(W#16#...)   | Erläuterung   |
|---|---|
| 0   | Kein Fehler   |
| 7000  | Kein Auftrag in Bearbeitung.  |
| 7001  | Erster Aufruf der asynchronen Anweisung "GetStationInfo". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0).   |
| 7002  | Weiterer Aufruf der asynchronen Anweisung "GetStationInfo". Die Ausführung der Anweisung ist noch nicht abgeschlossen (BUSY = 1, DONE = 0). |
| 8080  | Wert am Parameter MODE wird nicht unterstützt.  |
| 8090  | Die am Parameter LADDR angegebene Hardware-Kennung ist nicht konfiguriert.  |
| 8092  | Der Parameter LADDR adressiert kein PROFINET IO-Device.   |
| 8093  | Ungültiger Datentyp am Parameter DATA.  |
| 80A0  | Angeforderte Information kann nicht gelesen werden.   |
| 80C0  | Adressiertes IO-Device nicht erreichbar.  |
| 80C3  | Die maximale Anzahl simultaner Aufrufe der Anweisung "GetStationInfo" (10 Instanzen) ist zur Zeit erreicht.                                 |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### DeviceStates: Baugruppenzustandsinformation in einem IO-System lesen

#### Beschreibung

Über die Anweisung "DeviceStates" fragen Sie eine bestimmte Statusinformation für alle Baugruppen in einem IO-System ab, d. h.:

- Entweder zu allen IO-Devices innerhalb eines PROFINET-IO Systems
- Oder zu allen DP-Slaves innerhalb eines DP-Mastersystems



Ausgegeben wird als boolescher Wert, für welche Baugruppen der gewählte Status zutrifft. Zum Beispiel können Sie auslesen, welche IO-Devices in einem PROFINET-IO System gerade deaktiviert sind.

Darüber hinaus wird angezeigt, ob die zu lesende Statusinformation für mindestens eines der IO-Devices bzw. DP-Slaves zutrifft.

Die Anweisung kann sowohl im zyklischen OB, als auch in einem Alarm-OB (z. B. OB82 - Diagnosealarm) aufgerufen werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DeviceStates":

| Parameter | Deklaration | Datentyp    | Speicherbereich              | Beschreibung  |
|-----------|-------------|-------------|------------------------------|---|
| LADDR     | Input       | HW_IOSYSTEM | E, A, M, L oder Konstante    | Hardware-Kennung des PROFINET-IO- oder DP-Master-Systems (Beschreibung siehe unten) |
| MODE      | Input       | UINT        | E, A, M, D, L oder Konstante | Auswahl der zu lesenden Statusinformationen (Beschreibung siehe unten)              |
| RET_VAL   | Return      | INT         | E, A, M, D, L                | Status der Anweisung (Beschreibung siehe unten)                                     |
| STATE     | InOut       | VARIANT     | E, A, M, D, L                | Puffer für den Status der IO-Devices, bzw. der DP-Slaves (Beschreibung siehe unten) |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter LADDR

Das PROFINET-IO- oder DP-Master-System wählen Sie an dem Parameter LADDR über die Hardware-Kennung aus.

Die Hardware-Kennung entnehmen Sie:

- Entweder in der Netzsicht den Eigenschaften des PROFINET-IO- oder DP-Master-Systems.
- Oder In der PLC-Variablen-tabelle den aufgelisteten Systemkonstanten mit dem Datentyp HW\_IOSYSTEM.

### Parameter MODE

Die zu lesende Statusinformation wählen Sie über den Parameter MODE aus. Es kann jeweils eine der folgenden Statusinformationen für das gesamte PROFINET-IO- oder DP-Master-System gelesen werden:

- 1: IO-Devices/DP-Slaves sind konfiguriert
- 2: IO-Devices/DP-Slaves gestört
- 3: IO-Devices/DP-Slaves deaktiviert

- 4: IO-Devices/DP-Slaves vorhanden
- 5: IO-Devices/DP-Slaves, bei denen ein Problem aufgetreten ist. Zum Beispiel:
  - Wartung erforderlich oder empfohlen
  - Nicht erreichbar
  - Nicht verfügbar
  - Fehler aufgetreten

### Parameter STATE

Über den Parameter STATE wird der über den Parameter MODE ausgewählte Status der IO-Devices/DP-Slaves ausgegeben.

Trifft für ein IO-Device/DP-Slave der über MODE ausgewählte Status zu, werden am Parameter STATE die folgenden Bits auf "1" gesetzt:

- Bit 0 = 1: Sammelanzeige. Das Bit n von mindestens einem IO-Device/DP-Slave wurde auf "1" gesetzt.
- Bit n = 1: Der über MODE ausgewählte Status trifft für den IO-Device/DP-Slave zu.
  - Bei einem PROFINET-IO System entspricht das Bit n der Gerätenummer des jeweiligen IO-Device (siehe Eigenschaften der PROFINET-Schnittstelle in der Geräte- und Netzsicht)
  - Bei einem PROFIBUS-DP System entspricht das Bit n der PROFIBUS-Adresse des DP-Slaves (siehe Eigenschaften des DP-Slaves in der Geräte- und Netzsicht)

Als Datentyp verwenden Sie BOOL oder Array of BOOL:

- Um ausschließlich das Bit zur Sammelanzeige der Statusinformationen auszugeben, können Sie am Parameter STATE den Datentyp BOOL verwenden.
- Um die Statusinformation zu allen IO-Devices/DP-Slaves auszugeben, verwenden Sie Array of BOOL mit folgender Länge:
  - Bei PROFINET-IO System: 1024 Bit
  - Bei DP-Master System: 128 Bit

### Parameter RET\_VAL

| Fehlercode*<br>(W#16#...) | Beschreibung   |
|---------------------------|--|
| 0                         | Kein Fehler  |
| 8091                      | Hardware-Kennung des Parameters LADDR nicht vorhanden. Überprüfen Sie (z. B. in den Systemkonstanten), ob der Wert für LADDR im Projekt existiert. |
| 8092                      | LADDR adressiert kein PROFINET-IO- oder DP-Master-System.  |
| 8093                      | Ungültiger Datentyp am Parameter STATE.  |
| 80B1                      | Anweisung "DeviceStates" wird durch die CPU nicht unterstützt.   |
| 80B2                      | Der ausgewählte Parameter MODE wird von der verwendeten CPU für das am Parameter LADDR angegebene IO-System nicht unterstützt.                     |

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 8452  | Die vollständigen Status-Informationen passen nicht in die am Parameter STATE projizierte Variable. |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Beispiel - Auslesen der gestörten Stationen eines PROFINET IO Mastersystems

Ein PROFINET IO-System enthält 4 IO-Devices mit den Gerätenummern 1, 2, 3 und 4. Das IO-Device mit der Nummer 2 ist gestört.

Die Anweisung "DeviceStates" wird für das PROFINET IO-System mit MODE = 2 (gestört/nicht gestört) ausgeführt.

Am Parameter STATE werden die folgenden Bits gesetzt:

- Bit 0 = 1: Bei mindestens einem der IO-Devices liegt eine Störung vor.
- Bit 1 = 0: IO-Device mit der Gerätenummer 1 nicht gestört.
- Bit 2 = 1: IO-Device mit der Gerätenummer 2 gestört.
- Bit 3 = 0: IO-Device mit der Gerätenummer 3 nicht gestört.
- Bit 4 = 0: IO-Device mit der Gerätenummer 4 nicht gestört.
- Bit 5 = 0: irrelevant
- Bit 6 = 0: irrelevant
- ...

### Beispiel - Auslesen der gestörten Stationen eines PROFIBUS DP Mastersystems

Ein DP-Mastersystem enthält 4 DP-Slaves mit den PROFIBUS-Adressen 3, 4, 5 und 6. Der DP-Slave mit der Adresse 4 ist gestört.

Die Anweisung "DeviceStates" wird für das DP-Mastersystem mit MODE = 2 (gestört/nicht gestört) ausgeführt.

Am Parameter STATE werden die folgenden Bits gesetzt:

- Bit 0 = 1: Bei mindestens einem der DP-Slaves liegt eine Störung vor.
- Bit 1 = 0: irrelevant
- Bit 2 = 0: irrelevant
- Bit 3 = 0: DP-Slave mit der Adresse 3 nicht gestört.
- Bit 4 = 1: DP-Slave mit der Adresse 4 gestört.
- Bit 5 = 0: DP-Slave mit der Adresse 5 nicht gestört.
- Bit 6 = 0: DP-Slave mit der Adresse 6 nicht gestört.
- Bit 7 = 0: irrelevant
- Bit 8 = 0: irrelevant
- ...

## ModuleStates: Modulzustandsinformation einer Baugruppe lesen

### Beschreibung

Über die Anweisung "ModuleStates" lesen Sie Statusinformationen der Module eines PROFINET IO-Device oder PROFIBUS DP-Slave aus.

Ausgegeben wird als boolescher Wert, für welche Module der gewählte Status zutrifft. Zum Beispiel können Sie auslesen, welche Module eines PROFINET IO-Device gerade deaktiviert sind.

Darüber hinaus wird angezeigt, ob die zu lesende Statusinformation für mindestens eines der Module zutrifft.

Die Anweisung kann sowohl im zyklischen OB, als auch in einem Alarm-OB (z. B. OB82 - Diagnosealarm) aufgerufen werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ModuleStates":

| Parameter | Deklaration | Datentyp  | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------|------------------------------|---|
| LADDR     | Input       | HW_DEVICE | E, A, M, D, L oder Konstante | Hardware-Kennung der Station (Beschreibung siehe unten)                           |
| MODE      | Input       | UINT      | E, A, M, D, L oder Konstante | Auswahl der zu lesenden Statusinformationen der Module (Beschreibung siehe unten) |
| RET_VAL   | Return      | INT       | E, A, M, D, L                | Status der Anweisung (Beschreibung siehe unten)                                   |
| STATE     | InOut       | VARIANT   | E, A, M, D, L                | Puffer für den Status der Module (Beschreibung siehe unten)                       |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter LADDR

Das IO-Device oder den DP-Slave wählen Sie an dem Parameter LADDR über die Hardware-Kennung der Station aus.

Die Hardware-Kennung entnehmen Sie:

- Entweder in der Netzsicht den Eigenschaften der Station des IO-Device oder des DP-Slave.
- Oder in der PLC-Variablen-tabelle den aufgelisteten Systemkonstanten mit dem Datentyp HW\_DEVICE (für ein IO-Device) oder mit dem Datentyp HW\_DPSLAVE (für ein DP-Slave).

### Parameter MODE

Die zu lesende Statusinformation wählen Sie über den Parameter MODE aus. Es kann jeweils eine der folgenden Statusinformationen für die Module gelesen werden:

- 1: Module sind konfiguriert
- 2: Module gestört

- 3: Module deaktiviert
- 4: Module vorhanden
- 5: In den Modulen ist ein Problem aufgetreten. Zum Beispiel:
  - Wartung erforderlich oder empfohlen
  - Nicht erreichbar
  - Nicht verfügbar
  - Fehler aufgetreten

## Parameter STATE

Über den Parameter STATE wird der über den Parameter MODE ausgewählte Status der Module ausgegeben.

Trifft für eine Baugruppe der über MODE ausgewählte Status zu, werden die folgenden Bits auf "1" gesetzt:

- Bit 0 = 1: Sammelanzeige. Das Bit n von mindestens einem Modul wurde auf "1" gesetzt.
- Bit n = 1: Der über MODE ausgewählte Status trifft für das Modul im Slot n-1 zu (Beispiel: Bit 3 = Slot 2).

Als Datentyp verwenden Sie BOOL oder Array of BOOL:

- Um ausschließlich das Bit zur Sammelanzeige der Statusinformationen auszugeben, können Sie am Parameter STATE den Datentyp BOOL verwenden.
- Um die Statusinformation zu allen Modulen auszugeben, verwenden Sie Array of BOOL mit einer Länge von 128 Bit.

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 0   | Kein Fehler  |
| 8091  | Hardware-Kennung des Parameters LADDR nicht vorhanden. Überprüfen Sie (z. B. in den Systemkonstanten), ob der Wert für LADDR im Projekt existiert. |
| 8092  | LADDR adressiert kein IO-Device oder DP-Slave.   |
| 8093  | Ungültiger Datentyp am Parameter STATE.  |
| 80B1  | Die Anweisung "ModuleStates" wird durch die CPU nicht unterstützt.   |
| 80B2  | Der ausgewählte Parameter MODE wird von der verwendeten CPU für das IO-Device / den DP-Slave am Parameter LADDR nicht unterstützt.                 |
| 8452  | Die vollständigen Status-Informationen passen nicht in die am Parameter STATE projektierte Variable.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Beispiel

Ein IO-Device enthält 4 Module in den Slots 1 bis 4. Das Modul in Slot 2 ist gestört.

Die Anweisung "ModuleStates" wird für das IO-Device mit MODE = 2 (gestört/nicht gestört) ausgeführt.

Am Parameter STATE werden die folgenden Bits gesetzt:

- Bit 0 = 1: Bei mindestens einem Modul liegt eine Störung vor.
- Bit 1 = 0: Slot-Nummer 0 (durch IO-Device belegt)
- Bit 2 = 0: Modul in Slot Nummer 1 nicht gestört.
- Bit 3 = 1: Modul in Slot Nummer 2 gestört.
- Bit 4 = 0: Modul in Slot Nummer 3 nicht gestört.
- Bit 5 = 0: Modul in Slot Nummer 4 nicht gestört.
- Bit 6 = 0: irrelevant
- Bit 7 = 0: irrelevant

## GEN\_DIAG: Diagnoseinformation generieren

### Beschreibung

Die Anweisung "GEN\_DIAG" generiert Diagnoseinformationen für Hardware-Komponenten anderer Hersteller zur Verwendung in der Diagnose des TIA-Portals. Zur Verwendung der Anweisung muss die vom Hersteller mitgelieferte GSD(GSDL/GSDML)-Datei zuvor installiert werden.

Die Anweisung generiert alle Diagnose-Ereignisse (auch für Wartungsbedarf).

- Über den Parameter LADDR wählen Sie die Hardware-Komponente aus, für die Sie ein Diagnose-Ereignis generieren möchten.
- Über den Parameter MODE spezifizieren Sie, ob es sich um ein kommendes oder gehendes Ereignis handelt.
- Über den Parameter DiagEvent definieren Sie das Diagnose-Ereignis in der Struktur DiagnosticDetail. Die Struktur wird automatisch in der lokalen Schnittstelle des Bausteins angelegt, wenn Sie am Parameter DiagEvent eine Variable definieren.

Die Bereitstellung der Diagnoseinformationen erfolgt synchron. Das Übermitteln der Diagnoseinformationen und die Ausgabe von Meldungen erfolgen asynchron.

|  |
|--|
| <b>ACHTUNG</b>   |
| <b>Failsafe-spezifische Fehlermeldungen nicht zulässig</b>   |
| Wenn Sie am Parameter DiagEvent Failsafe-spezifischen Diagnoseinformationen definieren wird dies durch die Anweisung geprüft und der Fehlercode 80A1 ausgegeben. |

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GEN\_DIAG":

| Parameter | Deklaration | Datentyp         | Speicherbereich              | Beschreibung  |
|-----------|-------------|------------------|------------------------------|---|
| LADDR     | Input       | HW_ANY           | E, A, M, D, L oder Konstante | Identifikationsnummer der Hardware-Komponente   |
| MODE      | Input       | UINT             | E, A, M, D, L oder Konstante | Auswahl der Kommend-/Gehend-Information: <ul style="list-style-type: none"> <li>• 1: Das spezifizierte Diagnose-Ereignis ist ein kommendes Ereignis</li> <li>• 2: Das spezifizierte Diagnose-Ereignis ist ein gehendes Ereignis</li> <li>• 3: Alle Diagnose-Ereignisse sind gehend. Hiernach liegt bei der Hardware-Komponente keine Störung vor (grünes Diagnosesymbol). Der Parameter DiagEvent wird bei MODE = 3 nicht ausgewertet.</li> </ul> |
| DiagEvent | InOut       | DiagnosticDetail | L                            | Spezifiziert das Diagnose-Ereignis (siehe "Parameter DiagEvent").   |
| RET_VAL   | Return      | INT              | E, A, M, D, L                | Status der Anweisung / Fehlermeldung (siehe "Parameter RET_VAL").   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter DiagEvent

Die Struktur DiagnosticDetail ist ein Systemdatentyp zur Spezifizierung des Diagnose-Ereignisses mit folgendem Aufbau:

| Parameter        | Datentyp | Beschreibung  |
|------------------|----------|---|
| DiagnosticDetail | Struct   |   |
| ChannelInfo      | WORD     | Kanaleigenschaften (0...7)  |
| ALID             | UINT     | Lokale ID der Meldung. Anhand der ID wird die Meldung eindeutig identifiziert.                                    |
| TextID           | UINT     | ID eines Meldungstexts in einer Textliste.  |
| ChannelNumber    | UINT     | Herstellerspezifische Kanalnummer (0x0000 — 0x7FFF)   |
| Addval_0         | DWORD    | Platzhalter für zusätzliche Informationen. Der Wert / die Liste der Werte ist abhängig von dem Verbindungsfehler. |
| TextID2          | UINT     | Texte für die Reaktion der CPU (Betriebszustand, OB-Aufrufe, etc.).   |
| LADDR            | HW_ANY   | Identisch zu dem Parameter LADDR.   |
| TextListId       | UINT     | <ul style="list-style-type: none"> <li>• 0: Keine Textliste</li> <li>• ≠0: ID der Textliste</li> </ul>            |

| Parameter         | Datentyp | Beschreibung   |
|-------------------|----------|--|
| Channel Direction | UINT     | <ul style="list-style-type: none"> <li>• 0000: irrelevant</li> <li>• FFF1: Input</li> <li>• FFF2: Output</li> <li>• FFF3: In-/Output</li> </ul>  |
| Addval_1          | DWORD    | Platzhalter für zusätzliche Informationen zu dem Kanalfehler (abhängig von der GSD-Datei).<br>Zu den Kanalfehlertypen siehe auch: IEC-Norm 61158 (PROFINET IO Type 10 and PROFIBUS DP Type 3). |

**Parameter RET\_VAL**

| Fehlercode* (W#16#...) | Erläuterung  |
|------------------------|--|
| 0                      | Kein Fehler  |
| 8080                   | Wert am Parameter MODE nicht unterstützt.  |
| 8090                   | Identifikationsnummer für Hardware-Komponente am Parameter LADDR nicht vorhanden.  |
| 8091                   | Die Generierung von Diagnoseinformationen ist für die am Parameter LADDR adressierte Hardware-Komponente nicht möglich.  |
| 80A1                   | <ul style="list-style-type: none"> <li>• Inhalte der Struktur DiagnosticsDetail am Parameter DiagEvent ungültig oder inkonsistent.</li> <li>• Failsafe-spezifische Diagnoseinformationen am Parameter DiagEvent definiert (nicht zulässig).</li> </ul> |
| 80A4                   | Adressierte Hardware-Komponente nicht erreichbar.  |
| 80C1                   | Unzureichende Ressourcen für parallele Ausführung.   |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**GET\_DIAG: Diagnoseinformation lesen**

**Beschreibung**

Über die Anweisung "GET\_DIAG" können Sie die Diagnoseinformationen eines Hardwareobjekts auslesen. Die Auswahl des Hardwareobjekts erfolgt über den Parameter LADDR. Über den Parameter MODE wählen Sie aus, welche Diagnoseinformationen ausgelesen werden sollen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "GET\_DIAG":

| Parameter | Deklaration | Datentyp      | Speicherbereich              | Beschreibung   |
|-----------|-------------|---------------|------------------------------|--|
| MODE      | Input       | UINT          | E, A, M, D, L oder Konstante | Über den Parameter MODE wählen Sie aus, welche Diagnosedaten ausgegeben werden sollen. |
| LADDR     | Input       | HW_ANY (WORD) | E, A, M, L oder Konstante    | Hardwareerkennung des Geräts.  |



| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| RET_VAL   | Return      | INT      | E, A, M, D, L   | Status der Anweisung  |
| CNT_DIAG  | Output      | UINT     | E, A, M, D, L   | Anzahl der ausgegebenen Diagnosedetails   |
| DIAG      | InOut       | VARIANT  | E, A, M, D, L   | Diagnoseinformationen entsprechend dem ausgewählten Modus, siehe folgende Tabelle                               |
| DETAIL    | InOut       | VARIANT  | E, A, M, D, L   | Diagnosedetails entsprechend dem ausgewählten Modus<br>Parameter ist ausgeblendet (nur verwendet bei MODE = 3). |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter MODE

Abhängig von dem Wert am Parameter MODE, werden an den Ausgangsparametern DIAG, CNT\_DIAG und DETAIL unterschiedliche Diagnosedaten ausgegeben.

| MODE | Beschreibung  | DIAG   | CNT_DIAG                                  | DETAIL  |
|------|---|--|---|---|
| 0    | Ausgabe aller unterstützten Diagnoseinformationen für eine Baugruppe als DWORD, wobei Bit X=1 anzeigt, dass der Modus X unterstützt wird. | Bitfolge der unterstützten Modi als DWORD, wobei Bit X=1 anzeigt, dass der Modus X unterstützt wird.           | 0   | -   |
| 1    | Ausgabe des eigenen Status des adressierten Hardware-Objekts.   | Diagnosestatus.<br>Ausgabe des eigenen Status des adressierten Hardware-Objekts entsprechend der Struktur DIS. | 0   | -   |
| 2    | Ausgabe des Status aller untergeordneten Baugruppen des adressierten Hardware-Objekts.  | Diagnosestatus. Ausgabe entsprechend der Struktur DNN.   | 0   | -   |
| 3    | Ausgabe des eigenen Status des adressierten Hardware-Objekts sowie Detailinformationen zur Kanaldiagnose.                                 | Diagnosestatus. Ausgabe des eigenen Status des adressierten Hardware-Objekts entsprechend der Struktur DIS.    | Anzahl der Baugruppenzustandsinformation. | Baugruppenzustandsinformation entsprechend der Struktur DiagnosticDetail. |

### Struktur DIS

Bei Parameter MODE = 1 oder =3 werden die Diagnoseinformationen entsprechend der Struktur DIS ausgegeben. In diesem Fall geben Sie bei der Variablendeklaration den Systemdatentyp "DIS" als Datentyp an.

Die folgende Tabelle zeigt die Bedeutung der einzelnen Parameterwerte.

| Parameter            | Datentyp   | Wert           | Beschreibung  |
|----------------------|--|----------------|---|
| MaintenanceState     | DWORD  | Enum           |   |
|                      |  | 0              | Kein Wartungsbedarf   |
|                      |  | 1              | Die Baugruppe bzw. das Gerät ist deaktiviert.   |
|                      |  | 2              | -   |
|                      |  | 3              | -   |
|                      |  | 4              | -   |
|                      |  | 5              | Wartungsbedarf  |
|                      |  | 6              | Wartungsanforderung   |
|                      |  | 7              | Fehler  |
|                      |  | 8              | Status unbekannt / Fehler in untergeordneter Baugruppe  |
|                      |  | 9              | -   |
| 10                   | Eingänge/Ausgänge stehen nicht zur Verfügung.  |                |   |
| ComponentStateDetail | DWORD  | Bitarray       | Status der Baugruppenmodule: <ul style="list-style-type: none"> <li>• Bit 0 bis 15: Statusmeldung des Moduls</li> <li>• Bit 16 bis 31: Statusmeldung der CPU</li> </ul>   |
|                      |  | 0 bis 2 (enum) | Zusatzinformation: <ul style="list-style-type: none"> <li>• Bit 0: Keine Zusatzinformationen</li> <li>• Bit 1: Keine Übernahme zulässig</li> </ul>  |
|                      |  | 3              | Bit 3 = 1: Mindestens ein Kanal unterstützt Qualifier für Diagnose  |
|                      |  | 4              | Bit 4 = 1: Wartungsbedarf bei mindestens einem Kanal oder einer Komponente.   |
|                      |  | 5              | Bit 5 = 1: Wartungsanforderung bei mindestens einem Kanal oder einer Komponente.  |
|                      |  | 6              | Bit 6 = 1: Fehler bei mindestens einem Kanal oder einer Komponente.   |
|                      |  | 7 bis 10       | reserviert (immer "0")  |
|                      |  | 11 bis 14      | <ul style="list-style-type: none"> <li>• Bit 11 = 1: PNIO - Submodul korrekt</li> <li>• Bit 12 = 1: PNIO - Ersatzmodul</li> <li>• Bit 13 = 1: PNIO - Falsches Modul</li> <li>• Bit 14 = 1: PNIO - Modul nicht gesteckt</li> </ul> |
|                      |  | 15             | reserviert (immer "0")  |
| 16 bis 31            | Durch die CPU generierte Statusinformationen für Module: <ul style="list-style-type: none"> <li>• Bit 16 = 1: Modul deaktiviert</li> <li>• Bit 17 = 1: CiR-Operation aktiv</li> <li>• Bit 18 = 1: Eingang nicht verfügbar</li> <li>• Bit 19 = 1: Ausgang nicht verfügbarBit</li> <li>• 20 = 1: Überlauf DiagnosepufferBit</li> <li>• 21 = 1: Diagnose nicht verfügbarBit</li> <li>• 22 - 31: reserviert (immer 0)</li> </ul> |                |   |
| OwnState             | UINT   | Enum           | Der Wert des Parameters Ownstate beschreibt den Wartungszustand der Baugruppe.  |
|                      |  | 0              | Keine Störung   |
|                      |  | 1              | Die Baugruppe bzw. das Gerät ist deaktiviert.   |
|                      |  | 2              | Wartungsbedarf  |
|                      |  | 3              | Wartungsanforderung   |

| Parameter      | Datentyp | Wert     | Beschreibung   |
|----------------|----------|----------|--|
|                |          | 4        | Fehler   |
|                |          | 5        | Die Baugruppe bzw. das Gerät ist von der CPU aus nicht erreichbar (gültig für Baugruppen und Geräte unterhalb einer CPU).            |
|                |          | 6        | Eingänge/Ausgänge stehen nicht zur Verfügung.  |
|                |          | 7        | -  |
| IOState        | WORD     | Bitarray | E/A-Status der Baugruppe   |
|                |          | 0        | Bit 0 = 1: Kein Wartungsbedarf   |
|                |          | 1        | Bit 1 = 1: Die Baugruppe bzw. das Gerät ist deaktiviert.   |
|                |          | 2        | Bit 2 = 1: Wartungsbedarf  |
|                |          | 3        | Bit 3 = 1: Wartungsanforderung   |
|                |          | 4        | Bit 4 = 1: Fehler  |
|                |          | 5        | Bit 5 = 1: Die Baugruppe bzw. das Gerät ist von der CPU aus nicht erreichbar (gültig für Baugruppen und Geräte unterhalb einer CPU). |
|                |          | 6        | Eingänge/Ausgänge stehen nicht zur Verfügung.  |
|                |          | 7        | Qualifier; Bit 7 = 1, wenn Bit 0, 2, oder 3 gesetzt sind   |
|                |          | 8 bis 15 | reserviert (immer "0")   |
| OperatingState | UINT     | Enum     |  |
|                |          | 0        | -  |
|                |          | 1        | In STOP / Firmware-Update  |
|                |          | 2        | In STOP / Urlöschen  |
|                |          | 3        | In STOP / Selbststart  |
|                |          | 4        | In STOP  |
|                |          | 5        | Urlöschen  |
|                |          | 6        | In START   |
|                |          | 7        | In RUN   |
|                |          | 8        | -  |
|                |          | 9        | In HALT  |
|                |          | 10       | -  |
|                |          | 11       | -  |
|                |          | 12       | Baugruppe defekt   |
|                |          | 13       | -  |
|                |          | 14       | Keine Strom  |
|                |          | 15       | CiR  |
|                |          | 16       | In STOP / ohne ODIS  |
|                |          | 17       | In   |
|                |          | 18       |  |
| 19             |          |          |  |
| 20             |          |          |  |

### Struktur DNN

Bei Parameter MODE = 2 werden die Details der Diagnoseinformationen entsprechend der Struktur DNN ausgegeben. In diesem Fall geben Sie bei der Variablendeklaration den Systemdatentyp "DNN" als Datentyp an.

Die folgende Tabelle zeigt die Bedeutung der einzelnen Parameterwerte.

| Parameter          | Datentyp | Wert     | Beschreibung   |
|--------------------|----------|----------|--|
| SubordinateState   | UINT     | Enum     | Status der untergeordneten Baugruppe (siehe Parameter OwnState der Struktur DIS)   |
| SubordinateIOState | WORD     | Bitarray | Status der Ein- und Ausgänge der untergeordneten Baugruppe (siehe Parameter IO State der Struktur DIS)   |
| DNNmode            | WORD     | Bitarray | <ul style="list-style-type: none"> <li>• Bit 0 = 0: Diagnose aktiviert</li> <li>• Bit 0 = 1: Diagnose deaktiviert</li> <li>• Bit 1 bis 15: reserviert</li> </ul> |

### Struktur DiagnosticDetail

Bei Parameter MODE = 3 werden die Details der Diagnoseinformationen entsprechend der Struktur DiagnosticDetail ausgegeben. In diesem Fall geben Sie bei der Variablendeklaration den Systemdatentyp "DiagnosticDetail" als Datentyp an.

Die folgende Tabelle zeigt die Bedeutung der einzelnen Parameterwerte.

| Parameter        | Datentyp | Beschreibung  |
|------------------|----------|---|
| ChannelInfo      | WORD     | Kanaleigenschaften (0..7)   |
| ALID             | UInt     | Identifikations-ID des Alarms   |
| TextID           | UNIT     | ID eines Meldungstexts in einer Textliste.  |
| ChannelNumber    | UINT     | Herstellerspezifische Kanalnummer (0x0000 — 0x7FFF)   |
| Addval_0         | DWORD    | Platzhalter für zusätzliche Informationen. Der Wert / die Liste der Werte ist abhängig von dem Verbindungsfehler.   |
| TextID2          | UInt     | Texte für die Reaktion der CPU (Betriebszustand, OB-Aufrufe, etc.).   |
| LADDR            | HW_ANY   | Identisch zu dem Parameter LADDR.   |
| TextListId       | UInt     | <ul style="list-style-type: none"> <li>• 0: Keine Textliste</li> <li>• ≠0: ID der Textliste</li> </ul>  |
| ChannelDirection | UInt     | <ul style="list-style-type: none"> <li>• 000: Herstellerspezifisch</li> <li>• 001: Input</li> <li>• 002: Output</li> <li>• 003: In-/Output</li> <li>• 004 - 007: Reserviert</li> </ul>                    |
| AddVal_1         | DWORD    | <p>Platzhalter für zusätzliche Informationen zu dem Kanalfehler (abhängig von der GSD-Datei).</p> <p>Zu den Kanalfehlertypen siehe auch: IEC-Norm 61158 (PROFINET IO Type 10 and PROFIBUS DP Type 3).</p> |

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0   | Kein Fehler   |
| n   | Datenbereich am Parameter DETAIL ist zu klein. Nicht alle Details der Diagnosedaten können ausgegeben werden. |
| 8080  | Wert am Parameter MODE nicht unterstützt.   |
| 8081  | Typ am Parameter DIAG wird mit dem ausgewählten Modus (Parameter MODE) nicht unterstützt.                     |
| 8082  | Typ am Parameter DETAIL wird mit dem ausgewählten Modus (Parameter MODE) nicht unterstützt.                   |
| 8090  | LADDR nicht vorhanden   |
| 80C1  | Unzureichende Ressourcen für parallele Ausführung.  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### 9.7.3.10 Puls

#### CTRL\_PWM: Pulsweitenmodulation

##### Beschreibung

Mit der Anweisung "CTRL\_PWM" können Sie einen von der CPU unterstützten Impulsgenerator über die Software aktivieren und deaktivieren.

---

##### Hinweis

Die Parametrierung eines Impulsgenerators erfolgt ausschließlich in der Gerätekonfiguration und nicht über die Anweisung "CTRL\_PWM". Eine Parameteränderung, die auf der CPU wirksam werden soll, ist daher nur im Betriebszustand STOP der CPU möglich.

---

Die Hardware-Kennung des Impulsgenerators, den Sie über die Anweisung steuern wollen, geben Sie am Eingang PWM ein. Eine Voraussetzung für die fehlerfreie Ausführung der Anweisung ist, dass der angegebene Impulsgenerator in der Hardware-Konfiguration frei geschaltet ist.

Am Eingang PWM können nur Variablen angegeben werden, die vom Datentyp HW\_PWM sind. Der Hardware-Datentyp HW\_PWM hat eine Länge von einem WORD.

Der Impulsgenerator wird aktiviert, wenn das Bit am Eingang ENABLE der Anweisung gesetzt ist. Wenn ENABLE den Wert TRUE hat, erzeugt der Impulsgenerator Impulse, die die in der Gerätekonfiguration festgelegten Eigenschaften haben. Wenn das Bit am Eingang ENABLE zurückgesetzt wird oder die CPU in STOP geht, wird der Impulsgenerator deaktiviert und es werden keine Impulse mehr erzeugt.

Die Anweisung "CTRL\_PWM" wird nur ausgeführt, wenn der Signalzustand am Eingang EN "1" ist.

Da die S7-1200 den Impulsgenerator aktiviert, wenn die Anweisung "CTRL\_PWM" ausgeführt wird, hat BUSY bei S7-1200 stets den Wert FALSE.

Der Freigabeausgang ENO wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und keine Fehler während der Bearbeitung der Anweisung aufgetreten sind.

**Hinweis**

**Verwendung der Forcetabelle bei PWM und PTO**

Digitale Ein- und Ausgänge, die für PWM und PTO verwendet werden, können nicht geforct werden. Digitale Ein- und Ausgänge, die über die Gerätekonfiguration zugewiesen wurden, können weder durch die Forcetabelle noch durch die Beobachtungstabelle gesteuert werden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "CTRL\_PWM":

| Parameter | Deklaration | Datentyp      | Speicherbereich              | Beschreibung   |
|-----------|-------------|---------------|------------------------------|--|
| PWM       | Input       | HW_PWM (WORD) | E, A, M, L oder Konstante    | Hardware-Kennung des Impulsgenerators  |
| ENABLE    | Input       | BOOL          | E, A, M, D, L oder Konstante | Der Impulsgenerator wird bei ENABLE = TRUE aktiviert und bei ENABLE = FALSE deaktiviert. |
| BUSY      | Output      | BOOL          | E, A, M, D, L                | Bearbeitungsstatus   |
| STATUS    | Output      | WORD          | E, A, M, D, L                | Status der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter STATUS**

| Fehlercode* (W#16#...)   | Beschreibung                                   |
|--|--|
| 0  | Keine Fehler                                   |
| 80A1   | Hardware-Kennung des Impulsgenerators ungültig |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### 9.7.3.11 Rezepturen und Data Logging

#### Rezeptfunktionen

#### RecipeExport: Rezept exportieren

#### Beschreibung

Die Anweisung "RecipeExport" exportiert die Rezept-Daten aus einem Datenbaustein in eine CSV-Datei auf die Memory-Card der CPU.

Der Export wird durch den Parameter REQ angestoßen. Während des Exports wird der Parameter BUSY auf "1" gesetzt. Beim Exportieren wird die CSV-Datei in dem Ordner "Recipes" im Hauptverzeichnis der Memory-Card erstellt. Als Dateiname der erstellten CSV-Datei wird der Name des Datenbausteins verwendet. Wenn bereits eine CSV-Datei mit dem gleichen Namen vorhanden ist, wird diese beim Export überschrieben.

Nach Ausführung der Anweisung wird BUSY zurück auf "0" gesetzt und der Abschluss der Ausführung am Parameter DONE mit "1" angezeigt. Sollte bei der Ausführung ein Fehler auftreten, wird dieser über die Parameter ERROR und STATUS gemeldet.

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RecipeExport":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung  |
|-----------|-------------|----------|--|---|
| REQ       | Input       | BOOL     | E, A, M, D, L, T, Z<br>oder Konstante<br>(T und Z sind nur in<br>KOP und FUP<br>verfügbar) | Steuerparameter REQUEST: Aktiviert den Export bei steigender Flanke.  |
| RECIPE_DB | InOut       | VARIANT  |  | Zeiger auf den Rezept-Datenbaustein. Zum Aufbau des Datenbausteins siehe: Aufbau eines Rezept-DBs (Seite 2588)  |
| DONE      | Output      | BOOL     | E, A, M, D, L  | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>• 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul> |
| BUSY      | Output      | BOOL     | E, A, M, D, L  | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Die Anweisung wird nicht ausgeführt.</li> <li>• 1: Die Anweisung wird ausgeführt.</li> </ul>                                |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter<br><ul style="list-style-type: none"> <li>• 0: Weder Warnung noch Fehler.</li> <li>• 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul> |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Zustandsparameter<br>Siehe Tabelle "Parameter STATUS".   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

| Fehlercode*<br>(W#16#...) | Erläuterung  |
|---------------------------|--|
| 0                         | Kein Fehler aufgetreten  |
| 7000                      | Keine Auftragsbearbeitung aktiv  |
| 7001                      | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.  |
| 7002                      | Zwischenaufwurf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".  |
| 8090                      | Dateiname der CSV-Datei enthält ungültige Zeichen. Der Dateiname für die CSV-Datei ist identisch mit der des Datenbausteins. |
| 8091                      | Die mittels RECIPE_DB referenzierte Datenstruktur kann nicht verarbeitet werden.   |
| 8092                      | Datenstruktur am Parameter RECIPE_DB überschreitet 5000 Byte.  |
| 80B3                      | Kein ausreichender Speicherplatz auf der Memory-Card oder dem internen Ladespeicher.   |
| 80B4                      | Die Memory-Card ist schreibgeschützt.  |
| 80C0                      | CSV-Datei vorübergehend gesperrt.  |
| 80C1                      | Datenbaustein vorübergehend gesperrt.  |

\* Die Fehlercodes im Programmreditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### RecipeImport: Rezept importieren

#### Beschreibung

Die Anweisung "RecipeImport" importiert die Rezept-Daten aus einer CSV-Datei auf der Memory-Card in den Datenbaustein am Parameter RECIPE\_DB. Die Werte in dem Datenbaustein werden dabei überschrieben.

Beachten Sie für den Import der CSV-Datei Folgendes:

- Die CSV-Datei muss sich in dem Verzeichnis "Recipes" der Memory Card befinden.
- Der Name der CSV-Datei muss mit dem Namen des Datenbausteins am Parameter RECIPE\_DB identisch sein.



- Die erste Zeile (Kopfzeile) der CSV-Datei enthält die Namen der Rezeptbestandteile (siehe auch: Aufbau eines Rezept-DBs (Seite 2588)). Die erste Zeile wird beim Import ignoriert. Ein Abgleich der Namen der Rezeptbestandteile in der CSV-Datei und dem Datenbaustein wird beim Import nicht durchgeführt.
- Der jeweils erste Wert in jeder Zeile der CSV-Datei ist die Indexnummer des Rezepts. Die einzelnen Rezepte werden in Reihenfolge des Index importiert. Der Index in der CSV-Datei muss hierzu aufsteigend sein und darf keine Lücken enthalten (ist dies der Fall, wird am Parameter STATUS die Fehlermeldung 80B0 ausgegeben).
- Die CSV-Datei darf nicht mehr Rezepturdatensätze enthalten, als im Datenbaustein vorgesehen. Die maximale Anzahl an Datensätzen ist im Datenbaustein durch die Array-Grenzen vorgegeben.

Der Import wird durch den Parameter REQ angestoßen. Während des Imports wird der Parameter BUSY auf "1" gesetzt. Nach Ausführung der Anweisung wird BUSY zurück auf "0" gesetzt und der Abschluss der Ausführung am Parameter DONE mit "1" angezeigt. Sollte bei der Ausführung ein Fehler auftreten, wird dieser über die Parameter ERROR und STATUS gemeldet.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RecipeImport":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung  |
|-----------|-------------|----------|--|---|
| REQ       | Input       | BOOL     | E, A, M, D, L, T, Z<br>oder Konstante<br>(T und Z sind nur in<br>KOP und FUP<br>verfügbar) | Steuerparameter REQUEST: Aktiviert den Import bei steigender Flanke   |
| RECIPE_DB | InOut       | VARIANT  | D  | Zeiger auf den Rezept-Datenbaustein. Zum Aufbau des Datenbausteins siehe:   |
| DONE      | Output      | BOOL     | E, A, M, D, L  | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>• 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>                 |
| BUSY      | Output      | BOOL     | E, A, M, D, L  | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Die Anweisung wird nicht ausgeführt.</li> <li>• 1: Die Anweisung wird ausgeführt.</li> </ul>  |
| ERROR     | Output      | BOOL     | E, A, M, D, L  | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Weder Warnung noch Fehler.</li> <li>• 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul> |
| STATUS    | Output      | WORD     | E, A, M, D, L  | Zustandsparameter<br>Siehe Tabelle "Parameter STATUS".  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

| Fehlercode*<br>(W#16#...) | Erläuterung   |
|---------------------------|---|
| 0                         | Kein Fehler aufgetreten   |
| 7000                      | Keine Auftragsbearbeitung aktiv   |
| 7001                      | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.   |
| 7002                      | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".  |
| 8090                      | Der Dateiname enthält ungültige Zeichen.  |
| 8092                      | Keine passende CSV-Datei für den Import gefunden. Möglich Ursache: Der Name der CSV-Datei stimmt nicht mit dem Namen des Rezept-DBs überein.      |
| 80C0                      | CSV-Datei ist vorübergehend gesperrt.   |
| 80C1                      | Datenbaustein ist vorübergehend gesperrt.   |
| 80B0                      | Nummerierung im Index der CSV-Datei ist nicht durchgängig, nicht aufsteigend oder übersteigt die maximale Anzahl (Array-Grenze) im Datenbaustein. |
| 80B1                      | Struktur des Rezept-Datenbausteins und CSV-Datei stimmen nicht überein: Die CSV-Datei enthält zu viele Felder.                                    |
| 80B2                      | Struktur des Rezept-Datenbausteins und CSV-Datei stimmen nicht überein: Die CSV-Datei enthält zu wenige Felder.                                   |
| 80D0 +n                   | Struktur des Rezept-Datenbausteins und CSV-Datei stimmen nicht überein: Datentyp im Feld n stimmt nicht überein (n<=46).                          |
| 80FF                      | Struktur des Rezept-Datenbausteins und CSV-Datei stimmen nicht überein: Datentyp im Feld n stimmt nicht überein (n>46).                           |

\* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Aufbau eines Rezept-DBs

#### Einführung

Im Folgenden wird der Aufbau eines Rezept-DBs anhand eines einfachen Beispiels beschrieben. Der Rezept-DB besteht aus fünf Datensätzen, von denen drei verwendet werden. Der vierte und der fünfte Datensatz werden für spätere Erweiterungen freigelassen. Ein Datensatz enthält jeweils ein Rezept, welches sich aus dem Rezeptnamen und acht Zutaten zusammensetzt.

| product name | water | barley | wheat | hops | yeast | waterT mp | mashT mp | mashTime | QTest |
|--------------|-------|--------|-------|------|-------|-----------|----------|----------|-------|
| Pils         | 10    | 9      | 3     | 280  | 39    | 40        | 30       | 100      | 0     |
| Lager        | 10    | 9      | 3     | 150  | 33    | 50        | 30       | 120      | 0     |
| BlackBeer    | 10    | 9      | 3     | 410  | 47    | 60        | 30       | 90       | 1     |
| Not_used     | 0     | 0      | 0     | 0    | 0     | 0         | 0        | 0        | 0     |
| Not_used     | 0     | 0      | 0     | 0    | 0     | 0         | 0        | 0        | 0     |

## Aufbau des Rezept-Datenbausteins

In einem globalen Datenbaustein werden die Rezeptdaten folgendermaßen umgesetzt:

- Die Vorlage für alle Rezepte ist der PLC-Datentyp "Beer\_Recipe" mit den Rezeptbestandteilen "procutname", "water", usw. mit den entsprechenden Datentypen.
- In einem globalen Datenbaustein wird der PLC-Datentyp als Array [1.. 5] of "Beer\_Recipe" verwendet. Über die Array-Grenzen (hier 1 bis 5) wird die maximale Anzahl von Rezepten vorgegeben, die der DB enthalten darf.
- Die Werte für die Rezeptbestandteile werden als Startwert in dem Datenbaustein ergänzt.
- Über den InOut-Parameter RECIPE\_DB wird der globale DB mit der Anweisung verschaltet.

| Recipe_DB |             |                                 |        |             |
|-----------|-------------|---------------------------------|--------|-------------|
|           | Name        | Data type                       | Offset | Start value |
| 1         | Static      |                                 |        |             |
| 2         | Products    | Array [1 .. 5] of "Beer_Recipe" | ...    |             |
| 3         | Products[1] | "Beer_Recipe"                   | ...    |             |
| 4         | Products[2] | "Beer_Recipe"                   | ...    |             |
| 5         | Products[3] | "Beer_Recipe"                   | ...    |             |
| 6         | productname | String[20]                      | ...    | 'BlackBeer' |
| 7         | water       | UInt                            | ...    | 10          |
| 8         | barley      | UInt                            | ...    | 9           |
| 9         | wheat       | UInt                            | ...    | 3           |
| 10        | hops        | UInt                            | ...    | 410         |
| 11        | yeast       | UInt                            | ...    | 47          |
| 12        | waterTmp    | UInt                            | ...    | 60          |
| 13        | mashTmp     | UInt                            | ...    | 30          |
| 14        | mashTime    | UInt                            | ...    | 90          |
| 15        | QTest       | UInt                            | ...    | 1           |
| 16        | Products[4] | "Beer_Recipe"                   | ...    |             |
| 17        | Products[5] | "Beer_Recipe"                   | ...    |             |

## Export in CSV-Datei

Nach Ausführung der Anweisung "RecipeExport (Seite 2585)" werden die Daten des DBs in eine CSV-Datei mit dem folgenden Aufbau geschrieben:

### Recipe\_DB.csv

```
index,productname,water,barley,wheat,hops,yeast,waterTmp,mashTmp,mashTime,QTest
```

```
1,"Pils",10,9,3,280,39,40,30,100,0
2,"Lager",10,9,3,150,33,50,30,120,0
3,"BlackBeer",10,9,3,410,47,60,30,90,1
4,"Not_used",0,0,0,0,0,0,0,0,0
5,"Not_used",0,0,0,0,0,0,0,0,0
```

Über den WEB-Browser kann die CSV-Datei auf den PC/PG geladen und editiert werden:

- Sollen die Daten danach wieder in den DB geladen werden, dürfen keine Änderungen an der Struktur vorgenommen werden (zum Beispiel durch Hinzufügen von Zutaten in einer neuen Spalte).
- Wenn Sie die Datei um zusätzliche Datensätze ergänzen, achten Sie beim Import in den Datenbaustein darauf, dass die Array-Grenzen, über die Sie die maximale Anzahl von Datensätzen vorgeben, mindestens der Anzahl der Datensätze entspricht.
- Beim Export in die CSV-Datei wird automatisch ein Index erzeugt. Wenn Sie zusätzliche Datensätze erstellen, ergänzen Sie die fortlaufende Index-Nummer entsprechend.

Nach der Bearbeitung kann die geänderte Datei wieder in die CPU geladen werden. Die ursprüngliche CSV-Datei wird hierbei überschrieben. Über die Anweisung "RecipelImport (Seite 2586)" können die geänderten Daten aus der CSV-Datei wieder in den Datenbaustein importiert werden.

**Darstellung in Excel**

Die CSV-Datei kann zur übersichtlichen Darstellung und Bearbeitung mit Excel geöffnet werden. Sollte beim Öffnen die Kommas als Trennzeichen nicht erkannt werden, nutzen Sie die Importfunktion von Excel, um die Daten strukturiert auszugeben:

|   | A     | B           | C     | D      | E     | F    | G     | H        | I       | J        | K     |
|---|-------|-------------|-------|--------|-------|------|-------|----------|---------|----------|-------|
| 1 | index | product     | water | barley | wheat | hops | yeast | waterTmp | mashTmp | mashTime | QTest |
| 2 | 1     | "Pils"      | 10    | 9      | 3     | 280  | 39    | 40       | 30      | 100      | 0     |
| 3 | 2     | "Lager"     | 10    | 9      | 3     | 150  | 33    | 50       | 30      | 120      | 0     |
| 4 | 3     | "BlackBeer" | 10    | 9      | 3     | 410  | 47    | 60       | 30      | 90       | 1     |
| 5 | 4     | "Not_used"  | 0     | 0      | 0     | 0    | 0     | 0        | 0       | 0        | 0     |
| 6 | 5     | "Not_used"  | 0     | 0      | 0     | 0    | 0     | 0        | 0       | 0        | 0     |

**Data Logging**

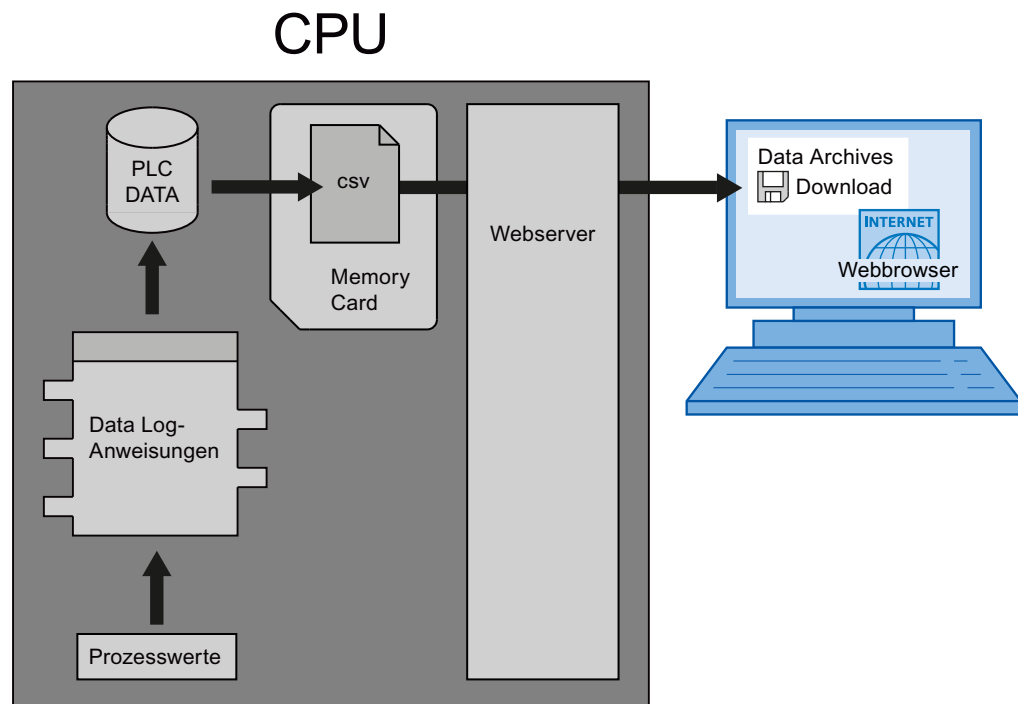
**Data Logging - Übersicht**

**Speichern von Prozesswerten**

Die Data Logging-Anweisungen werden im Anwenderprogramm genutzt, um Prozesswerte in Data Logs zu speichern. Data Logs können auf der Memory Card (MC) oder im internen Ladespeicher gespeichert werden. Das Speichern der Data Logs erfolgt im CSV-Format (Comma Separated Value).

Die Data Logging-Anweisungen verwenden Sie in Ihrem Programm zum Erstellen oder Öffnen eines Data Logs, zum Schreiben eines Eintrags und zum Schließen der Data Log-Datei.

Welche Programmwerte im Data Log beim Erstellen des Datenpuffers abgelegt werden entscheiden Sie durch das Erstellen des Datenpuffers. Der Datenpuffer wird als Speicher für neue Data Log-Einträge genutzt. Neue Werte müssen vor dem Aufruf von "DataLogWrite (Seite 2602)" in den Puffer geschrieben werden. Bei der Ausführung der Anweisung "DataLogWrite (Seite 2602)" werden die Daten vom Puffer in einen Data Log-Datensatz geschrieben.



Data Log-Dateien können folgendermaßen in den PC kopiert werden:

- Wenn die PROFINET-Schnittstelle mit dem PC verbunden ist, benutzen Sie einen Webbrowser, um auf die Data Logs über den Webserver zuzugreifen. Die CPU kann dabei im "RUN"- oder im "STOP"-Betrieb sein. Wenn die CPU in der Betriebsart "RUN" ist, läuft das Programm weiter, während der Webserver einen Datentransfer durchführt.
- Befindet sich in der CPU eine Memory Card, dann können Sie diese Karte herausnehmen und diese in einen Standardeinschub für SD-Cards (Secure Digital) oder MMC-Cards (MultiMediaCard) an einem PC oder Programmiergerät stecken. Nutzen Sie den Dateimanager, um die Data Log-Dateien von der Memory Card in den PC zu übertragen. Die CPU geht in "STOP", wenn Sie die Memory Card herausnehmen.

### Eigenschaften des Data Logs

Das Schreiben der Datensätze eines Data Logs erfolgt nach dem Prinzip eines Ringpuffers. Neue Datensätze werden angehängt, bis die maximale Anzahl Datensätze erreicht ist (Parameter RECORD). Der nächste Datensatz überschreibt dann den "ältesten" Datensatz des Data Logs.

Wenn Sie verhindern wollen, dass Datensätze überschrieben werden, dann nutzen Sie die Anweisung "DataLogNewFile (Seite 2608)", um eine neue Data Log-Datei basierend auf dem aktuellen Data Log zu erstellen. Neue Datensätze werden dann in den neuen Data Log geschrieben.

## Erstellen von Data Logs

Mit der Anweisung "DataLogCreate (Seite 2593)" erzeugen Sie eine neue Data Log-Datei im Verzeichnis "\\DataLogs" des Ladespeichers.

- Der am Parameter NAME vergebene Name ist die Benennung für das Data Log und wird auch als Dateiname für die CSV-Datei verwendet. Die Datei wird im Verzeichnis "DataLogs" abgelegt.
- Der Bausteinparameter DATA bestimmt den Datenpuffer für das neue Data Log-Objekt und die Spalten und Datentypen in dem Data Log. Die Spalten und Datentypen eines Datensatzes in dem Data Log entstehen durch die Elemente der Strukturdeklaration oder der Arraydeklaration dieses Datenpuffers. Jedes Element einer Struktur oder eines Arrays entspricht einer Spalte in einer Zeile im Data Log.
- Mit Hilfe des Bausteinparameters HEADER können Sie den Spalten jeweils eine Überschrift in der Kopfzeile zuweisen.
- Die Anweisung "DataLogCreate (Seite 2593)" gibt eine ID zurück. Diese ID wird von den anderen Data Logging -Anweisungen als Referenz für das erzeugte Data Log verwendet.

## Öffnen von Data Logs

Über die Anweisung "DataLogOpen (S7-1200)" und "DataLogTypedOpen" (S7-1500) öffnen Sie ein vorhandenes Data Log auf der Memory-Card. Ein Data Log muss geöffnet sein, um neue Datensätze in das Data Log zu schreiben.

Bei der Ausführung der Anweisungen "DataLogCreate (Seite 2593)" und "DataLogNewFile (Seite 2608)" wird das Data Log automatisch geöffnet.

Es können maximal 10 Data Logs gleichzeitig geöffnet sein. Das zu öffnende Data Log können Sie über die ID oder Namen des Data Logs auswählen.

- Wenn Sie an den Parametern ID und NAME jeweils die ID und den Namen des Data Logs angeben, wird das Data Log anhand der ID identifiziert. Ein Vergleich des Data Log-Namens wird nicht durchgeführt.
- Wenn Sie das Data Log über den Parameter NAME auswählen und keine ID angeben, wird die ID beim Öffnen des Data Logs am Parameter ID angezeigt.
- Wenn Sie das Data Log über den Parameter ID auswählen und keinen Namen angeben, wird der Name beim Öffnen des Data Logs am Parameter NAME nicht angezeigt.

Über den Parameter MODE wählen Sie aus, ob die Datensätze des Data Logs beim Öffnen gelöscht werden.

## Schreiben in Data Logs

Voraussetzung, um in einen Data Log einen Datensatz zu schreiben, ist ein geöffnetes Data Log (Anweisung "DataLogOpen (Seite 2597)"). Die Anweisung "DataLogWrite (Seite 2602)" schreibt einen Datensatz in das Data Log.

## Schließen von Data Logs

Mit der Anweisung "DataLogClose (Seite 2605)" schließen Sie ein geöffnetes Data Log. Das Data Log wählen Sie über den Parameter ID aus.

Beim Übergang der CPU in STOP oder bei einem Neustart wird das Data Log automatisch geschlossen.

## Löschen von Data Logs

Mit der Anweisung "DataLogDelete (Seite 2606)" (S7-1500) löschen Sie eine Data Log-Datei auf der Memory Card. Das Data Log und die enthaltenen Datensätze können nur gelöscht werden, wenn es über die Anweisung "DataLogCreate (Seite 2593)" erstellt wurde.

Über die Parameter NAME und ID wählen Sie das zu löschende Data Log aus. Der Parameter ID wird zuerst ausgewertet. Ist ein Data Log mit der entsprechenden ID vorhanden, wird der Parameter NAME nicht ausgewertet. Wird am Parameter ID der Wert "0" verwendet, muss am Parameter NAME ein Wert mit dem Datentyp STRING verwendet werden.

## Leeren von Data Logs

Mit der Anweisung "DataLogClear (Seite 2601)" (S7-1500) löschen Sie alle Datensätze eines vorhandenen Data Logs. Nicht gelöscht wird die optionale Kopfzeile der CSV-Datei (siehe Beschreibung des Parameters HEADER der Anweisung "DataLogCreate (Seite 2593)").

Über den Parameter ID wählen Sie das Data Log aus, deren Datensätze gelöscht werden sollen. Um die Datensätze zu löschen, muss das Data Log geöffnet sein.

## Neue Datei für Data Logs

Über die Anweisung "DataLogNewFile (Seite 2608)" (S7-1200) und "DataLogTypedNewFile (Seite 2609)" (S7-1500) erstellen Sie ein neues Data Log mit den gleichen Eigenschaften wie ein bereits vorhandenes Data Log. Hierdurch ist es möglich, die Inhalte eines bereits vorhandenen Data Logs zu erhalten.

Die Anweisung erstellt bei Aufruf ein neues Data Log auf der Memory Card bzw. dem internen Ladespeicher mit dem am Parameter NAME definierten Namen. Über den Parameter ID geben Sie die ID des alten Data Logs an, dessen Eigenschaften Sie für das neue Data Log übernehmen wollen. Über den Parameter ID wird anschließend die ID des neuen Data Logs ausgegeben.

Die Dateigröße für das neue Data Log geben Sie über den Parameter RECORDS der Anweisung an.

Für "DataLogTypedNewFile (Seite 2609)" (S7-1500) ist eine Konsistenzprüfung möglich.

## DataLogCreate: Data Log erstellen

### Beschreibung

Mit der Anweisung "DataLogCreate" erstellen Sie ein Data Log. Das Data Log wird auf der Memory Card bzw. im internen Ladespeicher im Verzeichnis "\\DataLogs" gespeichert. Sie können die Data Logging-Anweisungen verwenden, um Prozessdaten zu speichern. Die Datenmenge, die in einem Data Log abgelegt werden kann ist dabei abhängig von dem zur Verfügung stehenden Speicherplatz auf der Memory Card bzw. des Speicherplatz im internen Ladespeicher der eingesetzten CPU.

Die maximale Anzahl der Datensätze, die in einem Data Log abgelegt werden kann, geben Sie am Parameter RECORDS vor. Wird die vorgegebene maximale Anzahl von Datensätzen in dem Data Log erreicht, wird der jeweils älteste Datensatz überschrieben. Um ein Überschreiben der vorhandenen Datensätze zu vermeiden, können Sie die Anweisung "DataLogNewFile (Seite 2608)" nutzen. Die Anweisung kann bei Erreichen der am Parameter RECORDS vorgegebenen Anzahl (Returnwert 1 am Parameter STATUS der Anweisung "DataLogWrite (Seite 2602)") zum Erstellen eines neuen Data Log mit der gleichen Struktur genutzt werden. Die Datensätze werden danach in dem neuen Data Log gespeichert.

Den Namen für das Data Log können Sie am Parameter NAME vergeben. Das Data Log wird im CSV-Format (Comma Separated Values) erstellt. Über den Parameter HEADER können Sie eine (optionale) Kopfzeile für das Data Log erstellen. Als Trennzeichen muss ein Komma verwendet werden.

Nach der Erstellung wird das Data Log automatisch geöffnet.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogCreate":

| Parameter | Deklaration | Datentyp | Speicherbereich   | Beschreibung  |
|-----------|-------------|----------|---|---|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z oder Konstante<br>(T und Z sind nur in KOP und FUP verfügbar) | Ausführung der Anweisung bei steigender Flanke.   |
| RECORDS   | Input       | UDInt    | E, A, M, L, D oder Konstante  | Anzahl der Datensätze im Data Log   |
| FORMAT    | Input       | UInt     | E, A, M, L, D oder Konstante  | Datenformat:<br><ul style="list-style-type: none"> <li>• 0: Intern (wird nicht unterstützt)</li> <li>• 1: CSV (Comma separated values)</li> </ul>   |
| TIMESTAMP | Input       | UInt     | E, A, M, L, D oder Konstante  | Zeitstempelung:<br><ul style="list-style-type: none"> <li>• 0: Keine Zeitstempelung</li> <li>• 1: Datum und Uhrzeit</li> </ul> Bei der Zeitstempelung wird kein zusätzlicher Header für das Data Log benötigt.  |
| NAME      | Input       | VARIANT  | L, D  | Name des Data Logs. Der vergebene Name wird auch als Dateiname der CSV-Datei verwendet. Bei der Vergabe des Namens gelten Restriktionen für die Windows-Dateinamen. Die folgenden Zeichen dürfen nicht verwendet werden:<br>"\", "/", ":", "*", "?", "<", ">", " ", "Leerzeichen" |
| ID        | InOut       | DWORD    | E, A, M, L, D   | Objekt-ID des Data Logs (nur Output). Die ID des Data Logs wird für die weiteren Data Logging - Anweisungen benötigt.   |
| HEADER    | InOut       | VARIANT  | L, D  | Kopfzeile der CSV-Datei (optional). Der Parameter ist nach Einfügen der Anweisung ausgeblendet.   |
| DATA      | InOut       | VARIANT  | L, D  | Datenpuffer für einen Data Log - Datensatz.   |
| DONE      | Output      | BOOL     | E, A, M, L, D   | Die Anweisung wurde erfolgreich ausgeführt.   |



| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| BUSY      | Output      | BOOL     | E, A, M, L, D   | Erstellung des Data Logs noch nicht abgeschlossen.  |
| ERROR     | Output      | BOOL     | E, A, M, L, D   | <ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, L, D   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.                                    |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter HEADER

Der Parameter HEADER ist ein VARIANT-Pointer auf einen Datenbaustein, der eine Kopfzeile für die CSV-Datei (Header) definiert. Die Kopfzeile ist in der Darstellung einer CSV-Datei jeweils die erste Zeile. Bei der Erstellung einer Kopfzeile müssen die einzelnen Spalten jeweils mit einem Komma (S7-1200) oder mit einem Semikolon (S7-1500) getrennt sein. Als Datentyp kann für die einzelne Spaltenbenennung ein STRING, Array of BYTE oder Array of CHAR verwendet werden. Bei der Verwendung des Datentyps Array [...] of type ist eine längere Zeichenfolge möglich als bei der Verwendung des Datentyps STRING. Bei Verwendung von STRING ist die Länge auf 254 Bytes beschränkt.

Soll keine Kopfzeile erstellt werden, geben Sie am Parameter HEADER keinen Wert an.

## Parameter DATA

Der Parameter DATA ist ein VARIANT-Pointer auf eine Struktur oder ein Array in einem Datenbaustein. Ein Element einer Struktur oder eines Arrays entspricht einer Spalte im Data Log mit einem bestimmten Datentyp.

Beachten Sie bei der Erstellung des Datenbausteins Folgendes:

- Die Anzahl der Spalten muss der Anzahl an Spalten entsprechen, die am Parameter HEADER definiert sind.
- Bei Verwendung des Datentyps STRUCT dürfen keine Struktur-Verschachtelungen (STRUCT in STRUCT) verwendet werden.
- Arrays (nur 1-dimensional) können als einziges Element oder als Strukturkomponente verwendet werden. Jedes Arrayelement erzeugt eine eigene Spalte im Data Log.
- Die Variablen des Datenbausteins können als remanent oder nicht remanent eingestellt sein. Die Remanenzeinstellung für alle Variablen des Datenbausteins muss jedoch gleich sein.

**Parameter STATUS (S7-1200)**

| <b>Fehlercode*<br/>(W#16#...)</b>   | <b>Beschreibung</b>  |
|---|--|
| 0   | Keine Fehler.  |
| 7000  | Keine Auftragsbearbeitung aktiv.   |
| 7001  | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.                    |
| 7002  | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1". |
| 8070  | Gesamter interner Instanzspeicher ist belegt.                                    |
| 8090  | Unzulässiger Dateiname (siehe Beschreibung des Parameters NAME).                 |
| 8093  | Data Log existiert bereits.  |
| 8097  | Länge der Datei überschreitet Begrenzung des Dateisystems.                       |
| 80B3  | Speicherplatz auf der Memory Card nicht ausreichend.                             |
| 80B4  | Memory Card ist schreibgeschützt.  |
| 80C1  | Zu viele Dateien geöffnet.   |
| 8453  | Ungültige Formatauswahl.   |
| 8553  | Ungültige Zeitstempelung.  |
| 8B51  | Unzulässiger Datentyp am Parameter HEADER.                                       |
| 8C20  | String mit Längenangabe ungleich 254 verwendet.                                  |
| 8C51  | Unzulässiger Datentyp am Parameter DATA.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**Parameter STATUS (S7-1500)**

| <b>Fehlercode*<br/>(W#16#...)</b> | <b>Beschreibung</b>  |
|-----------------------------------|--|
| 0                                 | Keine Fehler.  |
| 7000                              | Keine Auftragsbearbeitung aktiv.                                 |
| 7001                              | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.    |
| 8070                              | Gesamter interner Instanzspeicher ist belegt.                    |
| 8090                              | Unzulässiger Dateiname (siehe Beschreibung des Parameters NAME). |
| 8091                              | Parameter "NAME" ist kein String.                                |
| 8093                              | Data Log existiert bereits.                                      |
| 8097                              | Länge der Datei überschreitet Begrenzung des Dateisystems.       |
| 80B3                              | Speicherplatz auf der Memory Card nicht ausreichend.             |
| 80B4                              | Memory Card ist schreibgeschützt.                                |
| 80C0                              | Zugriff momentan nicht möglich.                                  |
| 80C1                              | Zu viele Dateien geöffnet.                                       |
| 8253                              | Ungültiger Wert am Parameter RECORDS.                            |
| 8353                              | Ungültige Formatauswahl.   |
| 8453                              | Ungültige Zeitstempelung.  |

| Fehlercode*<br>(W#16#...)   | Beschreibung                               |
|---|--|
| 8B51  | Unzulässiger Datentyp am Parameter HEADER. |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

Data Logging - Übersicht (Seite 2590)

## DataLogOpen: Data Log öffnen

## DataLogOpen: Data Log öffnen

## Beschreibung

Über die Anweisung "DataLogOpen" öffnen Sie ein vorhandenes Data Log auf der Memory-Card. Ein Data Log muss geöffnet sein, um neue Datensätze in das Data Log zu schreiben.

Bei der Ausführung der Anweisungen "DataLogCreate (Seite 2593)" und "DataLogNewFile (Seite 2608)" wird das Data Log automatisch geöffnet.

Es können maximal 10 Data Logs gleichzeitig geöffnet sein. Das zu öffnende Data Log können Sie über die ID oder Namen des Data Logs auswählen.

- Wenn Sie an den Parametern ID und NAME jeweils die ID und den Namen des Data Logs angeben, wird das Data Log anhand der ID identifiziert. Ein Vergleich des Data Log-Namens wird nicht durchgeführt.
- Wenn Sie das Data Log über den Parameter NAME auswählen und keine ID angeben, wird die ID beim Öffnen des Data Logs am Parameter ID angezeigt.
- Wenn Sie das Data Log über den Parameter ID auswählen und keinen Namen angeben, wird der Name beim Öffnen des Data Logs am Parameter NAME nicht angezeigt.

Über den Parameter MODE wählen Sie aus, ob die Datensätze des Data Logs beim Öffnen gelöscht werden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogOpen":

| Parameter | Deklaration | Datentyp | Speicherbereich   | Beschreibung  |
|-----------|-------------|----------|---|---|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z oder Konstante<br>(T und Z sind nur in KOP und FUP verfügbar) | Ausführen der Anweisung bei steigender Flanke.  |
| MODE      | Input       | UInt     | E, A, M, L, D oder Konstante  | Modus zum Öffnen des Data Logs:<br><ul style="list-style-type: none"> <li>• MODE= "0"<br/>Datensätze des Data Logs bleiben erhalten</li> <li>• MODE= "1"<br/>Datensätze des Data Logs werden gelöscht, die Kopfzeile bleibt erhalten</li> </ul> |
| NAME      | Input       | VARIANT  | L, D  | (Datei-)name des Data Logs.   |
| ID        | InOut       | DWORD    | E, A, M, L, D   | Objekt-ID des Data Logs.  |
| DONE      | Output      | BOOL     | E, A, M, L, D   | Anweisung wurde erfolgreich ausgeführt.   |
| BUSY      | Output      | BOOL     | E, A, M, L, D   | Ausführung der Anweisung noch nicht beendet.  |
| ERROR     | Output      | BOOL     | E, A, M, L, D   | <ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben.                                 |
| STATUS    | Output      | WORD     | E, A, M, L, D   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter STATUS**

| Fehlercode* (W#16#...) | Beschreibung   |
|------------------------|--|
| 0                      | Keine Fehler.  |
| 2                      | Warnung: Data Log-Datei wurde bereits durch diese Anwendung geöffnet.                    |
| 7000                   | Keine Auftragsbearbeitung aktiv.   |
| 7001                   | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.                            |
| 7002                   | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".         |
| 8070                   | Gesamter interner Instanzspeicher ist belegt.  |
| 8090                   | Zwischen der Data Log-Definition und bestehenden Data Log-Daten bestehen Inkonsistenzen. |
| 8091                   | Am Parameter NAME wurde ein anderer Datentyp als String verwendet.                       |
| 8092                   | Data Log existiert nicht.  |

| Fehlercode*<br>(W#16#...)   | Beschreibung                      |
|---|-----------------------------------|
| 80B4  | Memory Card ist schreibgeschützt. |
| 80C0  | Data Log-Datei ist gesperrt.      |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                                   |

## Siehe auch

Data Logging - Übersicht (Seite 2590)

## DataLogOpen: Data Log öffnen

### Beschreibung

Über die Anweisung "DataLogOpen" öffnen Sie ein vorhandenes Data Log auf der Memory-Card. Ein Data Log muss geöffnet sein, um neue Datensätze in das Data Log schreiben zu können.

Bei der Ausführung der Anweisungen "DataLogCreate (Seite 2593)" und "DataLogNewFile (Seite 2608)" wird das Data Log automatisch geöffnet.

Es können maximal 10 Data Logs gleichzeitig geöffnet sein. Das zu öffnende Data Log können Sie über die ID oder Namen des Data Logs auswählen.

- Wenn Sie an den Parametern ID und NAME jeweils die ID und den Namen des Data Logs angeben, wird das Data Log anhand der ID identifiziert. Ein Vergleich des Data Log-Namens wird nicht durchgeführt.
- Wenn Sie das Data Log über den Parameter NAME auswählen und keine ID angeben, wird die ID beim Öffnen des Data Logs am Parameter ID angezeigt.
- Wenn Sie das Data Log über den Parameter ID auswählen und keinen Namen angeben, wird der Name beim Öffnen des Data Logs am Parameter NAME nicht angezeigt.

Über den Parameter MODE wählen Sie aus, ob die Datensätze des Data Logs beim Öffnen gelöscht werden sollen.

Der Parameter DATA ermöglicht eine Konsistenzprüfung zwischen dem zu öffnenden Data Log und der Definition des Data Logs der Anweisung "DataLogCreate (Seite 2593)". Die Konsistenzprüfung kann nur durchgeführt werden, wenn das Data Log mit der Anweisung "DataLogCreate (Seite 2593)" erstellt wurde:

- Wenn Sie am Parameter DATA den gleichen Zeiger verwenden, wie am Parameter DATA der Anweisung "DataLogCreate (Seite 2593)", wird überprüft, ob die Datentypen übereinstimmen. Ist dies nicht der Fall, wird am Parameter STATUS der Fehlercode W#16#80A0 ausgegeben.
- Wurde das zu öffnende Data Log nicht mit "DataLogCreate (Seite 2593)" erstellt, ist eine Konsistenzprüfung nicht möglich. Geben Sie in diesem Fall am Parameter DATA den Wert "NULL" an.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogOpen":

| Parameter | Deklaration | Datentyp | Speicherbereich   | Beschreibung  |
|-----------|-------------|----------|---|---|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z oder Konstante<br>(T und Z sind nur in KOP und FUP verfügbar) | Ausführen der Anweisung bei steigender Flanke.  |
| MODE      | Input       | UInt     | E, A, M, L, D oder Konstante  | Modus zum Öffnen des Data Logs:<br><ul style="list-style-type: none"> <li>• MODE= "0"<br/>Datensätze des Data Logs bleiben erhalten</li> <li>• MODE= "1"<br/>Datensätze des Data Logs werden gelöscht, die Kopfzeile bleibt erhalten</li> </ul> |
| NAME      | Input       | VARIANT  | L, D  | (Datei-)name des Data Logs.   |
| ID        | InOut       | DWORD    | E, A, M, L, D   | Objekt-ID des Data Logs.  |
| DATA      | InOut       | VARIANT  | L, D  | Bei Konsistenzprüfung: Zeiger auf den Datenbereich am Parameter DATA der Anweisung "DataLogCreate (Seite 2593)".  |
| DONE      | Output      | BOOL     | E, A, M, L, D   | Anweisung wurde erfolgreich ausgeführt.   |
| BUSY      | Output      | BOOL     | E, A, M, L, D   | Ausführung der Anweisung noch nicht beendet.  |
| ERROR     | Output      | BOOL     | E, A, M, L, D   | <ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben.                                 |
| STATUS    | Output      | WORD     | E, A, M, L, D   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter STATUS**

| Fehlercode* (W#16#...) | Beschreibung   |
|------------------------|--|
| 0                      | Keine Fehler.  |
| 2                      | Warnung: Data Log-Datei wurde bereits durch diese Anwendung geöffnet.                    |
| 7000                   | Keine Auftragsbearbeitung aktiv.   |
| 7001                   | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.                            |
| 7002                   | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".         |
| 8070                   | Gesamter interner Instanzspeicher ist belegt.  |
| 8090                   | Zwischen der Data Log-Definition und bestehenden Data Log-Daten bestehen Inkonsistenzen. |

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 8091  | Am Parameter NAME wurde ein anderer Datentyp als String verwendet.   |
| 8092  | Data Log existiert nicht.  |
| 80A0  | Datentypen inkonsistent. Das Data Log am Parameter ID verwendet andere Datentypen als am Parameter DATA angegeben. |
| 80B4  | Memory Card ist schreibgeschützt.  |
| 80C1  | Zu viele Dateien geöffnet.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## Siehe auch

Data Logging - Übersicht (Seite 2590)

## DataLogClear: Data Log leeren

### Beschreibung

Mit der Anweisung "DataLogClear" löschen Sie alle Datensätze eines vorhandenen Data Logs. Nicht gelöscht wird die optionale Kopfzeile der CSV-Datei (siehe Beschreibung des Parameters HEADER der Anweisung "DataLogCreate (Seite 2593)").

Über den Parameter ID wählen Sie das Data Log aus, dessen Datensätze gelöscht werden sollen.

### Voraussetzung

Um die Datensätze zu löschen, muss das Data Log geöffnet sein (siehe Anweisung "DataLogOpen (Seite 2597)").

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogClear":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung                                       |
|-----------|-------------|----------|--|--|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z<br>oder Konstante<br>(T und Z sind nur in<br>KOP und FUP<br>verfügbar) | Ausführung der Anweisung bei steigender Flanke.    |
| ID        | InOut       | DWORD    | E, A, M, D, L  | Objekt-ID des Data Logs                            |
| DONE      | Output      | BOOL     | E, A, M, D, L  | Anweisung wurde erfolgreich ausgeführt.            |
| BUSY      | Output      | BOOL     | E, A, M, D, L  | Ausführung der Anweisung noch nicht abgeschlossen. |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| ERROR     | Output      | BOOL     | E, A, M, D, L   | <ul style="list-style-type: none"> <li>0: Kein Fehler.</li> <li>1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden an dem Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.                              |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

| Fehlercode* (W#16#...) | Erläuterung  |
|------------------------|--|
| 0000                   | Kein Fehler.   |
| 7000                   | Keine Auftragsbearbeitung aktiv.   |
| 7001                   | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.                    |
| 7002                   | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1". |
| 8080                   | Data Log-Datei passt nicht zur Anweisung.  |
| 8092                   | Data Log existiert nicht.  |
| 80B0                   | Data Log ist nicht geöffnet.   |
| 80B4                   | Memory-Card ist schreibgeschützt.  |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Siehe auch

Data Logging - Übersicht (Seite 2590)

### DataLogWrite: Data Log schreiben

#### Beschreibung

Mit der Anweisung "DataLogWrite" schreiben Sie einen Datensatz in ein vorhandenes Data Log. Über den Parameter ID wählen Sie das Data Log aus, in welches der Datensatz geschrieben werden soll. Um einen neuen Datensatz zu erstellen, muss das Data Log geöffnet sein. Die Anweisung erstellt einen neuen Datensatz in dem Format, das am Parameter DATA bei der Erstellung des Data Logs angegeben wurde.

Übertragen Sie vor dem Aufruf der Anweisung "DataLogWrite" die Daten an die Variable, welche sie am Parameter DATA der Anweisung "DataLogCreate" verschaltet haben. Bei der



Ausführung der Anweisung "DataLogWrite" werden die übertragenen Daten in das Data Log kopiert.

### ACHTUNG

#### Data Log-Datenverlust bei Unterbrechung der Stromversorgung der CPU

Wird die Stromversorgung während der Ausführung der Anweisung "DataLogWrite" unterbrochen, geht der zu übertragende Datensatz verloren.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogWrite":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung  |
|-----------|-------------|----------|--|---|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z<br>oder Konstante<br>(T und Z sind nur in<br>KOP und FUP<br>verfügbar) | Ausführung der Anweisung bei steigender Flanke.   |
| ID        | InOut       | DWORD    | E, A, M, L, D  | Objekt-ID des Data Logs   |
| DONE      | Output      | BOOL     | E, A, M, L, D  | Anweisung wurde erfolgreich ausgeführt.   |
| BUSY      | Output      | BOOL     | E, A, M, L, D  | Ausführung der Anweisung noch nicht abgeschlossen.  |
| ERROR     | Output      | BOOL     | E, A, M, L, D  | <ul style="list-style-type: none"> <li>0: Kein Fehler.</li> <li>1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, L, D  | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt.<br>Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.                             |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS (S7-1200)

| Fehlercode*<br>(W#16#...) | Beschreibung  |
|---------------------------|---|
| 0                         | Keine Fehler  |
| 0001                      | Letzter möglicher Datensatz am Ende der Datei erstellt. Beim Erstellen eines weiteren Datensatzes wird ein älterer überschrieben. |
| 7000                      | Keine Auftragsbearbeitung aktiv.  |
| 7001                      | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0  |

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 7002  | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1". |
| 8070  | Gesamter interner Instanzspeicher ist belegt.                                    |
| 8092  | Data Log existiert nicht.  |
| 80B0  | Data Log ist nicht geöffnet.   |
| 80B3  | Speicherplatz auf der Memory Card nicht ausreichend.                             |
| 80B4  | Memory Card ist schreibgeschützt.  |
| 80C0  | Data Log ist gesperrt.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### Parameter STATUS (S7-1500)

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 0   | Keine Fehler  |
| 0001  | Letzter möglicher Datensatz am Ende der Datei erstellt. Beim Erstellen eines weiteren Datensatzes wird ein älterer überschrieben. |
| 7000  | Keine Auftragsbearbeitung aktiv.  |
| 7001  | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0  |
| 8070  | Gesamter interner Instanzspeicher ist belegt.   |
| 8092  | Data Log existiert nicht.   |
| 80B0  | Data Log ist nicht geöffnet.  |
| 80B3  | Speicherplatz auf der Memory Card nicht ausreichend.  |
| 80B4  | Memory Card ist schreibgeschützt.   |
| 80C0  | Data Log ist gesperrt.  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Siehe auch

Data Logging - Übersicht (Seite 2590)

## DataLogClose: Data Log schließen

### Beschreibung

Mit der Anweisung "DataLogClose" schließen Sie ein geöffnetes Data Log. Das Data Log wählen Sie über den Parameter ID aus.

#### Hinweis

##### Automatisches Schließen von Data Logs

Beim Übergang der CPU in STOP oder einem Neustart wird das Data Log automatisch geschlossen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogClose":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung  |
|-----------|-------------|----------|--|---|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z<br>oder Konstante<br>(T und Z sind nur in<br>KOP und FUP<br>verfügbar) | Funktion bei steigender Flanke ausführen.   |
| ID        | InOut       | DWORD    | E, A, M, L, D  | Objekt-ID des Data Logs   |
| DONE      | Output      | BOOL     | E, A, M, L, D  | Anweisung wurde erfolgreich ausgeführt.   |
| BUSY      | Output      | BOOL     | E, A, M, L, D  | Ausführung der Anweisung noch nicht abgeschlossen.  |
| ERROR     | Output      | BOOL     | E, A, M, L, D  | <ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, L, D  | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt.<br>Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.                                 |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter STATUS (S7-1200)**

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 0   | Keine Fehler   |
| 1   | Data Log nicht geöffnet  |
| 7000  | Keine Auftragsbearbeitung aktiv.   |
| 7001  | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0                     |
| 7002  | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1". |
| 8092  | Data Log existiert nicht.  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**Parameter STATUS (S7-1500)**

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 0   | Keine Fehler   |
| 1   | Data Log nicht geöffnet                                      |
| 7000  | Keine Auftragsbearbeitung aktiv.                             |
| 7001  | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0 |
| 8092  | Data Log existiert nicht.                                    |
| 80B4  | Memory Card ist schreibgeschützt.                            |
| 80C0  | Zugriff momentan nicht möglich.                              |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**DataLogDelete: Data Log löschen**

**Beschreibung**

Mit der Anweisung "DataLogDelete" löschen Sie eine Data Log-Datei auf der Memory Card. Das Data Log und die enthaltenen Datensätze können nur gelöscht werden, wenn es über die Anweisung "DataLogCreate (Seite 2593)" erstellt wurde.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogDelete":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung   |
|-----------|-------------|----------|--|--|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z<br>oder Konstante<br>(T und Z sind nur in<br>KOP und FUP<br>verfügbar) | Ausführung der Anweisung bei steigender<br>Flanke.   |
| NAME      | Input       | VARIANT  | L, D   | Dateiname des Data Logs  |
| DELFILE   | Input       | BOOL     | E, A, M, D, L  | <ul style="list-style-type: none"> <li>0: Data Log bleibt erhalten.</li> <li>1: Data Log wird gelöscht.</li> </ul> Detailinformationen werden an dem<br>Parameter STATUS ausgegeben.                             |
| ID        | InOut       | DWORD    | E, A, M, D, L  | Objekt-ID des Data Logs  |
| DONE      | Output      | BOOL     | E, A, M, D, L  | Anweisung wurde erfolgreich ausgeführt.  |
| BUSY      | Output      | BOOL     | E, A, M, D, L  | Löschen des Data Logs noch nicht<br>abgeschlossen.   |
| ERROR     | Output      | BOOL     | E, A, M, D, L  | <ul style="list-style-type: none"> <li>0: Kein Fehler.</li> <li>1: Bei der Ausführung der Anweisung ist<br/>ein Fehler aufgetreten.</li> </ul> Detailinformationen werden an dem<br>Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, D, L  | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang<br>gesetzt. Zur Anzeige des Status sollten Sie<br>deshalb den Parameter STATUS in einen<br>freien Datenbereich kopieren.                            |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter NAME und ID

Über die Parameter NAME und ID wählen Sie das zu löschende Data Log aus. Der Parameter ID wird zuerst ausgewertet. Ist ein Data Log mit der entsprechenden ID vorhanden, wird der Parameter NAME nicht ausgewertet. Wird am Parameter ID der Wert "0" verwendet, muss am Parameter NAME ein Wert mit dem Datentyp STRING verwendet werden.

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...) | Erläuterung  |
|---------------------------|--|
| 0                         | Kein Fehler.   |
| 7000                      | Keine Auftragsbearbeitung aktiv.   |
| 7001                      | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0                     |
| 7002                      | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1". |

| Fehlercode*<br>(W#16#...)   | Erläuterung   |
|---|---|
| 8091  | Am Parameter NAME wird ein anderer Datentyp als STRING verwendet. |
| 8092  | Data Log existiert nicht.   |
| 80A2  | Schreibfehler   |
| 80B4  | Memory-Card ist schreibgeschützt.                                 |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### DataLogNewFile: Data Log in neuer Datei

### DataLogNewFile: Data Log in neuer Datei

#### Beschreibung

Über die Anweisung "DataLogNewFile" erstellen Sie ein neues Data Log mit den gleichen Eigenschaften wie ein bereits vorhandenes Data Log. Hierdurch ist es möglich, die Inhalte eines bereits vorhandenen Data Logs zu erhalten.

Die Anweisung erstellt bei Aufruf ein neues Data Log auf der Memory Card bzw. dem internen Ladespeicher mit dem am Parameter NAME definierten Namen. Über den Parameter ID geben Sie die ID des alten Data Logs an, dessen Eigenschaften Sie für das neue Data Log übernehmen wollen. Über den Parameter ID wird anschließend die ID des neuen Data Logs ausgegeben.

Die Dateigröße für das neue Data Log geben Sie über den Parameter RECORDS der Anweisung an.

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogNewFile":

| Parameter | Deklaration | Datentyp | Speicherbereich  | Beschreibung   |
|-----------|-------------|----------|--|--|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z<br>oder Konstante<br>(T und Z sind nur in<br>KOP und FUP<br>verfügbar) | Ausführung der Anweisung bei steigender Flanke.  |
| RECORDS   | Input       | UDInt    | E, A, M, L, D oder<br>Konstante  | Anzahl der Datensätze im neuen Data Log.   |
| NAME      | Input       | VARIANT  | L, D   | Dateiname des neuen Data Logs.   |
| ID        | InOut       | DWORD    | E, A, M, L, D  | Objekt-ID des Data Logs<br><ul style="list-style-type: none"> <li>• In: ID des bereits vorhandenen Data Logs</li> <li>• Out: ID des neuen Data Logs</li> </ul> |
| DONE      | Output      | BOOL     | E, A, M, L, D  | Anweisung wurde erfolgreich ausgeführt.  |
| BUSY      | Output      | BOOL     | E, A, M, L, D  | Ausführung der Anweisung noch nicht abgeschlossen.   |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| ERROR     | Output      | BOOL     | E, A, M, L, D   | <ul style="list-style-type: none"> <li>0: Kein Fehler.</li> <li>1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, L, D   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.                                |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Fehlercode*<br>(W#16#...) | Beschreibung   |
|---------------------------|--|
| 0                         | Keine Fehler.  |
| 7000                      | Keine Auftragsbearbeitung aktiv.   |
| 7001                      | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.                    |
| 7002                      | Zwischenaufruf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1". |
| 8070                      | Gesamter interner Instanzspeicher ist belegt.                                    |
| 8090                      | Ungültiger Dateiname.  |
| 8091                      | Pfad existiert nicht.  |
| 8092                      | Quellen- Data Log existiert nicht.   |
| 8093                      | Neues Data Log existiert bereits.  |
| 8097                      | Länge der Datei überschreitet Begrenzung des Dateisystems.                       |
| 80B3                      | Ladespeicher nicht ausreichend.  |
| 80B4                      | Memory Card ist schreibgeschützt.  |
| 80C1                      | Zu viele Dateien geöffnet.   |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## DataLogNewFile: Data Log in neuer Datei

### Beschreibung

Über die Anweisung "DataLogNewFile" erstellen Sie ein neues Data Log mit den gleichen Eigenschaften wie ein bereits vorhandenes Data Log. Hierdurch ist es möglich, die Inhalte eines bereits vorhandenen Data Logs zu erhalten.

Die Anweisung erstellt bei Aufruf ein neues Data Log auf der Memory Card bzw. dem internen Ladespeicher mit dem am Parameter NAME definierten Namen. Über den Parameter ID geben Sie die ID des alten Data Logs an, dessen Eigenschaften Sie für das neue Data Log

übernehmen wollen. Über den Parameter ID wird anschließend die ID des neuen Data Logs ausgegeben.

Die Dateigröße für das neue Data Log geben Sie über den Parameter RECORDS der Anweisung an.

Der Parameter DATA ermöglicht eine Konsistenzprüfung zwischen dem neu zu erstellenden Data Log und der Definition des Data Logs der Anweisung "DataLogCreate (Seite 2593)". Die Konsistenzprüfung kann nur durchgeführt werden, wenn das Data Log mit der Anweisung "DataLogCreate (Seite 2593)" erstellt wurde:

- Wenn Sie am Parameter DATA den gleichen Zeiger verwenden, wie am Parameter DATA der Anweisung "DataLogCreate (Seite 2593)", wird überprüft, ob die Datentypen übereinstimmen. Ist dies nicht der Fall, wird am Parameter STATUS der Fehlercode W#16#80A0 ausgegeben.
- Wurde das zu öffnende Data Log nicht mit "DataLogCreate (Seite 2593)" erstellt, ist eine Konsistenzprüfung nicht möglich. Geben Sie in diesem Fall am Parameter DATA den Wert "NULL" an.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "DataLogNewFile":

| Parameter | Deklaration | Datentyp | Speicherbereich   | Beschreibung  |
|-----------|-------------|----------|---|---|
| REQ       | Input       | BOOL     | E, A, M, L, D, T, Z oder Konstante<br>(T und Z sind nur in KOP und FUP verfügbar) | Ausführung der Anweisung bei steigender Flanke.   |
| RECORDS   | Input       | UDInt    | E, A, M, L, D oder Konstante  | Anzahl der Datensätze im neuen Data Log.  |
| NAME      | Input       | VARIANT  | L, D  | Dateiname des neuen Data Logs.  |
| ID        | InOut       | DWORD    | E, A, M, L, D   | Objekt-ID des Data Logs <ul style="list-style-type: none"> <li>• In: ID des bereits vorhandenen Data Logs</li> <li>• Out: ID des neuen Data Logs</li> </ul>   |
| DATA      | InOut       | VARIANT  | L, D  | Datentyp für Konsistenzprüfung  |
| DONE      | Output      | BOOL     | E, A, M, L, D   | Anweisung wurde erfolgreich ausgeführt.   |
| BUSY      | Output      | BOOL     | E, A, M, L, D   | Ausführung der Anweisung noch nicht abgeschlossen.  |
| ERROR     | Output      | BOOL     | E, A, M, L, D   | <ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD     | E, A, M, L, D   | Zustandsparameter<br>Der Parameter ist nur einen Aufruf lang gesetzt. Zur Anzeige des Status sollten Sie deshalb den Parameter STATUS in einen freien Datenbereich kopieren.                                    |



Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 0   | Keine Fehler.  |
| 7000  | Keine Auftragsbearbeitung aktiv.   |
| 7001  | Beginn der Auftragsbearbeitung. Parameter BUSY = 1, DONE = 0.  |
| 7002  | Zwischenaufwurf (REQ irrelevant): Anweisung bereits aktiv; BUSY hat den Wert "1".                                  |
| 8070  | Gesamter interner Instanzspeicher ist belegt.  |
| 8090  | Ungültiger Dateiname.  |
| 8091  | Pfad existiert nicht.  |
| 8092  | Quellen- Data Log existiert nicht.   |
| 8093  | Neues Data Log existiert bereits.  |
| 8097  | Länge der Datei überschreitet Begrenzung des Dateisystems.   |
| 80A0  | Datentypen inkonsistent. Das Data Log am Parameter ID verwendet andere Datentypen als am Parameter DATA angegeben. |
| 80B3  | Ladespeicher nicht ausreichend.  |
| 80B4  | Memory Card ist schreibgeschützt.  |
| 80C0  | Zugriff momentan nicht möglich.  |
| 80C1  | Zu viele Dateien geöffnet.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### 9.7.3.12 Datenbausteinfunktionen

#### CREATE\_DB: Datenbaustein erzeugen

##### Beschreibung

Mit der Anweisung "CREATE\_DB" erstellen Sie im Lade- und/oder Arbeitsspeicher einen neuen Datenbaustein.

Die Anweisung "CREATE\_DB" verändert nicht die Prüfsumme des Anwenderprogramms.

##### Nummer des Datenbausteins

Der erzeugte Datenbaustein erhält eine Nummer aus dem an den Parametern LOW\_LIMIT (Untergrenze) und UP\_LIMIT (Obergrenze) definierten Bereich. Aus dem angegebenen Bereich vergibt "CREATE\_DB" die kleinste mögliche Nummer an den DB. Die Nummern der im Anwenderprogramm bereits enthaltenen DBs können nicht mehr vergeben werden.

Einen DB mit einer bestimmten Nummer erzeugen Sie, in dem Sie für die Ober- und Untergrenze des anzugebenden Bereichs die gleiche Nummer vergeben. Ist bereits ein DB mit gleicher Nummer im Arbeitsspeicher und/oder im Ladespeicher vorhanden oder liegt der DB als kopierte Variante vor, wird die Anweisung beendet und eine Fehlermeldung am Parameter RET\_VAL ausgegeben.

### Startwerte des Datenbausteins

Über den Parameter SRCBLK können Sie für den zu erzeugenden DB Startwerte definieren. Der Parameter SRCBLK ist ein Zeiger auf ein DB oder ein DB-Bereich, aus dem Sie die Startwerte übernehmen. Der am Parameter SRCBLK adressierte DB muss mit Standardzugriff erstellt sein (Attribut "Optimierter Bausteinzugriff" deaktiviert).

- Wenn der am Parameter SRCBLK angegebene Bereich größer ist als der erzeugte DB werden die Werte bis zur Länge des erzeugten DBs als Startwerte übernommen.
- Wenn der am Parameter SRCBLK angegebene Bereich kleiner ist als der erzeugte DB werden die übrigen Werte mit "0" aufgefüllt.

Um Datenkonsistenz zu gewährleisten, dürfen Sie diesen Datenbereich während der Bearbeitung von "CREATE\_DB" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "CREATE\_DB":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Erzeugen des DB   |
| LOW_LIMIT | Input       | UINT     | E, A, M, D, L oder Konstante | Untergrenze des Bereichs, aus dem "CREATE_DB" Ihrem DB eine Nummer vergibt (Kleinst mögliche DB-Nummer: 60000)   |
| UP_LIMIT  | Input       | UINT     | E, A, M, D, L oder Konstante | Obergrenze des Bereichs, aus dem "CREATE_DB" Ihrem DB eine Nummer vergibt (Größt mögliche DB-Nummer: 60999)  |
| COUNT     | Input       | UDINT    | E, A, M, D, L oder Konstante | Der Zählwert gibt die Anzahl der Bytes an, die Sie für den erzeugten DB reservieren wollen. Für die Anzahl der Bytes muss eine gerade Nummer vergeben werden. Die maximale Länge beträgt 65534 Bytes.  |
| ATTRIB    | Input       | BYTE     | E, A, M, D, L oder Konstante | Über die ersten 4 Bit des Bytes am Parameter ATTRIB definieren Sie die Eigenschaften des Datenbausteins: <ul style="list-style-type: none"> <li>• Bit 0 = 0: DB nur im Arbeitsspeicher</li> <li>• Bit 0 = 1: DB nur im Ladespeicher</li> <li>• Bit 1 = 0: DB nicht schreibgeschützt</li> <li>• Bit 1 = 1: DB schreibgeschützt</li> <li>• Bit 2 = 0: DB remanent (nur bei DBs, die im Ladespeicher erzeugt werden)</li> <li>• Bit 2 = 1: DB nicht remanent</li> </ul> |

| Parameter | Deklaration | Datentyp                  | Speicherbereich | Beschreibung   |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
|-----------|-------------|---------------------------|-----------------|--|-------|-------|---------------|---|---|------------------------|---|---|---------------------|---|---|---------------------------|---|---|---------------------------|
|           |             |                           |                 | <ul style="list-style-type: none"> <li>• Bit 3= 0: Erstellung des DBs entweder im Lade- oder im Arbeitsspeicher</li> <li>• Bit 3= 1: Erstellung des DBs sowohl im Lade- als auch im Arbeitsspeicher</li> </ul> <p>Aus Kompatibilitätsgründen zu STEP7 V5.x müssen die Bits 1 und 3 in Kombination verwendet werden:</p> <table border="1"> <thead> <tr> <th>Bit 0</th> <th>Bit 3</th> <th>DB Erstellung</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Nur im Arbeitsspeicher</td> </tr> <tr> <td>1</td> <td>0</td> <td>Nur im Ladespeicher</td> </tr> <tr> <td>0</td> <td>1</td> <td>Arbeits- und Ladespeicher</td> </tr> <tr> <td>1</td> <td>1</td> <td>Arbeits- und Ladespeicher</td> </tr> </tbody> </table> <ul style="list-style-type: none"> <li>• Bit 4 = 0 - Keine Startwerte vorgegeben (Eingangswerte am Parameter SRCBLK werden ignoriert).</li> <li>• Bit 4 = 1 - Startwerte vorgeben (Werte entsprechen den über Parameter SRCBLK adressierten DB).</li> </ul> | Bit 0 | Bit 3 | DB Erstellung | 0 | 0 | Nur im Arbeitsspeicher | 1 | 0 | Nur im Ladespeicher | 0 | 1 | Arbeits- und Ladespeicher | 1 | 1 | Arbeits- und Ladespeicher |
| Bit 0     | Bit 3       | DB Erstellung             |                 |  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
| 0         | 0           | Nur im Arbeitsspeicher    |                 |  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
| 1         | 0           | Nur im Ladespeicher       |                 |  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
| 0         | 1           | Arbeits- und Ladespeicher |                 |  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
| 1         | 1           | Arbeits- und Ladespeicher |                 |  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
| SRCBLK    | Input       | VARIANT                   | D               | Zeiger auf den Datenbaustein, mit dessen Werten der zu erzeugende Datenbaustein initialisiert wird.  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
| RET_VAL   | Return      | INT                       | E, A, M, D, L   | Fehlerinformation  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
| BUSY      | Output      | BOOL                      | E, A, M, D, L   | BUSY = 1: Der Vorgang ist noch nicht beendet.  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |
| DB_NUM    | Output      | DB_DYN (UINT)             | E, A, M, D, L   | Nummer des erzeugten DBs.  |       |       |               |   |   |                        |   |   |                     |   |   |                           |   |   |                           |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode* (W#16#...) | Beschreibung  |
|------------------------|---|
| 0000                   | kein Fehler   |
| 0081                   | Der Zielbereich ist größer als der Quellbereich.<br>Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs werden mit 0 aufgefüllt.   |
| 7000                   | Erstaufruf mit REQ = 0: keine Datenübertragung aktiv; BUSY hat den Wert "0".  |
| 7001                   | Erstaufruf mit REQ = 1: Datenübertragung angestoßen; BUSY hat den Wert "1".   |
| 7002                   | Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1"  |
| 8081                   | Der Quellbereich ist größer als der Zielbereich.<br>Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.   |
| 8091                   | Sie haben "CREATE_DB" geschachtelt aufgerufen.  |
| 8092                   | Die Funktion "Erzeugen eines DB" ist momentan nicht durchführbar, weil <ul style="list-style-type: none"> <li>• die Funktion "Komprimieren des Anwenderspeichers" gerade aktiv ist</li> <li>• die maximale Anzahl an Bausteinen auf Ihrer CPU bereits erreicht ist</li> </ul> |
| 8093                   | Beim Parameter SRCBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.  |
| 8094                   | Beim Parameter ATTRIB wurde ein noch nicht unterstütztes Attribut angegeben.  |

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 80A1  | Fehler bei der DB-Nummer: <ul style="list-style-type: none"> <li>• die Nummer ist "0"</li> <li>• Untergrenze &gt; Obergrenze</li> </ul>   |
| 80A2  | Fehler bei der DB-Länge: <ul style="list-style-type: none"> <li>• die Länge ist "0"</li> <li>• die Länge ist eine ungerade Zahl</li> <li>• die Länge ist größer, als die CPU zulässt</li> </ul> |
| 80A3  | Der Datenbaustein am Parameter SRCBLK wurde nicht mit Standardzugriff erstellt.   |
| 80B1  | Es ist keine DB-Nummer frei.  |
| 80B2  | Zu wenig Arbeitsspeicher frei.  |
| 80B4  | Die Memory-Card ist schreibgeschützt.   |
| 80BB  | Nicht genügend Ladespeicher frei.   |
| 80C0  | Das Ziel wird gerade durch eine andere Anweisung oder eine Kommunikationsfunktion bearbeitet.   |
| 80C1  | Löschen eines DBs mit dieser DB-Nummer wird zur Zeit durchgeführt.  |
| 80C3  | Die maximale Anzahl gleichzeitig aktiver "CREATE_DB"-Anweisungen ist momentan bereits erreicht.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## READ\_DBL: Aus Datenbaustein im Ladespeicher lesen

### Beschreibung

Mit der Anweisung kopieren Sie einen DB oder einen Teil eines DB, der im Ladespeicher (Micro Memory Card) liegt, in den Datenbereich eines Ziel-DB. Der Ziel-DB muss ablaufrelevant sein, d. h. er darf nicht mit dem Attribut UNLINKED erzeugt worden sein. Der Inhalt des Ladespeichers wird beim Kopiervorgang nicht verändert.

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Zielbereich während der Bearbeitung von "READ\_DBL" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

Für die Parameter SRCBLK und DSTBLK (Quellenbaustein und Zielbaustein) gelten folgende Einschränkungen:

- Die Länge des VARIANT-Pointers muss durch 8 teilbar sein.
- Bei einem VARIANT-Pointer vom Typ STRING muss die Länge gleich 1 sein.
- Quell- und Zielbaustein müssen mit dem gleichen Bausteinzugriff erstellt worden sein, d.h. beide müssen entweder die Zugriffsart "Optimiert" oder "Standard" verwenden.

---

#### Hinweis

"READ\_DBL" wird asynchron bearbeitet und eignet sich deshalb nicht dazu, Variablen häufig (bzw. zyklisch) im Ladespeicher zu lesen.

Ein einmal begonnener Auftrag wird stets zu Ende geführt. Wenn die maximale Anzahl gleichzeitig aktiver "READ\_DBL"-Anweisungen erreicht ist und Sie "READ\_DBL" zu diesem Zeitpunkt in einer Prioritätsklasse hoher Priorität erneut aufrufen, dann wird der Fehlercode W#16#80C3 zurückgeliefert. Es ist daher nicht sinnvoll, den hoch prioren Auftrag sofort wieder zu starten.

---

### Arbeitsweise

Die Anweisung "READ\_DBL" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "READ\_DBL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET\_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620)

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "READ\_DBL":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| REQ       | Input       | BOOL     | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Lesen  |
| SRCBLK    | Input       | VARIANT  | D                            | Zeiger auf den Datenbaustein im Ladespeicher, aus dem gelesen wird              |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlerinformation   |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | BUSY = 1: Der Lesevorgang ist noch nicht beendet.                               |
| DSTBLK    | Output      | VARIANT  | D                            | Zeiger auf den Datenbaustein im Arbeitsspeicher, in den geschrieben werden soll |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode*<br>(W#16#...)  | Beschreibung  |
|--|---|
| 0000   | kein Fehler   |
| 0081   | Der Zielbereich ist größer als der Quellbereich. Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs werden nicht verändert. |
| 7000   | Erstaufruf mit REQ=0: keine Datenübertragung aktiv; BUSY hat den Wert "0".  |
| 7001   | Erstaufruf mit REQ=1: Datenübertragung angestoßen; BUSY hat den Wert "1".   |
| 7002   | Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".   |
| 8081   | Der Quellbereich ist größer als der Zielbereich.<br>Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.                       |
| 8082   | Ziel-DB und Quell-DB sind von unterschiedlichem Typ (optimierter Zugriff/Standardzugriff).  |
| 8093   | Beim Parameter DSTBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.  |
| 80B1   | Beim Parameter SRCBLK ist kein Datenbaustein angegeben, oder der dort angegebene Datenbaustein ist kein Ladespeicherobjekt.   |
| 80B4   | DB mit F-Attribut darf nicht gelesen werden.  |
| 80C3   | Die maximale Anzahl gleichzeitig aktiver "READ_DBL"-Anweisungen ist momentan bereits erreicht.  |
| 80C0   | Der Ziel-DB wird gerade durch eine andere Anweisung oder eine Kommunikationsfunktion bearbeitet.  |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Siehe auch**

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 2304)

**WRIT\_DBL: In Datenbaustein im Ladespeicher schreiben**

**Beschreibung**

Mit der Anweisung "WRIT\_DBL" übertragen Sie den Inhalt eines DBs oder eines DB-Bereichs aus dem Arbeitsspeicher in einen DB oder einen DB-Bereich im Ladespeicher (Micro-Memory-Card). Der Quell-DB muss ablaufrelevant sein, d. h. er darf nicht mit dem Attribut UNLINKED erzeugt worden sein.

Um Datenkonsistenz zu gewährleisten, dürfen Sie den Quellbereich während der Bearbeitung von "WRIT\_DBL" (d. h. solange der Parameter BUSY den Wert TRUE hat) nicht verändern.

Für die Parameter SRCBLK und DSTBLK (Quellbaustein und Zielbaustein) gelten folgende Einschränkungen:

- Bei einem VARIANT-Pointer vom Typ BOOL muss die Länge durch 8 teilbar sein.
- Bei einem VARIANT-Pointer vom Typ STRING muss die Länge gleich 1 sein.
- Quell- und Zielbaustein müssen mit dem gleichen Bausteinzugriff erstellt worden sein, d. h. beide müssen entweder die Zugriffsart "Optimierter Bausteinzugriff" oder "Standard-kompatibel mit S7-300/400" verwenden.

Die Anweisung "WRIT\_DBL" verändert nicht die Prüfsumme des Anwenderprogramms, wenn Sie einen mittels Anweisung erzeugten DB beschreiben. Beim Beschreiben eines geladenen DB hingegen verändert das erste Schreiben auf diesen DB die Prüfsumme des Anwenderprogramms.

#### Hinweis

"WRIT\_DBL" eignet sich nicht dazu, Variablen häufig (bzw. zyklisch) im Ladespeicher zu schreiben, da auf eine Memory-Card technologiebedingt nur eine bestimmte Anzahl von Schreibzugriffen möglich ist.

### Arbeitsweise

Die Anweisung "WRIT\_DBL" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag, indem Sie "WRIT\_DBL" mit REQ = 1 aufrufen.

Über die Ausgangsparameter RET\_VAL und BUSY wird der Zustand des Auftrags angezeigt.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WRIT\_DBL":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Schreiben   |
| SRCBLK    | Input       | VARIANT  | D                            | Zeiger auf den Datenbaustein im Arbeitsspeicher, aus dem gelesen wird        |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlerinformation  |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | BUSY = 1: Der Schreibvorgang ist noch nicht beendet.                         |
| DSTBLK    | Output      | VARIANT  | D                            | Zeiger auf den Datenbaustein im Ladespeicher, in den geschrieben werden soll |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode* (W#16#...) | Beschreibung  |
|------------------------|---|
| 0000                   | kein Fehler   |
| 0081                   | Der Zielbereich ist größer als der Quellbereich. Der Quellbereich wird komplett in den Zielbereich geschrieben, die restlichen Bytes des Zielbereichs werden nicht verändert. |
| 7000                   | Erstauf Ruf mit REQ = 0: keine Datenübertragung aktiv; BUSY hat den Wert "0".   |

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 7001  | Erstaufruf mit REQ = 1: Datenübertragung angestoßen; BUSY hat den Wert "1".  |
| 7002  | Zwischenaufruf (REQ irrelevant): Datenübertragung bereits aktiv; BUSY hat den Wert "1".  |
| 8081  | Der Quellbereich ist größer als der Zielbereich.<br><b>Der Zielbereich wird komplett beschrieben, die restlichen Bytes des Quellbereichs werden ignoriert.</b>                       |
| 8082  | Ziel-DB und Quell-DB sind von unterschiedlichem Typ (optimierter/nicht optimierter Zugriff).   |
| 8092  | Falsche Betriebsart: Während "WRIT_DBL" aktiv war, ist die CPU in STOPP gegangen. Beim nächsten Übergang nach RUN wird dieser Fehlercode geliefert. Rufen Sie "WRIT_DBL" erneut auf. |
| 8093  | Beim Parameter SRCBLK ist kein Datenbaustein angegeben oder ein Datenbaustein, der nicht im Arbeitsspeicher steht.   |
| 80B1  | Beim Parameter DSTBLK ist kein Datenbaustein angegeben oder der dort angegebene Datenbaustein ist kein Ladespeicherobjekt.   |
| 80B4  | DB mit F-Attribut darf nicht gelesen werden.   |
| 80C3  | Die maximale Anzahl gleichzeitig aktiver "WRIT_DBL"-Anweisungen ist momentan bereits erreicht.   |
| 80C0  | Der Ziel-DB wird gerade durch eine andere Anweisung oder eine Kommunikationsfunktion bearbeitet.   |
| allgemeine Fehlercodes  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## ATTR\_DB: Attribute eines Datenbausteins lesen

### Beschreibung

Mit der Anweisung "ATTR\_DB" erhalten Sie Informationen über einen im Arbeitsspeicher der CPU liegenden Datenbaustein (DB). Die Anweisung ermittelt für den ausgewählten DB die gesetzten Attribute am Parameter ATTRIB.

Für Datenbausteine mit optimiertem Zugriff kann die Länge nicht ausgelesen werden, der Parameter DB\_LENGTH enthält für DBs mit optimiertem Zugriff die Länge "0".

Datenbausteine für Motion Control können mit der Anweisung "ATTR\_DB" nicht ausgelesen werden. Für diese wird der Fehlercode 80B2 ausgegeben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "ATTR\_DB":

| Parameter | Deklaration | Datentyp      | Speicherbereich              | Beschreibung   |
|-----------|-------------|---------------|------------------------------|--|
| REQ       | Input       | BOOL          | E, A, M, D, L oder Konstante | REQ = 1: Anforderung zum Lesen der Bausteinattribute   |
| DB_NUMBER | Input       | DB_ANY (UINT) | E, A, M, D, L oder Konstante | Nummer des zu überprüfenden DB                         |
| RET_VAL   | Output      | INT           | E, A, M, D, L                | Fehlerinformation                                      |
| DB_LENGTH | Output      | UDINT         | E, A, M, D, L                | Anzahl der Datenbytes, die der ausgewählte DB enthält. |



| Parameter   | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|---|-------------|----------|-----------------|---|
| ATTRIB  | Output      | BYTE     | E, A, M, D, L   | DB-Eigenschaften: <ul style="list-style-type: none"> <li>• Bit 0* = 0: LINKED - Der DB ist nur im Arbeitsspeicher</li> <li>• Bit 0* = 1: UNLINKED - Der DB ist nur im Ladespeicher</li> <li>• Bit 1 = 0: READ_ONLY - Der DB ist nicht schreibgeschützt.</li> <li>• Bit 1 = 1: READ_ONLY - Der DB ist schreibgeschützt.</li> <li>• Bit 2 = 0: RETAIN - Der DB ist remanent.</li> <li>• Bit 2 = 1: NON_RETAIN - Der DB ist nicht remanent.</li> <li>• Bit 3*= 0: Der DB ist entweder im Lade- oder im Arbeitsspeicher</li> <li>• Bit 3*= 1: Der DB ist sowohl im Lade- als auch im Arbeitsspeicher</li> </ul> |
| * Der Zusammenhang zwischen Bit 0 und Bit 3 ist in den Parametern der Anweisung "CREATE_DB: Datenbaustein erzeugen (Seite 2611)" erläutert. |             |          |                 |   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Erläuterung  |
|---|--|
| 0000  | Es ist kein Fehler aufgetreten.  |
| 80A1  | Fehler beim Eingangsparameter DB_NUMBER: Der gewählte Aktualparameter <ul style="list-style-type: none"> <li>• hat den Wert "0"</li> <li>• ist größer als die für die eingesetzte CPU maximal mögliche DB-Nummer.</li> </ul> |
| 80B1  | Der DB mit der angegebenen Nummer ist auf der CPU nicht vorhanden.   |
| 80B2  | Auf DBs der Technologieobjekte Motion Control kann mit der Anweisung "ATTR_DB" nicht zugegriffen werden.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### DELETE\_DB: Datenbaustein löschen

#### Beschreibung

Mit der Anweisung "DELETE\_DB" löschen Sie einen Datenbaustein (DB), der aus dem Anwenderprogramm durch Aufruf der Anweisung "CREATE\_DB (Seite 2611)" erzeugt wurde.

Wurde der Datenbaustein nicht durch "CREATE\_DB" erzeugt, wird am Parameter RET\_VAL der Fehlercode W#16#80B5 ausgegeben.

Der zu löschende DB darf sich weder in der aktuellen noch in einer niedrigeren Ablaufebe-  
 nen befinden. Bei Aufruf der Anweisung "DELETE\_DB" startet die CPU sonst den OB 121. Wenn  
 der OB 121 nicht vorhanden ist, wechselt die CPU in STOP. Die Anweisung "DELETE\_DB"  
 ist von höherprioritären Ablaufebenen unterbrechbar. Falls dort die Anweisung erneut aufgerufen  
 wird, wird dieser zweite Aufruf abgebrochen, und in RET\_VAL wird W#16#8091 eingetragen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "DELETE\_DB":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L oder Konstante | REQ =1: Anforderung zum Löschen des DB mit der Nummer am Parameter DB_NUMBER |
| DB_NUMBER | Input       | UINT     | E, A, M, D, L oder Konstante | Nummer des zu löschenden DBs   |
| RET_VAL   | Output      | INT      | E, A, M, D, L                | Fehlerinformation (siehe "Parameter RET_VAL")                                |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | BUSY =1: Der Vorgang ist noch nicht beendet.                                 |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode* (W#16#...)   | Erläuterung  |
|--|--|
| 0000   | Es ist kein Fehler aufgetreten.  |
| 8091   | Bei ineinander geschachtelten "DELETE_DB"-Aufrufen wurde die maximale Schachtelungstiefe der verwendeten CPU überschritten.  |
| 8092   | Die Anweisung kann momentan nicht ausgeführt werden, weil: <ul style="list-style-type: none"> <li>• Die Funktion "Komprimieren des Anwenderspeichers" gerade aktiv ist.</li> <li>• Sie den zu löschenden DB gerade von der CPU in ein Offline-Projekt kopieren.</li> </ul> |
| 80A1   | Fehler beim Eingangsparameter DB_NUMBER: <ul style="list-style-type: none"> <li>• Der Wert am Parameter ist "0".</li> <li>• Der Wert am Parameter ist größer als die für die eingesetzte CPU maximal mögliche DB-Nummer.</li> </ul>  |
| 80B1   | Der DB mit der angegebenen Nummer ist auf der CPU nicht vorhanden.   |
| 80B4   | Der DB kann nicht gelöscht werden, da die Memory-Card der CPU schreibgeschützt ist.  |
| 80B5   | Der DB wurde nicht über "CREATE_DB" erzeugt.   |
| 80C1   | Die Funktion "Löschen eines DB" ist wegen temporären Ressourcenengpässes momentan nicht durchführbar.  |
| allgemeine Fehlerinformation   | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

### 9.7.3.13 Adressierung

#### Anweisungen zur Adressumwandlung

##### Beschreibung

Es bestehen unterschiedliche Möglichkeiten zur Adressierung einer Baugruppe (EA-Adresse, Hardware-Kennung, Steckplatz).

Die Adressdaten können Sie mit den folgenden Anweisungen umwandeln:

- GEO2LOG: Aus dem Steckplatz die Hardware-Kennung ermitteln (Seite 2622)
- LOG2GEO: Aus der Hardware-Kennung den Steckplatz ermitteln (Seite 2624)
- LOG2MOD: Aus der Adressierung von STEP 7 V5.5 SPx die Hardware-Kennung ermitteln (Seite 2625)
- IO2MOD: Aus einer EA-Adresse die Hardware-Kennung ermitteln (Seite 2626)
- RD\_ADDR: Aus der Hardware-Kennung die EA-Adressen ermitteln (Seite 2627)

Für migrierte Projekte werden darüber hinaus die folgenden Anweisungen unterstützt:

- GEO\_LOG: Aus dem Steckplatz die Hardware-Kennung ermitteln (Seite 2628)
- LOG\_GEO: Aus der Hardware-Kennung den Steckplatz ermitteln (Seite 2630)
- RD\_LGADR: Aus der Hardware-Kennung die EA-Adressen ermitteln (Seite 2631)
- GADR\_LGC: Aus dem Steckplatz und dem Offset im Nutzdatenadressraum die Hardware-Kennung ermitteln (Seite 2633)
- LGC\_GADR: Aus der Hardware-Kennung den Steckplatz ermitteln (Seite 2634)

### Art der Adressumwandlung

Das folgende Bild zeigt, welche Anweisung welche Adressumwandlung durchführt.

| Name     | Typ | EA-Adresse(n) | HW-Kennung | Steckplatz |
|----------|-----|---------------|------------|------------|
| GEO2LOG  | SFC |               | ←          | ●          |
| LOG2GEO  | SFC |               | ●          | →          |
| LOG2MOD  | SFC | ●             | →          |            |
| IO2MOD   | SFC | ●             | →          |            |
| RD_ADDR  | SFC | ←             | ●          |            |
| GEO_LOG  | FC  |               | ←          | ●          |
| LOG_GEO  | FC  |               | ●          | →          |
| RD_LGADR | FC  | ←             | ●          |            |
| GADR_LGC | FC  |               | ←          | ●          |
| LGC_GADR | FC  |               | ●          | →          |

### GEO2LOG: Aus dem Steckplatz die Hardware-Kennung ermitteln

#### Beschreibung

Mit der Anweisung "LOG2GEO" ermitteln Sie die logische Adresse anhand der Steckplatzinformationen, welche Sie über den Systemdatentyp GEOADDR definieren.

Abhängig von dem Hardwaretyp, den Sie am Parameter HWTYPE definieren, werden folgende Informationen aus den übrigen Parameter von GEOADDR ausgewertet:

- Bei HWTYPE = 1 (PROFINET IO-System):
  - Es wird nur IOSYSTEM ausgewertet. Die übrigen Parameter von GEOADDR werden nicht berücksichtigt.
  - Die Hardware-Kennung des PROFINET IO-Systems wird ausgegeben.
- Bei HWTYPE = 2 (PROFINET IO-Device):
  - Es werden IOSYSTEM und STATION ausgewertet. Die übrigen Parameter von GEOADDR werden nicht berücksichtigt.
  - Die Hardware-Kennung des PROFINET IO-Device wird ausgegeben.
- Bei HWTYPE = 3 (Baugruppenträger):
  - Es werden nur IOSYSTEM und STATION ausgewertet. Die übrigen Parameter von GEOADDR werden nicht berücksichtigt.
  - Die Hardware-Kennung des Baugruppenträgers wird ausgegeben.

- Bei HWTYPE = 4 (Modul):
  - Es werden IOSYSTEM, STATION und SLOT ausgewertet. Der Parameter SUBSLOT von GEOADDR wird nicht berücksichtigt.
  - Die Hardware-Kennung des Moduls wird ausgegeben.
- Bei HWTYPE = 5 (Submodul):
  - Es werden alle Parameter von GEOADDR ausgewertet.
  - Die Hardware-Kennung des Submoduls wird ausgegeben.

Der Parameter AREA des Systemdatentyps GEOADDR wird nicht ausgewertet.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GEO2LOG":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| GEOADDR   | Input       | VARIANT  | D, L            | Zeiger auf die Struktur des Systemdatentyps GEOADDR.  |
| RET_VAL   | Return      | INT      | E, A, M, D, L   | Ausgabe der Fehlerinformation.  |
| LADDR     | Output      | HW_ANY   | E, A, M, D, L   | Hardware-Kennung der Baugruppe oder des Moduls.<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder des Interface in der Hardware-Konfiguration hinterlegt. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Systemdatentyp GEOADDR

Der Systemdatentyp GEOADDR ist eine Struktur mit dem folgenden Aufbau:

| Parametername | Datentyp | Beschreibung   |
|---------------|----------|--|
| GEOADDR       | STRUCT   |  |
| HWTYPE        | UNIT     | Hardwaretyp: <ul style="list-style-type: none"> <li>• 1: PROFINET IO-System</li> <li>• 2: PROFINET IO-Device</li> <li>• 3: Baugruppenträger</li> <li>• 4: Modul</li> <li>• 5: Submodul</li> </ul> Wird der Hardwaretyp von der Anweisung nicht unterstützt, wird an HWTYPE "0" ausgegeben. |
| AREA          | UNIT     | Bereichskennung (0=Zentralbaugruppe)   |
| IOSYSTEM      | UNIT     | PROFINET IO-System (0=Zentralgerät im Baugruppenträger 0-3)  |
| STATION       | UNIT     | <ul style="list-style-type: none"> <li>• Nummer des Baugruppenträgers, falls Bereichskennung AREA = 0.</li> <li>• Stationsnummer, falls Bereichskennung AREA &gt; 0.</li> </ul>  |

| Parametername | Datentyp | Beschreibung   |
|---------------|----------|--|
| SLOT          | UNIT     | Steckplatz-Nummer  |
| SUBSLOT       | UNIT     | Nummer des Submoduls. Falls kein Submodul gesteckt werden kann, hat dieser Parameter den Wert "0". |

### Parameter RET\_VAL

| Fehlercode* (W#16#...) | Erläuterung                              |
|------------------------|--|
| 0                      | Kein Fehler aufgetreten.                 |
| 8091                   | Ungültiger Wert in GEOADDR für HWTYPE.   |
| 8094                   | Ungültiger Wert in GEOADDR für IOSYSTEM. |
| 8095                   | Ungültiger Wert in GEOADDR für STATION.  |
| 8096                   | Ungültiger Wert in GEOADDR für SLOT.     |
| 8097                   | Ungültiger Wert in GEOADDR für SUBSLOT.  |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Siehe auch

Anweisungen zur Adressumwandlung (Seite 2621)

### LOG2GEO: Aus der Hardware-Kennung den Steckplatz ermitteln

#### Beschreibung

Mit der Anweisung "LOG2GEO" ermitteln Sie den zu einer Hardware-Kennung gehörenden Baugruppensteckplatz.

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LOG2GEO":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| LADDR     | Input       | HW_ANY   | E, A, M, D, L oder Konstante | Hardware-Kennung des IO-Systems oder des Moduls.<br>Die Nummer wird automatisch vergeben und ist in den Eigenschaften der CPU oder des Interface in der Hardware-Konfiguration hinterlegt. |
| RET_VAL   | Output      | INT      | E, A, M, D, L                | Ausgabe der Fehlerinformation.   |
| GEOADDR   | InOut       | VARIANT  | D                            | Zeiger auf den Systemdatentyp GEOADDR.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Systemdatentyp GEOADDR

Der Systemdatentyp GEOADDR ist eine Struktur mit dem folgenden Aufbau:

| Parametername | Datentyp | Beschreibung  |
|---------------|----------|---|
| GEOADDR       | STRUCT   |   |
| HWTYPE        | UNIT     | Hardwaretyp: <ul style="list-style-type: none"> <li>• 1: IO-System (PROFINET/PROFIBUS)</li> <li>• 2: IO-Device / DP-Slave</li> <li>• 3: Baugruppenträger</li> <li>• 4: Modul</li> <li>• 5: Submodul</li> </ul> Wird der Hardwaretyp von der Anweisung nicht unterstützt, wird an HWTYPE "0" ausgegeben. |
| AREA          | UNIT     | Bereichskennung: <ul style="list-style-type: none"> <li>• 0 = Zentralbaugruppe</li> <li>• 1 = PROFINET IO</li> <li>• 2 = PROFIBUS DP</li> </ul>   |
| IOSYSTEM      | UNIT     | PROFINET IO-System (0=Zentralgerät im Baugruppenträger 0-3)   |
| STATION       | UNIT     | <ul style="list-style-type: none"> <li>• Nummer des Baugruppenträgers, falls Bereichskennung AREA = 0.</li> <li>• Stationsnummer, falls Bereichskennung AREA &gt; 0.</li> </ul>   |
| SLOT          | UNIT     | Steckplatz-Nummer   |
| SUBSLOT       | UNIT     | Nummer des Submoduls. Falls kein Submodul gesteckt werden kann, hat dieser Parameter den Wert "0".  |

## Parameter RET\_VAL

| Fehlercode* (W#16#...) | Erläuterung   |
|------------------------|---|
| 0                      | Kein Fehler aufgetreten.                                |
| 8090                   | Die am Parameter LADDR angegebene Adresse ist ungültig. |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## LOG2MOD: Aus der Adressierung von STEP 7 V5.5 SPx die Hardware-Kennung ermitteln

### Beschreibung

Mit der Anweisung "LOG2MOD" ermitteln Sie für ein EA-(Sub)Modul aus der Adressierung von STEP 7 5.5 SPx (EA-Datenadresse oder Diagnoseadresse) die Hardware-Kennung.

Die Hardware-Kennung wird an dem Eingangsparameter LADDR von diversen Anweisungen zur Adressierung verwendet. Durch den vorhergehenden Aufruf von "LOG2MOD" können Sie die Adressierungsparameter aus STEP 7 5.5 SPx konvertieren.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "LOG2MOD":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| IOID      | Input       | BYTE     | E, A, M, D, L oder Konstante | Kennung des Adressbereichs wie in STEP 7 5.5 SPx: <ul style="list-style-type: none"> <li>• B#16#00: Bit15 von ADDR gibt an, ob Ein- (Bit15=0) oder Ausgangsadresse (Bit15=1) vorliegt.</li> <li>• B#16#54 = Peripherie Eingang (PE)</li> <li>• B#16#55 = Peripherie Ausgang (PA)</li> </ul> |
| ADDR      | Input       | WORD     | E, A, M, D, L oder Konstante | Logische Adresse der EA-Daten des Moduls als Offset (entsprechend Adressierung in STEP 7 5.5 SPx) oder Diagnoseadresse.   |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlercode der Anweisung.   |
| HWID      | Output      | HW_IO    | E, A, M, D, L                | Ermittelte Hardware-Kennung (logische Adresse) des EA-(Sub)Moduls.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Erläuterung   |
|---|---|
| 0   | Kein Fehler aufgetreten.  |
| 8093  | <ul style="list-style-type: none"> <li>• Angegebene Adresse wird von keiner Hardware-Komponente verwendet.</li> <li>• Angegebener Wert am Parameter IOID ist ungültig.</li> </ul> |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## IO2MOD: Aus einer EA-Adresse die Hardware-Kennung ermitteln

### Beschreibung

Die Anweisung "IO2MOD" ermittelt aus einer EA-Adresse (E, A, PE, PA) eines (Sub)Moduls die Hardware-Kennung des Moduls.

Am Parameter ADDR geben Sie die EA-Adresse an. Wird am diesem Parameter ein Reihe von EA-Adressen verwendet, wird nur die erste Adresse ausgewertet, um die Hardware-Kennung zu ermitteln. Wird die erste Adresse korrekt angegeben, spielt die Länge für die Adressangabe am Parameter ADDR keine Rolle. Auch wenn ein Adressbereich verwendet wird, der mehrere Module umfasst oder nicht verwendete Adressen enthält, kann die Hardware-Kennung des ersten Moduls ermitteln werden.

Wird an dem Parameter ADDR keine EA-Adresse eines Sub(Modul) angegeben, wird am Parameter RET\_VAL der Fehlercode 8090 ausgegeben.



## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "IO2MOD":

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| ADDR      | Input       | VARIANT  | E, A, M, D, L   | EA-Adresse (E, A, PE, PA) innerhalb eines (Sub)Moduls.             |
| RET_VAL   | Return      | INT      | E, A, M, D, L   | Fehlercode der Anweisung.  |
| LADDR     | Output      | HW_IO    | E, A, M, D, L   | Ermittelte Hardware-Kennung (logische Adresse) des EA-(Sub)Moduls. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Erläuterung  |
|---|--|
| 0   | Kein Fehler aufgetreten.   |
| 8090  | Angegebene EA-Adresse am Parameter ADDR wird von keiner Hardware-Komponente verwendet. |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

## RD\_ADDR: Aus der Hardware-Kennung die EA-Adressen ermitteln

### Beschreibung

Die Anweisung "RD\_ADDR" ermittelt anhand der Hardware-Kennung eines (Sub-)Moduls die Länge und die Anfangsadresse der Ein- bzw. Ausgänge.

- Über den Parameter LADDR wählen Sie anhand der Hardware-Kennung das Ein- oder Ausgangsmodul aus.
- Je nachdem, ob es sich um ein Ein- oder Ausgangsmodul handelt, werden die folgenden Ausgangsparameter verwendet:
  - Bei einem Eingabemodul werden die ermittelten Werte an den Parametern PIADDR und PICOUNT ausgegeben.
  - Bei einem Ausgabemodul werden die ermittelten Werte an den Parametern PQADDR und PQCOUNT ausgegeben.
- Die Parameter PIADDR und PQADDR enthalten jeweils die Anfangsadresse der E/A-Adressen des Moduls.
- Die Parameter PICOUNT und PQCOUNT enthalten jeweils die Anzahl Bytes der Ein- bzw. Ausgänge (1 Byte bei 8 Ein-/Ausgängen, 2 Byte bei 16 Ein-/Ausgängen).

**Hinweis**

**Adressierung von gepackten Modulen**

Werden gepackte Module (nicht das erste Modul der gepackten Modulgruppe) adressiert, so weichen die angezeigten Daten von der Hardware-Konfiguration ab. Es wird für PIADDR bzw. PQADDR und PICOUNT bzw. PQCOUNT "0" zurückgegeben. RET\_VAL zeigt keinen Fehler an (16#0000).

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_ADDR":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung                       |
|-----------|-------------|----------|------------------------------|------------------------------------|
| LADDR     | Input       | HW_IO    | E, A, M, D, L oder Konstante | Hardware-Kennung des (Sub-)Moduls. |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlercode der Anweisung.          |
| PIADDR    | Output      | UDINT    | E, A, M, D, L                | Anfangsadresse des Eingabemoduls.  |
| PICOUNT   | Output      | UINT     | E, A, M, D, L                | Anzahl Bytes der Eingänge.         |
| PQADDR    | Output      | UDINT    | E, A, M, D, L                | Anfangsadresse des Ausgabemoduls.  |
| PQCOUNT   | Output      | UINT     | E, A, M, D, L                | Anzahl Bytes der Ausgänge.         |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode* (W#16#...)   | Erläuterung  |
|--|--|
| 0  | Kein Fehler aufgetreten.                                 |
| 8090   | Hardware-Kennung des Moduls am Parameter LADDR ungültig. |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**Weitere**

**GEO\_LOG: Aus dem Steckplatz die Hardware-Kennung ermitteln**

**Beschreibung**

Vom Kanal einer Signalbaugruppe sei der zugehörige Baugruppensteckplatz der Baugruppe bekannt. Mit der Anweisung "GEO\_LOG" ermitteln Sie daraus die zugehörige Hardware-Kennung der Baugruppe.

Wenn Sie die Anweisung "GEO\_LOG" auf Power-Module oder Module mit gepackten Adressen (ET 200S) anwenden, wird die Diagnoseadresse zurückgeliefert.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GEO\_LOG":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| MASTER    | Input       | INT      | E, A, M, D, L oder Konstante | Bereichskennung: <ul style="list-style-type: none"> <li>• 0, falls sich der Steckplatz im zentralen Aufbau befindet.</li> <li>• 1 bis 32: DP-Mastersystem-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFIBUS befindet</li> <li>• 100 bis 115: PROFINET IO-System-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFINET befindet</li> </ul> |
| STATION   | Input       | INT      | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• Wenn MASTER = 0: Nummer des Baugruppenträgers</li> <li>• Wenn MASTER &gt; 0: Stationsnummer des Feldgeräts</li> </ul>  |
| SLOT      | Input       | INT      | E, A, M, D, L oder Konstante | Steckplatz-Nummer   |
| SUBSLOT   | Input       | INT      | E, A, M, D, L oder Konstante | Der Parameter SUBLOT wird von der Anweisung nicht ausgewertet.  |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlerinformation   |
| LADDR     | Output      | HW_IO    | E, A, M, D, L                | Hardware-Kennung oder Diagnoseadresse der Baugruppe   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...) | Erläuterung   |
|---------------------------|---|
| 0000                      | Es ist kein Fehler aufgetreten.                                   |
| 8094                      | Es wurde kein Subnetz mit der angegebenen SUBNETID konfiguriert.  |
| 8095                      | Unzulässiger Wert beim Parameter STATION                          |
| 8096                      | Unzulässiger Wert beim Parameter SLOT                             |
| 8099                      | Der Steckplatz ist nicht projektiert.                             |
| 809A                      | Die Nummer für den ausgewählten Steckplatz ist nicht projektiert. |

|   |   |
|---|---|
| <b>Fehlercode*<br/>(W#16#...)</b>   | <b>Erläuterung</b>  |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832) |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Siehe auch**

Anweisungen zur Adressumwandlung (Seite 2621)

**LOG\_GEO: Aus der Hardware-Kennung den Steckplatz ermitteln**

**Beschreibung**

Mit der Anweisung "LOG\_GEO" ermitteln Sie den zu einer Hardware-Kennung gehörenden Baugruppensteckplatz.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "LOG\_GEO":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| LADDR     | Input       | HW_IO    | E, A, M, D, L oder Konstante | Hardware-Kennung des Moduls, zu dem der Steckplatz ermittelt werden soll.  |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlerinformation  |
| AREA      | Output      | INT      | E, A, M, D, L                | Bereichskennung: Sie gibt an, wie die restlichen Ausgangsparameter zu interpretieren sind: <ul style="list-style-type: none"> <li>• 0: Zentralgerät</li> <li>• 2: PROFIBUS DP / PROFINET IO</li> </ul>   |
| MASTER    | Output      | INT      | E, A, M, D, L oder Konstante | Bei AREA = 0: <ul style="list-style-type: none"> <li>• 0: Falls sich der Steckplatz in einem der Baugruppenträger (Zentralgerät) befindet.</li> </ul> Bei AREA = 2: <ul style="list-style-type: none"> <li>• 1 bis 32: DP-Mastersystem-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFIBUS befindet</li> <li>• 100 bis 115: PROFINET IO-System-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFINET befindet</li> </ul> |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| STATION   | Output      | INT      | E, A, M, D, L   | <ul style="list-style-type: none"> <li>Bei MASTER = 0: Nummer des Baugruppenträgers</li> <li>Bei MASTER &gt; 0: Stationsnummer des Feldgeräts</li> </ul> |
| SLOT      | Output      | INT      | E, A, M, D, L   | Steckplatz-Nummer  |
| SUBSLOT   | Output      | INT      | E, A, M, D, L   | Der Parameter SUBSLOT wird von der Anweisung nicht ausgegeben (immer "0").   |
| OFFSET    | Output      | INT      | E, A, M, D, L   | Der Parameter OFFSET wird von der Anweisung nicht ausgegeben (immer "0").  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode* (W#16#...)  | Erläuterung   |
|---|---|
| 0000  | Es ist kein Fehler aufgetreten.                               |
| 8090  | Angegebene logische Adresse ungültig                          |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832) |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## Siehe auch

Anweisungen zur Adressumwandlung (Seite 2621)

## RD\_LGADR: Aus der Hardware-Kennung die EA-Adressen ermitteln

### Beschreibung

Sie gehen von einer Hardware-Kennung einer Baugruppe, eines zentralen Submoduls oder eines Submoduls bei PNIO aus. Mit der Anweisung ermitteln Sie alle vereinbarten logischen Adressen dieser Baugruppe bzw. dieses Submoduls. Die Anweisung "RD\_LGADR" trägt die ermittelten logischen Adressen in den Parametern PEADDR bzw. PAADDR in aufsteigender Reihenfolge ein.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "RD\_LGADR":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| IOID      | Input       | BYTE     | E, A, M, D, L oder Konstante | Kennung des Adressbereichs:<br><ul style="list-style-type: none"> <li>• B#16#54 = Peripherie Eingang (PE)</li> <li>• B#16#55 = Peripherie Ausgang (PA)</li> </ul> |
| LADDR     | Input       | HW_ANY   | E, A, M, D, L oder Konstante | Hardware-Kennung der Baugruppe bzw. des Submoduls.  |
| RET_VAL   | Return      | INT      | E, A, M, D, L                | Fehlerinformation   |
| PEADDR    | Output      | ANY      | E, A, M, D, L                | Feld für die PE-Adressen, Feldelemente müssen vom Datentyp WORD sein.   |
| PECOUNT   | Output      | INT      | E, A, M, D, L                | Anzahl der zurückgelieferten PE-Adressen  |
| PAADDR    | Output      | ANY      | E, A, M, D, L                | Feld für die PA-Adressen, Feldelemente müssen vom Datentyp WORD sein.   |
| PACOUNT   | Output      | INT      | E, A, M, D, L                | Anzahl der zurückgelieferten PA-Adressen  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode (W#16#...)        | Erläuterung   |
|------------------------------|---|
| 0000                         | Es ist kein Fehler aufgetreten.   |
| 8090                         | Angegebene logische Adresse ungültig oder unzulässiger Wert beim Parameter IOID                           |
| 80A0                         | Fehler beim Ausgangsparameter PEADDR: Der Datentyp der Feldelemente ist nicht WORD.                       |
| 80A1                         | Fehler beim Ausgangsparameter PAADDR: Der Datentyp der Feldelemente ist nicht WORD.                       |
| 80A2                         | Fehler beim Ausgangsparameter PEADDR: Das angegebene Feld konnte nicht alle logischen Adressen aufnehmen. |
| 80A3                         | Fehler beim Ausgangsparameter PAADDR: Das angegebene Feld konnte nicht alle logischen Adressen aufnehmen. |
| allgemeine Fehlerinformation | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   |

**Siehe auch**

Anweisungen zur Adressumwandlung (Seite 2621)

## GADR\_LGC: Aus dem Steckplatz und dem Offset im Nutzdatenadressraum die Hardware-Kennung ermitteln

### Beschreibung

Mit der Anweisung "GADR\_LGC" ermitteln Sie die Hardware-Kennung einer Signalbaugruppe. Die Hardware-Kennung wird aus dem Baugruppensteckplatz und dem Offset im Nutzdatenadressraum der Baugruppe ermittelt.

Wenn Sie die Anweisung "GADR\_LGC" auf Power-Module oder Module mit gepackten Adressen (ET 200S) anwenden, wird die Diagnoseadresse zurückgeliefert.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GADR\_LGC":

| Parameter | Deklaration | Datentyp  | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------|------------------------------|---|
| SUBNETID  | Input       | BYTE      | E, A, M, D, L oder Konstante | Bereichskennung: <ul style="list-style-type: none"> <li>• 0: Falls sich der Steckplatz in der Zentralbaugruppe befindet</li> <li>• 1 bis 32: DP-Mastersystem-ID des zugehörigen dezentralen Peripheriesystems, falls sich der Steckplatz in einem dezentralen Peripheriegerät befindet</li> <li>• 100 bis 115: PROFINET IO-System-ID des zugehörigen Feldgeräts, falls sich der Steckplatz in einem Feldgerät am PROFINET befindet</li> </ul> |
| RACK      | Input       | WORD      | E, A, M, D, L oder Konstante | <ul style="list-style-type: none"> <li>• Nummer des Baugruppenträgers, falls Bereichskennung 0</li> <li>• Stationsnummer des dezentralen Peripheriegeräts, falls Bereichskennung &gt; 0</li> </ul>  |
| SLOT      | Input       | WORD      | E, A, M, D, L oder Konstante | Steckplatz-Nr.  |
| SUBSLOT   | Input       | BYTE      | E, A, M, D, L oder Konstante | Submodulsteckplatz (falls kein Submodul gesteckt werden kann, ist hier 0 anzugeben)   |
| SUBADDR   | Input       | WORD      | E, A, M, D, L oder Konstante | Offset im Nutzdatenadressraum der Baugruppe   |
| RET_VAL   | Return      | INT       | E, A, M, D, L                | Fehlerinformation   |
| IOID      | Output      | BYTE      | E, A, M, D, L                | Der Ausgangsparameter IOID wird nicht beschrieben (immer "0").  |
| LADDR     | Output      | HW_MODULE | E, A, M, D, L                | Hardware-Kennung der Baugruppe  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter RET\_VAL**

| Fehlercode*<br>(W#16#...)   | Erläuterung   |
|---|---|
| 0000  | Es ist kein Fehler aufgetreten.   |
| 8089  | Unzulässiger Wert beim Parameter SUBADDR.   |
| 8093  | Unzulässiger Wert beim Parameter SUBNETID   |
| 8094  | Es wurde kein Subnetz mit der angegebenen SUBNETID konfiguriert.  |
| 8095  | Unzulässiger Wert beim Parameter RACK.  |
| 8096  | Unzulässiger Wert beim Parameter SLOT.  |
| 8097  | Unzulässiger Wert beim Parameter SUBSLOT.   |
| 8099  | Der Steckplatz ist nicht projektiert.   |
| 809A  | Die Subadresse für den ausgewählten Steckplatz ist nicht projektiert (nur möglich bei zentraler Peripherie für CPU und IM). |
| allgemeine Fehlerinformation  | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Siehe auch**

Anweisungen zur Adressumwandlung (Seite 2621)

**LGC\_GADR: Aus der Hardware-Kennung den Steckplatz ermitteln**

**Beschreibung**

Mit der Anweisung "LGC\_GADR" ermitteln Sie den zu einer Hardware-Kennung gehörenden Baugruppensteckplatz einer Baugruppe.

**Hinweis**

Die Anwendung der Anweisung "LGC\_GADR" auf ein Modul mit gepackten Adressen (ET 200S) ist nicht möglich.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "LGC\_GADR":

| Parameter | Deklaration | Datentyp      | Speicherbereich              | Beschreibung                   |
|-----------|-------------|---------------|------------------------------|--------------------------------|
| IOID      | Input       | BYTE          | E, A, M, D, L oder Konstante | Wird nicht ausgewertet.        |
| LADDR     | Input       | HW_MODUL<br>E | E, A, M, D, L oder Konstante | Hardware-Kennung der Baugruppe |



| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| RET_VAL   | Return      | INT      | E, A, M, D, L   | Fehlerinformation  |
| AREA      | Output      | BYTE     | E, A, M, D, L   | Bereichskennung: Sie gibt an, wie die restlichen Ausgangsparameter zu interpretieren sind: <ul style="list-style-type: none"> <li>• 0: Zentralbaugruppe</li> <li>• 2: PROFIBUS DP</li> </ul>   |
| RACK      | Output      | WORD     | E, A, M, D, L   | Baugruppenträger-Nummer: <ul style="list-style-type: none"> <li>• Bei Zentralbaugruppe (AREA=0): <ul style="list-style-type: none"> <li>– Baugruppenträger-Nummer</li> </ul> </li> <li>• Bei PROFIBUS DP (AREA=2): <ul style="list-style-type: none"> <li>– Low Byte: Stationsnummer</li> <li>– High Byte: DP-Mastersystem-ID</li> </ul> </li> </ul> |
| SLOT      | Output      | WORD     | E, A, M, D, L   | Steckplatz-Nummer: <ul style="list-style-type: none"> <li>• Bei Zentralbaugruppe (AREA=0): <ul style="list-style-type: none"> <li>– Steckplatz-Nummer</li> </ul> </li> <li>• Bei PROFIBUS DP (AREA=2): <ul style="list-style-type: none"> <li>– Steckplatz-Nr. in der Station</li> </ul> </li> </ul>   |
| SUBADDR   | Output      | WORD     | E, A, M, D, L   | Wird nicht ausgegeben (immer "0").   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter RET\_VAL

| Fehlercode*<br>(W#16#...)  | Erläuterung   |
|--|---|
| 0000   | Es ist kein Fehler aufgetreten.   |
| 8090   | Angegebene logische Adresse ungültig oder unzulässiger Wert beim Parameter IOID                     |
| 8093   | Für die über die Parameter IOID und LADDR ausgewählte Baugruppe ist diese Anweisung nicht zulässig. |
| allgemeine Fehlerinformation   | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)                                       |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## Siehe auch

Anweisungen zur Adressumwandlung (Seite 2621)

## 9.7.4 Technologie

### 9.7.4.1 S7-1200 Motion Control

#### MC\_Power

#### MC\_Power: Achsen freigeben, sperren

#### Beschreibung

Die Motion Control-Anweisung "MC\_Power" gibt eine Achse frei, bzw. sperrt diese.

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.
- Es steht kein freigabeverhindernder Fehler an.

#### Ablöseverhalten

Die Bearbeitung des "MC\_Power" kann durch keinen Motion Control-Auftrag abgebrochen werden.

Mit dem Sperren der Achse (Eingangsparameter "Enable" = FALSE ) werden alle Motion Control-Aufträge entsprechend dem gewählten "StopMode" am zugehörigen Technologieobjekt abgebrochen.

#### Parameter

| Parameter | Deklaration | Datentyp  | Defaultwert | Beschreibung                |  |
|-----------|-------------|-----------|-------------|-----------------------------|--|
| Axis      | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse |  |
| Enable    | INPUT       | BOOL      | FALSE       | TRUE                        | Motion Control versucht die Achse freizugeben.   |
|           |             |           |             | FALSE                       | Alle laufenden Aufträge werden entsprechend dem parametrisierten "StopMode" abgebrochen und die Achse wird gestoppt..  |
| StopMode  | INPUT       | INT       | 0           | 0                           | Notstopp<br>Steht eine Anforderung zum Sperren der Achse an, so bremst die Achse mit der konfigurierten Notstopp-Verzögerung ab. Beim Erreichen des Stillstands wird die Achse gesperrt. |
|           |             |           |             | 1                           | Sofort-Aus<br>Steht eine Anforderung zum Sperren der Achse an, so wird diese ohne Verzögerung gesperrt. Die Impulsausgabe wird sofort gestoppt.  |

| Parameter | Deklaration | Datentyp | Defaultwert | Beschreibung  |
|-----------|-------------|----------|-------------|---|
|           |             |          |             | 2<br>Notstopp mit Ruckbegrenzung<br>Steht eine Anforderung zum Sperren der Achse an, so bremst die Achse mit der konfigurierten Notstopp-Verzögerung ab. Ist die Ruckbegrenzung aktiviert, so wird der konfigurierte Ruck berücksichtigt. Beim Erreichen des Stillstands wird die Achse gesperrt.   |
| Status    | OUTPUT      | BOOL     | FALSE       | Status der Achsfreigabe   |
|           |             |          |             | FALSE<br>Die Achse ist gesperrt<br>Die Achse führt keine Motion Control-Aufträge aus und nimmt keine neuen Aufträge an (Ausnahme: MC_Reset-Auftrag).<br>Die Achse ist nicht referenziert.<br>Beim Sperren wechselt der Status erst mit dem Erreichen des Stillstandes der Achse auf FALSE.  |
|           |             |          |             | TRUE<br>Die Achse ist freigegeben<br>Die Achse ist bereit Motion Control-Aufträge auszuführen.<br>Beim Freigeben der Achse wechselt der Status erst mit dem Anstehen des Signals "Antrieb bereit" auf TRUE. Wurde das Antriebssignal "Antrieb bereit" in der Achskonfiguration nicht konfiguriert, so wechselt der Status sofort nach TRUE. |
| Busy      | OUTPUT      | BOOL     | FALSE       | TRUE<br>MC-Power ist aktiv  |
| Error     | OUTPUT      | BOOL     | FALSE       | TRUE<br>An der Motion Control-Anweisung "MC_Power" oder am zugehörigen Technologieobjekt ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.   |
| ErrorID   | OUTPUT      | WORD     | 16#0000     | Fehlerkennung (Seite 4233) zum Parameter "Error"  |
| ErrorInfo | OUTPUT      | WORD     | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID"  |

### Hinweis

Wird die Achse durch einen Fehler ausgeschaltet, so wird sie nach dem Beseitigen und Quittieren des Fehlers automatisch wieder freigegeben. Voraussetzung hierfür ist, dass der Eingangsparameter "Enable" während dieses Vorgangs den Wert TRUE beibehalten hat.

### Freigeben einer Achse mit konfigurierten Antriebssignalen

Gehen Sie zum Freigeben der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie den Eingangsparameter "StopMode" mit dem gewünschten Wert. Setzen Sie den Eingangsparameter "Enable" auf TRUE.  
Der Freigabe-Ausgang für die "Antriebsfreigabe" wechselt auf TRUE, um dem Antrieb die Leistungsfreigabe zu erteilen. Die CPU wartet auf das Signal "Antrieb bereit" des Antriebs. Mit dem Anstehen des Signals "Antrieb bereit" am konfigurierten Bereit-Eingang der CPU wird die Achse freigegeben. Der Ausgangsparameter "Status", sowie die Variable des Technologieobjekts <Achurname>.StatusBits.Enable zeigen den Wert TRUE.

### Freigeben einer Achse ohne konfigurierte Antriebssignale

Gehen Sie zum Freigeben der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie den Eingangsparameter "StopMode" mit dem gewünschten Wert. Setzen Sie den Eingangsparameter "Enable" auf TRUE. Die Achse wird freigegeben. Der Ausgangsparameter "Status", sowie die Variable des Technologieobjekts <Achurname>.StatusBits.Enable zeigen den Wert TRUE.

### Sperren einer Achse

Zum Sperren einer Achse können Sie wie nachfolgend beschrieben vorgehen:

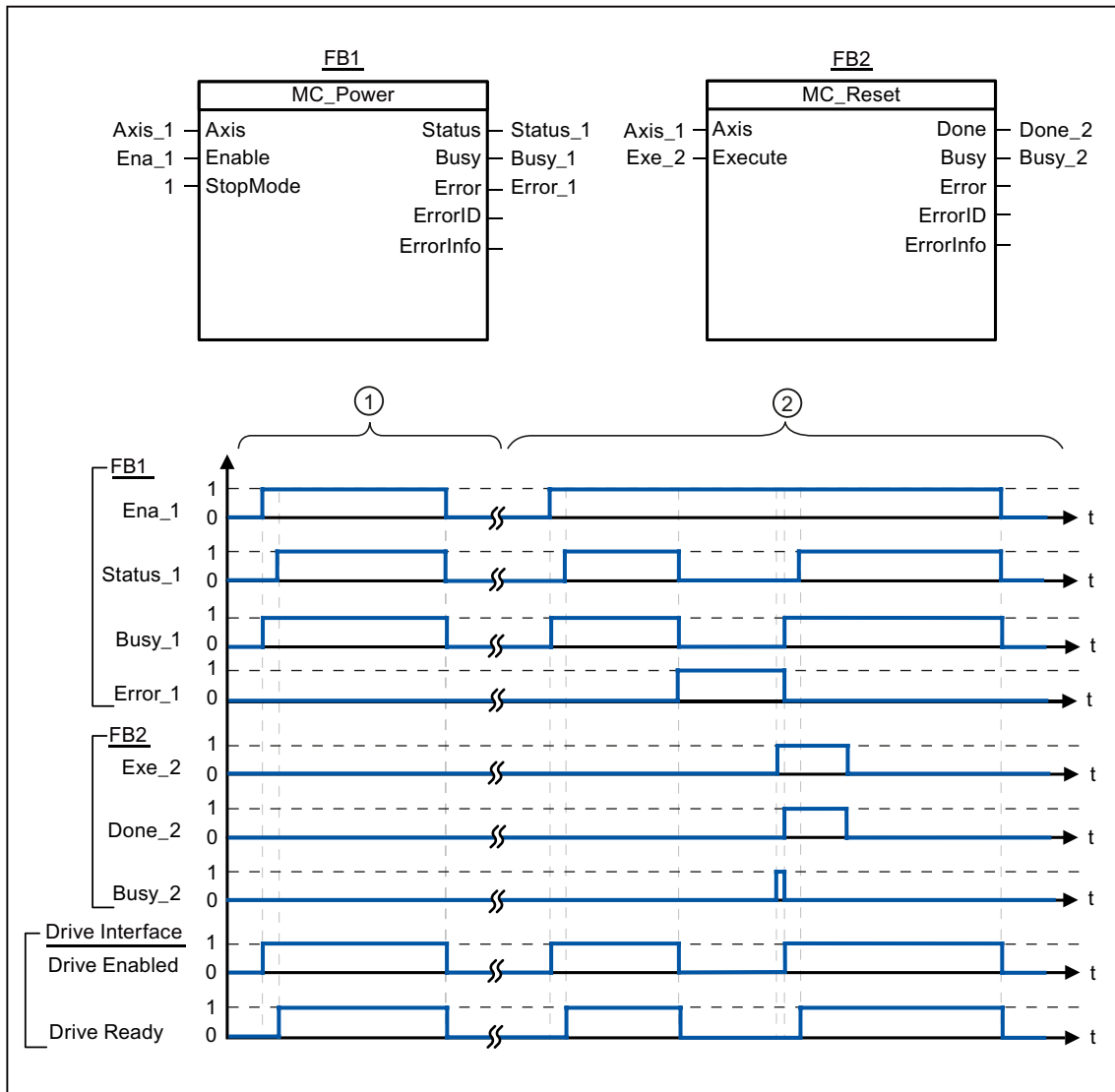
1. Bringen Sie die Achse in den Stillstand.  
Wann sich die Achse im Stillstand befindet, erkennen Sie an der Variable des Technologieobjekts <Achurname>.StatusBits.StandStill.
2. Setzen Sie den Eingangsparameters "Enable" nach dem Erreichen des Stillstandes auf FALSE.
3. Zeigen die Ausgangsparameter "Busy" und "Status", sowie die Variable des Technologieobjekts <Achurname>.StatusBits.Enable den Wert FALSE, so ist das Sperren der Achse abgeschlossen.

## Siehe auch

- Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)
- MC\_Power: Funktionsdiagramm (Seite 2640)
- MC\_Reset: Fehler quittieren (Seite 2641)
- MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)
- MC\_Halt: Achsen anhalten (Seite 2646)
- MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)
- MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)
- MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)
- MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)
- MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)
- MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)

### MC\_Power: Funktionsdiagramm

#### Funktionsdiagramm



- |   |  |
|---|--|
| ① | Eine Achse wird freigegeben und anschließend wieder gesperrt. Nachdem der Antrieb das Signal "Antrieb bereit" an die CPU zurück gemeldet hat, kann die erfolgreiche Freigabe über "Status_1" abgelesen werden. |
| ② | Nach dem Freigeben einer Achse tritt ein Fehler auf, der das Sperren der Achse zur Folge hat. Der Fehler wird beseitigt und mit "MC_Reset" quittiert. Daraufhin wird die Achse wieder freigegeben.             |

#### Siehe auch

MC\_Power: Achsen freigeben, sperren (Seite 2636)

## MC\_Reset

### MC\_Reset: Fehler quittieren

#### Beschreibung

Mit der Motion Control-Anweisung "MC\_Reset" können "Betriebsfehler mit Stopp der Achse" und "Konfigurationsfehler" quittiert werden. Welche Fehler quittiert werden müssen können Sie der "Liste der ErrorIDs und ErrorInfos" unter "Abhilfe" entnehmen.

Ab Version V3.0 kann nach einem Laden im Betriebszustand RUN die Konfiguration der Achse in den Arbeitsspeicher geladen werden.

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.
- Bei einem anstehenden quittierbaren Konfigurationsfehler wurde die Fehlerursache beseitigt (z. B. die Beschleunigung im Technologieobjekt "Achse" auf einen gültigen Wert geändert).

#### Ablöseverhalten

Der MC\_Reset-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Der neue MC\_Reset-Auftrag bricht keine anderen laufenden Motion Control-Aufträge ab.

#### Parameter

| Parameter | Deklaration | Datentyp  | Defaultwert | Beschreibung  |  |
|-----------|-------------|-----------|-------------|---|--|
| Axis      | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse   |  |
| Execute   | INPUT       | BOOL      | FALSE       | Start des Auftrags mit steigender Flanke  |  |
| Restart   | INPUT       | BOOL      | FALSE       | (Ab Version V3.0)   |  |
|           |             |           |             | TRUE  | Lädt die Konfiguration der Achse aus dem Ladespeicher in den Arbeitsspeicher. Der Auftrag ist nur bei gesperrter Achse ausführbar. Beachten Sie hierzu die Hinweise zum Laden in die CPU (Seite 4190). |
|           |             |           |             | FALSE   | Quittiert anstehende Fehler  |
| Done      | OUTPUT      | BOOL      | FALSE       | TRUE Fehler wurde quittiert.  |  |
| Busy      | OUTPUT      | BOOL      | FALSE       | TRUE Auftrag ist in Bearbeitung   |  |
| Error     | OUTPUT      | BOOL      | FALSE       | TRUE Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen. |  |
| ErrorID   | OUTPUT      | WORD      | 16#0000     | Fehlererkennung (Seite 4233) zum Parameter "Error"  |  |
| ErrorInfo | OUTPUT      | WORD      | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID"  |  |

### Einen mit MC\_Reset zu quittierenden Fehler quittieren

Gehen Sie zum Quittieren eines Fehlers folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Starten Sie das Quittieren des Fehlers durch eine steigende Flanke am Eingangsparameter "Execute".
3. Zeigt der Ausgangsparameter "Done" den Wert TRUE sowie die Variable des Technologieobjekts <Achse>.StatusBits.Error den Wert FALSE, so wurde der Fehler quittiert.

### Siehe auch

Laden in CPU (Seite 4190)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)

MC\_Power: Achsen freigeben, sperren (Seite 2636)

MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)

MC\_Halt: Achsen anhalten (Seite 2646)

MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)

MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)

MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)

MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)

MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)

MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)



## MC\_Home

### MC\_Home: Achsen referenzieren, Referenzpunkt setzen

#### Beschreibung

Mit Hilfe der Motion Control-Anweisung "MC\_Home" gleichen Sie die Achskoordinate auf die reale, physikalische Position des Antriebs ab. Soll die Achse absolut positioniert werden, so ist eine Referenzierung nötig. Folgende Referenzierarten können durchgeführt werden:

- Aktives Referenzieren (Mode = 3)  
Die Referenzpunktfahrt wird automatisch durchgeführt.
- Passives Referenzieren (Mode = 2)  
Beim passiven Referenzieren führt die Motion Control-Anweisung "MC\_Home" keine Referenzierbewegung durch. Die dafür notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen realisiert werden. Beim Erkennen des Referenzpunktschalters wird die Achse referenziert.
- Direktes Referenzieren Absolut (Mode = 0)  
Die aktuelle Achsposition wird auf den Wert des Parameters "Position" gesetzt.
- Direktes Referenzieren Relativ (Mode = 1)  
Die aktuelle Achsposition wird um den Wert des Parameters "Position" verschoben.

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.
- Die Achse ist freigegeben.
- Beim Start mit Mode = 0, 1 und 2 darf kein MC\_CommandTable-Auftrag aktiv sein.

#### Ablöseverhalten

Das Ablöseverhalten ist vom gewählten Mode abhängig:

##### **Mode = 0, 1**

Der MC\_Home-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Der MC\_Home-Auftrag bricht keine laufenden Motion Control-Aufträge ab. Positionsbezogene Verfahrtaufträge werden nach dem Referenzieren entsprechend der neuen Referenzierposition (Wert am Eingangsparameter: "Position") fortgesetzt.

##### **Mode = 2**

Der MC\_Home-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC\_Home-Auftrag Mode = 2, 3

Der neue MC\_Home-Auftrag bricht folgenden laufenden Motion Control-Auftrag ab:

- MC\_Home-Auftrag Mode = 2

Positionsbezogene Verfahrtaufträge werden nach dem Referenzieren entsprechend der neuen Referenzierposition (Wert am Eingangsparameter: "Position") fortgesetzt.

**Mode = 3**

Der MC\_Home-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

Der neue MC\_Home-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC\_Home-Auftrag Mode = 2, 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

**Parameter**

| Parameter | Deklaration | Datentyp  | Defaultwert | Beschreibung  |  |
|-----------|-------------|-----------|-------------|---|--|
| Axis      | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse   |  |
| Execute   | INPUT       | BOOL      | FALSE       | Start des Auftrags mit steigender Flanke  |  |
| Position  | INPUT       | REAL      | 0.0         | <ul style="list-style-type: none"> <li>• Mode = 0, 2 und 3<br/>Absolute Position der Achse nach dem Abschluss des Referenziervorgangs</li> <li>• Mode = 1<br/>Korrekturwert zur aktuellen Achsposition</li> </ul> Grenzwerte:<br>$-1.0e^{12} \leq \text{Position} \leq 1.0e^{12}$ |  |
| Mode      | INPUT       | INT       | 0           | Referenziermodus  |  |
|           |             |           |             | 0   | Direktes Referenzieren Absolut<br>Neue Achsposition ist der Positionswert des Parameters "Position".                         |
|           |             |           |             | 1   | Direktes Referenzieren Relativ<br>Neue Achsposition ist die aktuelle Achsposition + Positionswert des Parameters "Position". |

| Parameter      | Deklaration | Datentyp | Defaultwert | Beschreibung                              |  |
|----------------|-------------|----------|-------------|---|--|
|                |             |          |             | 2   | Passives Referenzieren<br>Referenzieren gemäß der Achskonfiguration. Als neue Achsposition nach dem Referenzieren wird der Wert des Parameters "Position" gesetzt. |
|                |             |          |             | 3   | Aktives Referenzieren<br>Referenzpunktfahrt gemäß Achskonfiguration. Als neue Achsposition nach dem Referenzieren wird der Wert des Parameters "Position" gesetzt. |
| Done           | OUTPUT      | BOOL     | FALSE       | TRUE                                      | Auftrag ist abgeschlossen  |
| Busy           | OUTPUT      | BOOL     | FALSE       | TRUE                                      | Auftrag ist in Bearbeitung   |
| CommandAborted | OUTPUT      | BOOL     | FALSE       | TRUE                                      | Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.   |
| Error          | OUTPUT      | BOOL     | FALSE       | TRUE                                      | Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.                 |
| ErrorID        | OUTPUT      | WORD     | 16#0000     | Fehlerkennung zum Parameter "Error"       |  |
| ErrorInfo      | OUTPUT      | WORD     | 16#0000     | Fehlerinfokennung zum Parameter "ErrorID" |  |

### Hinweis

Die Referenzierung einer Achse geht unter folgenden Bedingungen verloren:

- Sperren der Achse durch die Motion Control-Anweisung "MC\_Power"
- Wechsel zwischen Automatikbetrieb und Handsteuerung
- Mit dem Starten des aktiven Referenzierens. Nach dem erfolgreichen Abschluss des Referenzvorgangs, ist die Referenzierung der Achse wieder vorhanden.
- Nach NETZ-AUS -> NETZ-EIN der CPU
- Nach Neustart der CPU (RUN-STOP -> STOP-RUN)

### Eine Achse referenzieren

Gehen Sie beim Referenzieren der Achse folgendermaßen vor:

1. Prüfen Sie die oben genannten Voraussetzungen.
2. Versorgen Sie die notwendigen Eingangsparameter mit Werten und starten Sie das Referenzieren durch eine steigende Flanke am Eingangsparameter "Execute"
3. Zeigt der Ausgangsparameter "Done" sowie die Variable des Technologieobjekts <Achsname>.StatusBits.HomingDone den Wert TRUE, so ist das Referenzieren abgeschlossen.

## Siehe auch

- Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)
- MC\_Power: Achsen freigeben, sperren (Seite 2636)
- MC\_Reset: Fehler quittieren (Seite 2641)
- MC\_Halt: Achsen anhalten (Seite 2646)
- MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)
- MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)
- MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)
- MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)
- MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)
- MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

## MC\_Halt

### MC\_Halt: Achsen anhalten

#### Beschreibung

Die Motion Control-Anweisung "MC\_Halt" bricht alle Bewegungsvorgänge ab und bremst die Achse mit der konfigurierten Verzögerung bis zum Stillstand. Die Stillstandsposition ist nicht definiert.

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

#### Ablöseverhalten

Der MC\_Halt-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag

- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

Der neue MC\_Halt-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

## Parameter

| Parameter      | Deklaration | Datentyp  | Defaultwert | Beschreibung  |
|----------------|-------------|-----------|-------------|---|
| Axis           | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse   |
| Execute        | INPUT       | BOOL      | FALSE       | Start des Auftrags mit steigender Flanke  |
| Done           | OUTPUT      | BOOL      | FALSE       | TRUE   Geschwindigkeit Null erreicht  |
| Busy           | OUTPUT      | BOOL      | FALSE       | TRUE   Auftrag ist in Bearbeitung   |
| CommandAborted | OUTPUT      | BOOL      | FALSE       | TRUE   Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.   |
| Error          | OUTPUT      | BOOL      | FALSE       | TRUE   Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen. |
| ErrorID        | OUTPUT      | WORD      | 16#0000     | Fehlerkennung (Seite 4233) zum Parameter "Error"  |
| ErrorInfo      | OUTPUT      | WORD      | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID"  |

**Siehe auch**

MC\_Halt: Funktionsdiagramm (Seite 2649)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)

MC\_Power: Achsen freigeben, sperren (Seite 2636)

MC\_Reset: Fehler quittieren (Seite 2641)

MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)

MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)

MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)

MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)

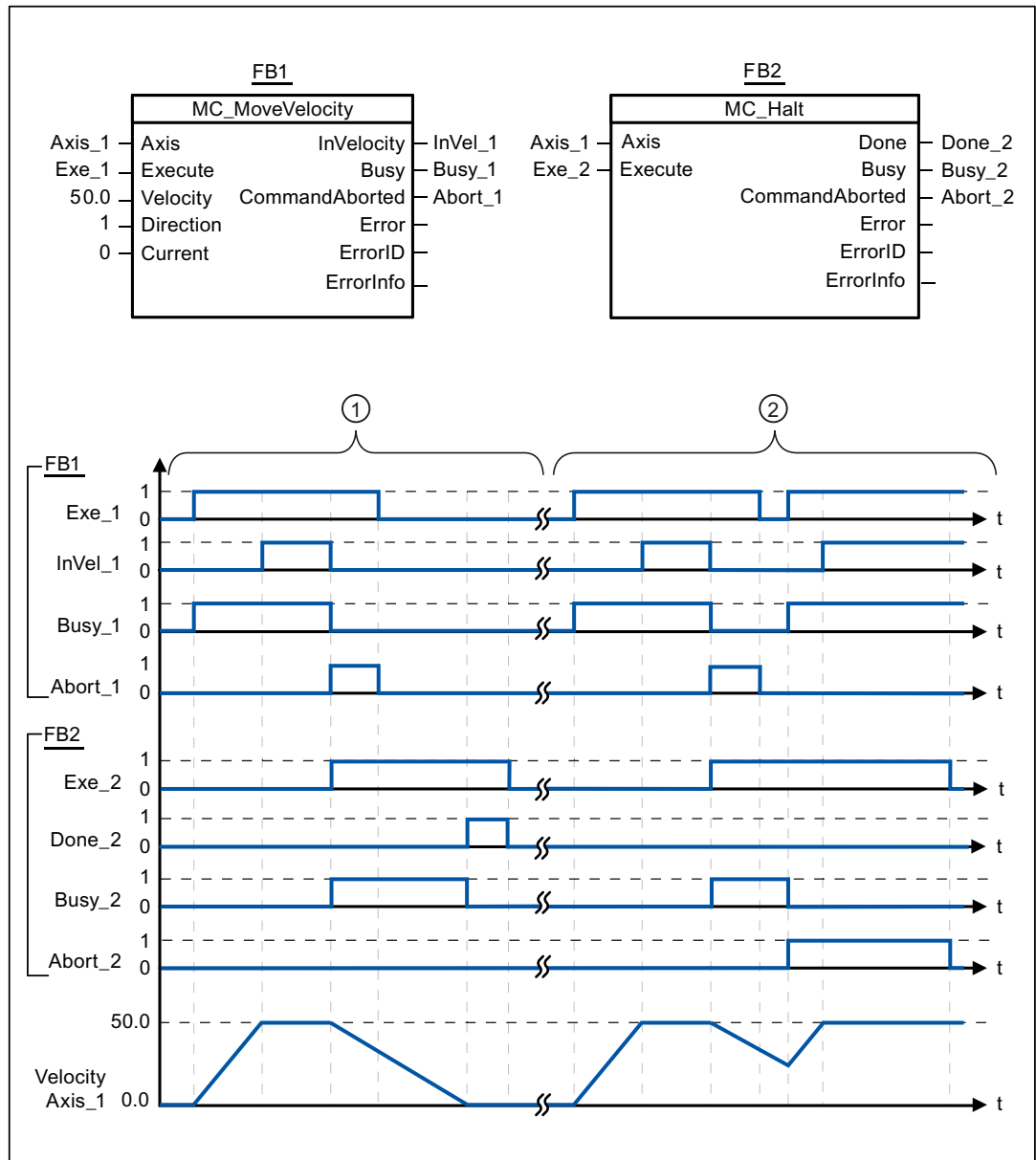
MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)

MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)

MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

## MC\_Halt: Funktionsdiagramm

### Funktionsdiagramm



Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 5.0

|   |   |
|---|---|
| ① | Die Achse wird durch einen MC_Halt-Auftrag bis zum Stillstand abgebremst. Das Erreichen des Stillstandes wird über "Done_2" gemeldet.                         |
| ② | Während ein MC_Halt-Auftrag die Achse abbremst wird dieser Auftrag durch einen anderen Verfahrtauftrag abgebrochen. Der Abbruch wird über "Abort_2" gemeldet. |

**Siehe auch**

MC\_Halt: Achsen anhalten (Seite 2646)

**MC\_MoveAbsolute**

**MC\_MoveAbsolute: Achsen absolut positionieren**

**Beschreibung**

Die Motion Control-Anweisung "MC\_MoveAbsolute" startet eine Positionierbewegung der Achse zu einer absoluten Position.

**Voraussetzungen**

- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.
- Die Achse ist freigegeben.
- Die Achse ist referenziert.

**Ablöseverhalten**

Der MC\_MoveAbsolute-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

Der neue MC\_MoveAbsolute-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag



## Parameter

| Parameter      | Deklaration | Datentyp  | Defaultwert | Beschreibung  |
|----------------|-------------|-----------|-------------|---|
| Axis           | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse   |
| Execute        | INPUT       | BOOL      | FALSE       | Start des Auftrags mit steigender Flanke  |
| Position       | INPUT       | REAL      | 0.0         | Absolute Zielposition<br>Grenzwerte:<br>$-1.0e^{12} \leq \text{Position} \leq 1.0e^{12}$  |
| Velocity       | INPUT       | REAL      | 10.0        | Geschwindigkeit der Achse<br>Diese Geschwindigkeit wird aufgrund der konfigurierten Beschleunigung und Verzögerung und der anzufahrenden Zielposition nicht immer erreicht.<br>Grenzwerte:<br>Start/Stopp-Geschwindigkeit $\leq$ Velocity $\leq$ Maximale Geschwindigkeit |
| Done           | OUTPUT      | BOOL      | FALSE       | TRUE Absolute Zielposition erreicht   |
| Busy           | OUTPUT      | BOOL      | FALSE       | TRUE Auftrag ist in Bearbeitung   |
| CommandAborted | OUTPUT      | BOOL      | FALSE       | TRUE Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.   |
| Error          | OUTPUT      | BOOL      | FALSE       | TRUE Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.   |
| ErrorID        | OUTPUT      | WORD      | 16#0000     | Fehlerkennung (Seite 4233) zum Parameter "Error"  |
| ErrorInfo      | OUTPUT      | WORD      | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID"  |

## Siehe auch

MC\_MoveAbsolute: Funktionsdiagramm (Seite 2652)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)

MC\_Power: Achsen freigeben, sperren (Seite 2636)

MC\_Reset: Fehler quittieren (Seite 2641)

MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)

MC\_Halt: Achsen anhalten (Seite 2646)

MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)

MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)

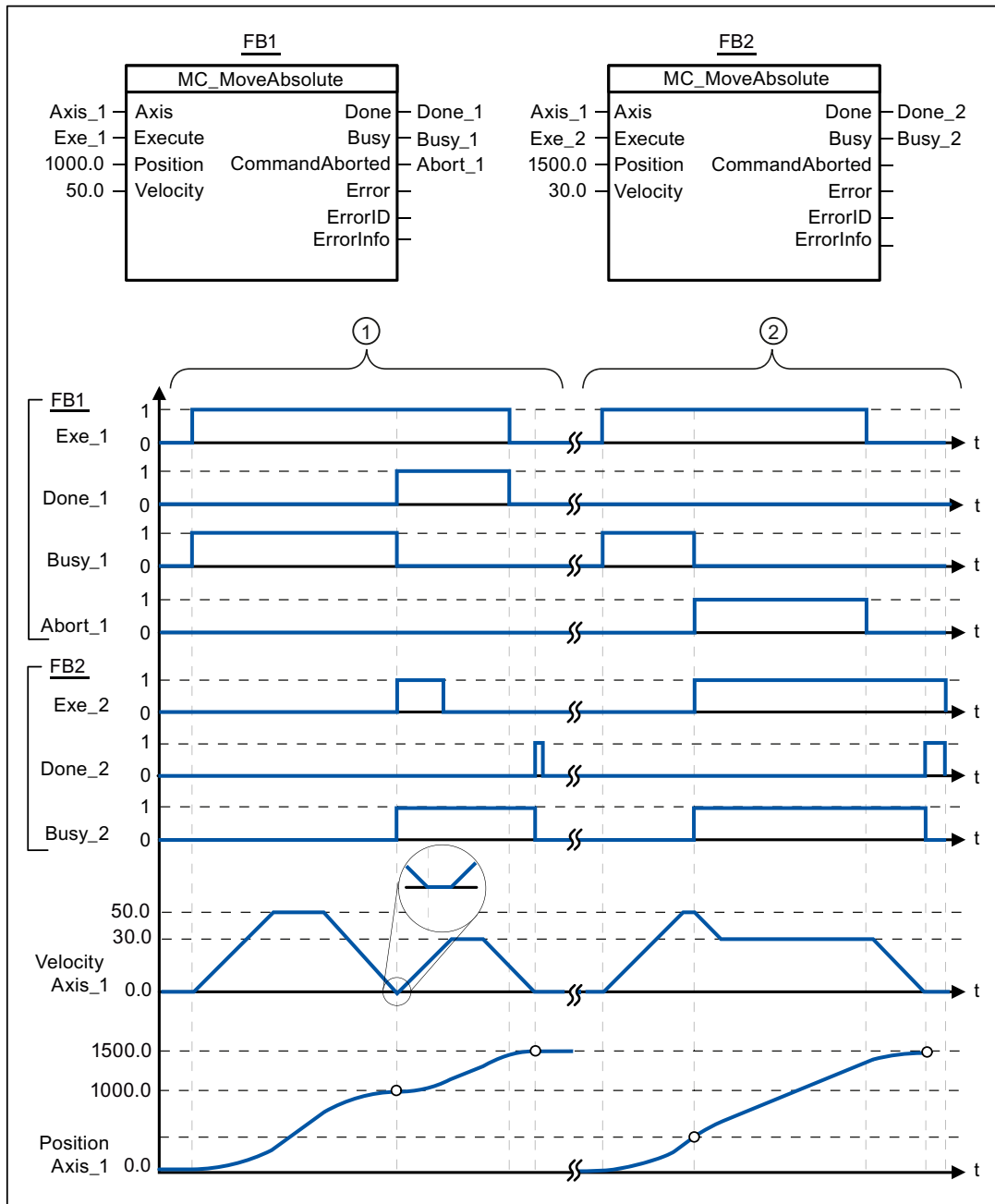
MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)

MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)

MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

### MC\_MoveAbsolute: Funktionsdiagramm

#### Funktionsdiagramm



Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

|   |  |
|---|--|
| ① | Eine Achse wird durch einen MC_MoveAbsolute-Auftrag auf die absolute Position 1000.0 verfahren. Das Erreichen der Zielposition wird über "Done_1" gemeldet. Mit "Done_1" = TRUE wird ein weiterer MC_MoveAbsolute-Auftrag, mit Zielposition 1500.0, gestartet. Aufgrund von Reaktionszeiten (z. B. Zykluszeit des Anwenderprogramms, ...) kommt es zu einem kurzen Stillstand der Achse (siehe Lupendarstellung). Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet. |
| ② | Ein laufender MC_MoveAbsolute-Auftrag wird durch einen weiteren MC_MoveAbsolute-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit auf die neue Zielposition 1500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.   |

## Siehe auch

MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)

## MC\_MoveRelative

### MC\_MoveRelative: Achsen relativ positionieren

#### Beschreibung

Die Motion Control-Anweisung "MC\_MoveRelative" startet eine Positionierbewegung relativ zur Startposition.

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

#### Ablöseverhalten

Der MC\_MoveRelative-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

Der neue MC\_MoveRelative-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag

- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

**Parameter**

| Parameter      | Deklaration | Datentyp  | Defaultwert | Beschreibung   |
|----------------|-------------|-----------|-------------|--|
| Axis           | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse  |
| Execute        | INPUT       | BOOL      | FALSE       | Start des Auftrags mit steigender Flanke   |
| Distance       | INPUT       | REAL      | 0.0         | Wegstrecke für den Positioniervorgang<br>Grenzwerte:<br>$-1.0e^{12} \leq \text{Distance} \leq 1.0e^{12}$   |
| Velocity       | INPUT       | REAL      | 10.0        | Geschwindigkeit der Achse<br>Diese Geschwindigkeit wird aufgrund der konfigurierten Beschleunigung und Verzögerung und der zu fahrenden Wegstrecke nicht immer erreicht.<br>Grenzwerte:<br>$\text{Start/Stop-Geschwindigkeit} \leq \text{Velocity} \leq \text{Maximale Geschwindigkeit}$ |
| Done           | OUTPUT      | BOOL      | FALSE       | TRUE   Zielposition erreicht   |
| Busy           | OUTPUT      | BOOL      | FALSE       | TRUE   Auftrag ist in Bearbeitung  |
| CommandAborted | OUTPUT      | BOOL      | FALSE       | TRUE   Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.  |
| Error          | OUTPUT      | BOOL      | FALSE       | TRUE   Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.  |
| ErrorID        | OUTPUT      | WORD      | 16#0000     | Fehlerkennung (Seite 4233) zum Parameter "Error"   |
| ErrorInfo      | OUTPUT      | WORD      | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID"   |

## Siehe auch

MC\_MoveRelative: Funktionsdiagramm (Seite 2656)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)

MC\_Power: Achsen freigeben, sperren (Seite 2636)

MC\_Reset: Fehler quittieren (Seite 2641)

MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)

MC\_Halt: Achsen anhalten (Seite 2646)

MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)

MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)

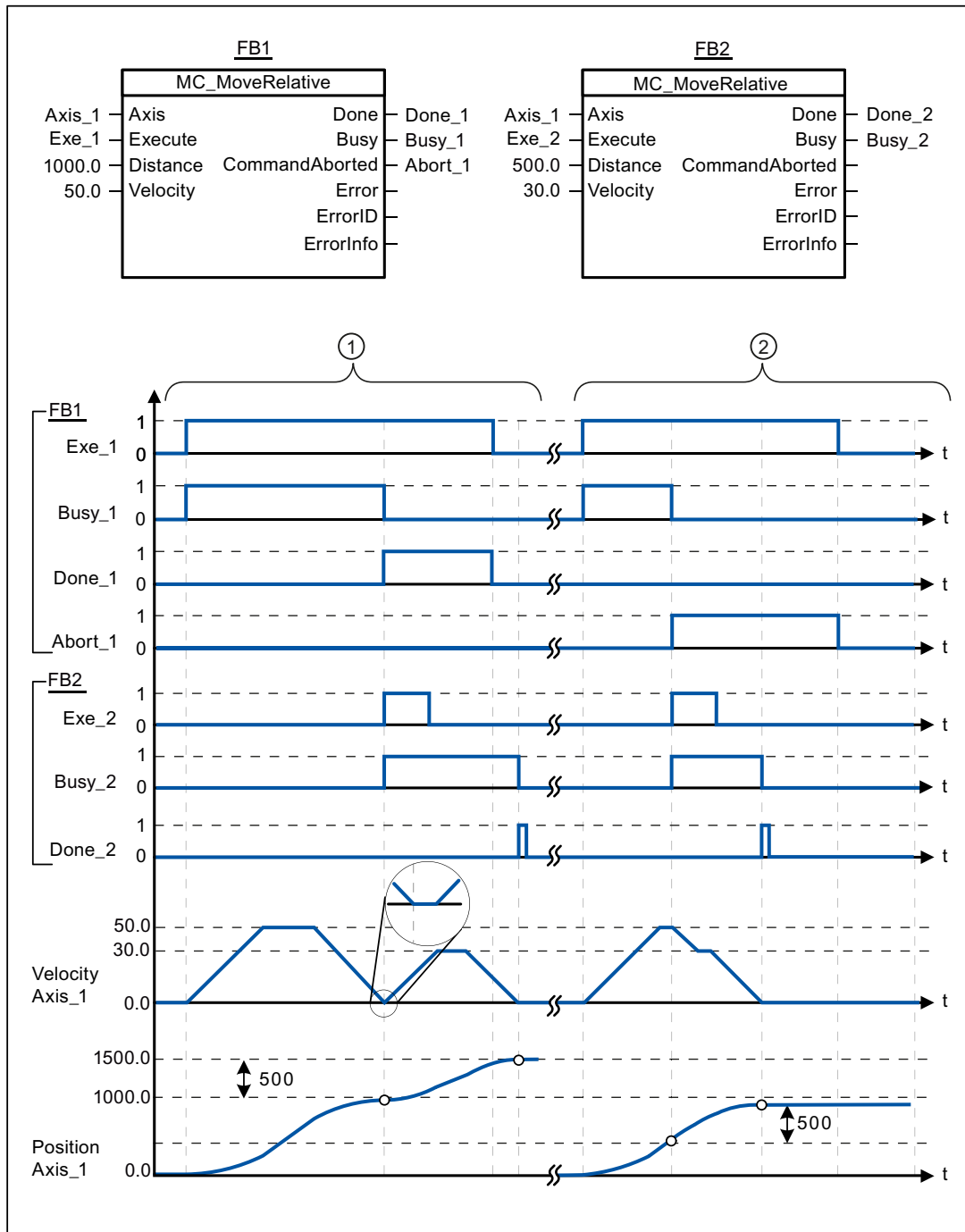
MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)

MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)

MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

### MC\_MoveRelative: Funktionsdiagramm

#### Funktionsdiagramm



Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

|   |   |
|---|---|
| ① | Die Achse wird durch einen MC_MoveRelative-Auftrag um die Wegstrecke ("Distance") 1000.0 verfahren. Das Erreichen der Zielposition wird über "Done_1" gemeldet. Mit "Done_1" = TRUE wird ein weiterer MC_MoveRelative-Auftrag, mit der Wegstrecke 500.0, gestartet. Aufgrund von Reaktionszeiten (z. B. Zykluszeit des Anwenderprogramms, ...) kommt es zu einem kurzen Stillstand der Achse (siehe Lupendarstellung). Das erfolgreiche Erreichen der neuen Zielposition wird über "Done_2" gemeldet. |
| ② | Ein laufender MC_MoveRelative-Auftrag wird durch einen weiteren MC_MoveRelative-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit um die neue Wegstrecke ("Distance") 500.0 verfahren. Das Erreichen der neuen Zielposition wird über "Done_2" gemeldet.   |

## Siehe auch

MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)

## MC\_MoveVelocity

### MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren

#### Beschreibung

Die Motion Control-Anweisung "MC\_MoveVelocity" verfährt die Achse konstant mit der vorgegebenen Geschwindigkeit

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

#### Ablöseverhalten

Der MC\_MoveVelocity kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

Der neue MC\_MoveVelocity-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag

- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

**Parameter**

| Parameter  | Deklaration | Datentyp  | Defaultwert | Beschreibung   |  |
|------------|-------------|-----------|-------------|--|--|
| Axis       | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse  |  |
| Execute    | INPUT       | BOOL      | FALSE       | Start des Auftrags mit steigender Flanke   |  |
| Velocity   | INPUT       | REAL      | 10.0        | Geschwindigkeitsvorgabe für das Verfahren der Achse<br>Grenzwerte:<br>Start/Stopp-Geschwindigkeit ≤  Velocity  ≤ Maximale Geschwindigkeit<br>(Velocity = 0.0 ist erlaubt)  |  |
| Direction  | INPUT       | INT       | 0           | Richtungsvorgabe   |  |
|            |             |           |             | 0  | Drehrichtung entsprechend dem Vorzeichen des Werts am Parameter "Velocity"   |
|            |             |           |             | 1  | Drehrichtung positiv<br>(Das Vorzeichen des Werts am Parameter "Velocity" wird ignoriert)  |
|            |             |           |             | 2  | Drehrichtung negativ<br>(Das Vorzeichen des Werts am Parameter "Velocity" wird ignoriert)  |
| Current    | INPUT       | BOOL      | FALSE       | Aktuelle Geschwindigkeit beibehalten   |  |
|            |             |           |             | FALSE  | "Aktuelle Geschwindigkeit beibehalten" deaktiviert. Es werden die Werte der Parameter "Velocity" und "Direction" verwendet.  |
|            |             |           |             | TRUE   | "Aktuelle Geschwindigkeit beibehalten" aktiviert. Die Werte an den Parametern "Velocity" und "Direction" bleiben unberücksichtigt.<br>Sobald die Achse mit der aktuellen Geschwindigkeit weiterfährt liefert der Parameter "InVelocity" den Wert TRUE. |
| InVelocity | OUTPUT      | BOOL      | FALSE       | TRUE <ul style="list-style-type: none"> <li>• "Current" = FALSE:<br/>Die am Parameter "Velocity" angegebene Geschwindigkeit wurde erreicht.</li> <li>• "Current" = TRUE:<br/>Die Achse verfährt mit der zum Startzeitpunkt aktuellen Geschwindigkeit.</li> </ul> |  |
| Busy       | OUTPUT      | BOOL      | FALSE       | TRUE Auftrag ist in Bearbeitung  |  |



| Parameter      | Deklaration | Datentyp | Defaultwert | Beschreibung   |
|----------------|-------------|----------|-------------|--|
| CommandAborted | OUTPUT      | BOOL     | FALSE       | TRUE<br>Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.   |
| Error          | OUTPUT      | BOOL     | FALSE       | TRUE<br>Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen. |
| ErrorID        | OUTPUT      | WORD     | 16#0000     | Fehlerkennung (Seite 4233) zum Parameter "Error"   |
| ErrorInfo      | OUTPUT      | WORD     | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID"   |

### Verhalten bei Sollgeschwindigkeit Null (Velocity = 0.0)

Ein MC\_MoveVelocity-Auftrag mit "Velocity" = 0.0 bricht (wie ein MC\_Halt-Auftrag) aktive Bewegungsaufträge ab und stoppt die Achse mit der konfigurierten Verzögerung.

Mit dem Erreichen des Stillstands wird am Ausgangsparameter "InVelocity" für mindestens einen Programmzyklus TRUE angezeigt.

"Busy" zeigt während des Verzögerungsvorgangs den Wert TRUE und wechselt diesen zusammen mit "InVelocity" zu FALSE. Ist der Parameter "Execute" = TRUE gesetzt, so wird "InVelocity" und "Busy" speichernd angezeigt.

Mit dem Start des MC\_MoveVelocity-Auftrags wird das Statusbit "SpeedCommand" im Technologieobjekt gesetzt. Das Statusbit "ConstantVelocity" mit dem Stillstand der Achse. Beide Bits werden mit dem Start eines neuen Verfahrungsantrags an die neue Situation angepasst.

### Siehe auch

MC\_MoveVelocity: Funktionsdiagramm (Seite 2660)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)

MC\_Power: Achsen freigeben, sperren (Seite 2636)

MC\_Reset: Fehler quittieren (Seite 2641)

MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)

MC\_Halt: Achsen anhalten (Seite 2646)

MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)

MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)

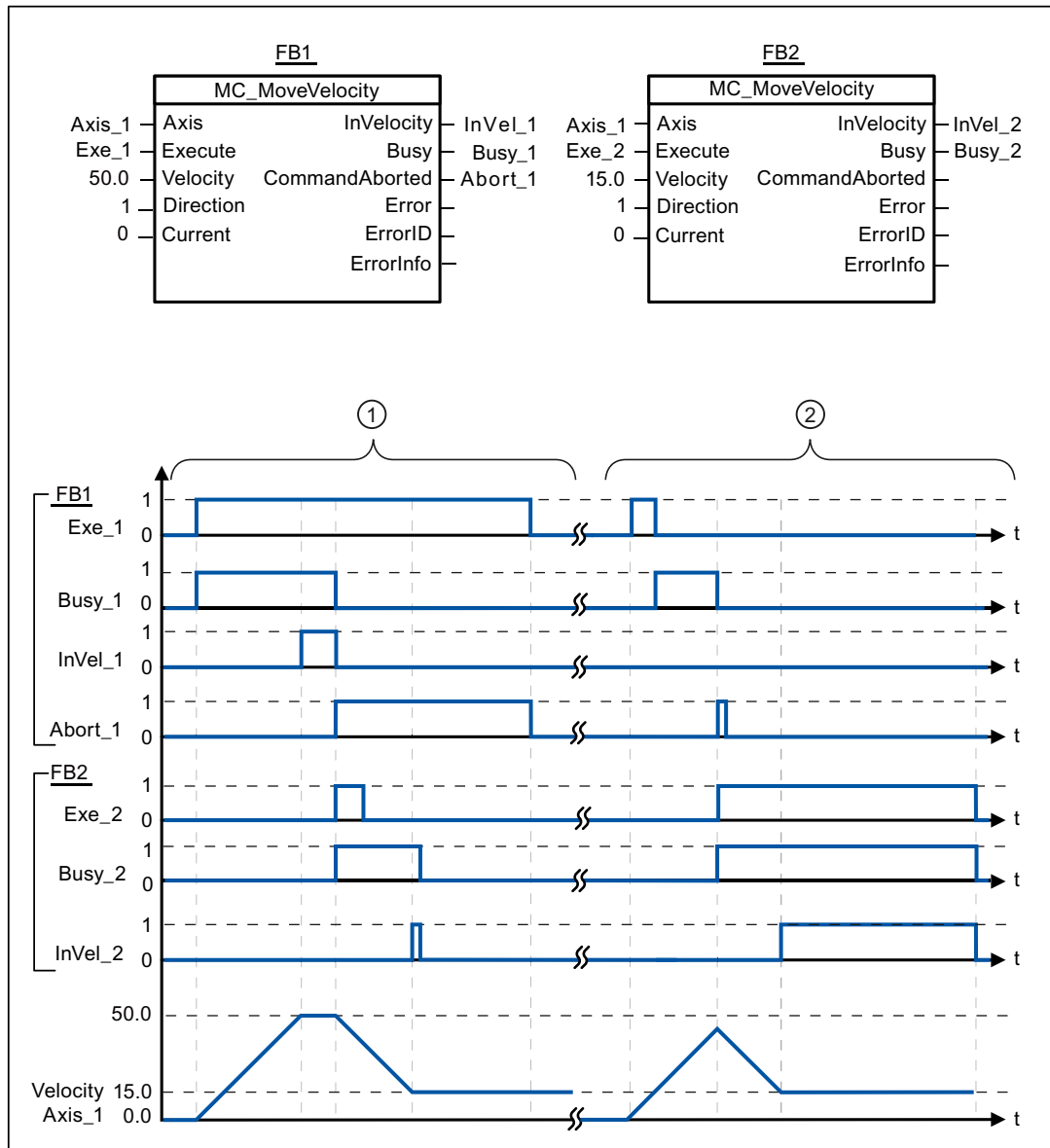
MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)

MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)

MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

### MC\_MoveVelocity: Funktionsdiagramm

#### Funktionsdiagramm



Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 10.0

|   |  |
|---|--|
| ① | Ein laufender MC_MoveVelocity-Auftrag meldet über "InVel_1" das Erreichen seiner Zielgeschwindigkeit. Anschließend wird er durch einen weiteren MC_MoveVelocity-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Zielgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit konstant weiter verfahren. |
| ② | Ein laufender MC_MoveVelocity-Auftrag wird vor dem Erreichen seiner Zielgeschwindigkeit durch einen weiteren MC_MoveVelocity-Auftrag abgebrochen. Der Abbruch wird über "Abort_1" gemeldet. Das Erreichen der neuen Zielgeschwindigkeit 15.0 wird über "InVel_2" gemeldet. Die Achse wird anschließend mit der neuen Geschwindigkeit konstant weiter verfahren.                                    |

## Siehe auch

MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)

## MC\_MoveJog

### MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen

#### Beschreibung

Die Motion Control-Anweisung "MC\_MoveJog" verfährt die Achse konstant mit der vorgegebenen Geschwindigkeit im Tippbetrieb. Verwenden Sie diese Motion Control Anweisung z. B. für Test- und Inbetriebnahmezwecke.

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.
- Die Achse ist freigegeben.

#### Ablöseverhalten

Der MC\_MoveJog-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

Der neue MC\_MoveJog-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag

- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag

**Parameter**

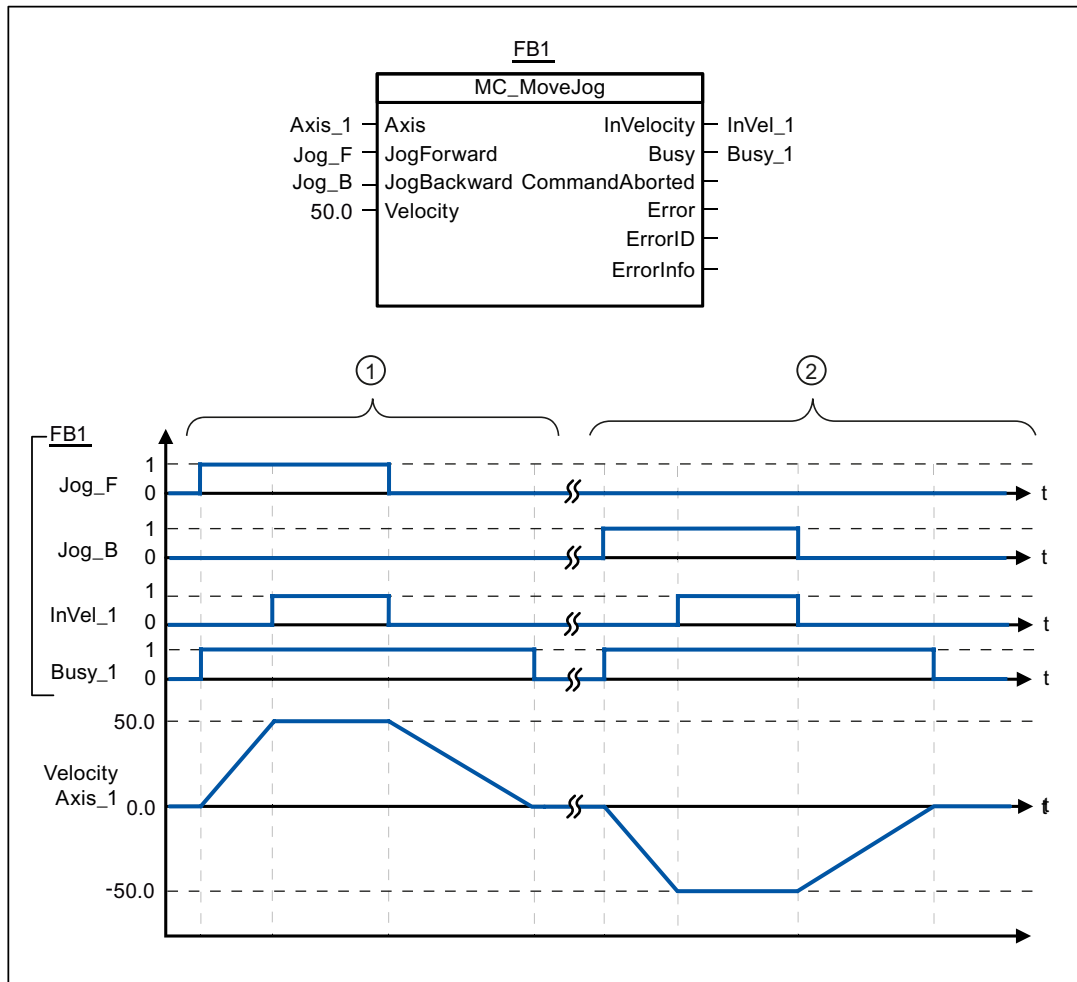
| Parameter   | Deklaration | Datentyp  | Defaultwert | Beschreibung  |
|---|-------------|-----------|-------------|---|
| Axis  | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse   |
| JogForward  | INPUT       | BOOL      | FALSE       | Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in positive Richtung.                   |
| JogBackward   | INPUT       | BOOL      | FALSE       | Solange der Parameter TRUE ist, verfährt die Achse mit der am Parameter "Velocity" vorgegebenen Geschwindigkeit in negative Richtung.                   |
| Sind beide Parameter gleichzeitig TRUE, so stoppt die Achse mit der konfigurierten Verzögerung. An den Parametern "Error", "ErrorID" und "ErrorInfo" wird ein Fehler angezeigt. |             |           |             |   |
| Velocity  | INPUT       | REAL      | 10.0        | Geschwindigkeitsvorgabe für den Tipbetrieb.   |
|   |             |           |             | Grenzwerte Anweisungsversion V1.0:<br>Start/Stop-Geschwindigkeit ≤  Velocity  ≤ Maximale Geschwindigkeit  |
|   |             |           |             | Grenzwerte Anweisungsversion V2.0:<br>Start/Stop-Geschwindigkeit ≤ Velocity ≤ Maximale Geschwindigkeit  |
| InVelocity  | OUTPUT      | BOOL      | FALSE       | TRUE Die am Parameter "Velocity" angegebene Geschwindigkeit wurde erreicht.   |
| Busy  | OUTPUT      | BOOL      | FALSE       | TRUE Auftrag ist in Bearbeitung   |
| CommandAborted  | OUTPUT      | BOOL      | FALSE       | TRUE Der Auftrag wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.   |
| Error   | OUTPUT      | BOOL      | FALSE       | TRUE Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen. |
| ErrorID   | OUTPUT      | WORD      | 16#0000     | Fehlerkennung (Seite 4233) zum Parameter "Error"  |
| ErrorInfo   | OUTPUT      | WORD      | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID"  |

## Siehe auch

- MC\_MoveJog: Funktionsdiagramm (Seite 2664)
- Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)
- MC\_Power: Achsen freigeben, sperren (Seite 2636)
- MC\_Reset: Fehler quittieren (Seite 2641)
- MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)
- MC\_Halt: Achsen anhalten (Seite 2646)
- MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)
- MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)
- MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)
- MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)
- MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

### MC\_MoveJog: Funktionsdiagramm

#### Funktionsdiagramm



Folgende Werte wurden im Konfigurationsfenster **Dynamik > Allgemein** konfiguriert:

- Beschleunigung: 10.0
- Verzögerung: 5.0

|   |  |
|---|--|
| ① | Über "Jog_F" wird die Achse im Tippbetrieb in positiver Richtung verfahren. Das Erreichen der Zielgeschwindigkeit 50.0 wird über "InVelo_1" gemeldet. Nach dem Rücksetzen von "Jog_F" bremst die Achse wieder bis zum Stillstand ab. |
| ② | Über "Jog_B" wird die Achse im Tippbetrieb in negativer Richtung verfahren. Das Erreichen der Zielgeschwindigkeit 50.0 wird über "InVelo_1" gemeldet. Nach dem Rücksetzen von "Jog_B" bremst die Achse wieder bis zum Stillstand ab. |

#### Siehe auch

MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)

## MC\_CommandTable

### MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0)

#### Beschreibung

Die Motion Control-Anweisung "MC\_CommandTable" fasst mehrere Einzelaufträge zur Steuerung einer Achse zu einer Bewegungssequenz zusammenfassen.

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde in der Version V2.0 eingefügt und korrekt konfiguriert.
- Das Technologieobjekt "Auftragstabelle" wurde eingefügt und korrekt konfiguriert.
- Die Achse ist freigegeben

#### Ablöseverhalten

Der MC\_CommandTable-Auftrag kann durch folgende Motion Control-Aufträge abgebrochen werden:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag
- MC\_CommandTable-Auftrag

Der neue MC\_CommandTable-Auftrag bricht folgende laufende Motion Control-Aufträge ab:

- MC\_Home-Auftrag Mode = 3
- MC\_Halt-Auftrag
- MC\_MoveAbsolute-Auftrag
- MC\_MoveRelative-Auftrag
- MC\_MoveVelocity-Auftrag
- MC\_MoveJog-Auftrag
- MC\_CommandTable-Auftrag

Der laufende Motion Control-Auftrag wird mit dem Start des ersten "Positioning Relative", "Positioning Absolute", "Velocity set point" oder "Halt" Auftrag abgebrochen.

**Parameter**

| Parameter      | Deklaration | Datentyp          | Defaultwert | Beschreibung   |
|----------------|-------------|-------------------|-------------|--|
| Axis           | INPUT       | TO_Axis_1         | -           | Technologieobjekt der Achse  |
| CommandTable   | INPUT       | TO_CommandTable_1 | -           | Technologieobjekt der Auftrags-tabelle   |
| Execute        | INPUT       | BOOL              | FALSE       | Start der Auftrags-tabelle mit steigender Flanke   |
| StartStep      | INPUT       | INT               | 1           | Vorgabe ab welchem Schritt die Auftrags-tabelle abgearbeitet werden soll<br>Grenzwerte:<br>$1 \leq \text{StartStep} \leq \text{EndStep}$                       |
| EndStep        | INPUT       | INT               | 32          | Vorgabe bis zu welchem Schritt die Auftrags-tabelle abgearbeitet werden soll<br>Grenzwerte:<br>$\text{StartStep} \leq \text{EndStep} \leq 32$                  |
| Done           | OUTPUT      | BOOL              | FALSE       | TRUE Auftragstabelle wurde erfolgreich abgearbeitet  |
| Busy           | OUTPUT      | BOOL              | FALSE       | TRUE Auftragstabelle ist in Bearbeitung  |
| CommandAborted | OUTPUT      | BOOL              | FALSE       | TRUE Die Auftrags-tabelle wurde während der Bearbeitung durch einen anderen Auftrag abgebrochen.   |
| Error          | OUTPUT      | BOOL              | FALSE       | TRUE Während der Bearbeitung der Auftrags-tabelle ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen. |
| ErrorID        | OUTPUT      | WORD              | 16#0000     | Fehlerkennung (Seite 4233) zum Parameter "Error"   |
| ErrorInfo      | OUTPUT      | WORD              | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID"   |
| CurrentStep    | OUTPUT      | INT               | 0           | Aktuell in Bearbeitung befindlicher Schritt der Auftrags-tabelle   |
| StepCode       | OUTPUT      | WORD              | 16#0000     | Benutzerdefinierte Zahlenwert / Bitmuster des aktuell in Bearbeitung befindlichen Schritts   |



## Siehe auch

- Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)
- Übersicht über die Motion Control-Anweisungen (Seite 4194)
- MC\_Power: Achsen freigeben, sperren (Seite 2636)
- MC\_Reset: Fehler quittieren (Seite 2641)
- MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)
- MC\_Halt: Achsen anhalten (Seite 2646)
- MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)
- MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)
- MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)
- MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)
- MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

## MC\_ChangeDynamic

### MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0)

#### Beschreibung

Die Motion Control-Anweisung "MC\_ChangeDynamic" ermöglicht die Änderung folgender Einstellungen der Achse:

- Wert für Hochlaufzeit (Beschleunigung) ändern
- Wert für Rücklaufzeit (Verzögerung) ändern
- Wert für Notstopp-Rücklaufzeit (Notstopp-Verzögerung) ändern
- Wert für Verrundungszeit (Ruck) ändern

Die Wirksamkeit der Änderung entnehmen Sie der Beschreibung der Variablen (Seite 4242).

#### Voraussetzungen

- Das Technologieobjekt "Achse" wurde in der Version V2.0 eingefügt.
- Das Technologieobjekt "Achse" wurde korrekt konfiguriert.

#### Ablöseverhalten

Ein MC\_ChangeDynamic-Auftrag kann durch keinen anderen Motion Control-Auftrag abgebrochen werden.

Ein neuer MC\_ChangeDynamic-Auftrag bricht keine laufenden Motion Control-Aufträge ab.

**Parameter**

| Parameter         | Deklaration | Datentyp  | Defaultwert | Beschreibung   |
|-------------------|-------------|-----------|-------------|--|
| Axis              | INPUT       | TO_Axis_1 | -           | Technologieobjekt der Achse  |
| Execute           | INPUT       | BOOL      | FALSE       | Start des Auftrags mit steigender Flanke   |
| ChangeRampUp      | INPUT       | BOOL      | FALSE       | TRUE   Hochlaufzeit entsprechend Eingangsparameter "RampUpTime" ändern   |
| RampUpTime        | INPUT       | REAL      | 5.00        | Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung vom Stillstand auf die konfigurierte maximale Geschwindigkeit zu beschleunigen<br>Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Acceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.                            |
| ChangeRampDown    | INPUT       | BOOL      | FALSE       | TRUE   Rücklaufzeit entsprechend Eingangsparameter "RampDownTime" geändern   |
| RampDownTime      | INPUT       | REAL      | 5.00        | Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung von der konfigurierten maximalen Geschwindigkeit auf Stillstand zu verzögern<br>Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Deceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.                              |
| ChangeEmergency   | INPUT       | BOOL      | FALSE       | TRUE   Notstopp-Rücklaufzeit entsprechend Eingangsparameter "EmergencyRampTime" ändern   |
| EmergencyRampTime | INPUT       | REAL      | 2.00        | Zeit (in Sekunden) um die Achse ohne Ruckbegrenzung im Notstopp-Betrieb von der konfigurierten maximalen Geschwindigkeit auf Stillstand zu verzögern<br>Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.EmergencyDeceleration beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen. |
| ChangeJerkTime    | INPUT       | BOOL      | FALSE       | TRUE   Verrundungszeit entsprechend Eingangsparameter "JerkTime" ändern  |
| JerkTime          | INPUT       | REAL      | 0.25        | Verrundungszeit (in Sekunden), die auf die Beschleunigungs- und Verzögerungsrampe der Achse angewendet wird<br>Durch die Änderung wird der Wert der Variable <Achsname>.Config.DynamicDefaults.Jerk beeinflusst. Die Wirksamkeit der Änderung können Sie der Beschreibung dieser Variable entnehmen.   |
| Done              | OUTPUT      | BOOL      | FALSE       | TRUE   Die geänderten Werte wurden in den Technologie-DB geschrieben. Wann die Änderung wirksam wird, können Sie der Beschreibung der Variablen entnehmen.   |
| Error             | OUTPUT      | BOOL      | FALSE       | TRUE   Während der Bearbeitung des Auftrages ist ein Fehler aufgetreten. Die Fehlerursache können Sie den Parametern "ErrorID" und "ErrorInfo" entnehmen.  |

| Parameter | Deklaration | Datentyp | Defaultwert | Beschreibung   |
|-----------|-------------|----------|-------------|--|
| ErrorID   | OUTPUT      | WORD     | 16#0000     | Fehlerkennung (Seite 4233) zum Parameter "Error"       |
| ErrorInfo | OUTPUT      | WORD     | 16#0000     | Fehlerinfokennung (Seite 4233) zum Parameter "ErrorID" |

---

### Hinweis

An den Eingangsparametern "RampUpTime", "RampDownTime", "EmergencyRampTime" und "JerkTime" können Werte angegeben werden, welche die zulässigen Grenzwerte der resultierenden Parameter: "Beschleunigung", "Verzögerung", "Notstopp-Verzögerung" und "Ruck" überschreiten.

Bitte beachten Sie unter Berücksichtigung der Gleichungen und Grenzwerte in Kapitel: "Technologieobjekt Achse" -> "Konfigurieren des Technologieobjekts" -> "Dynamik" dass Ihre Eingaben innerhalb des gültigen Bereichs liegen.

---

### Siehe auch

- Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)
- Übersicht über die Motion Control-Anweisungen (Seite 4194)
- Konfiguration der Dynamikwerte im Anwenderprogramm ändern (Seite 4162)
- Konfiguration zum Referenzieren im Anwenderprogramm ändern (Seite 4169)
- MC\_Power: Achsen freigeben, sperren (Seite 2636)
- MC\_Reset: Fehler quittieren (Seite 2641)
- MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)
- MC\_Halt: Achsen anhalten (Seite 2646)
- MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)
- MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)
- MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)
- MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)
- MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)
- Variable des Technologieobjekts Achse (Seite 4242)

### 9.7.4.2 Schnelle Zähler

#### CTRL\_HSC: Schnelle Zähler steuern

##### Parameter

| Parameter  | Datentyp | Speicherbereich              | Beschreibung  |
|------------|----------|------------------------------|---|
| EN         | BOOL     | E, A, M, D, L                | Freigabeeingang   |
| ENO        | BOOL     | E, A, M, D, L                | Freigabeausgang   |
| HSC        | HW_HSC   | E, A, M oder Konstante       | Hardware-Kennung des Schnellen Zählers (HW-ID)                        |
| DIR        | BOOL     | E, A, M, D, L oder Konstante | Freigabe der neuen Zählrichtung (siehe NEW_DIR)                       |
| CV         | BOOL     | E, A, M, D, L oder Konstante | Freigabe des neuen Zählerwerts (siehe NEW_CV)                         |
| RV         | BOOL     | E, A, M, D, L oder Konstante | Freigabe des neuen Referenzwerts (siehe NEW_RV)                       |
| PERIOD     | BOOL     | E, A, M, D, L oder Konstante | Freigabe der neuen Zeitdauer einer Frequenzmessung (siehe NEW_PERIOD) |
| NEW_DIR    | INT      | E, A, M, D, L oder Konstante | Zählrichtung, die bei DIR = TRUE geladen wird.                        |
| NEW_CV     | DINT     | E, A, M, D, L oder Konstante | Zählerwert, der bei CV = TRUE geladen wird.                           |
| NEW_RV     | DINT     | E, A, M, D, L oder Konstante | Referenzwert, der bei RV = TRUE geladen wird.                         |
| NEW_PERIOD | INT      | E, A, M, D, L oder Konstante | Zeitdauer der Frequenzmessung, die bei PERIOD = TRUE geladen wird.    |
| BUSY       | BOOL     | E, A, M, D, L                | Bearbeitungsstatus  |
| STATUS     | WORD     | E, A, M, D, L                | Status der Operation  |

##### Beschreibung

Mit der Anweisung "Schnelle Zähler steuern" können Sie die von der CPU unterstützten Schnellen Zähler über die Software parametrieren und steuern, indem Sie neue Werte in die Zähler laden. Eine Voraussetzung für die Ausführung der Anweisung ist, dass ein zu steuernder Schneller Zähler aktiviert ist. Pro Schnellen Zähler können Sie im Programm nicht mehrere Anweisungen "Schnelle Zähler steuern" gleichzeitig ausführen.

Die folgenden Parameterwerte können Sie mithilfe der Anweisung "Schnelle Zähler steuern" in einen Schnellen Zähler laden:

- **Zählrichtung (NEW\_DIR):** Die Zählrichtung definiert, ob ein Schneller Zähler vorwärts oder rückwärts zählt. Die Zählrichtung wird durch die folgenden Werte am Eingang NEW\_DIR definiert: 1 = Vorwärts, -1=Rückwärts.  
Ein Wechsel der Zählrichtung über die Anweisung "Schnelle Zähler steuern" ist nur dann möglich, wenn für den angegebenen Zähler eine Richtungssteuerung über das Programm parametrier ist. Die am Eingang NEW\_DIR angegebene Zählrichtung wird in einen Schnellen Zähler geladen, wenn das Bit am Eingang DIR gesetzt ist.
- **Zählerwert (NEW\_CV):** Der Zählwert ist der Anfangswert, mit dem ein Schneller Zähler neu zum Zählen beginnt. Der Zählwert kann in einem Bereich von -2147483648 bis 2147483647 liegen.  
Der am Eingang NEW\_CV angegebene Zählwert wird in einen Schnellen Zähler geladen, wenn das Bit am Eingang CV gesetzt ist.
- **Referenzwert (NEW\_RV):** Der Referenzwert kann mit dem aktuellen Zählerwert verglichen werden, um einen Alarm auszulösen. Der Referenzwert kann wie der Zählerwert in einem Bereich von -2147483648 bis 2147483647 liegen.  
Der am Eingang NEW\_RV angegebene Referenzwert wird in einen Schnellen Zähler geladen, wenn das Bit am Eingang RV gesetzt ist.
- **Zeitdauer der Frequenzmessung (NEW\_PERIOD):** Die Zeitdauer der Frequenzmessung wird durch die folgenden Werte am Eingang NEW\_PERIOD festgelegt: 10 = 0,01s, 100 = 0,1s, 1000 = 1s.  
Eine Aktualisierung der Zeitdauer ist dann möglich, wenn die Funktion "Frequenzmessen" für den angegebenen Schnellen Zähler parametrier ist. Die am Eingang NEW\_PERIOD angegebene Zeitdauer wird in einen Schnellen Zähler geladen, wenn das Bit am Eingang PERIOD gesetzt ist.

Die Anweisung "Schnelle Zähler steuern" wird nur ausgeführt, wenn der Signalzustand am Eingang EN "1" ist. Solange die Operation bearbeitet wird, bleibt das Bit am Ausgang BUSY gesetzt. Wenn die Operation vollständig ausgeführt ist, wird das Bit am Ausgang BUSY zurückgesetzt.

Der Freigabeausgang ENO wird nur gesetzt, wenn der Freigabeeingang EN den Signalzustand "1" liefert und keine Fehler während der Bearbeitung der Operation auftreten.

Beim Einfügen der Anweisung "Schnelle Zähler steuern" wird ein Instanz-Datenbaustein angelegt, in dem die Daten der Operation gespeichert werden.

## Parameter STATUS

Am Ausgang STATUS kann abgefragt werden, ob Fehler während der Bearbeitung der Anweisung "Schnelle Zähler steuern" aufgetreten sind. Die folgende Tabelle zeigt die Bedeutung der Werte, die am Ausgang STATUS ausgegeben werden:

| Fehlercode (Hexadezimal) | Beschreibung                                   |
|--------------------------|--|
| 0                        | Keine Fehler                                   |
| 80A1                     | Hardware-Kennung der Schnellen Zähler ungültig |
| 80B1                     | Zählrichtung (NEW_DIR) ungültig                |
| 80B2                     | Zählwert (NEW_CV) ungültig                     |

| Fehlercode (Hexadezimal) | Beschreibung   |
|--------------------------|--|
| 80B3                     | Referenzwert (NEW_RV) ungültig   |
| 80B4                     | Zeitdauer der Frequenzmessung (NEW_PERIOD) ungültig                              |
| 80C0                     | Mehrfacher Zugriff auf den Schnellen Zähler                                      |
| 80D0                     | Der Schnelle Zähler (HSC) ist in der CPU Hardware Konfiguration nicht aktiviert. |

### 9.7.4.3 PID Control

#### PID\_Compact

#### Neuerungen PID\_Compact

##### PID\_Compact V2.1

- Einsatz mit S7-1200**  
 Ab PID\_Compact V2.1 ist die Anweisung mit V2-Funktionalität auch auf einer S7-1200 ab der Firmware-Version 4.0 einsetzbar.

##### PID\_Compact V2.0

- Verhalten im Fehlerfall**  
 Das Verhalten im Fehlerfall wurde grundlegend überarbeitet. PID\_Compact verhält sich in der Default-Einstellung Fehler toleranter. Dieses Verhalten wird beim Kopieren von PID\_Compact V1.X von einer S7-1200 CPU auf eine S7-1500 CPU eingestellt.

| ACHTUNG  |
|--|
| <p><b>Ihre Anlage kann beschädigt werden.</b></p> <p>Wenn Sie die Default-Einstellung verwenden, bleibt PID_Compact beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden.</p> <p>Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.</p> |

Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. Mit ErrorAck quittieren Sie die Fehler und Warnungen, ohne den Regler neu zu starten oder den I-Anteil zu löschen. Durch einen Wechsel der Betriebsart werden nicht mehr anstehende Fehler nicht mehr gelöscht.

Sie konfigurieren das Verhalten im Fehlerfall mit SetSubstituteOutput und ActivateRecoverMode.

- Ersatzausgangswert**  
 Sie können einen Ersatzausgangswert konfigurieren, der im Fehlerfall ausgegeben wird.
- Wechsel der Betriebsarten**  
 Die Betriebsart geben Sie am Durchgangsparameter Mode vor und starten diese über eine steigende Flanke an ModeActivate. Die Variable sRet.i\_Mode ist entfallen.

- **Multiinstanzfähigkeit**  
Sie können PID\_Compact als Multiinstanz-DB aufrufen. Dann wird kein Technologieobjekt angelegt und es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID\_Compact direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.
- **Anlaufverhalten**  
Die an Mode vorgegebene Betriebsart wird auch gestartet bei einer fallenden Flanke an Reset und bei Kaltstart der CPU, wenn RunModeByStartup = TRUE ist.
- **ENO-Verhalten**  
ENO wird in Abhängigkeit von der Betriebsart gesetzt.  
Wenn State = 0 ist, dann ist ENO = FALSE.  
Wenn State ≠ 0 ist, dann ist ENO = TRUE.
- **Sollwertvorgabe während der Optimierung**  
Die erlaubte Schwankung des Sollwerts während der Optimierung konfigurieren Sie an der Variable CancelTuningLevel.
- **Wertebereich für Ausgangswertgrenzen**  
Der Wert 0.0 muss nicht mehr innerhalb der Ausgangswertgrenzen liegen.
- **I-Anteil vorbelegen**  
Mit den Variablen IntegralResetMode und OverwriteInitialOutputValue bestimmen Sie, wie der I-Anteil beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb" vorbelegt wird.
- **Aufschalten einer Störgröße**  
Sie können am Parameter Disturbance eine Störgröße aufschalten.
- **Vorbelegung der PID-Parameter**  
Folgende Vorbelegungen wurden geändert:
  - Gewichtung des P-Anteils (PWeighting) von 0.0 auf 1.0
  - Gewichtung des D-Anteils (DWeighting) von 0.0 auf 1.0
  - Koeffizient für den Differenzierverszug (TdFiltRatio) von 0.0 auf 0.2
- **Umbenennung der Variablen**  
Die statischen Variablen wurden kompatibel zu PID\_3Step neu benannt.

## PID\_Compact V1.2

- **Handbetrieb beim Anlauf der CPU**  
Wenn beim Start der CPU ManualEnable = TRUE, startet PID\_Compact im Handbetrieb. Eine steigende Flanke an ManualEnable ist nicht notwendig.
- **Erstoptimierung**  
Wenn die CPU während der Erstoptimierung ausgeschaltet wird, wird die Erstoptimierung beim Einschalten der CPU erneut gestartet.

### PID\_Compact V1.1

- **Handbetrieb beim Anlauf der CPU**  
 Beim Start der CPU schaltet PID\_Compact nur bei einer steigenden Flanke an ManualEnable in den Handbetrieb. Ohne steigende Flanke startet PID\_Compact in der letzten Betriebsart bei der ManualEnable FALSE war.
- **Verhalten bei Reset**  
 Durch eine steigende Flanke an Reset werden Fehler und Warnungen zurückgesetzt und der I-Anteil gelöscht. Durch eine fallende Flanke an Reset wird in die zuletzt aktive Betriebsart gewechselt.
- **Vorbelegung der Obergrenze Istwert**  
 Die Vorbelegung von r\_Pv\_Hlm wurde zu 120.0 geändert.
- **Überwachung der Abtastzeit**
  - Es wird kein Fehler mehr ausgegeben, wenn aktuelle Abtastzeit  $\geq 1,5 \times$  aktueller Mittelwert oder Aktuelle Abtastzeit  $\leq 0,5 \times$  aktueller Mittelwert. Im Automatikbetrieb darf die Abtastzeit stärker abweichen.
  - PID\_Compact ist kompatibel mit FW ab V2.0.
- **Zugriff auf Variablen**  
 Die folgenden Variablen können jetzt im Anwenderprogramm verwendet werden.
  - i\_Event\_SUT
  - i\_Event\_TIR
  - r\_Ctrl\_Ioutv
- **Fehlerbehebung**  
 PID\_Compact gibt jetzt korrekte Impulse aus, wenn die kleinste Einschaltzeit ungleich der kleinsten Ausschaltzeit ist.

### Kompatibilität mit CPU und FW

Die folgende Tabelle zeigt, auf welcher CPU Sie welche Version von PID\_Compact einsetzen können.

| CPU     | FW   | PID_Compact  |
|---------|------|--------------|
| S7-1200 | V4.x | V2.1<br>V1.2 |
| S7-1200 | V3.X | V1.2<br>V1.1 |
| S7-1200 | V2.X | V1.2<br>V1.1 |
| S7-1200 | V1.X | V1.0         |
| S7-1500 | V1.1 | V2.1<br>V2.0 |
| S7-1500 | V1.0 | V2.0         |



## PID\_Compact V2

### Beschreibung PID\_Compact V2

#### Beschreibung

Die Anweisung PID\_Compact stellt einen PID-Regler mit integrierter Optimierung für proportional wirkende Stellglieder zur Verfügung.

Folgende Betriebsarten sind möglich:

- Inaktiv
- Erstopptimierung
- Nachoptimierung
- Automatikbetrieb
- Handbetrieb
- Ersatzausgangswert mit Fehlerüberwachung

Die Betriebsarten sind detailliert beschrieben beim Parameter State.

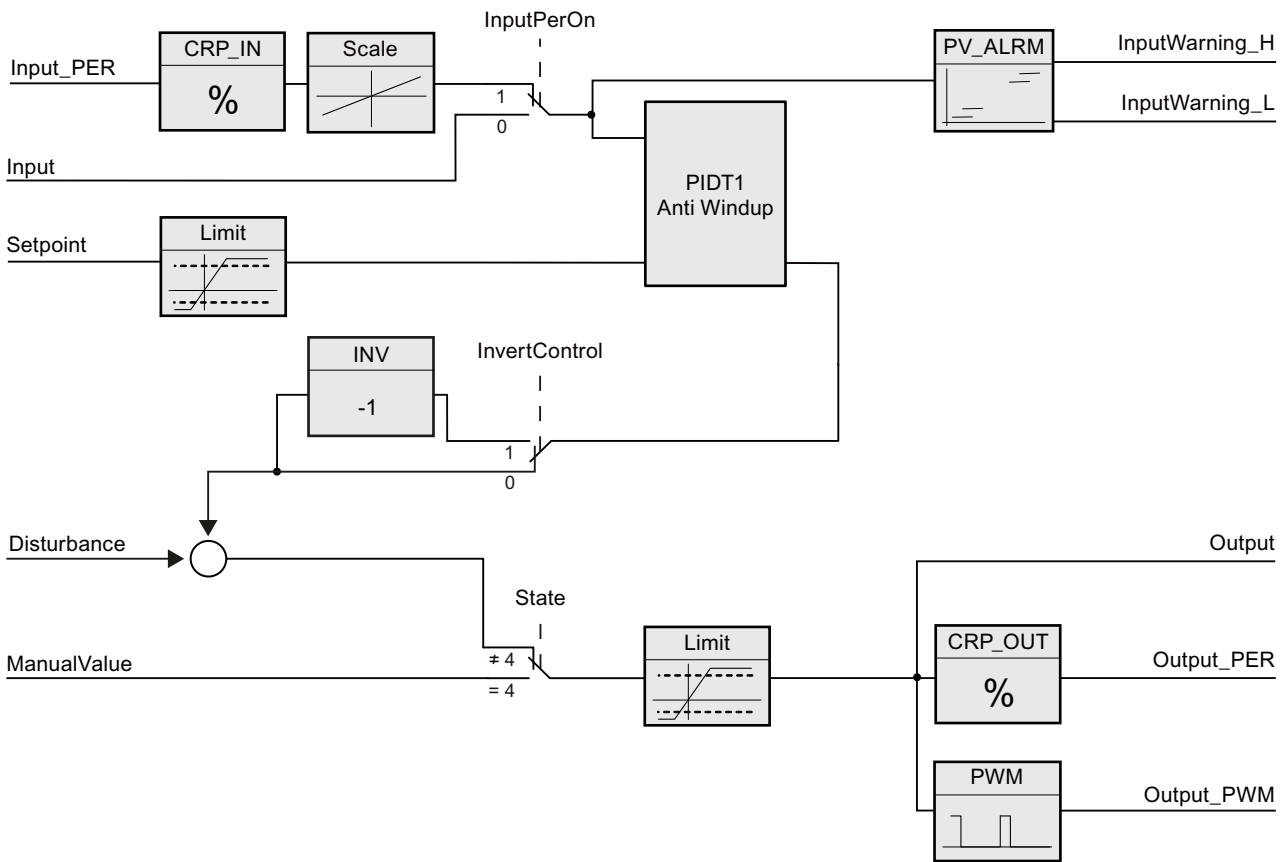
#### PID-Algorithmus

PID\_Compact ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der PID-Algorithmus arbeitet nach folgender Formel:

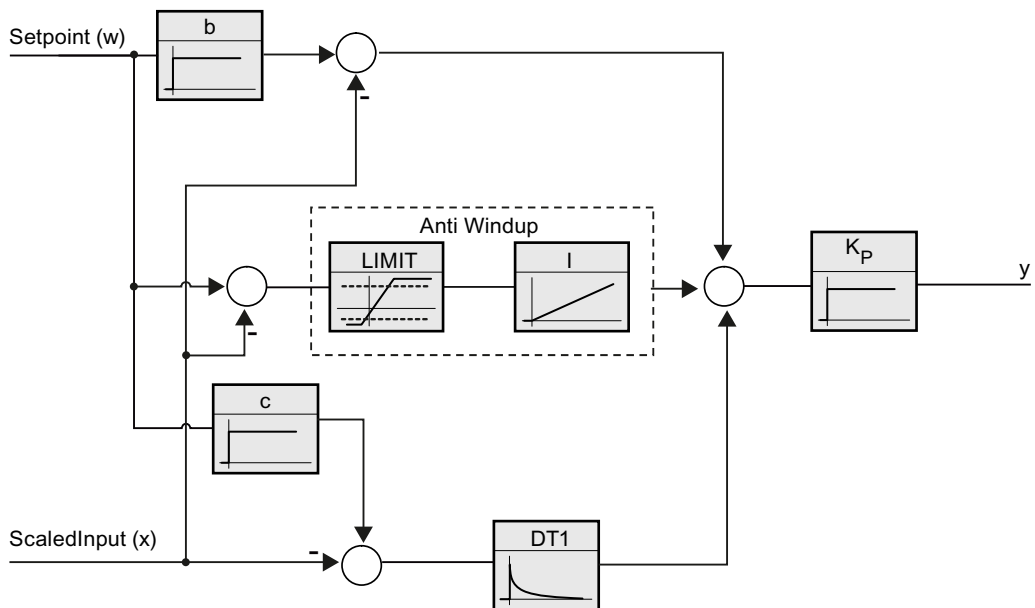
$$y = K_p \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

| Symbol         | Beschreibung   |
|----------------|--|
| y              | Ausgangswert des PID-Algorithmus   |
| K <sub>p</sub> | Proportionalverstärkung  |
| s              | Laplace-Operator   |
| b              | Gewichtung des P-Anteils   |
| w              | Sollwert   |
| x              | Istwert  |
| T <sub>i</sub> | Integrationszeit   |
| T <sub>D</sub> | Differenzierzeit   |
| a              | Koeffizient für den Differenzierverszug (Differenzierverszug T1 = a × T <sub>D</sub> ) |
| c              | Gewichtung des D-Anteils   |

**Blockschaltbild PID\_Compact**



**Blockschaltbild PIDT1 mit Anti-Windup**



## Aufruf

PID\_Compact wird im konstanten Zeitraster eines Weckalarm-OBs aufgerufen.

Wenn Sie PID\_Compact als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID\_Compact direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

## Laden in Gerät

Die Aktualwerte permanenter Variablen werden nur aktualisiert, wenn Sie PID\_Compact vollständig laden.

Technologieobjekte in Gerät laden (Seite 4056)

## Anlauf

Bei Anlauf der CPU startet PID\_Compact in der Betriebsart, die am Durchgangsparameter Mode gespeichert ist. Um beim Anlauf in die Betriebsart "Inaktiv" zu wechseln, setzen Sie RunModeByStartup = FALSE.

## Verhalten im Fehlerfall

Im Automatikbetrieb und während der Inbetriebnahme hängt das Verhalten im Fehlerfall von den Variablen SetSubstituteOutput und ActivateRecoverMode ab. Im Handbetrieb ist das Verhalten unabhängig von SetSubstituteOutput und ActivateRecoverMode. Wenn ActivateRecoverMode = TRUE ist, hängt das Verhalten zusätzlich vom aufgetretenen Fehler ab.

| SetSubstituteOutput | ActivateRecoverMode | Konfigurationseditor<br>> Ausgangswert<br>> Output setzen auf | Verhalten  |
|---------------------|---------------------|---|--|
| nicht relevant      | FALSE               | Null (Inaktiv)  | Wechsel in Betriebsart "Inaktiv" (State = 0)<br>Der Wert 0.0 0 wird an das Stellglied übertragen.  |
| FALSE               | TRUE                | Aktuellen Ausgangswert für die Fehlerdauer                    | Wechsel in Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" (State = 5)<br>Der aktuelle Ausgangswert wird für die Fehlerdauer an das Stellglied übertragen.    |
| TRUE                | TRUE                | Ersatzausgangswert für die Fehlerdauer                        | Wechsel in Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" (State = 5)<br>Der Wert an SubstituteOutput wird für die Fehlerdauer an das Stellglied übertragen. |

PID\_Compact verwendet im Handbetrieb ManualValue als Ausgangswert, außer ManualValue ist ungültig. Wenn ManualValue ungültig ist, wird SubstituteOutput verwendet. Wenn ManualValue und SubstituteOutput ungültig sind, wird Config.OutputLowerLimit verwendet.

Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. ErrorBits wird durch eine steigende Flanke an Reset oder ErrorAck zurückgesetzt.

## Arbeitsweise PID\_Compact V2

### Istwertgrenzen überwachen

In den Variablen Config.InputUpperLimit und Config.InputLowerLimit legen Sie eine Ober- und Untergrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Grenzen liegt, tritt ein Fehler auf (ErrorBits = 0001h).

In den Variablen Config.InputUpperWarning und Config.InputLowerWarning legen Sie eine obere und untere Warngrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Warngrenzen liegt, tritt eine Warnung auf (Warning = 0040h) und der Ausgangsparameter InputWarning\_H oder InputWarning\_L wird TRUE.

### Sollwert begrenzen

In den Variablen Config.SetpointUpperLimit und Config.SetpointLowerLimit legen Sie eine Ober- und Untergrenze des Sollwerts fest. PID\_Compact begrenzt den Sollwert automatisch auf die Istwertgrenzen. Sie können den Sollwert auf einen kleineren Bereich begrenzen. PID\_Compact prüft, ob dieser Bereich innerhalb der Istwertgrenzen liegt. Wenn der Sollwert außerhalb dieser Grenzen liegt, wird die Ober- oder Untergrenze als Sollwert verwendet und der Ausgangsparameter SetpointLimit\_H oder SetpointLimit\_L wird TRUE.

Der Sollwert wird in allen Betriebsarten begrenzt.

### Ausgangswert begrenzen

In den Variablen Config.OutputUpperLimit und Config.OutputLowerLimit legen Sie eine Ober- und Untergrenze des Ausgangswerts fest. Output, ManualValue und SubstituteOutput werden auf diese Werte begrenzt. Die Ausgangswertgrenzen müssen zum Regelsinn passen.

Die gültigen Werte für die Ausgangswertgrenzen hängen vom verwendeten Output ab.

|            |                    |
|------------|--------------------|
| Output     | -100.0 bis 100.0 % |
| Output_PER | -100.0 bis 100.0 % |
| Output_PWM | 0.0 bis 100.0 %    |

Es muss gelten:

OutputUpperLimit > OutputLowerLimit

### Ersatzausgangswert

PID\_Compact kann im Fehlerfall einen Ersatzausgangswert ausgeben, den Sie an der Variablen SubstituteOutput vorgeben. Der Ersatzausgangswert muss innerhalb der Ausgangswertgrenzen liegen.

## Gültigkeit der Signale überwachen

Die Werte der folgenden Parameter werden bei Verwendung auf Gültigkeit überwacht:

- Setpoint
- Input
- Input\_PER
- Disturbance
- ManualValue
- SubstituteOutput
- Output
- Output\_PER
- Output\_PWM

## Überwachung der Abtastzeit PID\_Compact

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden OB. Die Anweisung PID\_Compact misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (Error = 0800h).

Während der Optimierung tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert  $\geq 1,1 \times$  alter Mittelwert
- Neuer Mittelwert  $\leq 0,9 \times$  alter Mittelwert

Im Automatikbetrieb tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert  $\geq 1,5 \times$  alter Mittelwert
- Neuer Mittelwert  $\leq 0,5 \times$  alter Mittelwert

Wenn Sie die Überwachung der Abtastzeit ausschalten (CycleTime.EnMonitoring = FALSE), können Sie PID\_Compact auch im OB1 aufrufen. Sie müssen dann aufgrund der schwankenden Abtastzeit eine schlechtere Regelqualität akzeptieren.

## Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID\_Compact werden bei jedem Aufruf durchgeführt.

Wenn Sie Output\_PWM verwenden, wird die Genauigkeit des Ausgangssignals bestimmt durch das Verhältnis von Abtastzeit PID-Algorithmus zu Zykluszeit des OB. Die Zykluszeit sollte mindestens das 10-fache der Abtastzeit PID-Algorithmus betragen.

### Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn. Für Kühlungen und Abflussregelungen kann es notwendig sein, den Regelsinn zu invertieren. PID\_Compact arbeitet nicht mit negativer Proportionalverstärkung. Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts. Der Regelsinn wird auch während Erst- und Nachoptimierung berücksichtigt.

### Eingangsparameter PID\_Compact V2

Tabelle 9-77

| Parameter    | Datentyp | Vorbelegung | Beschreibung  |
|--------------|----------|-------------|---|
| Setpoint     | REAL     | 0.0         | Sollwert des PID-Reglers im Automatikbetrieb  |
| Input        | REAL     | 0.0         | Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet.<br>Wenn Sie den Parameter Input verwenden, muss Config.InputPerOn = FALSE sein.   |
| Input_PER    | INT      | 0           | Ein Analogeingang wird als Quelle des Istwerts verwendet.<br>Wenn Sie den Parameter Input_PER verwenden, muss Config.InputPerOn = TRUE sein.  |
| Disturbance  | REAL     | 0.0         | Störgröße oder Vorsteuerungswert  |
| ManualEnable | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>• Flanke FALSE -&gt; TRUE aktiviert die Betriebsart "Handbetrieb", State = 4, Mode bleibt unverändert. Solange ManualEnable = TRUE ist, können Sie die Betriebsart nicht über eine steigende Flanke an ModeActivate ändern und den Inbetriebnahmedialog nicht nutzen.</li> <li>• Flanke TRUE -&gt; FALSE aktiviert die Betriebsart, die durch Mode vorgegeben wird.</li> </ul> Es wird empfohlen, die Betriebsart nur über ModeActivate zu ändern. |
| ManualValue  | REAL     | 0.0         | Handwert<br>Dieser Wert wird im Handbetrieb als Ausgangswert verwendet.<br>Zulässig sind Werte von Config.OutputLowerLimit bis Config.OutputUpperLimit  |
| ErrorAck     | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>• Flanke FALSE -&gt; TRUE<br/>ErrorBits und Warning werden zurückgesetzt.</li> </ul>   |

| Parameter    | Datentyp | Vorbelegung | Beschreibung  |
|--------------|----------|-------------|---|
| Reset        | BOOL     | FALSE       | Führt einen Neustart des Reglers durch. <ul style="list-style-type: none"> <li>• Flanke FALSE -&gt; TRUE <ul style="list-style-type: none"> <li>– Wechsel in Betriebsart "Inaktiv"</li> <li>– ErrorBits und Warnings werden zurückgesetzt.</li> <li>– I-Anteil wird gelöscht (PID-Parameter bleiben erhalten)</li> </ul> </li> <li>• Solange Reset = TRUE ist, bleibt PID_Compact in der Betriebsart "Inaktiv" (State = 0).</li> <li>• Flanke TRUE -&gt; FALSE<br/>PID_Compact wechselt in die Betriebsart, die an Mode gespeichert ist.</li> </ul> |
| ModeActivate | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>• Flanke FALSE -&gt; TRUE<br/>PID_Compact wechselt in die Betriebsart, die an Mode gespeichert ist.</li> </ul>   |

## Ausgangsparameter PID\_Compact V2

Tabelle 9-78

| Parameter  | Datentyp | Vorbelegung | Beschreibung   |
|--|----------|-------------|--|
| ScaledInput  | REAL     | 0.0         | Skalierter Istwert   |
| Die Ausgänge "Output", "Output_PER" und "Output_PWM" können parallel genutzt werden. |          |             |  |
| Output   | REAL     | 0.0         | Ausgangswert im REAL-Format  |
| Output_PER   | INT      | 0           | Analoger Ausgangswert  |
| Output_PWM   | BOOL     | FALSE       | Pulsweitenmodulierter Ausgangswert<br>Der Ausgangswert wird über variable Ein- und Ausschaltzeiten gebildet.   |
| SetpointLimit_H  | BOOL     | FALSE       | Wenn SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht (Setpoint $\geq$ Config.SetpointUpperLimit).<br>Der Sollwert wird auf Config.SetpointUpperLimit begrenzt.  |
| SetpointLimit_L  | BOOL     | FALSE       | Wenn SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht (Setpoint $\leq$ Config.SetpointLowerLimit).<br>Der Sollwert wird auf Config.SetpointLowerLimit begrenzt. |
| InputWarning_H   | BOOL     | FALSE       | Wenn InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten.   |
| InputWarning_L   | BOOL     | FALSE       | Wenn InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten.   |

| Parameter | Datentyp | Vorbelegung | Beschreibung   |
|-----------|----------|-------------|--|
| State     | INT      | 0           | <p>Der Parameter State (Seite 2691) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit dem Eingangsparameter Mode und einer steigenden Flanke an ModeActivate.</p> <ul style="list-style-type: none"> <li>• State = 0: Inaktiv</li> <li>• State = 1: Ersoptimierung</li> <li>• State = 2: Nachoptimierung</li> <li>• State = 3: Automatikbetrieb</li> <li>• State = 4: Handbetrieb</li> <li>• State = 5: Ersatzausgangswert mit Fehlerüberwachung</li> </ul> |
| Error     | BOOL     | FALSE       | Wenn Error = TRUE, liegt in diesem Takt mindestens eine Fehlermeldung vor.   |
| ErrorBits | DWORD    | DW#16#0     | Der Parameter ErrorBits (Seite 2695) zeigt, welche Fehlermeldungen vorliegen. ErrorBits ist remanent und wird bei einer steigenden Flanke an Reset oder ErrorAck zurückgesetzt.  |

### Durchgangparameter PID\_Compact V2

Tabelle 9-79

| Parameter | Datentyp | Vorbelegung | Beschreibung  |
|-----------|----------|-------------|---|
| Mode      | INT      | 4           | <p>An Mode geben Sie die Betriebsart vor, in die PID_Compact wechseln soll. Möglich sind:</p> <ul style="list-style-type: none"> <li>• Mode = 0: Inaktiv</li> <li>• Mode = 1: Ersoptimierung</li> <li>• Mode = 2: Nachoptimierung</li> <li>• Mode = 3: Automatikbetrieb</li> <li>• Mode = 4: Handbetrieb</li> </ul> <p>Die Betriebsart wird aktiviert durch:</p> <ul style="list-style-type: none"> <li>• Steigende Flanke an ModeActivate</li> <li>• Fallende Flanke an Reset</li> <li>• Fallende Flanke an ManualEnable</li> <li>• Kaltstart der CPU, wenn RunModeByStartup = TRUE</li> </ul> <p>Mode ist remanent.</p> <p>Eine detaillierte Beschreibung der Betriebsarten finden Sie unter Parameter State und Mode V2 (Seite 2691) .</p> |

**Siehe auch**

Parameter State und Mode V2 (Seite 2691)



## Statische Variablen PID\_Compact V2

Nicht aufgeführte Variablen dürfen Sie nicht verändern. Diese werden nur intern verwendet.

Tabelle 9-80

| Variable                    | Datentyp | Vorbelegung | Beschreibung   |
|-----------------------------|----------|-------------|--|
| IntegralResetMode           | INT      | 1           | <p>Die Variable IntegralResetMode bestimmt, wie PIDCtrl.IntegralSum beim Wechsel von der Betriebsart "Inaktiv" in "Automatikbetrieb" vorbelegt wird. Diese Einstellung wirkt nur für einen Zyklus.</p> <p>Möglich sind:</p> <ul style="list-style-type: none"> <li>• IntegralResetMode = 0: Glätten<br/>Der Wert von IntegralSum wird so vorbelegt, dass die Umschaltung stoßfrei erfolgt.</li> <li>• IntegralResetMode = 1: Löschen<br/>Der Wert von IntegralSum wird gelöscht. Wenn eine Regeldifferenz vorhanden ist, kommt es zu einem Sprung des Ausgangswerts.</li> <li>• IntegralResetMode = 2: Halten<br/>Der Wert von IntegralSum wird nicht verändert. Sie können über das Anwenderprogramm einen neuen Wert vorgeben.</li> <li>• IntegralResetMode = 3: Vorbelegen<br/>Der Wert von IntegralSum wird automatisch so vorbelegt, dass Output auf den Wert von OverwriteInitialOutputValue berechnet wird. Diese Einstellung ist z. B. für einen Ablöseregler sinnvoll.</li> </ul> |
| OverwriteInitialOutputValue | REAL     | 0.0         | Wenn IntegralResetMode = 3 ist, wird der Wert von IntegralSum automatisch so vorbelegt, dass im nächsten Zyklus Output = OverwriteInitialOutputValue ist.  |
| RunModeByStartup            | BOOL     | TRUE        | <p>Nach CPU Neustart Betriebsart an Mode aktivieren</p> <p>Wenn RunModeByStartup = TRUE, startet PID_Compact nach CPU-Anlauf in der Betriebsart, die an Mode gespeichert ist.</p> <p>Wenn RunModeByStartup = FALSE, bleibt PID_Compact nach CPU-Anlauf in der Betriebsart "Inaktiv".</p>   |
| LoadBackUp                  | BOOL     | FALSE       | Wenn LoadBackUp = TRUE, wird der letzte Satz PID-Parameter wieder geladen. Der Satz wurde vor der letzten Optimierung gespeichert. LoadBackUp wird automatisch wieder auf FALSE gesetzt.   |
| PhysicalUnit                | INT      | 0           | Physikalische Einheit des Ist- und Sollwerts, z. B. °C oder °F.  |
| PhysicalQuantity            | INT      | 0           | Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur.  |
| ActivateRecoverMode         | BOOL     | TRUE        | Die Variable ActivateRecoverMode V2 (Seite 2697) bestimmt das Verhalten im Fehlerfall.   |
| Warning                     | DWORD    | 0           | Die Variable Warning V2 (Seite 2699) zeigt die Warnungen seit Reset = TRUE oder ErrorAck = TRUE. Warning ist remanent.   |

| Variable               | Datentyp | Vorbelegung | Beschreibung  |
|------------------------|----------|-------------|---|
| Progress               | REAL     | 0.0         | Fortschritt der Optimierung in Prozent (0.0 - 100.0)  |
| CurrentSetpoint        | REAL     | 0.0         | CurrentSetpoint zeigt immer den aktuellen Sollwert an. Dieser Wert wird während der Optimierung eingefroren.  |
| CancelTuningLevel      | REAL     | 10.0        | Zulässige Schwankung des Sollwerts während der Optimierung. Die Optimierung wird erst abgebrochen, wenn gilt: <ul style="list-style-type: none"> <li>• Setpoint &gt; CurrentSetpoint + CancelTuningLevel oder</li> <li>• Setpoint &lt; CurrentSetpoint - CancelTuningLevel</li> </ul>   |
| SubstituteOutput       | REAL     | 0.0         | Ersatzausgangswert<br>Wenn die folgenden Bedingungen erfüllt sind, wird der Ersatzausgangswert verwendet: <ul style="list-style-type: none"> <li>• Im Automatikbetrieb ist ein Fehler aufgetren.</li> <li>• SetSubstituteOutput = TRUE</li> <li>• ActivateRecoverMode = TRUE</li> </ul>   |
| SetSubstituteOutput    | BOOL     | TRUE        | Wenn SetSubstituteOutput = TRUE und ActivateRecoverMode = TRUE sind, wird der konfigurierte Ersatzausgangswert ausgegeben, solange ein Fehler ansteht.<br>Wenn SetSubstituteOutput = FALSE und ActivateRecoverMode = TRUE sind, bleibt das Stellglied auf dem aktuellen Ausgangswert, solange ein Fehler ansteht.<br>Wenn ActivateRecoverMode = FALSE ist, wirkt sich SetSubstituteOutput nicht aus.<br>Wenn SubstituteOutput ungültig ist (ErrorBits = 20000h), kann der Ersatzausgangswert nicht ausgegeben werden. |
| Config.InputPerOn      | BOOL     | TRUE        | Wenn InputPerOn = TRUE, wird der Parameter Input_PER verwendet. Wenn InputPerOn = FALSE, wird der Parameter Input verwendet.  |
| Config.InvertControl   | BOOL     | FALSE       | Invertieren des Regelsinns<br>Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts.   |
| Config.InputUpperLimit | REAL     | 120.0       | Obergrenze des Istwert<br>Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht.<br>Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird mit der Voreinstellung kein Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_Compact verhält sich wie unter Verhalten im Fehlerfall konfiguriert.<br>InputUpperLimit > InputLowerLimit                               |
| Config.InputLowerLimit | REAL     | 0.0         | Untergrenze des Istwerts<br>Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht.<br>InputLowerLimit < InputUpperLimit   |

| Variable                  | Datentyp | Vorbelegung   | Beschreibung   |
|---------------------------|----------|---------------|--|
| Config.InputUpperWarning  | REAL     | 3.402822e+38  | <p>Obere Warngrenze des Istwerts</p> <p>Wenn Sie InputUpperWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Obergrenze Istwert als obere Warngrenze verwendet.</p> <p>Wenn Sie InputUpperWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obere Warngrenze verwendet.</p> <p><math>\text{InputUpperWarning} &gt; \text{InputLowerWarning}</math><br/> <math>\text{InputUpperWarning} \leq \text{InputUpperLimit}</math></p>     |
| Config.InputLowerWarning  | REAL     | -3.402822e+38 | <p>Untere Warngrenze des Istwerts</p> <p>Wenn Sie InputLowerWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Untergrenze Istwert als untere Warngrenze verwendet.</p> <p>Wenn Sie InputLowerWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet.</p> <p><math>\text{InputLowerWarning} &lt; \text{InputUpperWarning}</math><br/> <math>\text{InputLowerWarning} \geq \text{InputLowerLimit}</math></p> |
| Config.OutputUpperLimit   | REAL     | 100.0         | <p>Obergrenze des Ausgangswerts</p> <p>Details siehe OutputLowerLimit</p> <p><math>\text{OutputUpperLimit} &gt; \text{OutputLowerLimit}</math></p>   |
| Config.OutputLowerLimit   | REAL     | 0.0           | <p>Untergrenze des Ausgangswerts</p> <p>Für Output und Output_PER gilt der Wertebereich von -100.0 bis +100.0 einschließlich Null. Bei -100.0 beträgt Output_PER = -27648; bei +100.0 beträgt Output_PER = 27648.</p> <p>Für Output_PWM gilt der Wertebereich 0.0 bis +100.0.</p> <p>Die Ausgangswertgrenzen müssen zum Regelsinn passen.</p> <p><math>\text{OutputLowerLimit} &lt; \text{OutputUpperLimit}</math></p>   |
| Config.SetpointUpperLimit | REAL     | 3.402822e+38  | <p>Obergrenze des Sollwerts</p> <p>Wenn Sie SetpointUpperLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert verwendet.</p> <p>Wenn Sie SetpointUpperLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.</p>   |
| Config.SetpointLowerLimit | REAL     | -3.402822e+38 | <p>Untergrenze des Sollwerts</p> <p>Wenn Sie SetpointLowerLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert verwendet.</p> <p>Wenn Sie SetpointLowerLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.</p>   |

| Variable                          | Datentyp | Vorbelegung | Beschreibung   |
|-----------------------------------|----------|-------------|--|
| Config.MinimumOnTime              | REAL     | 0.0         | Minimale Einschaltzeit der Pulsweitenmodulation in Sekunden wird gerundet auf<br>MinimumOnTime = n×CycleTime.Value   |
| Config.MinimumOffTime             | REAL     | 0.0         | Minimale Ausschaltzeit der Pulsweitenmodulation in Sekunden wird gerundet auf<br>MinimumOffTime = n×CycleTime.Value  |
| Config.InputScaling.UpperPointIn  | REAL     | 27648.0     | Skalierung Input_PER Oben<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER in Prozent umgerechnet.   |
| Config.InputScaling.LowerPointIn  | REAL     | 0.0         | Skalierung Input_PER Unten<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER in Prozent umgerechnet.  |
| Config.InputScaling.UpperPointOut | REAL     | 100.0       | Skalierter oberer Istwert<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER in Prozent umgerechnet.   |
| Config.InputScaling.LowerPointOut | REAL     | 0.0         | Skalierter unterer Istwert<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn wird Input_PER in Prozent umgerechnet.  |
| CycleTime.StartEstimation         | BOOL     | TRUE        | Wenn CycleTime.StartEstimation = TRUE, wird die automatische Ermittlung der Zykluszeit gestartet. Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE.   |
| CycleTime.EnEstimation            | BOOL     | TRUE        | Wenn CycleTime.EnEstimation = TRUE, wird die Abtastzeit PID_Compact berechnet.<br>Wenn CycleTime.EnEstimation = FALSE, wird die Abtastzeit PID_Compact nicht berechnet und Sie müssen CycleTime.Value manuell korrekt konfigurieren.                                     |
| CycleTime.EnMonitoring            | BOOL     | TRUE        | Wenn CycleTime.EnMonitoring = FALSE, wird die Abtastzeit PID_Compact nicht überwacht. Wenn PID_Compact nicht innerhalb der Abtastzeit ausgeführt werden kann, wird kein Fehler (ErrorBits=0800h) ausgegeben und PID_Compact wechselt nicht in die Betriebsart "Inaktiv". |
| CycleTime.Value                   | REAL     | 0.1         | Abtastzeit PID_Compact in Sekunden<br>CycleTime.Value wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.   |
| CtrlParamsBackUp.Gain             | REAL     | 1.0         | gespeicherte Proportionalverstärkung<br>Werte aus der Struktur CtrlParamsBackUp können mit LoadBackUp = TRUE wieder geladen werden.  |
| CtrlParamsBackUp.Ti               | REAL     | 20.0        | gespeicherte Integrationszeit [s]  |
| CtrlParamsBackUp.Td               | REAL     | 0.0         | gespeicherte Differenzierzeit [s]  |
| CtrlParamsBackUp.TdFiltRatio      | REAL     | 0.0         | gespeicherter Koeffizient Differenzierversug   |
| CtrlParamsBackUp.PWeighting       | REAL     | 0.0         | gespeicherter Gewichtungsfaktor des P-Anteils  |

| Variable                        | Datentyp | Vorbelegung | Beschreibung  |
|---------------------------------|----------|-------------|---|
| CtrlParamsBackUp.DWeighting     | REAL     | 0.0         | gespeicherter Gewichtungsfaktor des D-Anteils   |
| CtrlParamsBackUp.Cycle          | REAL     | 1.0         | gespeicherte Abtastzeit PID-Algorithmus   |
| PIDSelfTune.SUT.CalculateParams | BOOL     | FALSE       | Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn SUT.CalculateParams = TRUE, werden anhand dieser Eigenschaften die Parameter für die Erstopptimierung neu berechnet. Dadurch kann die Methode für die Parameterberechnung geändert werden, ohne die Optimierung zu wiederholen.<br>SUT.CalculateParams wird nach der Berechnung auf FALSE gesetzt.  |
| PIDSelfTune.SUT.TuneRule        | INT      | 0           | Parameter während Erstopptimierung berechnen nach Methode: <ul style="list-style-type: none"> <li>• SUT.TuneRule = 0: PID nach Chien, Hrones und Reswick</li> <li>• SUT.TuneRule = 1: PI nach Chien, Hrones und Reswick</li> </ul>  |
| PIDSelfTune.SUT.State           | INT      | 0           | Die Variable SUT.State zeigt die aktuelle Phase der Erstopptimierung: <ul style="list-style-type: none"> <li>• State = 0: Erstopptimierung initialisieren</li> <li>• State = 100: Standardabweichung berechnen</li> <li>• State = 200: Wendepunkt ermitteln</li> <li>• State = 9900: Erstopptimierung erfolgreich</li> <li>• State = 1: Erstopptimierung nicht erfolgreich</li> </ul>   |
| PIDSelfTune.TIR.RunIn           | BOOL     | FALSE       | Mit der Variable RunIn können Sie festlegen, dass eine Nachoptimierung auch ohne Erstopptimierung durchgeführt wird. <ul style="list-style-type: none"> <li>• RunIn = FALSE<br/>Wenn die Nachoptimierung aus der Betriebsart Inaktiv oder Handbetrieb gestartet wird, wird eine Erstopptimierung gestartet. Wenn die Voraussetzungen für Erstopptimierung nicht erfüllt sind, verhält sich PID_Compact wie bei RunIn = TRUE.<br/>Wenn die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt.<br/>Erst dann startet die Nachoptimierung. Ist die Erstopptimierung nicht möglich, wechselt PID_Compact in die Betriebsart, aus der die Optimierung gestartet wurde.</li> <li>• RunIn = TRUE<br/>Die Erstopptimierung wird übersprungen. PID_Compact versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet anschließend automatisch. RunIn wird nach der Nachoptimierung auf FALSE gesetzt.</li> </ul> |

| Variable                        | Datentyp | Vorbelegung | Beschreibung  |
|---------------------------------|----------|-------------|---|
| PIDSelfTune.TIR.CalculateParams | BOOL     | FALSE       | Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn TIR.CalculateParams = TRUE, werden anhand dieser Eigenschaften die Parameter für die Nachoptimierung neu berechnet. Dadurch kann die Methode für die Parameterberechnung geändert werden, ohne die Optimierung zu wiederholen.<br>TIR.CalculateParams wird nach der Berechnung auf FALSE gesetzt.   |
| PIDSelfTune.TIR.TuneRule        | INT      | 0           | Parameter während Nachoptimierung berechnen nach Methode: <ul style="list-style-type: none"> <li>• TIR.TuneRule = 0: PID automatisch</li> <li>• TIR.TuneRule = 1: PID schnell</li> <li>• TIR.TuneRule = 2: PID langsam</li> <li>• TIR.TuneRule = 3: Ziegler-Nichols PID</li> <li>• TIR.TuneRule = 4: Ziegler-Nichols PI</li> <li>• TIR.TuneRule = 5: Ziegler-Nichols P</li> </ul>   |
| PIDSelfTune.TIR.State           | INT      | 0           | Die Variable TIR.State zeigt die aktuelle Phase der "Nachoptimierung": <ul style="list-style-type: none"> <li>• State = -100 Die Nachoptimierung ist nicht möglich. Es wird erst eine Erstopoptimierung durchgeführt.</li> <li>• State = 0: Nachoptimierung initialisieren</li> <li>• State = 200: Standardabweichung berechnen</li> <li>• State = 300: Versuchen Sollwert zu erreichen</li> <li>• State = 400: Versuchen Sollwert mit vorhandenen PID-Parametern zu erreichen (wenn Erstopoptimierung erfolgreich war)</li> <li>• State = 500: Oszillation ermitteln und Parameter berechnen</li> <li>• State = 9900: Nachoptimierung erfolgreich</li> <li>• State = 1: Nachoptimierung nicht erfolgreich</li> </ul> |
| PIDCtrl.IntegralSum             | REAL     | 0.0         | Aktueller I-Anteil  |
| Retain.CtrlParams.Gain          | REAL     | 1.0         | Aktive Proportionalverstärkung<br>Verwenden Sie zum Invertieren des Regelsinns die Variable Config.InvertControl. Negative Werte an Gain invertieren den Regelsinn ebenfalls. Es wird empfohlen den Regelsinn nur über InvertControl einzustellen. Wenn InvertControl = TRUE und Gain < 0.0 sind, ist der Regelsinn auch invertiert.<br>Gain ist remanent.  |
| Retain.CtrlParams.Ti            | REAL     | 20.0        | <ul style="list-style-type: none"> <li>• CtrlParams.Ti &gt; 0.0: Aktive Integrationszeit</li> <li>• CtrlParams.Ti = 0.0: I-Anteil ist ausgeschaltet</li> </ul> Ti ist remanent.   |
| Retain.CtrlParams.Td            | REAL     | 0.0         | <ul style="list-style-type: none"> <li>• CtrlParams.Td &gt; 0.0: Aktive Differenzierzeit</li> <li>• CtrlParams.Td = 0.0: D-Anteil ist ausgeschaltet</li> </ul> Td ist remanent.   |

| Variable                      | Datentyp | Vorbelegung | Beschreibung  |
|-------------------------------|----------|-------------|---|
| Retain.CtrlParams.TdFiltRatio | REAL     | 0.2         | <p>Aktiver Koeffizient Differenzierverzögerung</p> <p>Die Wirkung des D-Anteils wird durch den Koeffizient Differenzierverzögerung verzögert.</p> <p>Differenzierverzögerung = Differenzierzeit × Koeffizient Differenzierverzögerung</p> <ul style="list-style-type: none"> <li>• 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.</li> <li>• 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit <b>einer</b> dominierenden Zeitkonstanten bewährt.</li> <li>• &gt; 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.</li> </ul> <p>TdFiltRatio ist remanent.</p> |
| Retain.CtrlParams.PWeighting  | REAL     | 1.0         | <p>Aktive Gewichtung des P-Anteils</p> <p>Sie können bei Sollwertänderungen den P-Anteil abschwächen.</p> <p>Sinnvoll sind Werte von 0.0 bis 1.0.</p> <ul style="list-style-type: none"> <li>• 1.0: P-Anteil bei Sollwertänderung voll wirksam</li> <li>• 0.0: P-Anteil bei Sollwertänderung nicht wirksam</li> </ul> <p>Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.</p> <p>PWeighting ist remanent.</p>   |
| Retain.CtrlParams.DWeighting  | REAL     | 1.0         | <p>Aktive Gewichtung des D-Anteils</p> <p>Sie können bei Sollwertänderungen den D-Anteil abschwächen.</p> <p>Sinnvoll sind Werte von 0.0 bis 1.0.</p> <ul style="list-style-type: none"> <li>• 1.0: D-Anteil bei Sollwertänderung voll wirksam</li> <li>• 0.0: D-Anteil bei Sollwertänderung nicht wirksam</li> </ul> <p>Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.</p> <p>DWeighting ist remanent.</p>   |
| Retain.CtrlParams.Cycle       | REAL     | 1.0         | <p>Aktive Abtastzeit des PID-Algorithmus</p> <p>CtrlParams.Cycle wird während der Optimierung ermittelt und zu einem ganzzahligen Vielfachen von CycleTime.Value aufgerundet.</p> <p>Cycle ist remanent.</p>  |

### Hinweis

Verändern Sie die hier aufgelisteten Variablen in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden.

**Siehe auch**

Variable ActivateRecoverMode V2 (Seite 2697)

Variable Warning V2 (Seite 2699)

Technologieobjekte in Gerät laden (Seite 4056)

**Änderungen der Schnittstelle PID\_Compact V2**

Die folgende Tabelle zeigt, was sich an der Schnittstelle der Anweisung PID\_Compact geändert hat.

| PID_Compact V1      | PID_Compact V2              | Änderung                 |
|---------------------|-----------------------------|--------------------------|
| Input_PER           | Input_PER                   | Datentyp von Word zu Int |
|                     | Disturbance                 | Neu                      |
|                     | ErrorAck                    | Neu                      |
|                     | ModeActivate                | Neu                      |
| Output_PER          | Output_PER                  | Datentyp von Word zu Int |
| Error               | ErrorBits                   | Umbenannt                |
|                     | Error                       | Neu                      |
|                     | Mode                        | Neu                      |
| sb_RunModeByStartup | RunModeByStartup            | Funktion                 |
|                     | IntegralResetMode           |                          |
|                     | OverwriteInitialOutputValue | Neu                      |
|                     | SetSubstituteOutput         | Neu                      |
|                     | CancelTuningLevel           | Neu                      |
|                     | SubstituteOutput            | Neu                      |

Die folgende Tabelle zeigt, welche Variablen umbenannt wurden.

| PID_Compact V1.x    | PID_Compact V2               |
|---------------------|------------------------------|
| sb_GetCycleTime     | CycleTime.StartEstimation    |
| sb_EnCyclEstimation | CycleTime.EnEstimation       |
| sb_EnCyclMonitoring | CycleTime.EnMonitoring       |
| sb_RunModeByStartup | RunModeByStartup             |
| si_Unit             | PhysicalUnit                 |
| si_Type             | PhysicalQuantity             |
| sd_Warning          | Warning                      |
| sBackUp.r_Gain      | CtrlParamsBackUp.Gain        |
| sBackUp.r_Ti        | CtrlParamsBackUp.Ti          |
| sBackUp.r_Td        | CtrlParamsBackUp.Td          |
| sBackUp.r_A         | CtrlParamsBackUp.TdFiltRatio |
| sBackUp.r_B         | CtrlParamsBackUp.PWeighting  |
| sBackUp.r_C         | CtrlParamsBackUp.DWeighting  |



| PID_Compact V1.x          | PID_Compact V2  |
|---------------------------|---|
| sBackUp.r_Cycle           | CtrlParamsBackUp.Cycle  |
| sPid_Calc.r_Cycle         | CycleTime.Value   |
| sPid_Calc.b_RunIn         | PIDSelfTune.TIR.RunIn   |
| sPid_Calc.b_CalcParamSUT  | PIDSelfTune.SUT.CalculateParams   |
| sPid_Calc.b_CalcParamTIR  | PIDSelfTune.TIR.CalculateParams   |
| sPid_Calc.i_CtrlTypeSUT   | PIDSelfTune.SUT.TuneRule  |
| sPid_Calc.i_CtrlTypeTIR   | PIDSelfTune.TIR.TuneRule  |
| sPid_Calc.r_Progress      | Progress  |
| sPid_Cmpt.r_Sp_Hlm        | Config.SetpointUpperLimit   |
| sPid_Cmpt.r_Sp_Llm        | Config.SetpointLowerLimit   |
| sPid_Cmpt.r_Pv_Norm_IN_1  | Config.InputScaling.LowerPointIn  |
| sPid_Cmpt.r_Pv_Norm_IN_2  | Config.InputScaling.UpperPointIn  |
| sPid_Cmpt.r_Pv_Norm_OUT_1 | Config.InputScaling.LowerPointOut   |
| sPid_Cmpt.r_Pv_Norm_OUT_2 | Config.InputScaling.UpperPointOut   |
| sPid_Cmpt.r_Lmn_Hlm       | Config.OutputUpperLimit   |
| sPid_Cmpt.r_Lmn_Llm       | Config.OutputLowerLimit   |
| sPid_Cmpt.b_Input_PER_On  | Config.InputPerOn   |
| sPid_Cmpt.b_LoadBackUp    | LoadBackUp  |
| sPid_Cmpt.b_InvCtrl       | Config.InvertControl  |
| sPid_Cmpt.r_Lmn_Pwm_PPTm  | Config.MinimumOnTime  |
| sPid_Cmpt.r_Lmn_Pwm_PBTm  | Config.MinimumOffTime   |
| sPid_Cmpt.r_Pv_Hlm        | Config.InputUpperLimit  |
| sPid_Cmpt.r_Pv_Llm        | Config.InputLowerLimit  |
| sPid_Cmpt.r_Pv_HWrn       | Config.InputUpperWarning  |
| sPid_Cmpt.r_Pv_LWrn       | Config.InputLowerWarning  |
| sParamCalc.i_Event_SUT    | PIDSelfTune.SUT.State   |
| sParamCalc.i_Event_TIR    | PIDSelfTune.TIR.State   |
| sRet.i_Mode               | sRet.i_Mode ist entfallen. Die Betriebsart wird durch Mode und ModeActivate gewechselt. |
| sRet.r_Ctrl_Gain          | Retain.CtrlParams.Gain  |
| sRet.r_Ctrl_Ti            | Retain.CtrlParams.Ti  |
| sRet.r_Ctrl_Td            | Retain.CtrlParams.Td  |
| sRet.r_Ctrl_A             | Retain.CtrlParams.TdFiltRatio   |
| sRet.r_Ctrl_B             | Retain.CtrlParams.PWeighting  |
| sRet.r_Ctrl_C             | Retain.CtrlParams.DWeighting  |
| sRet.r_Ctrl_Cycle         | Retain.CtrlParams.Cycle   |

## Parameter State und Mode V2

### Zusammenhang der Parameter

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Mit einer steigenden Flanke an ModeActivate wechselt PID\_Compact in die Betriebsart, die am Durchgangsparameter Mode gespeichert ist.

Wenn die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID\_Compact in der Betriebsart, die an Mode gespeichert ist. Um PID\_Compact in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

### Bedeutung der Werte

| State / Mode | Beschreibung der Betriebsart  |
|--------------|---|
| 0            | <p>Inaktiv</p> <p>In der Betriebsart "Inaktiv" wird immer der Ausgangswert 0.0 ausgegeben, unabhängig von Config.OutputUpperLimit und Config.OutputLowerLimit. Die Pulsweitenmodulation ist ausgeschaltet.</p>  |
| 1            | <p>Erstoptimierung</p> <p>Die Erstoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.</p> <p>Voraussetzungen für die Erstoptimierung:</p> <ul style="list-style-type: none"> <li>• Betriebsart Inaktiv (State = 0), Handbetrieb (State = 4) oder Automatikbetrieb (State = 3)</li> <li>• ManualEnable = FALSE</li> <li>• Reset = FALSE</li> <li>• Der Istwert darf nicht zu nah am Sollwert sein.<br/> <math> Setpoint - Input  &gt; 0.3 *  Config.InputUpperLimit - Config.InputLowerLimit </math> und<br/> <math> Setpoint - Input  &gt; 0.5 *  Setpoint </math></li> <li>• Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen.</li> </ul> <p>Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen.</p> <p>Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt:</p> <ul style="list-style-type: none"> <li>• <math>Setpoint &gt; CurrentSetpoint + CancelTuningLevel</math><br/>oder</li> <li>• <math>Setpoint &lt; CurrentSetpoint - CancelTuningLevel</math></li> </ul> <p>Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit LoadBackUp reaktiviert werden.</p> <p>Nach erfolgreicher Erstoptimierung wird in den Automatikbetrieb geschaltet; nach erfolgloser Erstoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode.</p> <p>Die Phase der Erstoptimierung wird mit PIDSelfTune.SUT.State angezeigt.</p> |

| State / Mode | Beschreibung der Betriebsart  |
|--------------|---|
| 2            | <p>Nachoptimierung</p> <p>Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstopoptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen. PID_Compact versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwertes. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwertes beeinflusst.</p> <p>Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt:</p> <ul style="list-style-type: none"> <li>• Setpoint &gt; CurrentSetpoint + CancelTuningLevel<br/>oder</li> <li>• Setpoint &lt; CurrentSetpoint - CancelTuningLevel</li> </ul> <p>Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit LoadBackUp reaktiviert werden.</p> <p>Voraussetzungen für Nachoptimierung:</p> <ul style="list-style-type: none"> <li>• Es werden keine Störungen erwartet.</li> <li>• Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen</li> <li>• ManualEnable = FALSE</li> <li>• Reset = FALSE</li> <li>• Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4)</li> </ul> <p>Die Nachoptimierung verläuft folgendermaßen beim Start aus:</p> <ul style="list-style-type: none"> <li>• Automatikbetrieb (State = 3)<br/>Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.<br/>PID_Compact regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.</li> <li>• Inaktiv (State = 0) oder Handbetrieb (State = 4)<br/>Wenn die Voraussetzungen für eine Erstopoptimierung erfüllt sind, wird eine Erstopoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind.<br/>Wenn sich der Istwert für eine Erstopoptimierung bereits zu nah am Sollwert befindet oder PIDSelfTune.TIR.RunIn = TRUE, wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen.<br/>Erst dann startet die Nachoptimierung.</li> </ul> <p>Nach erfolgreicher Nachoptimierung wechselt der Regler in den Automatikbetrieb; nach erfolgloser Nachoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode.</p> <p>Die Phase der "Nachoptimierung" wird mit PIDSelfTune.TIR.State angezeigt.</p> |
| 3            | <p>Automatikbetrieb</p> <p>Im Automatikbetrieb regelt PID_Compact die Regelstrecke entsprechend der vorgegebenen Parameter aus. Wenn eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt:</p> <ul style="list-style-type: none"> <li>• Erstopoptimierung erfolgreich abgeschlossen</li> <li>• Nachoptimierung erfolgreich abgeschlossen</li> <li>• Ändern des Durchgangsparameters Mode auf den Wert 3 und eine steigende Flanke an ModeActivate.</li> </ul> <p>Der Wechsel vom Automatikbetrieb in den Handbetrieb erfolgt nur im Inbetriebnahmeditor stoßfrei.</p> <p>Im Automatikbetrieb wird die Variable ActivateRecoverMode berücksichtigt.</p>   |

| State / Mode | Beschreibung der Betriebsart   |
|--------------|--|
| 4            | <p>Handbetrieb</p> <p>Im Handbetrieb geben Sie einen manuellen Ausgangswert am Parameter ManualValue vor.</p> <p>Diese Betriebsart können Sie zusätzlich über ManualEnable = TRUE aktivieren. Es wird empfohlen, die Betriebsarten nur über Mode und ModeActivate zu wechseln.</p> <p>Der Wechsel vom Handbetrieb in den Automatikbetrieb erfolgt stoßfrei. Der Handbetrieb ist auch möglich, wenn ein Fehler ansteht.</p>   |
| 5            | <p>Ersatzausgangswert mit Fehlerüberwachung</p> <p>Der Regelalgorithmus ist ausgeschaltet. Die Variable SetSubstituteOutput bestimmt, welcher Ausgangswert während dieser Betriebsart ausgegeben wird.</p> <ul style="list-style-type: none"> <li>• SetSubstituteOutput = FALSE: letzter gültiger Ausgangswert</li> <li>• SetSubstituteOutput = TRUE: Ersatzausgangswert</li> </ul> <p>Diese Betriebsart können Sie nicht mit Mode = 5 aktivieren.</p> <p>Sie wird im Fehlerfall statt der Betriebsart "Inaktiv" aktiviert, wenn alle folgenden Bedingungen erfüllt sind:</p> <ul style="list-style-type: none"> <li>• Automatikbetrieb (Mode = 3)</li> <li>• ActivateRecoverMode = TRUE</li> <li>• Ein oder mehrere Fehler sind aufgetreten, bei denen ActivateRecoverMode wirkt.</li> </ul> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_Compact wieder in den Automatikbetrieb.</p> |

### ENO-Verhalten

Wenn State = 0 ist, dann ist ENO = FALSE.

Wenn State ≠ 0 ist, dann ist ENO = TRUE.

### Automatische Betriebsartwechsel während der Inbetriebnahme

Nach erfolgreicher Erst- oder Nachoptimierung wird der Automatikbetrieb aktiviert. Die folgende Tabelle zeigt, wie sich Mode und State während einer erfolgreichen Erstopoptimierung ändern.

| Zyklus-Nr. | Mode | State | Aktion  |
|------------|------|-------|---|
| 0          | 4    | 4     | Mode = 1 setzen   |
| 1          | 1    | 4     | ModeActivate = TRUE setzen  |
| 1          | 4    | 1     | Wert von State wird an Mode gespeichert<br>Erstopoptimierung wird gestartet |
| n          | 4    | 1     | Erstopoptimierung erfolgreich beendet                                       |
| n          | 3    | 3     | Automatikbetrieb wird gestartet   |

Im Fehlerfall wechselt PID\_Compact automatisch die Betriebsart. Die folgende Tabelle zeigt, wie sich Mode und State während einer fehlerhaften Erstopoptimierung ändern.

| Zyklus-Nr. | Mode | State | Aktion                     |
|------------|------|-------|----------------------------|
| 0          | 4    | 4     | Mode = 1 setzen            |
| 1          | 1    | 4     | ModeActivate = TRUE setzen |

| Zyklus-Nr. | Mode | State | Aktion  |
|------------|------|-------|---|
| 1          | 4    | 1     | Wert von State wird an Mode gespeichert<br>Erstoptimierung wird gestartet |
| n          | 4    | 1     | Erstoptimierung abgebrochen   |
| n          | 4    | 4     | Handbetrieb wird gestartet  |

Wenn ActivateRecoverMode = TRUE ist, wird die Betriebsart aktiviert, die an Mode gespeichert ist. Beim Start der Erst- oder Nachoptimierung hat PID\_Compact den Wert von State am Durchgangsparameter Mode gespeichert. PID\_Compact wechselt also in die Betriebsart, aus der die Optimierung gestartet wurde.

Wenn ActivateRecoverMode = FALSE ist, wird in die Betriebsart "Inaktiv" gewechselt.

### Siehe auch

Ausgangsparameter PID\_Compact V2 (Seite 2681)

### Parameter ErrorBits V2

Stehen gleichzeitig mehrere Fehler an, so werden die Werte der ErrorBits binär addiert angezeigt. Wird z. B. ErrorBits = 0003h angezeigt, so stehen gleichzeitig die Fehler 0001h und 0002h an.

PID\_Compact verwendet im Handbetrieb ManualValue als Ausgangswert. Ausnahme ist Errorbits = 10000h.

| ErrorBits<br>(DW#16#...) | Beschreibung   |
|--------------------------|--|
| 0000                     | Es steht kein Fehler an.   |
| 0001                     | Der Parameter "Input" ist außerhalb der Istwertgrenzen. <ul style="list-style-type: none"> <li>• Input &gt; Config.InputUpperLimit oder</li> <li>• Input &lt; Config.InputLowerLimit</li> </ul> Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Compact im Automatikbetrieb.<br>Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.                         |
| 0002                     | Ungültiger Wert am Parameter "Input_PER". Überprüfen Sie, ob am Analogeingang ein Fehler ansteht.<br>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Compact den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb.<br>Wenn vor Auftreten des Fehlers die Betriebsart Erstoptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist. |
| 0004                     | Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden.<br>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.   |

| ErrorBits<br>(DW#16#...) | Beschreibung  |
|--------------------------|---|
| 0008                     | <p>Fehler beim Starten der Erstopptimierung. Der Istwert ist zu nahe am Sollwert. Starten Sie die Nachoptimierung.</p> <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.</p>  |
| 0010                     | <p>Der Sollwert wurde während der Optimierung verändert.</p> <p>An der Variable CancelTuningLevel können Sie die zulässige Schwankung des Sollwerts einstellen.</p> <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.</p>   |
| 0020                     | <p>Die Erstopptimierung ist während der Nachoptimierung nicht erlaubt.</p> <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bleibt PID_Compact in der Betriebsart Nachoptimierung.</p>   |
| 0080                     | <p>Fehler bei der Erstopptimierung. Die Ausgangswertgrenzen sind nicht korrekt konfiguriert.</p> <p>Überprüfen Sie, ob die Grenzen des Ausgangswerts korrekt konfiguriert sind und zum Regelsinn passen.</p> <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.</p>  |
| 0100                     | <p>Fehler während der Nachoptimierung führte zu ungültigen Parametern.</p> <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.</p>  |
| 0200                     | <p>Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Compact den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erstopptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.</p>     |
| 0400                     | <p>Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, gibt PID_Compact den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erstopptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.</p> |
| 0800                     | <p>Abtastzeitfehler: PID_Compact wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_Compact im Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erstopptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.</p>   |
| 1000                     | <p>Ungültiger Wert am Parameter "Setpoint": Wert hat kein gültiges Zahlenformat.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv und ActivateRecoverMode = TRUE waren, gibt PID_Compact den konfigurierten Ersatzausgangswert aus. Sobald der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erstopptimierung oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist.</p>    |

| ErrorBits<br>(DW#16#...) | Beschreibung  |
|--------------------------|---|
| 10000                    | Ungültiger Wert am Parameter ManualValue. Wert hat kein gültiges Zahlenformat.<br>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, verwendet PID_Compact SubstituteOutput als Ausgangswert. Sobald Sie einen gültigen Wert an ManualValue vorgeben, verwendet PID_Compact diesen als Ausgangswert.  |
| 20000                    | Ungültiger Wert an der Variablen SubstituteOutput. Wert hat kein gültiges Zahlenformat.<br>PID_Compact verwendet die Untergrenze Ausgangswert als Ausgangswert.<br>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_Compact wieder in den Automatikbetrieb.  |
| 40000                    | Ungültiger Wert am Parameter Disturbance. Wert hat kein gültiges Zahlenformat.<br>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, wird Disturbance auf Null gesetzt. PID_Compact bleibt im Automatikbetrieb.<br>Wenn vor Auftreten des Fehlers die Betriebsart Erst- oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_Compact in die Betriebsart, die an Mode gespeichert ist. Wenn Disturbance in der aktuellen Phase keinen Einfluss auf den Ausgangswert hat, wird die Optimierung nicht abgebrochen. |

### Variable ActivateRecoverMode V2

Die Variable ActivateRecoverMode bestimmt das Verhalten im Fehlerfall. Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind.

### Automatikbetrieb

| <b>ACHTUNG</b>   |
|--|
| <p><b>Ihre Anlage kann beschädigt werden.</b></p> <p>Wenn ActivateRecoverMode = TRUE ist, bleibt PID_Compact im Fehlerfall auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden.</p> <p>Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.</p> |

| ActivateRecover Mode | Beschreibung  |
|----------------------|---|
| FALSE                | Im Fehlerfall wechselt PID_Compact in die Betriebsart "Inaktiv". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.   |
| TRUE                 | <p>Wenn im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert, da PID_Compact bei jedem Fehlerfall zwischen berechnetem Ausgangswert und dem Ersatzausgangswert wechselt. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache.</p> <p>Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID_Compact im Automatikbetrieb:</p> <ul style="list-style-type: none"> <li>• 0001h: Der Parameter "Input" ist außerhalb der Istwertgrenzen.</li> <li>• 0800h: Abtastzeitfehler</li> <li>• 40000h: Ungültiger Wert am Parameter Disturbance.</li> </ul> <p>Wenn einer oder mehrere der folgenden Fehler auftreten, wechselt PID_Compact in die Betriebsart " Ersatzausgangswert mit Fehlerüberwachung":</p> <ul style="list-style-type: none"> <li>• 0002h: Ungültiger Wert am Parameter Input_PER.</li> <li>• 0200h: Ungültiger Wert am Parameter Input.</li> <li>• 0400h: Berechnung des Ausgangswerts fehlgeschlagen.</li> <li>• 1000h: Ungültiger Wert am Parameter Setpoint.</li> </ul> <p>Wenn der folgende Fehler auftritt, wechselt PID_Compact in die Betriebsart " Ersatzausgangswert mit Fehlerüberwachung" und fährt das Stellglied auf Config.OutputLowerLimit:</p> <ul style="list-style-type: none"> <li>• 20000h: Ungültiger Wert an der Variable SubstituteOutput. Wert hat kein gültiges Zahlenformat.</li> </ul> <p>Dieses Verhalten ist unabhängig von SetSubstituteOutput.</p> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_Compact wieder in den Automatikbetrieb.</p> |

### Erstoptimierung und Nachoptimierung

| ActivateRecover Mode | Beschreibung   |
|----------------------|--|
| FALSE                | Im Fehlerfall wechselt PID_Compact in die Betriebsart "Inaktiv". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.  |
| TRUE                 | <p>Wenn der folgende Fehler auftritt, bleibt PID_Compact in der aktiven Betriebsart:</p> <ul style="list-style-type: none"> <li>• 0020h: Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt.</li> </ul> <p>Die folgenden Fehler werden ignoriert:</p> <ul style="list-style-type: none"> <li>• 10000h: Ungültiger Wert am Parameter ManualValue.</li> <li>• 20000h: Ungültiger Wert an der Variable SubstituteOutput.</li> </ul> <p>Bei allen anderen Fehlern bricht PID_Compact die Optimierung ab und wechselt in die Betriebsart, aus der die Optimierung gestartet wurde.</p> |

### Handbetrieb

Im Handbetrieb wirkt ActivateRecoverMode nicht.



## Variable Warning V2

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Variable Warning binär addiert angezeigt. Wenn z. B. die Warnung 0003h angezeigt, so stehen gleichzeitig die Warnungen 0001h und 0002h an.

| Warning<br>(DW#16#...) | Beschreibung  |
|------------------------|---|
| 0000                   | Es liegt keine Warnung vor.   |
| 0001                   | Während der Erstopoptimierung wurde der Wendepunkt nicht gefunden.  |
| 0004                   | Der Sollwert wurde begrenzt auf die eingestellten Grenzen.  |
| 0008                   | Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode TIR.TuneRule = 3 berechnet. |
| 0010                   | Die Betriebsart konnte nicht geändert werden, da Reset = TRUE oder ManualEnable = TRUE.   |
| 0020                   | Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.                     |
| 0040                   | Der Istwert hat eine seiner Warngrenzen überschritten.  |
| 0080                   | Ungültiger Wert an Mode. Die Betriebsart wird nicht gewechselt.   |
| 0100                   | Der Handwert wurde begrenzt auf die Grenzen des Reglerausgangs.   |
| 0200                   | Die angegebene Regel zur Optimierung wird nicht unterstützt. Es werden keine PID-Parameter berechnet.   |
| 1000                   | Der Ersatzausgangswert kann nicht erreicht werden, da er außerhalb der Ausgangswertgrenzen liegt.   |

Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist:

- 0001h
- 0004h
- 0008h
- 0040h
- 0100h

Alle anderen Warnungen werden bei steigender Flanke an Reset oder ErrorAck gelöscht.

## PID\_Compact V1

### Beschreibung PID\_Compact V1

#### Beschreibung

Die Anweisung PID\_Compact stellt einen PID-Regler mit integrierter Optimierung für Automatik- und Handbetrieb zur Verfügung.

#### Aufruf

PID\_Compact wird im konstanten Zeitraster der Zykluszeit des aufrufenden OBs aufgerufen (vorzugsweise in einem Weckalarm-OB).

## Laden in Gerät

Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PID\_Compact vollständig laden.

Technologieobjekte in Gerät laden (Seite 4056)

## Anlauf

Bei Anlauf der CPU startet PID\_Compact in der zuletzt aktiven Betriebsart. Um PID\_Compact in der Betriebsart "Inaktiv" zu belassen, setzen Sie sb\_RunModeByStartup = FALSE.

## Überwachung der Abtastzeit PID\_Compact

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden OB. Die Anweisung PID\_Compact misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (Error = 0800 hex) und PID\_Compact wechselt in die Betriebsart "Inaktiv"

Folgende Bedingungen versetzen PID\_Compact ab Version 1.1 während der Optimierung in die Betriebsart "Inaktiv":

- Neuer Mittelwert  $\geq 1,1 \times$  alter Mittelwert
- Neuer Mittelwert  $\leq 0,9 \times$  alter Mittelwert

Folgende Bedingungen versetzen PID\_Compact ab Version 1.1 bei Automatikbetrieb in die Betriebsart "Inaktiv":

- Neuer Mittelwert  $\geq 1,5 \times$  alter Mittelwert
- Neuer Mittelwert  $\leq 0,5 \times$  alter Mittelwert

Folgende Bedingungen versetzen PID\_Compact 1.0 während der Optimierung und bei Automatikbetrieb in die Betriebsart "Inaktiv":

- Neuer Mittelwert  $\geq 1,1 \times$  alter Mittelwert
- Neuer Mittelwert  $\leq 0,9 \times$  alter Mittelwert
- Aktuelle Abtastzeit  $\geq 1,5 \times$  aktueller Mittelwert
- Aktuelle Abtastzeit  $\leq 0,5 \times$  aktueller Mittelwert

## Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID\_Compact werden bei jedem Aufruf durchgeführt.

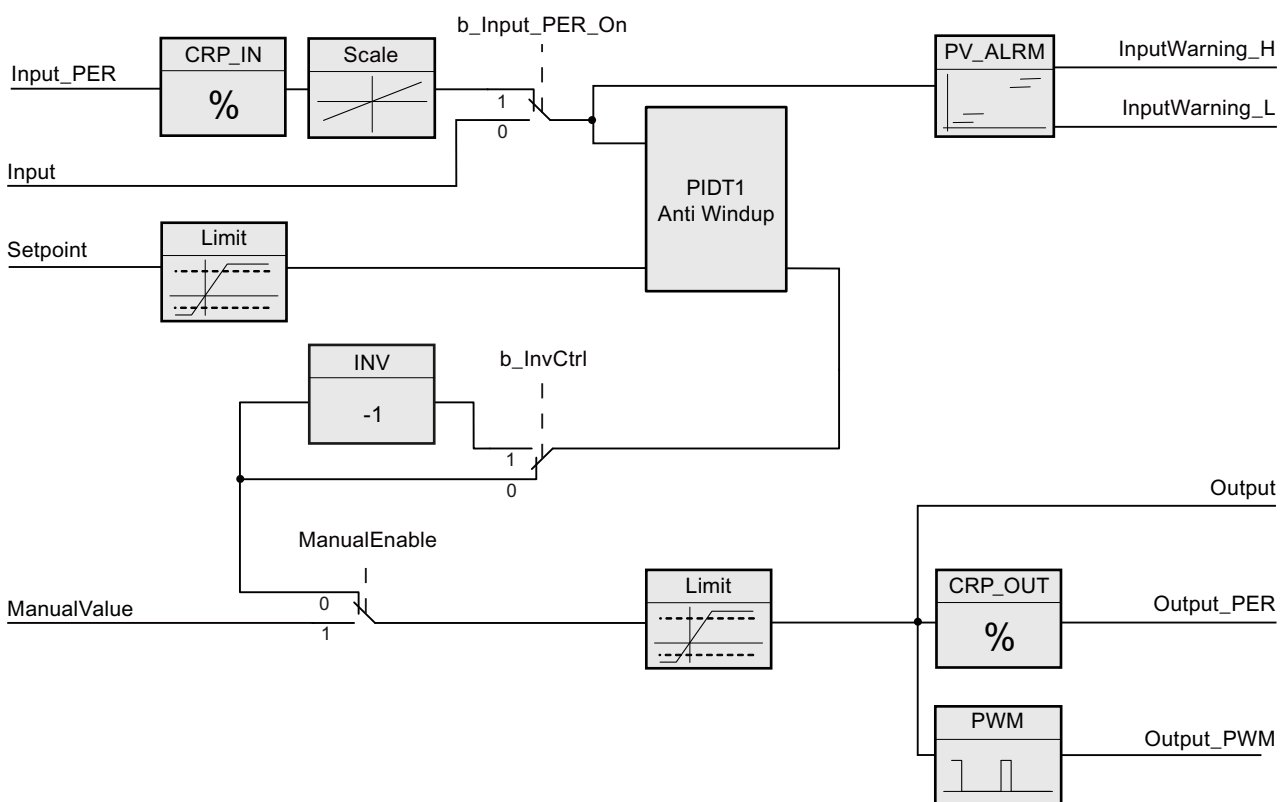
## PID-Algorithmus

PID\_Compact ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der Ausgangswert wird anhand folgender Formel berechnet.

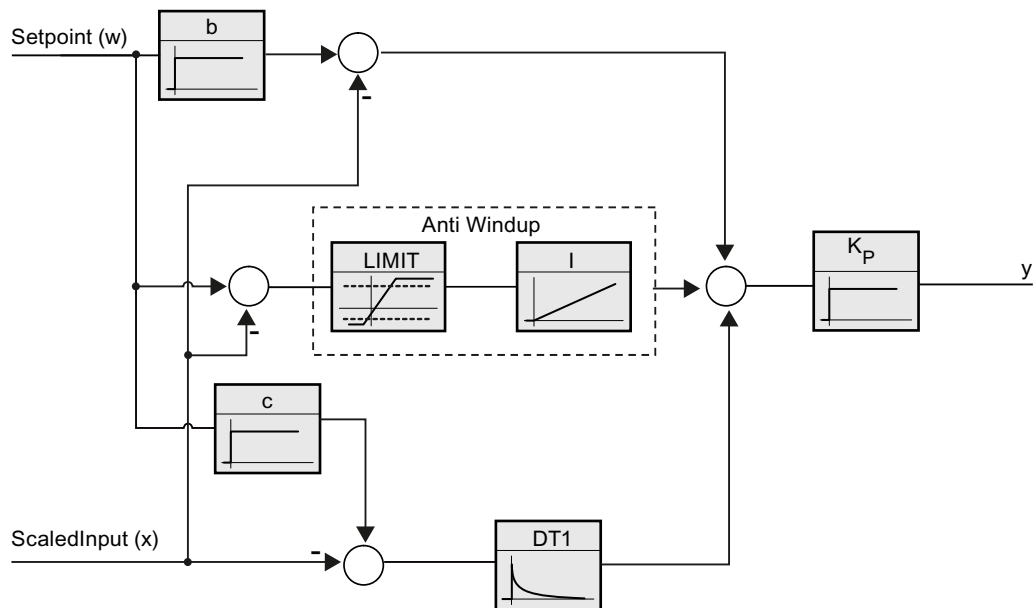
$$y = K_p \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

| Symbol         | Beschreibung  |
|----------------|---|
| y              | Ausgangswert  |
| K <sub>p</sub> | Proportionalverstärkung   |
| s              | Laplace-Operator  |
| b              | Gewichtung des P-Anteils  |
| w              | Sollwert  |
| x              | Istwert   |
| T <sub>i</sub> | Integrationszeit  |
| a              | Koeffizient für den Differenzierverzug (T1 = a × T <sub>D</sub> ) |
|                | Differenzierzeit  |
| c              | Gewichtung des D-Anteils  |

## Blockschaltbild PID\_Compact



**Blockschaltbild PIDT1 mit Anti-Windup**



**Verhalten im Fehlerfall**

Wenn Fehler auftreten, werden diese am Parameter Error ausgegeben und PID\_Compact wechselt in die Betriebsart "Inaktiv". Mit dem Parameter Reset setzen Sie die Fehler wieder zurück.

**Regelsinn**

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn. Für Kühlungen und Abflussregelungen kann es notwendig sein, den Regelsinn zu invertieren. PID\_Compact arbeitet nicht mit negativer Proportionalverstärkung. Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts. Der Regelsinn wird auch während Erst- und Nachoptimierung berücksichtigt.

**Siehe auch**

Regelungsart (Seite 4079)

## Eingangsparameter PID\_Compact V1

Tabelle 9-81

| Parameter    | Datentyp | Vorbelegung | Beschreibung   |
|--------------|----------|-------------|--|
| Setpoint     | REAL     | 0.0         | Sollwert des PID-Reglers im Automatikbetrieb   |
| Input        | REAL     | 0.0         | Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet.<br>Wenn Sie den Parameter Input verwenden, muss sPid_Cmpt.b_Input_PER_On = FALSE sein.   |
| Input_PER    | WORD     | W#16#0      | Analogeingang als Quelle des Istwerts<br>Wenn Sie den Parameter Input_PER verwenden, muss sPid_Cmpt.b_Input_PER_On = TRUE sein.  |
| ManualEnable | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>Flanke FALSE -&gt; TRUE wählt die Betriebsart "Handbetrieb", State = 4, sRet.i_Mode bleibt unverändert.</li> <li>Flanke TRUE -&gt; FALSE wählt die zuletzt aktive Betriebsart, State =sRet.i_Mode</li> </ul> <p>Während ManualEnable = TRUE wirkt sich eine Änderung von sRet.i_Mode nicht aus. Erst bei der Flanke TRUE -&gt; FALSE an ManualEnable wird die Änderung von sRet.i_Mode berücksichtigt.</p> <p><b>PID_Compact V1.2 und PID_Compact V1.0</b><br/>Wenn beim Start der CPU ManualEnable = TRUE, startet PID_Compact im Handbetrieb. Eine steigende Flanke (FALSE &gt; TRUE) an ManualEnable ist nicht notwendig.</p> <p><b>PID_Compact V1.1</b><br/>Beim Start der CPU schaltet PID_Compact nur bei einer steigenden Flanke (FALSE-&gt;TRUE) an ManualEnable in den Handbetrieb. Ohne steigende Flanke startet PID_Compact in der letzten Betriebsart bei der ManualEnable FALSE war.</p> |
| ManualValue  | REAL     | 0.0         | Handwert<br>Dieser Wert wird im Handbetrieb als Ausgangswert verwendet.  |
| Reset        | BOOL     | FALSE       | Der Parameter Reset (Seite 2713) führt einen Neustart des Reglers durch.   |

## Ausgangsparameter PID\_Compact V1

Tabelle 9-82

| Parameter  | Datentyp | Vorbelegung | Beschreibung   |
|--|----------|-------------|--|
| ScaledInput  | REAL     | 0.0         | Ausgabe des skalierten Istwerts  |
| Die Ausgänge "Output", "Output_PER" und "Output_PWM" können parallel genutzt werden. |          |             |  |
| Output   | REAL     | 0.0         | Ausgangswert im REAL-Format  |
| Output_PER   | WORD     | W#16#0      | Analoger Ausgangswert  |
| Output_PWM   | BOOL     | FALSE       | Pulsweitenmodulierter Ausgangswert<br>Der Ausgangswert wird über variable Ein- und Ausschaltzeiten gebildet. |

| Parameter       | Datentyp | Vorbelegung | Beschreibung  |
|-----------------|----------|-------------|---|
| SetpointLimit_H | BOOL     | FALSE       | Wenn SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht. In der CPU wird der Sollwert auf die konfigurierte absolute Obergrenze des Sollwerts begrenzt. Als Obergrenze Sollwert ist die konfigurierte absolute Obergrenze Istwert vorbelegt.<br>Wenn Sie sPid_Cmpt.r_Sp_Hlm auf einen Wert innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.      |
| SetpointLimit_L | BOOL     | FALSE       | Wenn SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht. In der CPU wird der Sollwert auf die konfigurierte absolute Untergrenze des Sollwerts begrenzt. Als Untergrenze Sollwert ist die konfigurierte absolute Untergrenze Istwert vorbelegt.<br>Wenn Sie sPid_Cmpt.r_Sp_Llm auf einen Wert innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet. |
| InputWarning_H  | BOOL     | FALSE       | Wenn InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten.  |
| InputWarning_L  | BOOL     | FALSE       | Wenn InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten.  |
| State           | INT      | 0           | Der Parameter State (Seite 2709) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit der Variable sRet.i_Mode. <ul style="list-style-type: none"> <li>• State = 0: Inaktiv</li> <li>• State = 1: Erstopptimierung</li> <li>• State = 2: Nachoptimierung</li> <li>• State = 3: Automatikbetrieb</li> <li>• State = 4: Handbetrieb</li> </ul>  |
| Error           | DWORD    | W#16#0      | Der Parameter Error (Seite 2712) zeigt die Fehlermeldungen.<br>Error = 0000: Es steht kein Fehler an.   |

### Statische Variablen PID\_Compact V1

Nicht aufgeführte Variablen dürfen Sie nicht verändern. Diese werden nur intern verwendet.

Tabelle 9-83

| Variable            | Datentyp | Vorbelegung | Beschreibung   |
|---------------------|----------|-------------|--|
| sb_GetCycleTime     | BOOL     | TRUE        | Wenn sb_GetCycleTime = TRUE, wird die automatische Ermittlung der Zykluszeit gestartet. Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE.   |
| sb_EnCyclEstimation | BOOL     | TRUE        | Wenn sb_EnCyclEstimation = TRUE, wird die Abtastzeit PID_Compact berechnet.  |
| sb_EnCyclMonitoring | BOOL     | TRUE        | Wenn sb_EnCyclMonitoring = FALSE, wird die Abtastzeit PID_Compact nicht überwacht. Wenn PID_Compact nicht innerhalb der Abtastzeit ausgeführt werden kann, wird kein Fehler 0800 ausgegeben und PID_Compact wechselt nicht in die Betriebsart "Inaktiv". |

| Variable            | Datentyp | Vorbelegung | Beschreibung   |
|---------------------|----------|-------------|--|
| sb_RunModeByStartup | BOOL     | TRUE        | Nach CPU Neustart letzte Betriebsart aktivieren<br>Wenn sb_RunModeByStartup = FALSE, bleibt der Regler nach CPU-Anlauf inaktiv.<br>Wenn sb_RunModeByStartup = TRUE, kehrt der Regler nach CPU-Anlauf in die zuletzt aktive Betriebsart zurück.   |
| si_Unit             | INT      | 0           | Physikalische Einheit des Ist- und Sollwerts, z. B. °C oder °F.  |
| si_Type             | INT      | 0           | Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur   |
| sd_Warning          | DWORD    | DW#16#0     | Die Variable sd_warning (Seite 2714) zeigt die Warnungen seit Reset oder dem letzten Wechsel der Betriebsart.  |
| sBackUp.r_Gain      | REAL     | 1.0         | gespeicherte Proportionalverstärkung<br>Werte aus der Struktur sBackUp können mit sPid_Cmpt.b_LoadBackUp = TRUE wieder geladen werden.   |
| sBackUp.r_Ti        | REAL     | 20.0        | gespeicherte Integrationszeit [s]  |
| sBackUp.r_Td        | REAL     | 0.0         | gespeicherte Differenzierzeit [s]  |
| sBackUp.r_A         | REAL     | 0.0         | gespeicherter Koeffizient Differenzierverszug  |
| sBackUp.r_B         | REAL     | 0.0         | gespeicherter Gewichtungsfaktor des P-Anteils  |
| sBackUp.r_C         | REAL     | 0.0         | gespeicherter Gewichtungsfaktor des D-Anteils  |
| sBackUp.r_Cycle     | REAL     | 1.0         | gespeicherte Abtastzeit PID-Algorithmus  |
| sPid_Calc.r_Cycle   | REAL     | 0.1         | Abtastzeit der Anweisung PID_Compact<br>r_Cycle wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.   |
| sPid_Calc.b_RunIn   | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>• b_RunIn = FALSE<br/>Wenn die Nachoptimierung aus der Betriebsart Inaktiv oder Handbetrieb gestartet wird, wird eine Erstopoptimierung gestartet. Wenn die Voraussetzungen für Erstopoptimierung nicht erfüllt sind, verhält sich PID_Compact wie bei b_RunIn = TRUE.<br/>Wenn die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt.<br/>Erst dann startet die Nachoptimierung. Ist die Erstopoptimierung nicht möglich, wechselt PID_Compact in die Betriebsart "Inaktiv".</li> <li>• b_RunIn = TRUE<br/>Die Erstopoptimierung wird übersprungen. PID_3Compact versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet anschließend automatisch.<br/>b_RunIn wird nach der Nachoptimierung auf FALSE gesetzt.</li> </ul> |

| Variable                 | Datentyp | Vorbelegung   | Beschreibung  |
|--------------------------|----------|---------------|---|
| sPid_Calc.b_CalcParamSUT | BOOL     | FALSE         | Wenn b_CalcParamSUT = TRUE, werden die Parameter für die Erstopoptimierung neu berechnet. Dadurch kann die Methode für die Parameterberechnung geändert werden, ohne die Optimierung zu wiederholen.<br>b_CalcParamSUT wird nach der Berechnung auf FALSE gesetzt.  |
| sPid_Calc.b_CalcParamTIR | BOOL     | FALSE         | Wenn b_CalcParamTIR = TRUE, werden die Parameter für die Nachoptimierung neu berechnet. Dadurch kann die Methode für die Parameterberechnung geändert werden, ohne die Optimierung zu wiederholen.<br>b_CalcParamTIR wird nach der Berechnung auf FALSE gesetzt.  |
| sPid_Calc.i_CtrlTypeSUT  | INT      | 0             | Parameter während Erstopoptimierung berechnen nach Methode: <ul style="list-style-type: none"> <li>• i_CtrlTypeSUT = 0: PID nach Chien, Hrones und Reswick</li> <li>• i_CtrlTypeSUT = 1: PI nach Chien, Hrones und Reswick</li> </ul>   |
| sPid_Calc.i_CtrlTypeTIR  | INT      | 0             | Parameter während Nachoptimierung berechnen nach Methode: <ul style="list-style-type: none"> <li>• i_CtrlTypeTIR = 0: PID automatisch</li> <li>• i_CtrlTypeTIR = 1: PID schnell</li> <li>• i_CtrlTypeTIR = 2: PID langsam</li> <li>• i_CtrlTypeTIR = 3: Ziegler-Nichols PID</li> <li>• i_CtrlTypeTIR = 4: Ziegler-Nichols PI</li> <li>• i_CtrlTypeTIR = 5: Ziegler-Nichols P</li> </ul> |
| sPid_Calc.r_Progress     | REAL     | 0.0           | Fortschritt der Optimierung in Prozent (0.0 - 100.0)  |
| sPid_Cmpt.r_Sp_Hlm       | REAL     | +3.402822e+38 | Obergrenze des Sollwerts<br>Wenn Sie sPid_Cmpt.r_Sp_Hlm außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert verwendet.<br>Wenn Sie sPid_Cmpt.r_Sp_Hlm innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.   |
| sPid_Cmpt.r_Sp_Llm       | REAL     | -3.402822e+38 | Untergrenze des Sollwerts<br>Wenn Sie sPid_Cmpt.r_Sp_Llm außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert verwendet.<br>Wenn Sie sPid_Cmpt.r_Sp_Llm innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.   |



| Variable                  | Datentyp | Vorbelegung | Beschreibung   |
|---------------------------|----------|-------------|--|
| sPid_Cmpt.r_Pv_Norm_IN_1  | REAL     | 0.0         | Skalierung Input_PER Unten<br>Anhand der zwei Wertepaare r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 und r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 der Struktur sPid_Cmpt wird Input_PER in Prozent umgerechnet. |
| sPid_Cmpt.r_Pv_Norm_IN_2  | REAL     | 27648.0     | Skalierung Input_PER Oben<br>Anhand der zwei Wertepaare r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 und r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 der Struktur sPid_Cmpt wird Input_PER in Prozent umgerechnet.  |
| sPid_Cmpt.r_Pv_Norm_OUT_1 | REAL     | 0.0         | Skalierter unterer Istwert<br>Anhand der zwei Wertepaare r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 und r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 der Struktur sPid_Cmpt wird Input_PER in Prozent umgerechnet. |
| sPid_Cmpt.r_Pv_Norm_OUT_2 | REAL     | 100.0       | Skalierter oberer Istwert<br>Anhand der zwei Wertepaare r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 und r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 der Struktur sPid_Cmpt wird Input_PER in Prozent umgerechnet.  |
| sPid_Cmpt.r_Lmn_Hlm       | REAL     | 100.0       | Obere Ausgangswertgrenze für den Ausgangsparameter "Output"  |
| sPid_Cmpt.r_Lmn_Llm       | REAL     | 0.0         | Untere Ausgangswertgrenze für den Ausgangsparameter "Output"   |
| sPid_Cmpt.b_Input_PER_On  | BOOL     | TRUE        | Wenn b_Input_PER_On = TRUE, wird der Parameter Input_PER verwendet. Wenn b_Input_PER_On = FALSE, wird der Parameter Input verwendet.   |
| sPid_Cmpt.b_LoadBackup    | BOOL     | FALSE       | Aktivieren des Backup Parametersatzes. Sollte eine Optimierung fehlgeschlagen sein, können Sie durch Setzen dieses Bits die vorigen PID Parameter wieder reaktivieren.                     |
| sPid_Cmpt.b_InvCtrl       | BOOL     | FALSE       | Invertieren des Regelsinns<br>Wenn b_InvCtrl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts.  |
| sPid_Cmpt.r_Lmn_Pwm_PPTm  | REAL     | 0.0         | Minimale Einschaltzeit der Pulsweitenmodulation in Sekunden wird gerundet auf<br>$r\_Lmn\_Pwm\_PPTm = r\_Cycle$ oder<br>$r\_Lmn\_Pwm\_PPTm = n * r\_Cycle$                                 |
| sPid_Cmpt.r_Lmn_Pwm_PBTm  | REAL     | 0.0         | Minimale Ausschaltzeit der Pulsweitenmodulation in Sekunden wird gerundet auf<br>$r\_Lmn\_Pwm\_PBTm = r\_Cycle$ oder<br>$r\_Lmn\_Pwm\_PBTm = n * r\_Cycle$                                 |

| Variable               | Datentyp | Vorbelegung   | Beschreibung   |
|------------------------|----------|---------------|--|
| sPid_Cmpt.r_Pv_Hlm     | REAL     | 120.0         | Obergrenze des Istwerts<br>Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird keine Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_Compact wechselt in die Betriebsart "Inaktiv".<br>$r\_Pv\_Hlm > r\_Pv\_Llm$   |
| sPid_Cmpt.r_Pv_Llm     | REAL     | 0.0           | Untergrenze des Istwerts<br>$r\_Pv\_Llm < r\_Pv\_Hlm$  |
| sPid_Cmpt.r_Pv_HWrn    | REAL     | +3.402822e+38 | Obere Warngrenze des Istwerts<br>Wenn Sie $r\_Pv\_HWrn$ außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obere Warngrenze verwendet.<br>Wenn Sie $r\_Pv\_HWrn$ innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obere Warngrenze verwendet.<br>$r\_Pv\_HWrn > r\_Pv\_LWrn$<br>$r\_Pv\_HWrn \leq r\_Pv\_Hlm$   |
| sPid_Cmpt.r_Pv_LWrn    | REAL     | -3.402822e+38 | Untere Warngrenze des Istwerts<br>Wenn Sie $r\_Pv\_LWrn$ außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untere Warngrenze verwendet.<br>Wenn Sie $r\_Pv\_LWrn$ innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet.<br>$r\_Pv\_LWrn < r\_Pv\_HWrn$<br>$r\_Pv\_LWrn \geq r\_Pv\_LWrn$  |
| sParamCalc.i_Event_SUT | INT      | 0             | Die Variable $i\_Event\_SUT$ (Seite 2715) zeigt die aktuelle Phase der "Erstoptimierung":  |
| sParamCalc.i_Event_TIR | INT      | 0             | Die Variable $i\_Event\_TIR$ (Seite 2715) zeigt die aktuelle Phase der "Nachoptimierung":  |
| sRet.i_Mode            | INT      | 0             | Die Betriebsart wird flankengesteuert gewechselt. Folgende Betriebsart wird aktiviert beim Wechsel zu <ul style="list-style-type: none"> <li>• <math>i\_Mode = 0</math>: Betriebsart "Inaktiv" (Stopp des Reglers)</li> <li>• <math>i\_Mode = 1</math>: Betriebsart "Erstoptimierung"</li> <li>• <math>i\_Mode = 2</math>: Betriebsart "Nachoptimierung"</li> <li>• <math>i\_Mode = 3</math>: Betriebsart "Automatikbetrieb"</li> <li>• <math>i\_Mode = 4</math>: Betriebsart "Handbetrieb"</li> </ul> $i\_Mode$ ist remanent. |
| sRet.r_Ctrl_Gain       | REAL     | 1.0           | Aktive Proportionalverstärkung<br>Gain ist remanent.   |
| sRet.r_Ctrl_Ti         | REAL     | 20.0          | <ul style="list-style-type: none"> <li>• <math>r\_Ctrl\_Ti &gt; 0.0</math>: Aktive Integrationszeit</li> <li>• <math>r\_Ctrl\_Ti = 0.0</math>: I-Anteil ist ausgeschaltet</li> </ul> $r\_Ctrl\_Ti$ ist remanent.   |

| Variable          | Datentyp | Vorbelegung | Beschreibung   |
|-------------------|----------|-------------|--|
| sRet.r_Ctrl_Td    | REAL     | 0.0         | <ul style="list-style-type: none"> <li>r_Ctrl_Td &gt; 0.0: Aktive Differenzierzeit</li> <li>r_Ctrl_Td = 0.0: D-Anteil ist ausgeschaltet</li> </ul> r_Ctrl_Td ist remanent.               |
| sRet.r_Ctrl_A     | REAL     | 0.0         | Aktiver Koeffizient Differenzierverszug<br>r_Ctrl_A ist remanent.  |
| sRet.r_Ctrl_B     | REAL     | 0.0         | Aktive Gewichtung des P-Anteils<br>r_Ctrl_B ist remanent.  |
| sRet.r_Ctrl_C     | REAL     | 0.0         | Aktive Gewichtung des D-Anteils<br>r_Ctrl_C ist remanent.  |
| sRet.r_Ctrl_Cycle | REAL     | 1.0         | Aktive Abtastzeit des PID-Algorithmus<br>r_Ctrl_Cycle wird während der Optimierung ermittelt und zu einem ganzzahligen Vielfachen von r_Cycle aufgerundet.<br>r_Ctrl_Cycle ist remanent. |

---

#### Hinweis

Verändern Sie die hier aufgelisteten Variablen in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden. Die Betriebsart "Inaktiv" erzwingen Sie durch den Wert "0" der Variable "sRet.i\_Mode".

---

#### Siehe auch

Technologieobjekte in Gerät laden (Seite 4056)

#### Parameter State und sRet.i\_Mode V1

##### Zusammenhang der Parameter

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Um die Betriebsart zu wechseln, müssen Sie die Variable sRet.i\_Mode ändern. Das gilt auch, wenn in sRet.i\_Mode bereits der Wert für die neue Betriebsart steht. Setzen Sie dann z. B. erst sRet.i\_Mode = 0 und anschließend sRet.i\_Mode = 3. Wenn die aktuelle Betriebsart des Reglers diesen Wechsel erlaubt, wird State auf den Wert von sRet.i\_Mode gesetzt.

Wenn PID\_Compact automatisch die Betriebsart wechselt gilt: State != sRet.i\_Mode.

Beispiele:

- Erfolgreiche Erstopptimierung  
State = 3 und sRet.i\_Mode = 1
- Im Fehlerfall  
State = 0 und sRet.i\_Mode bleibt auf bisherigem Wert, z. B. sRet.i\_Mode = 3
- ManualEnable = TRUE  
State = 4 und sRet.i\_Mode bleibt auf bisherigem Wert, z. B. sRet.i\_Mode = 3

**Hinweis**

Sie wollen z. B. eine erfolgreiche Nachoptimierung wiederholen ohne den Automatikbetrieb mit i\_Mode = 0 zu beenden.

Wenn Sie sRet.i\_Mode für einen Takt auf einen ungültigen Wert setzen, z. B. 9999, hat das keinen Einfluß auf State. Im nächsten Takt setzen Sie Mode = 2. So können Sie eine Änderung an sRet.i\_Mode erzeugen, ohne erst in die Betriebsart "Inaktiv" zu wechseln.

**Bedeutung der Werte**

| State / sRet.i_Mode | Beschreibung der Betriebsart   |
|---------------------|--|
| 0                   | <p>Inaktiv</p> <p>Der Regler ist ausgeschaltet.</p> <p>Bevor eine Erstopptimierung durchgeführt wurde, befindet sich der Regler in der Betriebsart "Inaktiv".</p> <p>Im laufenden Betrieb wechselt der PID-Regler in die Betriebsart "Inaktiv", wenn ein Fehler eintritt oder wenn im Inbetriebnahmefenster das Symbol "Regler deaktivieren" geklickt wird.</p>  |
| 1                   | <p>Erstopptimierung</p> <p>Die Erstopptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet.</p> <p>Voraussetzungen für die Erstopptimierung:</p> <ul style="list-style-type: none"> <li>• Der Regler befindet sich in der Betriebsart Inaktiv oder Handbetrieb</li> <li>• ManualEnable = FALSE</li> <li>• Der Istwert darf nicht zu nah am Sollwert sein.<br/> <math> \text{Setpoint} - \text{Input}  &gt; 0.3 *  sPid\_Cmpt.r\_Pv\_Hlm - sPid\_Cmpt.r\_Pv\_Llm </math> und<br/> <math> \text{Setpoint} - \text{Input}  &gt; 0.5 *  \text{Setpoint} </math></li> <li>• Der Sollwert darf während der Erstopptimierung nicht geändert werden.</li> </ul> <p>Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen.</p> <p>Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit sPid_Cmpt.b_LoadBackUp reaktiviert werden.</p> <p>Nach erfolgreicher Erstopptimierung wird in den Automatikbetrieb geschaltet; nach erfolgloser Erstopptimierung in die Betriebsart "Inaktiv".</p> <p>Die Phase der Erstopptimierung wird mit Variable i_Event_SUT V1 (Seite 2715) angezeigt.</p> |

| State / sRet.i_Mode | Beschreibung der Betriebsart  |
|---------------------|---|
| 2                   | <p>Nachoptimierung</p> <p>Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter optimiert. Die Unterschiede zwischen der Prozessantwort während der Erst- und Nachoptimierung werden analysiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstopptimierung.</p> <p>PID_Compact versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwertes. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwertes beeinflusst.</p> <p>Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit sPid_Cmpt.b_LoadBackUp reaktiviert werden.</p> <p>Voraussetzungen für Nachoptimierung:</p> <ul style="list-style-type: none"> <li>• Es werden keine Störungen erwartet.</li> <li>• Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen</li> <li>• Der Sollwert darf während der Nachoptimierung nicht geändert werden.</li> <li>• ManualEnable = FALSE</li> <li>• Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4)</li> </ul> <p>Die Nachoptimierung verläuft folgendermaßen beim Start aus:</p> <ul style="list-style-type: none"> <li>• Automatikbetrieb (State = 3)<br/>Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.<br/>PID_Compact regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.</li> <li>• Inaktiv (State = 0) oder Handbetrieb (State = 4)<br/>Wenn die Voraussetzungen für eine Erstopptimierung erfüllt sind, wird eine Erstopptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. Ist die Erstopptimierung nicht möglich, wechselt PID_Compact in die Betriebsart "Inaktiv".<br/>Wenn sich der Istwert für eine Erstopptimierung bereits zu nah am Sollwert befindet oder sPid_Calc.b_RunIn = TRUE, wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen.</li> </ul> <p>Nach erfolgreicher "Nachoptimierung" wechselt der Regler in die Betriebsart "Automatikbetrieb"; nach erfolgloser "Nachoptimierung" in die Betriebsart "Inaktiv".</p> <p>Die Phase der "Nachoptimierung" wird mit Variable i_Event_TIR V1 (Seite 2715) angezeigt.</p> |
| 3                   | <p>Automatikbetrieb</p> <p>Im Automatikbetrieb regelt PID_Compact die Regelstrecke entsprechend der vorgegebenen Parameter aus. Wenn eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt:</p> <ul style="list-style-type: none"> <li>• Erstopptimierung erfolgreich abgeschlossen</li> <li>• Nachoptimierung erfolgreich abgeschlossen</li> <li>• Ändern der Variable sRet.i_Mode auf den Wert 3.</li> </ul> <p>Wenn die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID_Compact in der zuletzt aktiven Betriebsart. Um PID_Compact in der Betriebsart "Inaktiv" zu belassen, setzen Sie sb_RunModeByStartup = FALSE.</p>  |
| 4                   | <p>Handbetrieb</p> <p>Im Handbetrieb geben Sie einen manuellen Ausgangswert am Parameter ManualValue vor.</p> <p>Diese Betriebsart wird aktiviert, wenn sRet.i_Mode = 4 oder bei steigender Flanke an ManualEnable. Wenn ManualEnable TRUE wird, ändert sich nur State. sRet.i_Mode bleibt auf dem bisherigen Wert. Bei einer fallenden Flanke an ManualEnable kehrt PID_Compact in die vorhergehende Betriebsart zurück.</p> <p>Der Wechsel in den Automatikbetrieb erfolgt stoßfrei.</p>  |

**Siehe auch**

- Ausgangsparameter PID\_Compact V1 (Seite 2703)
- Erstoptimierung (Seite 4089)
- Nachoptimierung (Seite 4091)
- Betriebsart "Handbetrieb" (Seite 4093)
- Variable i\_Event\_SUT V1 (Seite 2715)
- Variable i\_Event\_TIR V1 (Seite 2715)

**Parameter Error V1**

Stehen gleichzeitig mehrere Fehler an, so werden die Werte der Errorcodes binär addiert angezeigt. Wird z. B. der Errorcode 0003 angezeigt, so stehen gleichzeitig die Fehler 0001 und 0002 an.

| Error<br>(DW#16#...) | Beschreibung  |
|----------------------|---|
| 0000                 | Es steht kein Fehler an.  |
| 0001                 | Der Parameter "Input" ist außerhalb der Istwertgrenzen. <ul style="list-style-type: none"> <li>• Input &gt; sPid_Cmpt.r_Pv_Hlm oder</li> <li>• Input &lt; sPid_Cmpt.r_Pv_Llm</li> </ul> Sie können das Stellglied erst wieder verfahren, wenn der Fehler behoben ist. |
| 0002                 | Am Parameter "Input_PER" liegt ein ungültiger Wert an. Überprüfen Sie, ob am Analogeingang ein Fehler ansteht.  |
| 0004                 | Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden.  |
| 0008                 | Fehler beim Starten der Erstoptimierung. Der Istwert ist zu nahe am Sollwert. Starten Sie die Nachoptimierung.  |
| 0010                 | Der Sollwert wurde während der Optimierung verändert.   |
| 0020                 | Die Erstoptimierung ist im Automatikbetrieb und während der Nachoptimierung nicht erlaubt.  |
| 0080                 | Die Ausgangswertgrenzen sind nicht korrekt konfiguriert. Überprüfen Sie, ob die Grenzen des Ausgangswerts korrekt konfiguriert sind und zum Regelsinn passen.   |
| 0100                 | Fehler während der Optimierung führte zu ungültigen Parametern.   |
| 0200                 | Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat.  |
| 0400                 | Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter.  |
| 0800                 | Abtastzeitfehler: PID_Compact wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen.   |
| 1000                 | Ungültiger Wert am Parameter "Setpoint": Wert hat kein gültiges Zahlenformat.   |

**Siehe auch**

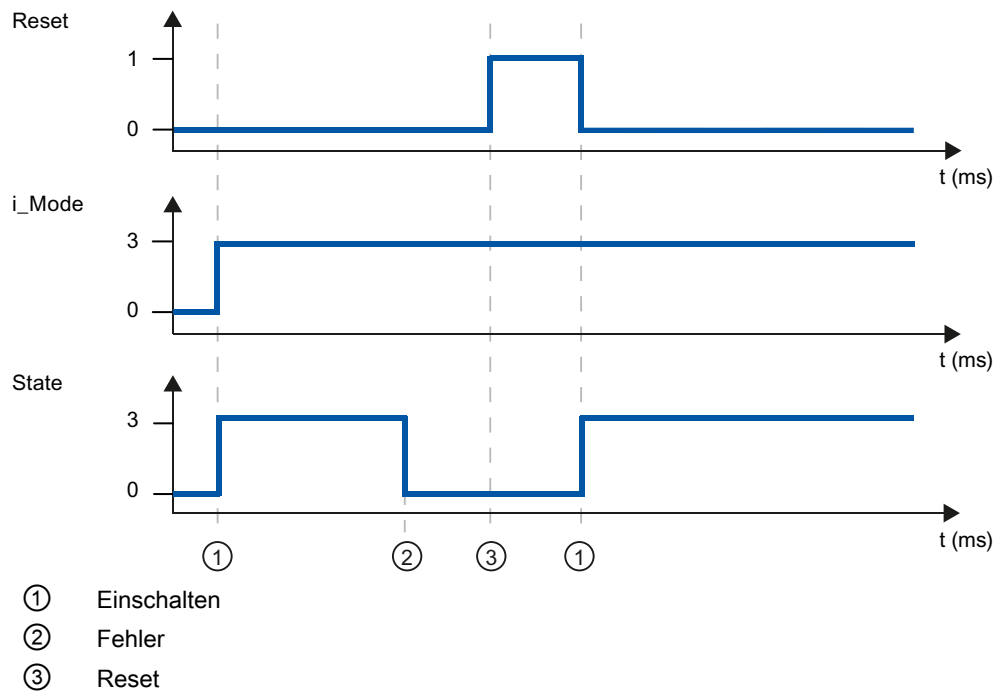
- Ausgangsparameter PID\_Compact V1 (Seite 2703)

## Parameter Reset V1

Das Verhalten bei Reset = TRUE ist abhängig von der Version der Anweisung PID\_Compact.

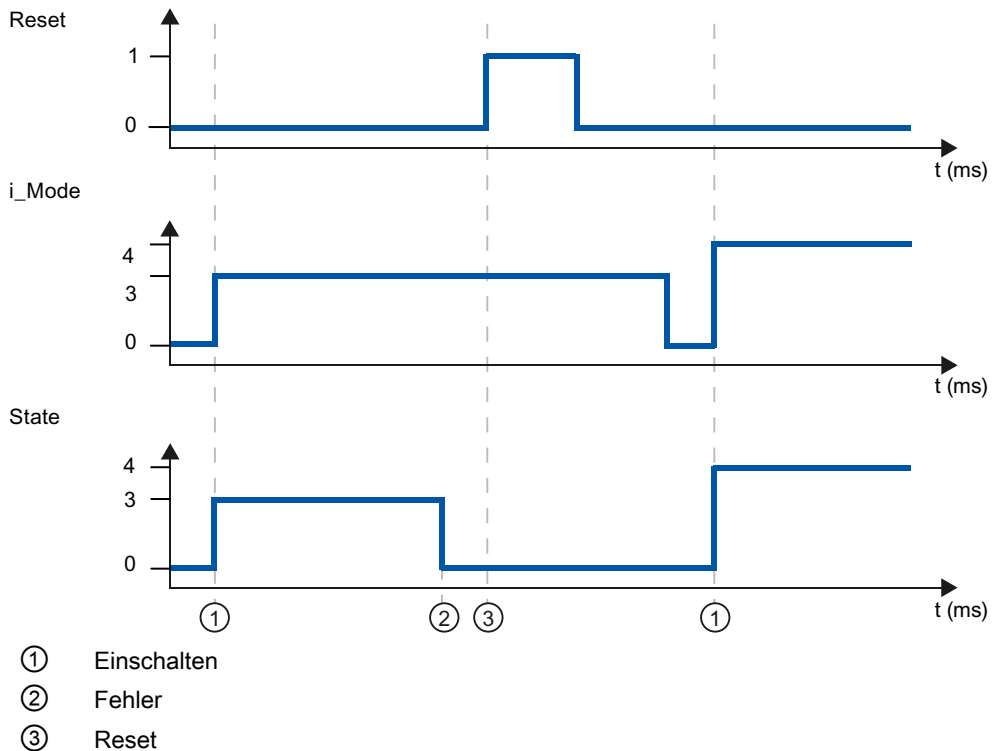
### Reset-Verhalten PID\_Compact ab V.1.1

Durch eine steigende Flanke an Reset werden Fehler und Warnungen zurückgesetzt und der I-Anteil gelöscht. Durch eine fallende Flanke an Reset wird in die zuletzt aktive Betriebsart gewechselt.



### Reset-Verhalten PID\_Compact V.1.0

Durch eine steigende Flanke an Reset werden Fehler und Warnungen zurückgesetzt und der I-Anteil gelöscht. Erst durch eine Flanke an i\_Mode wird der Regler wieder eingeschaltet.



### Variable sd\_warning V1

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Variable sd\_warning binär addiert angezeigt. Wenn z. B. die Warnung 0003 angezeigt, so stehen gleichzeitig die Warnungen 0001 und 0002 an.

| sd_warning (DW#16#...) | Beschreibung   |
|------------------------|--|
| 0000                   | Es liegt keine Warnung vor.  |
| 0001                   | Während der Ersoptimierung wurde der Wendepunkt nicht gefunden.  |
| 0002                   | Während der Nachoptimierung war die Oszillation verstärkt.   |
| 0004                   | Der Sollwert lag außerhalb der eingestellten Grenzen.  |
| 0008                   | Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode "i_CtrlTypeTIR = 3" berechnet. |
| 0010                   | Die Betriebsart konnte nicht geändert werden, da ManualEnable = TRUE.  |
| 0020                   | Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.                        |
| 0040                   | Der Istwert hat eine seiner Warngrenze überschritten.  |



Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist:

- 0004
- 0020
- 0040

Alle anderen Warnungen werden bei steigender Flanke an Reset gelöscht.

### Variable i\_Event\_SUT V1

| i_Event_SUT | Name        | Beschreibung                      |
|-------------|-------------|-----------------------------------|
| 0           | SUT_INIT    | Erstoptimierung initialisieren    |
| 100         | SUT_STDABW  | Standardabweichung berechnen      |
| 200         | SUT_GET_POI | Wendepunkt ermitteln              |
| 9900        | SUT_IO      | Erstoptimierung erfolgreich       |
| 1           | SUT_NIO     | Erstoptimierung nicht erfolgreich |

### Siehe auch

Statische Variablen PID\_Compact V1 (Seite 2704)

Parameter State und sRet.i\_Mode V1 (Seite 2709)

### Variable i\_Event\_TIR V1

| i_Event_TIR | Name          | Beschreibung  |
|-------------|---------------|---|
| -100        | TIR_FIRST_SUT | Die Nachoptimierung ist nicht möglich. Es wird erst eine Erstoptimierung durchgeführt.                |
| 0           | TIR_INIT      | Nachoptimierung initialisieren  |
| 200         | TIR_STDABW    | Standardabweichung berechnen  |
| 300         | TIR_RUN_IN    | Versuchen Sollwert zu erreichen   |
| 400         | TIR_CTRLN     | Versuchen Sollwert mit vorhandenen PID-Parametern zu erreichen (wenn Erstoptimierung erfolgreich war) |
| 500         | TIR_OSZIL     | Oszillation ermitteln und Parameter berechnen   |
| 9900        | TIR_IO        | Nachoptimierung erfolgreich   |
| 1           | TIR_NIO       | Nachoptimierung nicht erfolgreich   |

### Siehe auch

Statische Variablen PID\_Compact V1 (Seite 2704)

Parameter State und sRet.i\_Mode V1 (Seite 2709)

## PID\_3Step

### Neuerungen PID\_3Step

#### PID\_3Step V2.1

- **Einsatz mit S7-1200**  
Ab PID\_3Step V2.1 ist die Anweisung mit V2-Funktionalität auch auf einer S7-1200 ab der Firmware-Version 4.0 einsetzbar.

#### PID\_3Step V2.0

- **Verhalten im Fehlerfall**  
Das Verhalten bei ActivateRecoverMode = TRUE wurde grundlegend überarbeitet. PID\_3Step verhält sich in der Default-Einstellung Fehler toleranter.

|  |
|--|
| <b>ACHTUNG</b>   |
| <b>Ihre Anlage kann beschädigt werden.</b>   |
| Wenn Sie die Default-Einstellung verwenden, bleibt PID_3Step auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden. |
| Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.   |

Mit dem Eingangsparameter ErrorAck quittieren Sie die Fehler und Warnungen, ohne den Regler neu zu starten oder den I-Anteil zu löschen.  
Durch einen Wechsel der Betriebsart werden nicht mehr anstehende Fehler nicht quittiert.

- **Wechsel der Betriebsarten**  
Die Betriebsart geben Sie am Durchgangsparameter Mode vor und starten diese über eine steigende Flanke an ModeActivate. Die Variable Retain.Mode ist entfallen.  
Die Stellzeitmessung kann nicht mehr mit GetTransitTime.Start gestartet werden, sondern nur noch mit Mode = 6 und einer steigende Flanke an ModeActivate.
- **Multiinstanzfähigkeit**  
Sie können PID\_3Step als Multiinstanz-DB aufrufen. Dann wird kein Technologieobjekt angelegt und es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID\_3Step direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.
- **Anlaufverhalten**  
Die an Mode vorgegebene Betriebsart wird auch gestartet bei einer fallenden Flanke an Reset und bei Kaltstart der CPU, wenn RunModeByStartup = TRUE ist.
- **ENO-Verhalten**  
ENO wird in Abhängigkeit von der Betriebsart gesetzt.  
Wenn State = 0 ist, dann ist ENO = FALSE.  
Wenn State ≠ 0 ist, dann ist ENO = TRUE.

- **Handbetrieb**  
Die Eingangsparameter Manual\_UP und Manual\_DN arbeiten nicht mehr Flanken gesteuert. Der Flanken gesteuerte Handbetrieb ist weiterhin über die Variablen ManualUpInternal und ManualDnInternal möglich.  
Im "Handbetrieb ohne Anschlagssignale" (Mode = 10) werden die Anschlagssignale Actuator\_H und Actuator\_L ignoriert, obwohl sie eingeschaltet sind.
- **Vorbelegung der PID-Parameter**  
Folgende Vorbelegungen wurden geändert:
  - Gewichtung des P-Anteils (PWeighting) von 0.0 auf 1.0
  - Gewichtung des D-Anteils (DWeighting) von 0.0 auf 1.0
  - Koeffizient für den Differenzierverzögerung (TdFiltRatio) von 0.0 auf 0.2
- **Begrenzung der Motorstellzeit**  
Sie konfigurieren an der Variable Config.VirtualActuatorLimit, um wie viel % der Motorstellzeit das Stellglied maximal in eine Richtung verfahren wird.
- **Sollwertvorgabe während der Optimierung**  
Die erlaubte Schwankung des Sollwerts während der Optimierung konfigurieren Sie an der Variable CancelTuningLevel.
- **Aufschalten einer Störgröße**  
Sie können am Parameter Disturbance eine Störgröße aufschalten.
- **Fehlerbehebung**  
Wenn die Anschlagssignale nicht aktiviert sind (ActuatorEndStopOn = FALSE), werden Actuator\_H und Actuator\_L nicht mehr für die Ermittlung von ScaledFeedback berücksichtigt.

### PID\_3Step V1.1

- **Handbetrieb beim Anlauf der CPU**  
Wenn beim Start der CPU ManualEnable = TRUE, startet PID\_3Step im Handbetrieb. Eine steigende Flanke an ManualEnable ist nicht notwendig.
- **Verhalten im Fehlerfall**  
Die Variable ActivateRecoverMode wirkt im Handbetrieb nicht mehr.
- **Fehlerbehebung**  
Die Variable Progress wird nach erfolgreicher Optimierung oder Stellzeitmessung zurückgesetzt.

### Kompatibilität mit CPU und FW

Die folgende Tabelle zeigt, auf welcher CPU Sie welche Version von PID\_3Step einsetzen können.

| CPU     | FW   | PID_3Step    |
|---------|------|--------------|
| S7-1200 | V4.X | V2.1<br>V1.1 |
| S7-1200 | V3.X | V1.1<br>V1.0 |

| CPU     | FW   | PID_3Step    |
|---------|------|--------------|
| S7-1200 | V2.X | V1.1<br>V1.0 |
| S7-1200 | V1.X | -            |
| S7-1500 | V1.1 | V2.1<br>V2.0 |
| S7-1500 | V1.0 | V2.0         |

## PID\_3Step V2

### Beschreibung PID\_3Step V2

#### Beschreibung

Mit der Anweisung PID\_3Step konfigurieren Sie einen PID-Regler mit Selbstoptimierung für Ventile oder Stellglieder mit integrierendem Verhalten.

Folgende Betriebsarten sind möglich:

- Inaktiv
- Erstopoptimierung
- Nachoptimierung
- Automatikbetrieb
- Handbetrieb
- Ersatzausgangswert anfahren
- Stellzeitmessung
- Fehlerüberwachung
- Ersatzausgangswert anfahren mit Fehlerüberwachung
- Handbetrieb ohne Anschlagssignale

Die Betriebsarten sind detailliert beschrieben beim Parameter State.

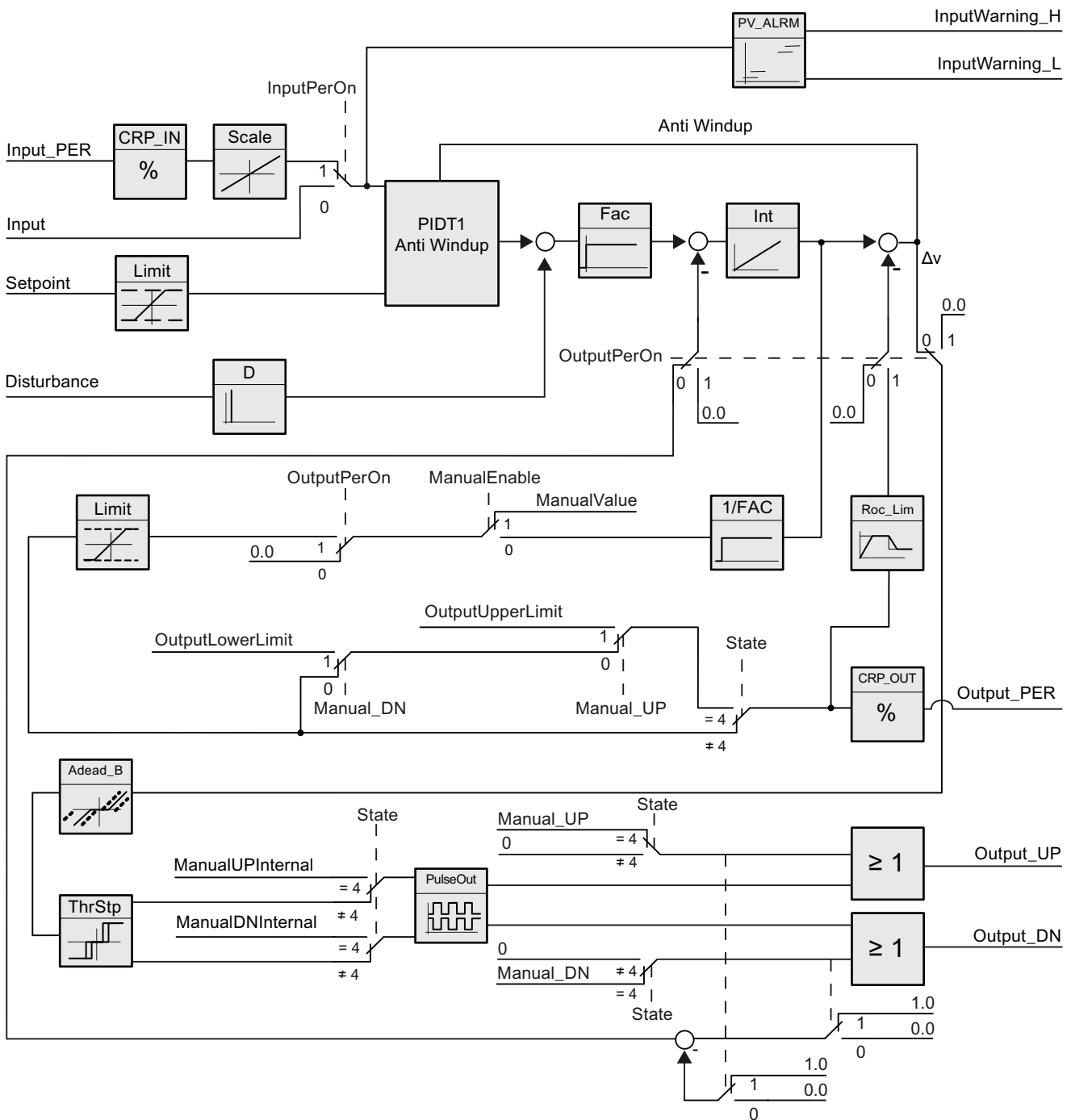
## PID-Algorithmus

PID\_3Step ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der PID-Algorithmus arbeitet nach folgender Formel:

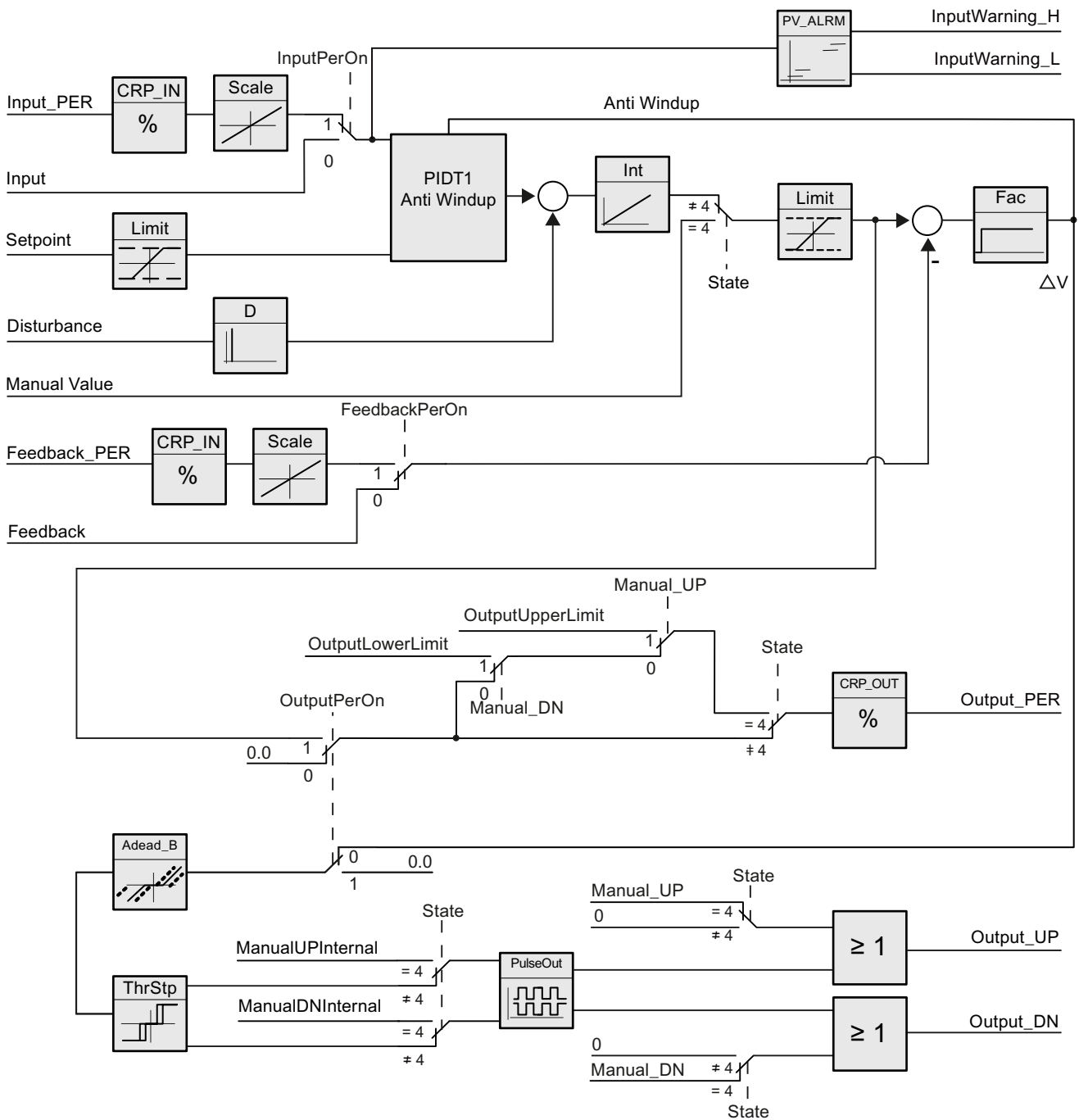
$$\Delta y = K_p \cdot s \cdot \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

| Symbol     | Beschreibung   |
|------------|--|
| $\Delta y$ | Ausgangswert des PID-Algorithmus   |
| $K_p$      | Proportionalverstärkung  |
| $s$        | Laplace-Operator   |
| $b$        | Gewichtung des P-Anteils   |
| $w$        | Sollwert   |
| $x$        | Istwert  |
| $T_i$      | Integrationszeit   |
| $T_D$      | Differenzierzeit   |
| $a$        | Koeffizient für den Differenzierverzug (Differenzierverzug $T1 = a \times T_D$ ) |
| $c$        | Gewichtung des D-Anteils   |

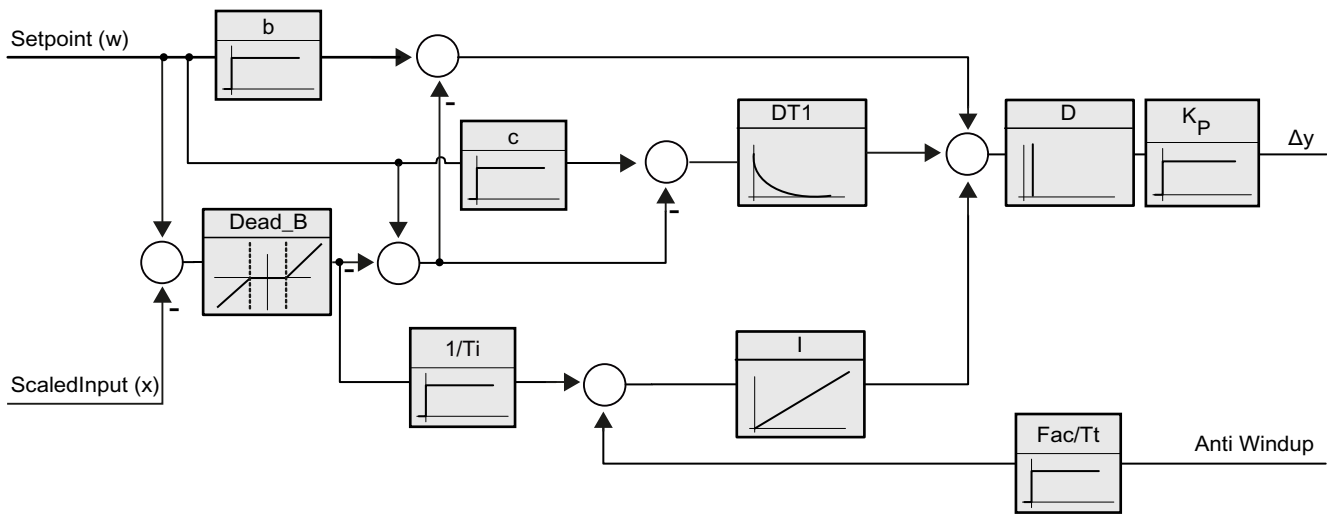
Blockschaltbild ohne Stellungsrückmeldung



Blockschaltbild mit Stellungsrückmeldung



**Blockschaltbild PIDT1 mit Anti-Windup**



**Aufruf**

PID\_3Step wird im konstanten Zeitraster eines Weckalarm-OBs aufgerufen.

Wenn Sie PID\_3Step als Multiinstanz-DB aufrufen, wird kein Technologieobjekt angelegt. Es steht Ihnen keine Parametrier- und Inbetriebnahmeoberfläche zur Verfügung. Sie müssen PID\_3Step direkt im Multiinstanz-DB parametrieren und über eine Beobachtungstabelle in Betrieb nehmen.

**Laden in Gerät**

Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PID\_3Step vollständig laden.

Technologieobjekte in Gerät laden (Seite 4056)

**Anlauf**

Bei Anlauf der CPU startet PID\_3Step in der Betriebsart, die am Durchgangparameter Mode gespeichert ist. Um PID\_3Step in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

**Verhalten im Fehlerfall**

Im Automatikbetrieb und während der Inbetriebnahme hängt das Verhalten im Fehlerfall von den Variablen ErrorBehaviour und ActivateRecoverMode ab. Im Handbetrieb ist das Verhalten



unabhängig von ErrorBehaviour und ActivateRecoverMode. Wenn ActivateRecoverMode = TRUE ist, hängt das Verhalten zusätzlich vom aufgetretenen Fehler ab.

| ErrorBehaviour | ActivateRecoverMode | Konfigurationseditor<br>> Stellgliedeinstellung<br>> Output setzen auf | Verhalten   |
|----------------|---------------------|--|---|
| FALSE          | FALSE               | Aktuellen Ausgangswert   | Wechsel in Betriebsart "Inaktiv" (State = 0)<br>Das Stellglied bleibt in der aktuellen Position.  |
| FALSE          | TRUE                | Aktuellen Ausgangswert für die Fehlerdauer                             | Wechsel in Betriebsart "Fehlerüberwachung" (State = 7)<br>Das Stellglied bleibt für die Fehlerdauer in der aktuellen Position.  |
| TRUE           | FALSE               | Ersatzausgangswert   | Wechsel in Betriebsart "Ersatzausgangswert anfahren" (State = 5)<br>Das Stellglied wird auf den konfigurierten Ersatzausgangswert gefahren.<br>Wechsel in Betriebsart "Inaktiv" (State = 0)<br>Das Stellglied bleibt in der aktuellen Position. |
| TRUE           | TRUE                | Ersatzausgangswert für die Fehlerdauer                                 | Wechsel in Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" (State = 8)<br>Das Stellglied wird auf den konfigurierten Ersatzausgangswert gefahren.<br>Wechsel in Betriebsart "Fehlerüberwachung" (State = 7)                     |

PID\_3Step verwendet im Handbetrieb ManualValue als Ausgangswert, außer bei den folgenden Fehlern:

- 2000h: Ungültiger Wert am Parameter Feedback\_PER.
- 4000h: Ungültiger Wert am Parameter Feedback.
- 8000h: Fehler bei der digitalen Stellungsrückmeldung.

Sie können die Position des Stellglieds dann nur mit Manual\_UP und Manual\_DN verändern, nicht mit ManualValue:

Der Parameter Error zeigt, ob in diesem Takt ein Fehler aufgetreten ist. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind. ErrorBits wird durch eine steigende Flanke an Reset oder ErrorAck zurückgesetzt.

## Siehe auch

Parameter State und Mode V2 (Seite 2740)

Parameter ErrorBits V2 (Seite 2744)

PID\_3Step V2 konfigurieren (Seite 4095)

## Arbeitsweise PID\_3Step V2

### Istwertgrenzen überwachen

In den Variablen Config.InputUpperLimit und Config.InputLowerLimit legen Sie eine Ober- und Untergrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Grenzen liegt, tritt ein Fehler auf (ErrorBits = 0001h).

In den Variablen Config.InputUpperWarning und Config.InputLowerWarning legen Sie eine obere und untere Warngrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Warngrenzen liegt, tritt eine Warnung auf (Warning = 0040h) und der Ausgangsparameter InputWarning\_H oder InputWarning\_L wird TRUE.

### Sollwert begrenzen

In den Variablen Config.SetpointUpperLimit und Config.SetpointLowerLimit legen Sie eine Ober- und Untergrenze des Sollwerts fest. PID\_3Step begrenzt den Sollwert automatisch auf die Istwertgrenzen. Sie können den Sollwert auf einen kleineren Bereich begrenzen. PID\_3Step prüft, ob dieser Bereich innerhalb der Istwertgrenzen liegt. Wenn der Sollwert außerhalb dieser Grenzen liegt, wird die Ober- oder Untergrenze als Sollwert verwendet und der Ausgangsparameter SetpointLimit\_H oder SetpointLimit\_L wird TRUE.

Der Sollwert wird in allen Betriebsarten begrenzt.

### Ausgangswert begrenzen

In den Variablen Config.OutputUpperLimit und Config.OutputLowerLimit legen Sie eine Ober- und Untergrenze des Ausgangswerts fest. Die Ausgangswertgrenzen müssen innerhalb "Unterer Anschlag" und "Oberer Anschlag" liegen.

- Oberer Anschlag: Config.FeedbackScaling.UpperPointOut
- Unter Anschlag: Config.FeedbackScaling.LowerPointOut

Es muss gelten:

$UpperPointOut \geq OutputUpperLimit > OutputLowerLimit \geq LowerPointOut$

Die gültigen Werte für "Oberer Anschlag" und "Unterer Anschlag" hängen ab von:

- FeedbackOn
- FeedbackPerOn
- OutputPerOn

| OutputPerOn | FeedbackOn | FeedbackPerOn | LowerPointOut             | UpperPointOut               |
|-------------|------------|---------------|---------------------------|-----------------------------|
| FALSE       | FALSE      | FALSE         | nicht einstellbar (0.0 %) | nicht einstellbar (100.0 %) |
| FALSE       | TRUE       | FALSE         | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |
| FALSE       | TRUE       | TRUE          | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |
| TRUE        | FALSE      | FALSE         | nicht einstellbar (0.0 %) | nicht einstellbar (100.0 %) |

| OutputPerOn | FeedbackOn | FeedbackPerOn | LowerPointOut       | UpperPointOut       |
|-------------|------------|---------------|---------------------|---------------------|
| TRUE        | TRUE       | FALSE         | -100.0 % oder 0.0 % | 0.0 % oder +100.0 % |
| TRUE        | TRUE       | TRUE          | -100.0 % oder 0.0 % | 0.0 % oder +100.0 % |

Wenn OutputPerOn = FALSE und FeedbackOn = FALSE, können Sie den Ausgangswert nicht begrenzen. Output\_UP und Output\_DN werden dann bei Actuator\_H = TRUE oder Actuator\_L = TRUE zurück gesetzt. Wenn auch keine Anschlagssignale vorhanden sind, werden Output\_UP und Output\_DN nach einer Verfahrzeit von  $\text{Config.VirtualActuatorLimit} \times \text{Retain.TransitTime}/100$  zurück gesetzt.

Der Ausgangswert beträgt 27648 bei 100 % und -27648 bei -100 %. PID\_3Step muss das Ventil vollständig schließen können.

### Ersatzausgangswert

PID\_3Step kann im Fehlerfall einen Ersatzausgangswert ausgeben und das Stellglied in eine sichere Position stellen, die Sie an der Variablen SavePosition vorgeben. Der Ersatzausgangswert muss innerhalb der Ausgangswertgrenzen liegen.

### Gültigkeit der Signale überwachen

Die Werte der folgenden Parameter werden bei Verwendung auf Gültigkeit überwacht:

- Setpoint
- Input
- Input\_PER
- Input\_PER
- Feedback
- Feedback\_PER
- Disturbance
- ManualValue
- SavePosition
- Output\_PER

### Abtastzeit PID\_3Step überwachen

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden OB. Die Anweisung PID\_3Step misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (ErrorBits = 0800h).

Während der Optimierung tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert  $\geq 1,1 \times$  alter Mittelwert
- Neuer Mittelwert  $\leq 0,9 \times$  alter Mittelwert

Im Automatikbetrieb tritt der Fehler auf, wenn gilt:

- Neuer Mittelwert  $\geq 1,5$  x alter Mittelwert
- Neuer Mittelwert  $\leq 0,5$  x alter Mittelwert

Wenn Sie die Überwachung der Abtastzeit ausschalten (`CycleTime.EnMonitoring = FALSE`), können Sie `PID_3Step` auch im `OB1` aufrufen. Sie müssen dann aufgrund der schwankenden Abtastzeit eine schlechtere Regelqualität akzeptieren.

### Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von `PID_3Step` werden bei jedem Aufruf durchgeführt.

### Motorstellzeit messen

Die Motorstellzeit ist die Zeit in Sekunden, die der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen. Das Stellglied wird maximal für  $\text{Config.VirtualActuatorLimit} \times \text{Retain.TransitTime}/100$  in eine Richtung bewegt. `PID_3Step` benötigt die Motorstellzeit so genau wie möglich, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren. Die Motorstellzeit können Sie während der Inbetriebnahme messen. Die Ausgangswertgrenzen werden während der Motorstellzeitmessung nicht berücksichtigt. Das Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

### Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn. Für Kühlungen und Abflussregelungen kann es notwendig sein, den Regelsinn zu invertieren. `PID_3Step` arbeitet nicht mit negativer Proportionalverstärkung. Wenn `InvertControl = TRUE`, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts. Der Regelsinn wird auch während Erst- und Nachoptimierung berücksichtigt.

### Siehe auch

`PID_3Step V1 konfigurieren` (Seite 4111)

### Änderungen der Schnittstelle `PID_3Step V2`

Die folgende Tabelle zeigt, was sich an der Schnittstelle der Anweisung `PID_3Step` geändert hat.

| PID_3Step V1           | PID_3Step V2          | Änderung   |
|------------------------|-----------------------|--|
| Input_PER              | Input_PER             | Datentyp von Word zu Int                                 |
| Feedback_PER           | Feedback_PER          | Datentyp von Word zu Int                                 |
|                        | Disturbance           | Neu  |
| Manual_UP              | Manual_UP             | Funktion   |
| Manual_DN              | Manual_DN             | Funktion   |
|                        | ErrorAck              | Neu  |
|                        | ModeActivate          | Neu  |
| Output_PER             | Output_PER            | Datentyp von Word zu Int                                 |
|                        | ManualUPInternal      | Neu  |
|                        | ManualDNInternal      | Neu  |
|                        | CancelTuningLevel     | Neu  |
|                        | VirtualActuatorLlimit | Neu  |
| Config.Loadbackup      | Loadbackup            | Umbenannt  |
| Config.TransitTime     | Retain.TransitTime    | Umbenannt und Remanenz hinzugefügt                       |
| GetTransitTime.Start   |                       | Ersetzt durch Mode und ModeActivate.                     |
| SUT.CalculateSUTParams | SUT.CalculateParams   | Umbenannt  |
| SUT.TuneRuleSUT        | SUT.TuneRule          | Umbenannt  |
| TIR.CalculateTIRParams | TIR.CalculateParams   | Umbenannt  |
| TIR.TuneRuleTIR        | TIR.TuneRule          | Umbenannt  |
| Retain.Mode            | Mode                  | Funktion<br>Deklaration von Static zu Durchgangparameter |

## Eingangsparameter PID\_3Step V2

Tabelle 9-84

| Parameter  | Datentyp | Vorbelegung | Beschreibung   |
|------------|----------|-------------|--|
| Setpoint   | REAL     | 0.0         | Sollwert des PID-Reglers im Automatikbetrieb   |
| Input      | REAL     | 0.0         | Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet.<br>Wenn Sie den Parameter Input verwenden, muss Config.InputPerOn = FALSE sein.  |
| Input_PER  | INT      | 0           | Ein Analogeingang wird als Quelle des Istwerts verwendet.<br>Wenn Sie den Parameter Input_PER verwenden, muss Config.InputPerOn = TRUE sein.   |
| Actuator_H | BOOL     | FALSE       | Digitale Stellungsrückmeldung des Ventils für den oberen Anschlag<br>Wenn Actuator_H = TRUE ist die Stellung des Ventils am oberen Anschlag und das Ventil wird nicht weiter in diese Richtung bewegt. |

| Parameter    | Datentyp | Vorbelegung | Beschreibung  |
|--------------|----------|-------------|---|
| Actuator_L   | BOOL     | FALSE       | Digitale Stellungsrückmeldung des Ventils für den unteren Anschlag<br>Wenn Actuator_L = TRUE ist die Stellung des Ventils am unteren Anschlag und das Ventil wird nicht weiter in diese Richtung bewegt.  |
| Feedback     | REAL     | 0.0         | Stellungsrückmeldung des Ventils<br>Wenn Sie den Parameter Feedback verwenden, muss Config.FeedbackPerOn = FALSE sein.  |
| Feedback_PER | INT      | 0           | Analoge Stellungsrückmeldung eines Ventils<br>Wenn Sie den Parameter Feedback_PER verwenden, muss Config.FeedbackPerOn = TRUE sein.<br>Feedback_PER wird skaliert anhand der Variablen: <ul style="list-style-type: none"> <li>• Config.FeedbackScaling.LowerPointIn</li> <li>• Config.FeedbackScaling.UpperPointIn</li> <li>• Config.FeedbackScaling.LowerPointOut</li> <li>• Config.FeedbackScaling.UpperPointOut</li> </ul>  |
| Disturbance  | REAL     | 0.0         | Störgröße oder Vorsteuerungswert  |
| ManualEnable | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>• Flanke FALSE -&gt; TRUE aktiviert die Betriebsart "Handbetrieb", State = 4, Mode bleibt unverändert. Solange ManualEnable = TRUE ist, können Sie die Betriebsart nicht über eine steigende Flanke an ModeActivate ändern und den Inbetriebnahmedialog nicht nutzen.</li> <li>• Flanke TRUE -&gt; FALSE aktiviert die Betriebsart, die durch Mode vorgegeben wird.</li> </ul> <p>Es wird empfohlen, die Betriebsart nur über ModeActivate zu ändern.</p>  |
| ManualValue  | REAL     | 0.0         | Im Handbetrieb wird die absolute Position des Ventils vorgegeben. ManualValue wird nur ausgewertet, wenn Sie Output_PER verwenden <b>oder</b> eine Stellungsrückmeldung verfügbar ist.  |
| Manual_UP    | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>• Manual_UP = TRUE<br/>Das Ventil wird geöffnet, auch wenn Sie Output_PER oder eine Stellungsrückmeldung verwenden. Das Ventil wird nicht mehr bewegt, wenn der obere Anschlag erreicht ist. Siehe auch Config.VirtualActuatorLimit</li> <li>• Manual_UP = FALSE<br/>Wenn Sie Output_PER oder eine Stellungsrückmeldung verwenden, wird das Ventil auf ManualValue gefahren. Andernfalls wird das Ventil nicht mehr bewegt.</li> </ul> <p>Wenn Manual_UP und Manual_DN gleichzeitig TRUE gesetzt werden, wird das Ventil nicht bewegt.</p> |

| Parameter    | Datentyp | Vorbelegung | Beschreibung  |
|--------------|----------|-------------|---|
| Manual_DN    | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>Manual_DN = TRUE<br/>Das Ventil wird geschlossen, auch wenn Sie Output_PER oder eine Stellungsrückmeldung verwenden. Das Ventil wird nicht mehr bewegt, wenn der untere Anschlag erreicht ist.<br/>Siehe auch Config.VirtualActuatorLimit</li> <li>Manual_DN = FALSE<br/>Wenn Sie Output_PER oder eine Stellungsrückmeldung verwenden, wird das Ventil auf ManualValue gefahren. Andernfalls wird das Ventil nicht mehr bewegt.</li> </ul>   |
| ErrorAck     | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>Flanke FALSE -&gt; TRUE<br/>ErrorBits und Warning werden zurückgesetzt.</li> </ul>   |
| Reset        | BOOL     | FALSE       | <p>Führt einen Neustart des Reglers durch.</p> <ul style="list-style-type: none"> <li>Flanke FALSE -&gt; TRUE <ul style="list-style-type: none"> <li>Wechsel in Betriebsart "Inaktiv"</li> <li>ErrorBits und Warning werden zurückgesetzt.</li> <li>I-Anteil wird gelöscht (PID-Parameter bleiben erhalten)</li> </ul> </li> <li>Solange Reset = TRUE ist, bleibt PID_3Step in der Betriebsart "Inaktiv" (State = 0).</li> <li>Flanke TRUE -&gt; FALSE<br/>PID_3Step wechselt in die Betriebsart, die an Mode gespeichert ist.</li> </ul> |
| ModeActivate | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>Flanke FALSE -&gt; TRUE<br/>PID_3Step wechselt in die Betriebsart, die an Mode gespeichert ist.</li> </ul>   |

## Ausgangsparameter PID\_3Step V2

Tabelle 9-85

| Parameter      | Datentyp | Vorbelegung | Beschreibung   |
|----------------|----------|-------------|--|
| ScaledInput    | REAL     | 0.0         | Skalierter Istwert   |
| ScaledFeedback | REAL     | 0.0         | Skalierte Stellungsrückmeldung<br>Bei einem Stellglied ohne Stellungsrückmeldung zeigt ScaledFeedback die Position des Stellglieds sehr ungenau an. Dann darf ScaledFeedback nur zur groben Abschätzung der aktuellen Position genutzt werden. |
| Output_UP      | BOOL     | FALSE       | Digitaler Ausgangswert zum Öffnen des Ventils<br>Wenn Config.OutputPerOn = FALSE, wird der Parameter Output_UP verwendet.  |
| Output_DN      | BOOL     | FALSE       | Digitaler Ausgangswert zum Schließen des Ventils<br>Wenn Config.OutputPerOn = FALSE, wird der Parameter Output_DN verwendet.   |
| Output_PER     | INT      | 0           | Analoger Ausgangswert<br>Wenn Config.OutputPerOn = TRUE, wird Output_PER verwendet.  |

| Parameter       | Datentyp | Vorbelegung | Beschreibung  |
|-----------------|----------|-------------|---|
| SetpointLimit_H | BOOL     | FALSE       | Wenn SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht (Setpoint ≥ Config.SetpointUpperLimit). Der Sollwert wird auf Config.SetpointUpperLimit begrenzt.   |
| SetpointLimit_L | BOOL     | FALSE       | Wenn SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht (Setpoint ≤ Config.SetpointLowerLimit). Der Sollwert wird auf Config.SetpointLowerLimit begrenzt.  |
| InputWarning_H  | BOOL     | FALSE       | Wenn InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten.  |
| InputWarning_L  | BOOL     | FALSE       | Wenn InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten.  |
| State           | INT      | 0           | Der Parameter State (Seite 2740) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit dem Eingangsparameter Mode und einer steigenden Flanke an ModeActivate. <ul style="list-style-type: none"> <li>• State = 0: Inaktiv</li> <li>• State = 1: Erstopoptimierung</li> <li>• State = 2: Nachoptimierung</li> <li>• State = 3: Automatikbetrieb</li> <li>• State = 4: Handbetrieb</li> <li>• State = 5: Ersatzausgangswert anfahren</li> <li>• State = 6: Stellzeitmessung</li> <li>• State = 7: Fehlerüberwachung</li> <li>• State = 8: Ersatzausgangswert anfahren mit Fehlerüberwachung</li> <li>• State = 10: Handbetrieb ohne Anschlagssignale</li> </ul> |
| Error           | BOOL     | FALSE       | Wenn Error = TRUE, liegt in diesem Takt mindestens eine Fehlermeldung vor.  |
| ErrorBits       | DWORD    | DW#16#0     | Der Parameter ErrorBits (Seite 2744) zeigt, welche Fehlermeldungen vorliegen. ErrorBits ist remanent und wird bei einer steigenden Flanke an Reset oder ErrorAck zurückgesetzt.   |

**Siehe auch**

Parameter State und Mode V2 (Seite 2740)

Parameter ErrorBits V2 (Seite 2744)



## Durchgangparameter PID\_3Step V2

Tabelle 9-86

| Parameter | Datentyp | Vorbelegung | Beschreibung  |
|-----------|----------|-------------|---|
| Mode      | INT      | 4           | <p>An Mode geben Sie die Betriebsart vor, in die PID_3Step wechseln soll. Möglich sind:</p> <ul style="list-style-type: none"> <li>• Mode = 0: Inaktiv</li> <li>• Mode = 1: Erstopptimierung</li> <li>• Mode = 2: Nachoptimierung</li> <li>• Mode = 3: Automatikbetrieb</li> <li>• Mode = 4: Handbetrieb</li> <li>• Mode = 6: Stellzeitmessung</li> <li>• Mode = 10: Handbetrieb ohne Anschlagssignale</li> </ul> <p>Die Betriebsart wird aktiviert durch:</p> <ul style="list-style-type: none"> <li>• Steigende Flanke an ModeActivate</li> <li>• Fallende Flanke an Reset</li> <li>• Fallende Flanke an ManualEnable</li> <li>• Kaltstart der CPU, wenn RunModeByStartup = TRUE</li> </ul> <p>Mode ist remanent.</p> <p>Eine detaillierte Beschreibung der Betriebsarten finden Sie unter Parameter State und Mode V2 (Seite 2740) .</p> |

## Statische Variablen PID\_3Step V2

Nicht aufgeführte Variablen dürfen Sie nicht verändern. Diese werden nur intern verwendet.

| Variable            | Datentyp | Vorbelegung | Beschreibung   |
|---------------------|----------|-------------|--|
| ManualUpInternal    | BOOL     | FALSE       | Im Handbetrieb öffnet jede steigende Flanke das Ventil um 5 % des gesamten Stellbereichs oder für die Mindeststellzeit des Motors. ManualUpInternal wird nur ausgewertet, wenn Sie weder Output_PER noch eine Stellungsrückmeldung verwenden. Diese Variable wird im Inbetriebnahmedialog verwendet.   |
| ManualDnInternal    | BOOL     | FALSE       | Im Handbetrieb schließt jede steigende Flanke das Ventil um 5 % des gesamten Stellbereichs oder für die Mindeststellzeit des Motors. ManualDnInternal wird nur ausgewertet, wenn Sie weder Output_PER noch eine Stellungsrückmeldung verwenden. Diese Variable wird im Inbetriebnahmedialog verwendet. |
| ActivateRecoverMode | BOOL     | TRUE        | Die Variable ActivateRecoverMode V2 (Seite 2747) bestimmt das Verhalten im Fehlerfall.   |
| RunModeByStartup    | BOOL     | TRUE        | Nach CPU Neustart Betriebsart an Mode aktivieren<br>Wenn RunModeByStartup = TRUE, startet PID_3Step nach CPU-Anlauf in der Betriebsart, die an Mode gespeichert ist.<br>Wenn RunModeByStartup = FALSE, bleibt PID_3Step nach CPU-Anlauf in der Betriebsart "Inaktiv".                                  |

| Variable             | Datentyp | Vorbelegung | Beschreibung  |
|----------------------|----------|-------------|---|
| LoadBackUp           | BOOL     | FALSE       | Wenn LoadBackUp = TRUE, wird der letzte Satz PID-Parameter wieder geladen. Der Satz wurde vor der letzten Optimierung gespeichert. LoadBackUp wird automatisch wieder auf FALSE gesetzt.  |
| PhysicalUnit         | INT      | 0           | Physikalische Einheit des Ist- und Sollwerts, z. B. °C oder °F.   |
| PhysicalQuantity     | INT      | 0           | Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur  |
| ErrorBehaviour       | BOOL     | FALSE       | Wenn ErrorBehaviour = FALSE, bleibt das Ventil im Fehlerfall in der aktuellen Position und der Regler wechselt direkt in die Betriebsart "Inaktiv" oder "Fehlerüberwachung".<br>Wenn ErrorBehaviour = TRUE, wird im Fehlerfall das Stellglied auf den Ersatzausgangswert gefahren und erst dann in die Betriebsart "Inaktiv" oder "Fehlerüberwachung" gewechselt.<br>Wenn die folgenden Fehler auftreten, kann das Ventil nicht mehr auf einen konfigurierten Ersatzausgangswert gefahren werden. <ul style="list-style-type: none"> <li>• 2000h: Ungültiger Wert am Parameter Feedback_PER.</li> <li>• 4000h: Ungültiger Wert am Parameter Feedback.</li> <li>• 8000h: Fehler bei der digitalen Stellungsrückmeldung.</li> <li>• 20000h: Ungültiger Wert an der Variablen SavePosition.</li> </ul> |
| Warning              | DWORD    | DW#16#0     | Die Variable Warning (Seite 2740) zeigt die Warnungen seit Reset = TRUE oder ErrorAck = TRUE. Warning ist remanent.<br>Zyklische Warnungen (z.B. Istwertwarnung) werden solange angezeigt wie die Ursache der Warnung anliegt. Entfällt der Grund, werden sie automatisch gelöscht. Nichtzyklische Warnungen (z.B. Wendepunkt nicht gefunden) bleiben bestehen und werden wie Fehler gelöscht.  |
| SavePosition         | REAL     | 0.0         | Ersatzausgangswert<br>Wenn ErrorBehaviour = TRUE, wird das Stellglied im Fehlerfall in eine für die Anlage sichere Position gefahren. Sobald der Ersatzausgangswert erreicht wurde, wechselt PID_3Step in Abhängigkeit von ActivateRecoverMode die Betriebsart.   |
| CurrentSetpoint      | REAL     | 0.0         | Aktuell aktiver Sollwert. Dieser Wert wird beim Start der Optimierung eingefroren.  |
| CancelTuningLevel    | REAL     | 10.0        | Zulässige Schwankung des Sollwerts während der Optimierung. Die Optimierung wird erst abgebrochen, wenn gilt: <ul style="list-style-type: none"> <li>• Setpoint &gt; CurrentSetpoint + CancelTuningLevel</li> <li>oder</li> <li>• Setpoint &lt; CurrentSetpoint - CancelTuningLevel</li> </ul>  |
| Progress             | REAL     | 0.0         | Fortschritt der Optimierung in Prozent (0.0 - 100.0)  |
| Config.InputPerOn    | BOOL     | TRUE        | Wenn InputPerOn = TRUE, wird der Parameter Input_PER verwendet. Wenn InputPerOn = FALSE, wird der Parameter Input verwendet.  |
| Config.OutputPerOn   | BOOL     | FALSE       | Wenn OutputPerOn = TRUE, wird der Parameter Output_PER verwendet. Wenn OutputPerOn = FALSE, werden die Parameter Output_UP und Output_DN verwendet.   |
| Config.InvertControl | BOOL     | FALSE       | Invertieren des Regelsinns<br>Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts.   |

| Variable                 | Datentyp | Vorbelegung       | Beschreibung  |
|--------------------------|----------|-------------------|---|
| Config.FeedbackOn        | BOOL     | FALSE             | Wenn FeedbackOn = FALSE, wird eine Stellungsrückmeldung simuliert.<br>Wenn FeedbackOn = TRUE, wird generell die Stellungsrückmeldung aktiviert.   |
| Config.FeedbackPerOn     | BOOL     | FALSE             | FeedbackPerOn ist nur wirksam, wenn FeedbackOn = TRUE.<br>Wenn FeedbackPerOn = TRUE, wird der Analogeingang für die Stellungsrückmeldung verwendet (Parameter Feedback_PER).<br>Wenn FeedbackPerOn = FALSE, wird der Parameter Feedback für die Stellungsrückmeldung verwendet.   |
| Config.ActuatorEndStopOn | BOOL     | FALSE             | Wenn ActuatorEndStopOn = TRUE, wird die digitale Stellungsrückmeldung Actuator_L und Actuator_H berücksichtigt.   |
| Config.InputUpperLimit   | REAL     | 120.0             | Obergrenze des Istwerts<br>Input und Input_PER werden auf die Einhaltung dieser Grenze überwacht.<br>Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird kein Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_3Step verhält sich wie unter Verhalten im Fehlerfall konfiguriert.<br>InputUpperLimit > InputLowerLimit |
| Config.InputLowerLimit   | REAL     | 0.0               | Untergrenze des Istwerts<br>InputLowerLimit < InputUpperLimit   |
| Config.InputUpperWarning | REAL     | +3.40282<br>2e+38 | Obere Warngrenze des Istwerts<br>Wenn Sie InputUpperWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Obergrenze Istwert als obere Warngrenze verwendet.<br>Wenn Sie InputUpperWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als obere Warngrenze verwendet.<br>InputUpperWarning > InputLowerWarning<br>InputUpperWarning ≤ InputUpperLimit   |
| Config.InputLowerWarning | REAL     | -3.40282<br>2e+38 | Untere Warngrenze des Istwerts<br>Wenn Sie InputLowerWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Untergrenze Istwert als Untere Warngrenze verwendet.<br>Wenn Sie InputLowerWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet.<br>InputLowerWarning < InputUpperWarning<br>InputLowerWarning ≥ InputLowerLimit   |
| Config.OutputUpperLimit  | REAL     | 100.0             | Obergrenze des Ausgangswerts<br>Details siehe OutputLowerLimit  |
| Config.OutputLowerLimit  | REAL     | 0.0               | Untergrenze des Ausgangswerts<br>Wenn OutputPerOn = TRUE oder FeedbackOn = TRUE, gilt der Wertebereich von -100 bis +100 % einschließlich Null. Bei -100 % beträgt Output = -27648; bei +100 % Output = 27648<br>Wenn OutputPerOn = FALSE, gilt der Wertebereich von 0 bis 100 %. Bei 0 % ist das Ventil vollständig geschlossen, bei 100 % vollständig geöffnet.   |

| Variable                          | Datentyp | Vorbelegung   | Beschreibung  |
|-----------------------------------|----------|---------------|---|
| Config.SetpointUpperLimit         | REAL     | +3.402822e+38 | Obergrenze des Sollwerts<br>Wenn Sie SetpointUpperLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert vorbelegt.<br>Wenn Sie SetpointUpperLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.   |
| Config.SetpointLowerLimit         | REAL     | -3.402822e+38 | Untergrenze des Sollwerts<br>Wenn Sie SetpointLowerLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert vorbelegt.<br>Wenn Sie SetpointLowerLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.   |
| Config.MinimumOnTime              | REAL     | 0.0           | Minimale Einschaltzeit<br>Zeit in Sekunden, die der Stellantrieb mindestens eingeschaltet sein muss.  |
| Config.MinimumOffTime             | REAL     | 0.0           | Minimale Ausschaltzeit<br>Zeit in Sekunden, die der Stellantrieb mindestens ausgeschaltet sein muss.  |
| Config.VirtualActuatorLimit       | REAL     | 150.0         | Wenn alle folgenden Bedingungen erfüllt sind, wird das Stellglied maximal für die Dauer von VirtualActuatorLimit × Retain.TransitTime/100 in eine Richtung bewegt und die Warnung 2000h ausgegeben: <ul style="list-style-type: none"> <li>• Config.OutputPerOn = FALSE</li> <li>• Config.ActuatorEndStopOn = FALSE</li> <li>• Config.FeedbackOn = FALSE</li> </ul> Wenn Config.OutputPerOn = FALSE und Config.ActuatorEndStopOn = TRUE oder Config.FeedbackOn = TRUE sind, wird nur die Warnung 2000h ausgegeben.<br>Wenn Config.OutputPerOn = TRUE ist, wird VirtualActuatorLimit nicht berücksichtigt. |
| Config.InputScaling.UpperPointIn  | REAL     | 27648.0       | Skalierung Input_PER Oben<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.  |
| Config.InputScaling.LowerPointIn  | REAL     | 0.0           | Skalierung Input_PER Unten<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.   |
| Config.InputScaling.UpperPointOut | REAL     | 100.0         | Skalierter oberer Istwert<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.  |
| Config.InputScaling.LowerPointOut | REAL     | 0.0           | Skalierter unterer Istwert<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.   |

| Variable                             | Datentyp | Vorbelegung | Beschreibung  |
|--------------------------------------|----------|-------------|---|
| Config.FeedbackScaling.UpperPointIn  | REAL     | 27648.0     | Skalierung Feedback_PER Oben<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.   |
| Config.FeedbackScaling.LowerPointIn  | REAL     | 0.0         | Skalierung Feedback_PER Unten<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.  |
| Config.FeedbackScaling.UpperPointOut | REAL     | 100.0       | Oberer Anschlag<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.  |
| Config.FeedbackScaling.LowerPointOut | REAL     | 0.0         | Unterer Anschlag<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.   |
| GetTransitTime.InvertDirection       | BOOL     | FALSE       | Wenn InvertDirection = FALSE, wird das Ventil zum Ermitteln der Stellzeit vollständig geöffnet, geschlossen und wieder geöffnet.<br>Wenn InvertDirection = TRUE, wird das Ventil vollständig geschlossen, geöffnet und wieder geschlossen.  |
| GetTransitTime.SelectFeedback        | BOOL     | FALSE       | Wenn SelectFeedback = TRUE, wird Feedback_PER oder Feedback bei der Stellzeitmessung berücksichtigt.<br>Wenn SelectFeedback = FALSE, wird Actuator_H und Actuator_L bei der Stellzeitmessung berücksichtigt.  |
| GetTransitTime.State                 | INT      | 0           | Aktuelle Phase der Stellzeitmessung <ul style="list-style-type: none"> <li>• State = 0: Inaktiv</li> <li>• State = 1: Ventil vollständig öffnen</li> <li>• State = 2: Ventil vollständig schließen</li> <li>• State = 3: Ventil auf Zielstellung (NewOutput) stellen</li> <li>• State = 4: Stellzeitmessung erfolgreich beendet</li> <li>• State = 5: Stellzeitmessung abgebrochen</li> </ul> |
| GetTransitTime.NewOutput             | REAL     | 0.0         | Zielstellung für die Stellzeitmessung mit Stellungsrückmeldung<br>Die Zielstellung muss innerhalb "Oberer Anschlag" und "Unterer Anschlag" liegen. Die Differenz zwischen NewOutput und ScaledFeedback muss mindestens 50 % des zulässigen Stellbereichs sein.  |
| CycleTime.StartEstimation            | BOOL     | TRUE        | Wenn StartEstimation = TRUE, wird die Messung der Abtastzeit PID_3Step gestartet. Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE.  |
| CycleTime.EnEstimation               | BOOL     | TRUE        | Wenn EnEstimation = TRUE, wird die Abtastzeit PID_3Step berechnet.<br>Wenn CycleTime.EnEstimation = FALSE, wird die Abtastzeit PID_3Step nicht berechnet und Sie müssen CycleTime.Value manuell korrekt konfigurieren.  |

| Variable                        | Datentyp | Vorbelegung | Beschreibung   |
|---------------------------------|----------|-------------|--|
| CycleTime.EnMonitoring          | BOOL     | TRUE        | Wenn EnMonitoring = TRUE, wird die Abtastzeit PID_3Step überwacht. Wenn PID_3Step nicht innerhalb der Abtastzeit ausgeführt werden kann, wird der Fehler 0800h ausgegeben und die Betriebsart gewechselt. In welche Betriebsart gewechselt wird, hängt ab von ActivateRecoverMode und ErrorBehaviour.<br>Wenn EnMonitoring = FALSE, wird die Abtastzeit PID_3Step nicht überwacht, der Fehler 0800h nicht ausgegeben und die Betriebsart nicht gewechselt. |
| CycleTime.Value                 | REAL     | 0.1         | Abtastzeit PID_3Step in Sekunden<br>CycleTime.Value wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.   |
| CtrlParamsBackUp.SetByUser      | BOOL     | FALSE       | Gespeicherter Wert von Retain.CtrlParams.SetByUser<br>Werte aus der Struktur CtrlParamsBackUp können mit LoadBackUp = TRUE wieder geladen werden.  |
| CtrlParamsBackUp.Gain           | REAL     | 1.0         | Gespeicherte Proportionalverstärkung   |
| CtrlParamsBackUp.Ti             | REAL     | 20.0        | Gespeicherte Integrationszeit in Sekunden  |
| CtrlParamsBackUp.Td             | REAL     | 0.0         | Gespeicherte Differenzierzeit in Sekunden  |
| CtrlParamsBackUp.TdFilterRatio  | REAL     | 0.0         | Gespeicherter Koeffizient für den Differenzierverszug  |
| CtrlParamsBackUp.PWeighting     | REAL     | 0.0         | Gespeicherte Gewichtung des P-Anteils  |
| CtrlParamsBackUp.DWeighting     | REAL     | 0.0         | Gespeicherte Gewichtung des D-Anteils  |
| CtrlParamsBackUp.Cycle          | REAL     | 1.0         | Gespeicherte Abtastzeit PID-Algorithmus in Sekunden  |
| CtrlParamsBackUp.InputDeadBand  | REAL     | 0.0         | Gespeicherte Totzonenbreite der Regeldifferenz   |
| PIDSelfTune.SUT.CalculateParams | BOOL     | FALSE       | Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn CalculateParams = TRUE, werden anhand dieser Eigenschaften die PID-Parameter neu berechnet. Die PID-Parameter werden nach der Methode berechnet, die in TuneRule eingestellt ist. CalculateParams wird nach der Berechnung auf FALSE gesetzt.  |
| PIDSelfTune.SUT.TuneRule        | INT      | 1           | Parameter während Ersoptimierung berechnen nach Methode: <ul style="list-style-type: none"> <li>• SUT.TuneRule = 0: PID schnell I</li> <li>• SUT.TuneRule = 1: PID langsam I</li> <li>• SUT.TuneRule = 2: Chien, Hrones, Reswick PID</li> <li>• SUT.TuneRule = 3: Chien, Hrones, Reswick PI</li> <li>• SUT.TuneRule = 4: PID schnell II</li> <li>• SUT.TuneRule = 5: PID langsam II</li> </ul>   |

| Variable                        | Datentyp | Vorbelegung | Beschreibung   |
|---------------------------------|----------|-------------|--|
| PIDSelfTune.SUT.State           | INT      | 0           | <p>Die Variable SUT.State zeigt die aktuelle Phase der Erstopptimierung:</p> <ul style="list-style-type: none"> <li>• State = 0: Erstopptimierung initialisieren</li> <li>• State = 50: Startposition ohne Stellungsrückmeldung ermitteln</li> <li>• State = 100: Standardabweichung berechnen</li> <li>• State = 200: Wendepunkt ermitteln</li> <li>• State = 300: Anregelzeit ermitteln</li> <li>• State = 9900: Erstopptimierung erfolgreich</li> <li>• State = 1: Erstopptimierung nicht erfolgreich</li> </ul>  |
| PIDSelfTune.TIR.RunIn           | BOOL     | FALSE       | <p>Mit der Variable RunIn können Sie festlegen, dass eine Nachoptimierung auch ohne Erstopptimierung durchgeführt wird.</p> <ul style="list-style-type: none"> <li>• RunIn = FALSE<br/>Wenn die Nachoptimierung aus der Betriebsart Inaktiv oder Handbetrieb gestartet wird, wird eine Erstopptimierung gestartet. Wenn die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt. Erst dann startet die Nachoptimierung. Ist die Erstopptimierung nicht möglich, wechselt PID_3Step in die Betriebsart, aus der die Optimierung gestartet wurde.</li> <li>• RunIn = TRUE<br/>Die Erstopptimierung wird übersprungen. PID_3Step versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Erst dann startet die Nachoptimierung. RunIn wird nach der Nachoptimierung auf FALSE gesetzt.</li> </ul> |
| PIDSelfTune.TIR.CalculateParams | BOOL     | FALSE       | <p>Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn CalculateParams = TRUE, werden anhand dieser Eigenschaften die PID-Parameter neu berechnet. Die PID-Parameter werden nach der Methode berechnet, die in TuneRule eingestellt ist. CalculateParams wird nach der Berechnung auf FALSE gesetzt.</p>   |
| PIDSelfTune.TIR.TuneRule        | INT      | 0           | <p>Parameter während Nachoptimierung berechnen nach Methode:</p> <ul style="list-style-type: none"> <li>• TIR.TuneRule = 0: PID automatisch</li> <li>• TIR.TuneRule = 1: PID schnell</li> <li>• TIR.TuneRule = 2: PID langsam</li> <li>• TIR.TuneRule = 3: Ziegler-Nichols PID</li> <li>• TIR.TuneRule = 4: Ziegler-Nichols PI</li> <li>• TIR.TuneRule = 5: Ziegler-Nichols P</li> </ul>   |

| Variable                    | Datentyp | Vorbelegung | Beschreibung   |
|-----------------------------|----------|-------------|--|
| PIDSelfTune.TIR.State       | INT      | 0           | <p>Die Variable TIR.State zeigt die aktuelle Phase der "Nachoptimierung":</p> <ul style="list-style-type: none"> <li>• State = -100 Die Nachoptimierung ist nicht möglich. Es wird erst eine Erstopoptimierung durchgeführt.</li> <li>• State = 0: Nachoptimierung initialisieren</li> <li>• State = 200: Standardabweichung berechnen</li> <li>• State = 300: Versuchen Sollwert mit dem maximalen oder minimalen Ausgangswert zu erreichen</li> <li>• State = 400: Versuchen Sollwert mit vorhandenen PID-Parametern zu erreichen (wenn Erstopoptimierung erfolgreich war)</li> <li>• State = 500: Oszillation ermitteln und Parameter berechnen</li> <li>• State = 9900: Nachoptimierung erfolgreich</li> <li>• State = 1: Nachoptimierung nicht erfolgreich</li> </ul> |
| Retain.TransitTime          | REAL     | 30.0        | <p>Motorstellzeit in Sekunden</p> <p>Zeit in Sekunden, die der Stellantrieb benötigt, um das Ventil vom geschlossenen in den geöffneten Zustand zu bewegen.</p> <p>TransitTime ist remanent.</p>   |
| Retain.CtrlParams.SetByUser | BOOL     | FALSE       | <p>Wenn SetByUser = FALSE, sind die PID-Parameter automatisch ermittelt und PID_3Step arbeitet mit einer Totzone am Ausgangswert. Die Totzonenbreite wird während der Optimierung anhand der Standardabweichung des Ausgangswerts berechnet und in Retain.CtrlParams.OutputDeadBand gespeichert.</p> <p>Wenn SetByUser = TRUE, sind die PID-Parameter manuell eingegeben und PID_3 Step arbeitet ohne Totzone am Ausgangswert. Retain.CtrlParams.OutputDeadBand = 0.0</p> <p>SetByUser ist remanent.</p>   |
| Retain.CtrlParams.Gain      | REAL     | 1.0         | <p>Aktive Proportionalverstärkung</p> <p>Verwenden Sie zum Invertieren des Regelsinns die Variable Config.InvertControl. Negative Werte an Gain invertieren den Regelsinn ebenfalls. Es wird empfohlen den Regelsinn nur über InvertControl einzustellen. Wenn InvertControl = TRUE und Gain &lt; 0.0 sind, ist der Regelsinn auch invertiert.</p> <p>Gain ist remanent.</p>   |
| Retain.CtrlParams.Ti        | REAL     | 20.0        | <ul style="list-style-type: none"> <li>• Ti &gt; 0.0: Aktive Integrationszeit in Sekunden</li> <li>• Ti = 0.0: I-Anteil ist ausgeschaltet</li> </ul> <p>Ti ist remanent.</p>   |
| Retain.CtrlParams.Td        | REAL     | 0.0         | <ul style="list-style-type: none"> <li>• Td &gt; 0.0: Aktive Differenzierzeit in Sekunden</li> <li>• Td = 0.0: D-Anteil ist ausgeschaltet</li> </ul> <p>Td ist remanent.</p>   |



| Variable                        | Datentyp | Vorbelegung | Beschreibung  |
|---------------------------------|----------|-------------|---|
| Retain.CtrlParams.TdFiltRatio   | REAL     | 0.2         | <p>Aktiver Koeffizient für den Differenzierverzögerung</p> <p>Die Wirkung des D-Anteils wird durch den Koeffizient Differenzierverzögerung verzögert.</p> <p>Differenzierverzögerung = Differenzierzeit × Koeffizient Differenzierverzögerung</p> <ul style="list-style-type: none"> <li>• 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.</li> <li>• 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit <b>einer</b> dominierenden Zeitkonstanten bewährt.</li> <li>• &gt; 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.</li> </ul> <p>TdFiltRatio ist remanent.</p> |
| Retain.CtrlParams.PWeighting    | REAL     | 1.0         | <p>Aktive Gewichtung des P-Anteils</p> <p>Sie können bei Sollwertänderungen den P-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0.</p> <ul style="list-style-type: none"> <li>• 1.0: P-Anteil bei Sollwertänderung voll wirksam</li> <li>• 0.0: P-Anteil bei Sollwertänderung nicht wirksam</li> </ul> <p>Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam. PWeighting ist remanent.</p>   |
| Retain.CtrlParams.DWeighting    | REAL     | 1.0         | <p>Aktive Gewichtung des D- Anteils</p> <p>Sie können bei Sollwertänderungen den D-Anteil abschwächen. Sinnvoll sind Werte von 0.0 bis 1.0.</p> <ul style="list-style-type: none"> <li>• 1.0: D-Anteil bei Sollwertänderung voll wirksam</li> <li>• 0.0: D-Anteil bei Sollwertänderung nicht wirksam</li> </ul> <p>Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam. DWeighting ist remanent.</p>  |
| Retain.CtrlParams.Cycle         | REAL     | 1.0         | <p>Aktive Abtastzeit PID-Algorithmus in Sekunden, die zu einem ganzzahligen Vielfachen der Zykluszeit des aufrufenden OB aufgerundet wird.</p> <p>Cycle ist remanent.</p>   |
| Retain.CtrlParams.InputDeadBand | REAL     | 0.0         | <p>Totzonenbreite der Regeldifferenz</p> <p>InputDeadBand ist remanent.</p>   |

---

### Hinweis

Verändern Sie die hier aufgelisteten Variablen in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden.

---

**Siehe auch**

- Parameter State und Mode V2 (Seite 2740)
- Variable ActivateRecoverMode V2 (Seite 2747)
- Technologieobjekte in Gerät laden (Seite 4056)

**Parameter State und Mode V2**

**Zusammenhang der Parameter**

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Mit einer steigenden Flanke an ModeActivate wechselt PID\_3Step in die Betriebsart, die am Durchgangsparameter Mode gespeichert ist.

Wenn die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID\_3Step in der Betriebsart, die an Mode gespeichert ist. Um PID\_3Step in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

**Bedeutung der Werte**

| State | Beschreibung der Betriebsart  |
|-------|---|
| 0     | Inaktiv<br>Der Regler ist ausgeschaltet und verändert die Position des Ventils nicht mehr.  |
| 1     | Erstoptimierung<br>Die Erstoptimierung ermittelt die Prozessantwort auf einen Impuls des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.<br>Voraussetzungen für die Erstoptimierung: <ul style="list-style-type: none"> <li>• Die Motorstellzeit ist konfiguriert oder gemessen.</li> <li>• Betriebsart Inaktiv (State = 0), Handbetrieb (State = 4) oder Automatikbetrieb (State = 3)</li> <li>• ManualEnable = FALSE</li> <li>• Reset = FALSE</li> <li>• Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen.</li> </ul> <p>Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Dies ist am ehesten in den Betriebsarten "Inaktiv" oder "Handbetrieb" gegeben.</p> <p>Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt:</p> <ul style="list-style-type: none"> <li>• <math>Setpoint &gt; CurrentSetpoint + CancelTuningLevel</math><br/>oder</li> <li>• <math>Setpoint &lt; CurrentSetpoint - CancelTuningLevel</math></li> </ul> <p>Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit LoadBackUp reaktiviert werden.</p> <p>Nach erfolgreicher Erstoptimierung wechselt der Regler in den Automatikbetrieb; nach erfolgloser Erstoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode und ErrorBehaviour.</p> <p>Die Phase der Erstoptimierung wird mit Variable SUT.State angezeigt.</p> |

| State | Beschreibung der Betriebsart  |
|-------|---|
| 2     | <p>Nachoptimierung</p> <p>Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstop Optimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen. PID_3Step versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwertes. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwertes beeinflusst.</p> <p>Der Sollwert wird in der Variablen CurrentSetpoint eingefroren. Die Optimierung wird abgebrochen, wenn gilt:</p> <ul style="list-style-type: none"> <li>• Setpoint &gt; CurrentSetpoint + CancelTuningLevel<br/>oder</li> <li>• Setpoint &lt; CurrentSetpoint - CancelTuningLevel</li> </ul> <p>Vor der Nachoptimierung werden die PID-Parameter gesichert. Sie können mit LoadBackUp reaktiviert werden.</p> <p>Voraussetzungen für die Nachoptimierung:</p> <ul style="list-style-type: none"> <li>• Die Motorstellzeit ist konfiguriert oder gemessen.</li> <li>• Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen.</li> <li>• ManualEnable = FALSE</li> <li>• Reset = FALSE</li> <li>• Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4)</li> </ul> <p>Die Nachoptimierung verläuft folgendermaßen beim Start aus:</p> <ul style="list-style-type: none"> <li>• Automatikbetrieb (State = 3)<br/>Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.<br/>PID_3Step regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.</li> <li>• Inaktiv (State = 0) oder Handbetrieb (State = 4)<br/>Wenn die Voraussetzungen für eine Erstop Optimierung erfüllt sind, wird eine Erstop Optimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind.<br/>Wenn PIDSelfTune.TIR.RunIn = TRUE, wird die Erstop Optimierung übersprungen und versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet dann automatisch.</li> </ul> <p>Nach erfolgreicher Nachoptimierung wechselt der Regler in den Automatikbetrieb; nach erfolgloser Nachoptimierung ist der Wechsel der Betriebsart abhängig von ActivateRecoverMode und ErrorBehaviour.</p> <p>Die Phase der Nachoptimierung wird mit Variable TIR.State angezeigt.</p> |
| 3     | <p>Automatikbetrieb</p> <p>Im Automatikbetrieb regelt PID_3Step die Regelstrecke entsprechend der vorgegebenen Parameter aus. Wenn eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt:</p> <ul style="list-style-type: none"> <li>• Erstop Optimierung erfolgreich abgeschlossen</li> <li>• Nachoptimierung erfolgreich abgeschlossen</li> <li>• Ändern des Durchgangsparameters Mode auf den Wert 3 und eine steigende Flanke an ModeActivate.</li> </ul> <p>Der Wechsel vom Automatikbetrieb in den Handbetrieb erfolgt nur im Inbetriebnahmeeditor stoßfrei.</p> <p>Im Automatikbetrieb wird die Variable ActivateRecoverMode berücksichtigt.</p>   |

| State | Beschreibung der Betriebsart   |
|-------|--|
| 4     | <p>Handbetrieb</p> <p>Im Handbetrieb geben Sie manuelle Ausgangswerte an den Parametern Manual_UP und Manual_DN oder ManualValue vor. Ob das Stellglied im Fehlerfall auf den Ausgangswert verfahren werden kann, ist beim Parameter ErrorBits beschrieben.</p> <p>Diese Betriebsart können Sie zusätzlich über ManualEnable = TRUE aktivieren. Es wird empfohlen, die Betriebsarten nur über Mode und ModeActivate zu wechseln.</p> <p>Der Wechsel vom Handbetrieb in den Automatikbetrieb erfolgt stoßfrei. Der Handbetrieb ist auch möglich, wenn ein Fehler ansteht.</p>   |
| 5     | <p>Ersatzausgangswert anfahren</p> <p>Diese Betriebsart wird im Fehlerfall aktiviert, wenn Errorbehaviour = TRUE und ActivateRecoverMode = FALSE.</p> <p>PID_3Step fährt das Stellglied auf den Ersatzausgangswert und wechselt anschließend in die Betriebsart "Inaktiv".</p>   |
| 6     | <p>Stellzeitmessung</p> <p>Es wird die Zeit ermittelt, die der Motor benötigt, um das Ventil aus dem geschlossenen Zustand vollständig zu öffnen.</p> <p>Diese Betriebsart wird aktiviert, wenn Mode = 6 und ModeActivate = TRUE gesetzt wird.</p> <p>Wenn zur Stellzeitmessung Anschlagssignale verwendet werden, wird das Ventil von der aktuellen Position aus vollständig geöffnet, vollständig geschlossen und erneut vollständig geöffnet. Wenn GetTransitTime.InvertDirection = TRUE, wird dieses Verhalten umgekehrt.</p> <p>Wenn zur Stellzeitmessung eine Stellungsrückmeldung verwendet wird, wird das Stellglied von der aktuellen Stellung in eine Zielstellung gefahren.</p> <p>Die Ausgangswertgrenzen werden während der Stellzeitmessung nicht berücksichtigt. Das Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.</p> |
| 7     | <p>Fehlerüberwachung</p> <p>Der Regelalgorithmus ist ausgeschaltet und verändert die Position des Ventils nicht mehr.</p> <p>Diese Betriebsart wird im Fehlerfall statt der Betriebsart "Inaktiv" aktiviert.</p> <p>Alle folgenden Bedingungen müssen erfüllt sein:</p> <ul style="list-style-type: none"> <li>• Automatikbetrieb (Mode = 3)</li> <li>• Errorbehaviour = FALSE</li> <li>• ActivateRecoverMode = TRUE</li> <li>• Einer oder mehrere Fehler sind aufgetreten, bei denen ActivateRecoverMode (Seite 2747) wirkt.</li> </ul> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.</p>  |
| 8     | <p>Ersatzausgangswert anfahren mit Fehlerüberwachung</p> <p>Diese Betriebsart wird im Fehlerfall statt der Betriebsart "Ersatzausgangswert anfahren" aktiviert. PID_3Step fährt das Stellglied auf den Ersatzausgangswert und wechselt anschließend in die Betriebsart "Fehlerüberwachung".</p> <p>Alle folgenden Bedingungen müssen erfüllt sein:</p> <ul style="list-style-type: none"> <li>• Automatikbetrieb (Mode = 3)</li> <li>• Errorbehaviour = TRUE</li> <li>• ActivateRecoverMode = TRUE</li> <li>• Ein oder mehrere Fehler sind aufgetreten, bei denen ActivateRecoverMode (Seite 2747) wirkt.</li> </ul> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.</p>  |
| 10    | <p>Handbetrieb ohne Anschlagssignale</p> <p>Die Anschlagssignale werden <b>nicht</b> berücksichtigt, obwohl Config.ActuatorEndStopOn = TRUE ist. Die Ausgangswertgrenzen werden nicht berücksichtigt. Sonst verhält sich PID_3Step genauso wie im Handbetrieb.</p>   |

## ENO-Verhalten

Wenn State = 0 ist, dann ist ENO = FALSE.

Wenn State ≠ 0 ist, dann ist ENO = TRUE.

## Automatische Betriebsartwechsel während der Inbetriebnahme

Nach erfolgreicher Erst- oder Nachoptimierung wird der Automatikbetrieb aktiviert. Die folgende Tabelle zeigt, wie sich Mode und State während einer erfolgreichen Erstoptimierung ändern.

| Zyklus-Nr. | Mode | State | Aktion  |
|------------|------|-------|---|
| 0          | 4    | 4     | Mode = 1 setzen   |
| 1          | 1    | 4     | ModeActivate = TRUE setzen  |
| 1          | 4    | 1     | Wert von State wird an Mode gespeichert<br>Erstoptimierung wird gestartet |
| n          | 4    | 1     | Erstoptimierung erfolgreich beendet                                       |
| n          | 3    | 3     | Automatikbetrieb wird gestartet   |

Im Fehlerfall wechselt PID\_3Step automatisch die Betriebsart. Die folgende Tabelle zeigt, wie sich Mode und State während einer fehlerhaften Erstoptimierung ändern.

| Zyklus-Nr. | Mode | State | Aktion  |
|------------|------|-------|---|
| 0          | 4    | 4     | Mode = 1 setzen   |
| 1          | 1    | 4     | ModeActivate = TRUE setzen  |
| 1          | 4    | 1     | Wert von State wird an Mode gespeichert<br>Erstoptimierung wird gestartet |
| n          | 4    | 1     | Erstoptimierung abgebrochen   |
| n          | 4    | 4     | Handbetrieb wird gestartet  |

Wenn ActivateRecoverMode = TRUE ist, wird die Betriebsart aktiviert, die an Mode gespeichert ist. Beim Start der Stellzeitmessung, Erst- oder Nachoptimierung hat PID\_3Step den Wert von State am Durchgangparameter Mode gespeichert. PID\_3Step wechselt also in die Betriebsart, aus der die Stellzeitmessung oder Optimierung gestartet wurde.

Wenn ActivateRecoverMode = FALSE ist, wird die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren" aktiviert.

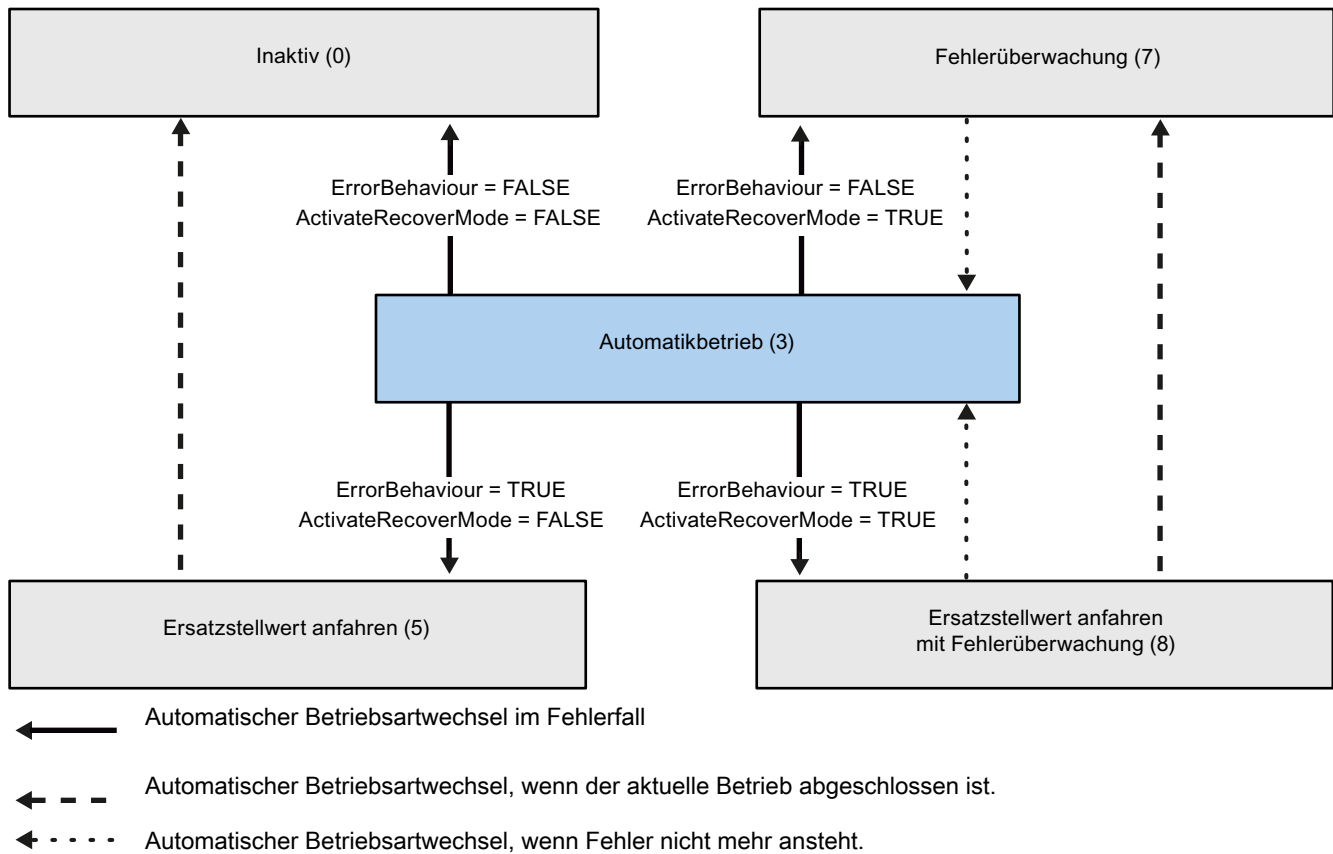
## Automatischer Betriebsartwechsel nach der Stellzeitmessung

Wenn ActivateRecoverMode = TRUE ist, wird nach erfolgreicher Stellzeitmessung die Betriebsart aktiviert, die an Mode gespeichert ist.

Wenn ActivateRecoverMode = FALSE ist, wird nach erfolgreicher Stellzeitmessung in die Betriebsart "Inaktiv" gewechselt.

### Automatische Betriebsartwechsel im Automatikbetrieb

Im Fehlerfall wechselt PID\_3Step automatisch die Betriebsart. Das folgende Diagramm zeigt den Einfluss von ErrorBehaviour und ActivateRecoverMode auf diesen Betriebsartwechsel.



### Siehe auch

Variable ActivateRecoverMode V2 (Seite 2747)

Parameter ErrorBits V2 (Seite 2744)

### Parameter ErrorBits V2

Stehen gleichzeitig mehrere Fehler an, so werden die Werte der ErrorBits binär addiert angezeigt. Wird z. B. ErrorBits = 0003h angezeigt, so stehen gleichzeitig die Fehler 0001h und 0002h an.

Wenn eine Stellungsrückmeldung vorhanden ist, verwendet PID\_3Step im Handbetrieb ManualValue als Ausgangswert. Ausnahme ist Errorbits = 1000h.

| ErrorBits<br>(DW#16#...) | Beschreibung  |
|--------------------------|---|
| 0000                     | Es steht kein Fehler an.  |
| 0001                     | Der Parameter "Input" ist außerhalb der Istwertgrenzen. <ul style="list-style-type: none"> <li>• Input &gt; Config.InputUpperLimit oder</li> <li>• Input &lt; Config.InputLowerLimit</li> </ul> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_3Step im Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erstopptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p>  |
| 0002                     | Ungültiger Wert am Parameter "Input_PER". Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung". Sobald der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erstopptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p> |
| 0004                     | Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden. <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.</p>  |
| 0010                     | Der Sollwert wurde während der Optimierung verändert. <p>An der Variable CancelTuningLevel können Sie die zulässige Schwankung des Sollwerts einstellen.</p> <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.</p>  |
| 0020                     | Die Erstopptimierung ist während der Nachoptimierung nicht erlaubt. <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bleibt PID_3Step in der Betriebsart Nachoptimierung.</p>  |
| 0080                     | Fehler bei der Erstopptimierung. Die Ausgangswertgrenzen sind nicht korrekt konfiguriert. <p>Überprüfen Sie, ob die Grenzen des Ausgangswerts korrekt konfiguriert sind und zum Regelsinn passen.</p> <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.</p>   |
| 0100                     | Fehler während der Nachoptimierung führte zu ungültigen Parametern. <p>Wenn vor Auftreten des Fehlers ActivateRecoverMode = TRUE war, bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, die an Mode gespeichert ist.</p>   |
| 0200                     | Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat. <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung". Sobald der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erstopptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p>                        |

| ErrorBits<br>(DW#16#...) | Beschreibung   |
|--------------------------|--|
| 0400                     | <p>Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung". Sobald der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Ersoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p>  |
| 0800                     | <p>Abtastzeitfehler: PID_3Step wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, bleibt PID_3Step im Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Ersoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p>  |
| 1000                     | <p>Ungültiger Wert am Parameter "Setpoint": Wert hat kein gültiges Zahlenformat.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE ist, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung". Sobald der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Ersoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p>   |
| 2000                     | <p>Ungültiger Wert am Parameter Feedback_PER.</p> <p>Überprüfen Sie, ob am Analogeingang ein Fehler ansteht.</p> <p>Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Im Handbetrieb können Sie die Position des Stellglieds nur mit Manual_UP und Manual_DN verändern, nicht mit ManualValue.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Ersoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p> |
| 4000                     | <p>Ungültiger Wert am Parameter Feedback. Wert hat kein gültiges Zahlenformat.</p> <p>Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Im Handbetrieb können Sie die Position des Stellglieds nur mit Manual_UP und Manual_DN verändern, nicht mit ManualValue.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Ersoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p>                               |



| ErrorBits<br>(DW#16#...) | Beschreibung  |
|--------------------------|---|
| 8000                     | <p>Fehler bei der digitalen Stellungsrückmeldung. Actuator_H = TRUE und Actuator_L = TRUE.</p> <p>Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Handbetrieb ist in diesem Zustand nicht möglich.</p> <p>Um das Stellglied aus diesem Zustand verfahren zu können, müssen Sie die "Anschlagsignale Stellglied" deaktivieren (Config.ActuatorEndStopOn = FALSE) oder in die Betriebsart "Handbetrieb ohne Anschlagsignale" (Mode = 10) wechseln.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erstopoptimierung, Nachoptimierung oder Stellzeitmessung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist.</p> |
| 10000                    | <p>Ungültiger Wert am Parameter ManualValue. Wert hat kein gültiges Zahlenformat.</p> <p>Das Stellglied kann nicht auf den Handwert verfahren werden und bleibt in der aktuellen Position stehen.</p> <p>Geben Sie einen gültigen Wert an ManualValue vor oder verfahren Sie das Stellglied im Handbetrieb mit Manual_UP und Manual_DN.</p>   |
| 20000                    | <p>Ungültiger Wert an der Variablen SavePosition. Wert hat kein gültiges Zahlenformat.</p> <p>Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen.</p>  |
| 40000                    | <p>Ungültiger Wert am Parameter Disturbance. Wert hat kein gültiges Zahlenformat.</p> <p>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war und ActivateRecoverMode = TRUE, wird Disturbance auf Null gesetzt. PID_3Step bleibt im Automatikbetrieb.</p> <p>Wenn vor Auftreten des Fehlers die Betriebsart Erst- oder Nachoptimierung aktiv und ActivateRecoverMode = TRUE waren, wechselt PID_3Step in die Betriebsart, die an Mode gespeichert ist. Wenn Disturbance in der aktuellen Phase keinen Einfluss auf den Ausgangswert hat, wird die Optimierung nicht abgebrochen.</p> <p>Während der Stellzeitmessung hat der Fehler keinen Einfluss.</p>  |

## Variable ActivateRecoverMode V2

Die Variable ActivateRecoverMode bestimmt das Verhalten im Fehlerfall. Der Parameter Error zeigt, ob aktuell ein Fehler ansteht. Wenn der Fehler nicht mehr ansteht, wird Error = FALSE. Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind.

### ACHTUNG

#### Ihre Anlage kann beschädigt werden.

Wenn ActivateRecoverMode = TRUE ist, bleibt PID\_3Step auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden.

Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

### Automatikbetrieb

| ActivateRecover Mode | Beschreibung  |
|----------------------|---|
| FALSE                | Im Fehlerfall wechselt PID_3Step in die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.  |
| TRUE                 | <p>Wenn im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert, da PID_3Step bei jedem Fehlerfall zwischen berechnetem Ausgangswert und dem Ersatzausgangswert wechselt. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache.</p> <p>Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID_3Step im Automatikbetrieb:</p> <ul style="list-style-type: none"> <li>• 0001h: Der Parameter "Input" ist außerhalb der Istwertgrenzen.</li> <li>• 0800h: Abtastzeitfehler</li> <li>• 4000h: Ungültiger Wert am Parameter Disturbance.</li> </ul> <p>Wenn einer oder mehrere der folgenden Fehler auftreten, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung":</p> <ul style="list-style-type: none"> <li>• 0002h: Ungültiger Wert am Parameter Input_PER.</li> <li>• 0200h: Ungültiger Wert am Parameter Input.</li> <li>• 0400h: Berechnung des Ausgangswerts fehlgeschlagen.</li> <li>• 1000h: Ungültiger Wert am Parameter Setpoint.</li> </ul> <p>Wenn einer oder mehrere der folgenden Fehler auftreten, kann PID_3Step das Stellglied nicht mehr bewegen:</p> <ul style="list-style-type: none"> <li>• 2000h: Ungültiger Wert am Parameter Feedback_PER.</li> <li>• 4000h: Ungültiger Wert am Parameter Feedback.</li> <li>• 8000h: Fehler bei der digitalen Stellungsrückmeldung.</li> <li>• 20000h: Ungültiger Wert an der Variablen SavePosition. Wert hat kein gültiges Zahlenformat.</li> </ul> <p>Dieses Verhalten ist unabhängig von ErrorBehaviour.</p> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.</p> |

### Erstoptimierung, Nachoptimierung und Stellzeitmessung

| ActivateRecover Mode | Beschreibung   |
|----------------------|--|
| FALSE                | <p>Im Fehlerfall wechselt PID_3Step in die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate aktiviert.</p> <p>Nach erfolgreicher Stellzeitmessung wechselt der Regler in die Betriebsart Inaktiv.</p>   |
| TRUE                 | <p>Wenn der folgende Fehler auftritt, bleibt PID_3Step in der aktiven Betriebsart:</p> <ul style="list-style-type: none"> <li>• 0020h: Die Erstoptimierung ist während der Nachoptimierung nicht erlaubt.</li> </ul> <p>Die folgenden Fehler werden ignoriert:</p> <ul style="list-style-type: none"> <li>• 10000h: Ungültiger Wert am Parameter ManualValue.</li> <li>• 20000h: Ungültiger Wert an der Variable SavePosition.</li> </ul> <p>Bei allen anderen Fehlern bricht PID_3Step die Optimierung ab und wechselt in die Betriebsart, aus der die Optimierung gestartet wurde.</p> |

## Handbetrieb

Im Handbetrieb wirkt ActivateRecoverMode nicht.

## Siehe auch

Statische Variablen PID\_3Step V2 (Seite 2731)

Parameter State und Mode V2 (Seite 2740)

## Variable Warning V2

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Warnungen binär addiert angezeigt. Wird z. B. die Warnung 0005h angezeigt, so stehen gleichzeitig die Warnungen 0001h und 0004h an.

| Warning<br>(DW#16#...) | Beschreibung  |
|------------------------|---|
| 0000                   | Es steht keine Warnung an.  |
| 0001                   | Während der Erstopptimierung wurde der Wendepunkt nicht gefunden.   |
| 0004                   | Der Sollwert wurde begrenzt auf die eingestellten Grenzen.  |
| 0008                   | Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode TIR.TuneRule = 3 berechnet.   |
| 0010                   | Die Betriebsart konnte nicht geändert werden, da Reset = TRUE oder ManualEnable = TRUE.   |
| 0020                   | Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.   |
| 0040                   | Der Istwert hat eine seiner Warn Grenzen überschritten.   |
| 0080                   | Ungültiger Wert an Mode. Die Betriebsart wird nicht gewechselt.   |
| 0100                   | Der Handwert wurde begrenzt auf die Grenzen des Reglerausgangs.   |
| 0200                   | Die angegebene Regel zur Optimierung wird nicht unterstützt. Es werden keine PID-Parameter berechnet.   |
| 0400                   | Die Stellzeit kann nicht gemessen werden, da die Einstellungen des Stellglieds nicht zur gewählten Messmethode passen.  |
| 0800                   | Bei der Stellzeitmessung ist die Differenz zwischen der aktuellen Position und dem neuen Ausgangswert zu klein. Dies kann zu falschen Ergebnissen führen. Die Differenz zwischen aktuellem Ausgangswert und neuem Ausgangswert muss mindestens 50% des gesamten Stellbereichs betragen. |
| 1000                   | Der Ersatzausgangswert kann nicht erreicht werden, da er außerhalb der Ausgangswertgrenzen liegt.   |
| 2000                   | Das Stellglied wurde länger als $\text{Config.VirtualActuatorLimit} \times \text{Retain.TransitTime}$ in eine Richtung bewegt. Kontrollieren Sie, ob das Stellglied ein Anschlagsignal erreicht hat.  |

Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist:

- 0001h
- 0004h
- 0008h
- 0040h
- 0100h

Alle anderen Warnungen werden bei steigender Flanke an Reset oder ErrorAck gelöscht.

## PID\_3Step V1

### Beschreibung PID\_3Step

#### Beschreibung

Mit der Anweisung PID\_3Step konfigurieren Sie einen PID-Regler mit Selbstoptimierung für Ventile oder Stellglieder mit integrierendem Verhalten.

Folgende Betriebsarten sind möglich:

- Inaktiv
- Ersoptimierung
- Nachoptimierung
- Automatikbetrieb
- Handbetrieb
- Ersatzausgangswert anfahren
- Stellzeitmessung
- Ersatzausgangswert anfahren mit Fehlerüberwachung
- Fehlerüberwachung

Die Betriebsarten sind detailliert beschrieben beim Parameter State.

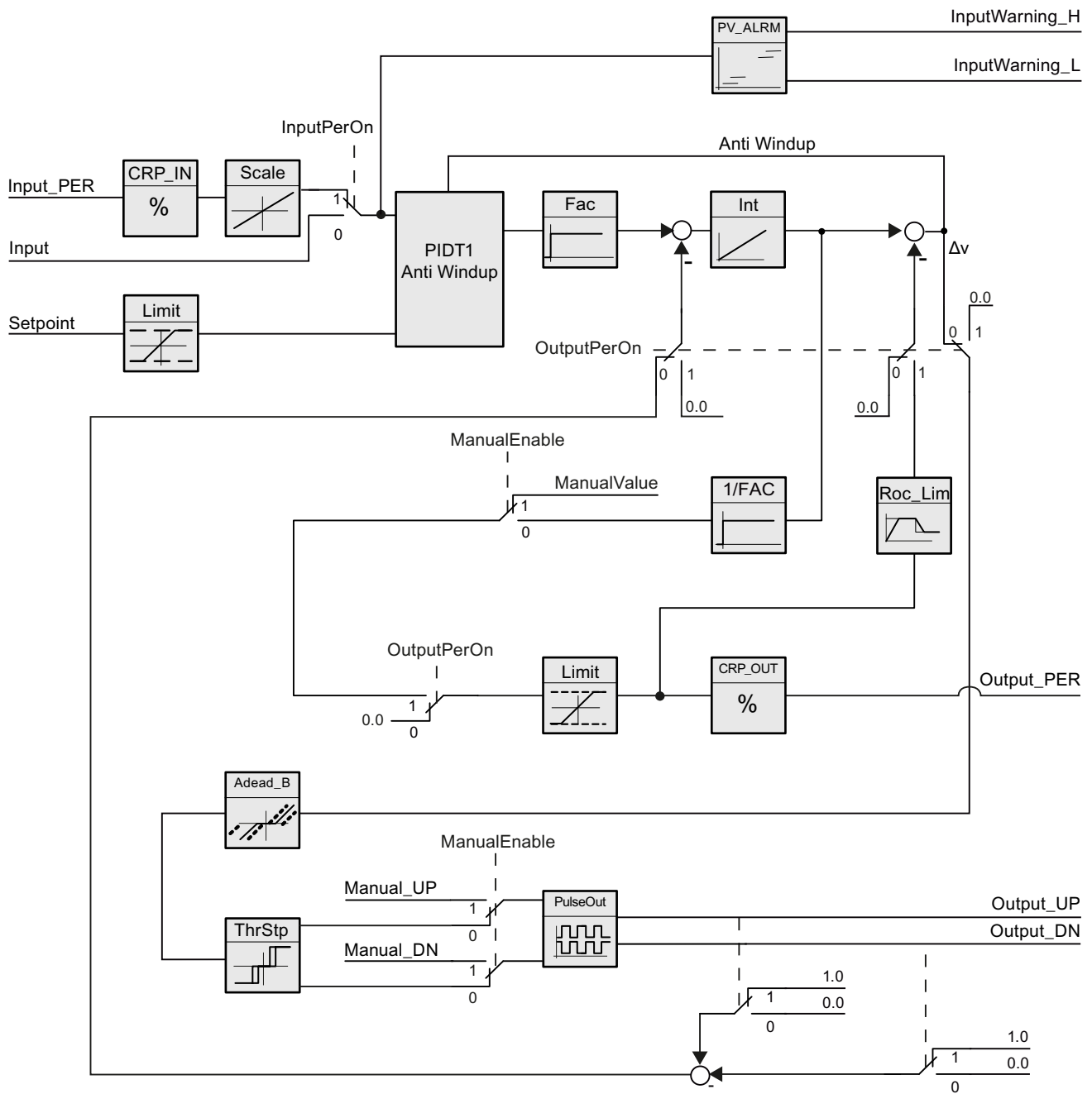
## PID-Algorithmus

PID\_3Step ist ein PIDT1-Regler mit Anti-Windup und Gewichtung des P- und D-Anteils. Der Ausgangswert wird anhand folgender Formel berechnet.

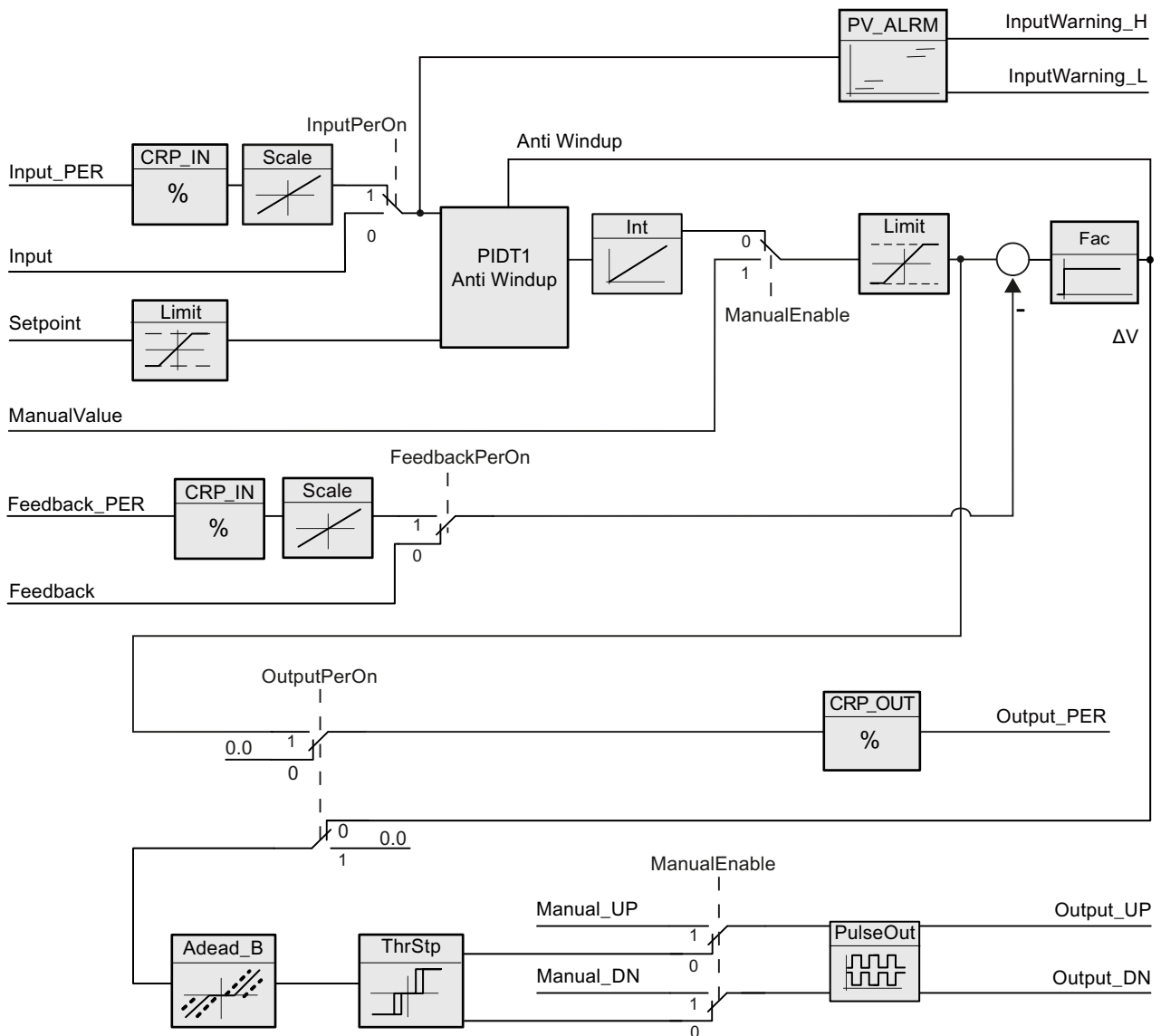
$$\Delta y = K_p \cdot s \cdot \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

| Symbol | Beschreibung  |
|--------|---|
| y      | Ausgangswert  |
| $K_p$  | Proportionalverstärkung   |
| s      | Laplace-Operator  |
| b      | Gewichtung des P-Anteils  |
| w      | Sollwert  |
| x      | Istwert   |
| $T_i$  | Integrationszeit  |
| a      | Koeffizient für den Differenzierversug ( $T_1 = a \times T_D$ ) |
| $T_D$  | Differenzierzeit  |
| c      | Gewichtung des D-Anteils  |

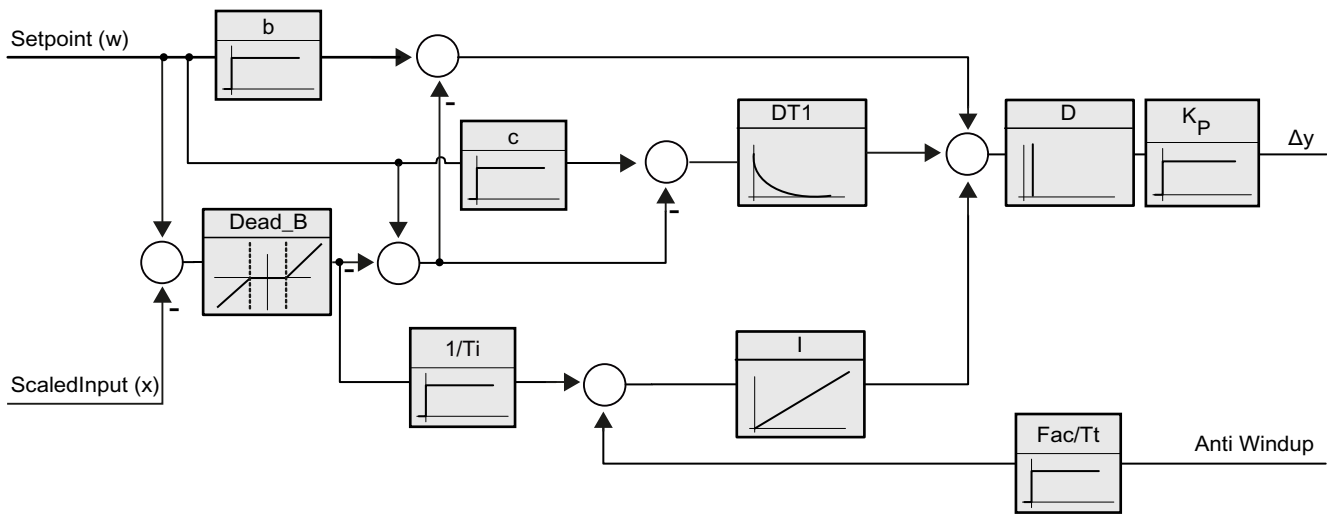
Blockschaltbild ohne Stellungsrückmeldung



Blockschaltbild mit Stellungsrückmeldung



**Blockschaltbild PIDT1 mit Anti-Windup**



**Aufruf**

PID\_3Step wird im konstanten Zeitraster der Zykluszeit des aufrufenden OBs aufgerufen (vorzugsweise in einem Weckalarm-OB).

**Laden in Gerät**

Die Aktualwerte remanenter Variablen werden nur aktualisiert, wenn Sie PID\_3Step vollständig laden.

Technologieobjekte in Gerät laden (Seite 4056)

**Anlauf**

Bei Anlauf der CPU startet PID\_3Step in der zuletzt aktiven Betriebsart. Um PID\_3Step in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.

**Verhalten im Fehlerfall**

Wenn Fehler auftreten, werden diese am Parameter Error ausgegeben. Das Verhalten von PID\_3Step konfigurieren Sie über die Variablen ErrorBehaviour und ActivateRecoverMode.

| ErrorBehaviour | ActivateRecoverMode | Konfiguration<br>Stellgliedeinstellung<br>Output setzen auf | Verhalten   |
|----------------|---------------------|---|---|
| 0              | FALSE               | Aktuellen Ausgangswert                                      | Wechsel in Betriebsart "Inaktiv" (Mode = 0)           |
| 0              | TRUE                | Aktuellen Ausgangswert für die Fehlerdauer                  | Wechsel in Betriebsart "Fehlerüberwachung" (Mode = 7) |



| ErrorBehavior | ActivateRecoverMode | Konfiguration<br>Stellgliedeinstellung<br>Output setzen auf | Verhalten   |
|---------------|---------------------|---|---|
| 1             | FALSE               | Ersatzausgangswert  | Wechsel in Betriebsart<br>"Ersatzausgangswert anfahren"<br>(Mode = 5)<br>Wechsel in Betriebsart "Inaktiv"<br>(Mode = 0)                                 |
| 1             | TRUE                | Ersatzausgangswert für die<br>Fehlerdauer                   | Wechsel in Betriebsart<br>"Ersatzausgangswert anfahren mit<br>Fehlerüberwachung" (Mode = 8)<br>Wechsel in Betriebsart<br>"Fehlerüberwachung" (Mode = 7) |

Der Parameter ErrorBits zeigt, welche Fehler aufgetreten sind.

### Siehe auch

Parameter State und Retain.Mode (Seite 2768)

Parameter ErrorBits (Seite 2776)

PID\_3Step V1 konfigurieren (Seite 4111)

### Arbeitsweise PID\_3Step

#### Istwertgrenzen überwachen

In den Variablen Config.InputUpperLimit und Config.InputLowerLimit legen Sie eine Ober- und Untergrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Grenzen liegt, tritt ein Fehler auf (ErrorBits = 0001hex).

In den Variablen Config.InputUpperWarning und Config.InputLowerWarning legen Sie eine obere und untere Warngrenze des Istwerts fest. Wenn der Istwert außerhalb dieser Warngrenzen liegt, tritt eine Warnung auf (Warnings = 0040hex) und der Ausgangsparameter InputWarning\_H oder InputWarning\_L wird TRUE.

#### Sollwert begrenzen

In den Variablen Config.SetpointUpperLimit und Config.SetpointLowerLimit legen Sie eine Ober- und Untergrenze des Sollwerts fest. PID\_3Step begrenzt den Sollwert automatisch auf die Istwertgrenzen. Sie können den Sollwert auf einen kleineren Bereich begrenzen. PID\_3Step prüft, ob dieser Bereich innerhalb der Istwertgrenzen liegt. Wenn der Sollwert außerhalb dieser Grenzen liegt, wird die Ober- oder Untergrenze als Sollwert verwendet und der Ausgangsparameter SetpointLimit\_H oder SetpointLimit\_L wird TRUE.

Der Sollwert wird in allen Betriebsarten begrenzt.

### Ausgangswert begrenzen

In den Variablen Config.OutputUpperLimit und Config.OutputLowerLimit legen Sie eine Ober- und Untergrenze des Ausgangswerts fest. Die Ausgangswertgrenzen müssen innerhalb "Unterer Anschlag" und "Oberer Anschlag" liegen.

- Oberer Anschlag: Config.FeedbackScaling.UpperPointOut
- Unter Anschlag: Config.FeedbackScaling.LowerPointOut

Es muss gelten:

$$\text{UpperPointOut} \geq \text{OutputUpperLimit} > \text{OutputLowerLimit} \geq \text{LowerPointOut}$$

Die gültigen Werte für "Oberer Anschlag" und "Unterer Anschlag" hängen ab von:

- FeedbackOn
- FeedbackPerOn
- OutputPerOn

| OutputPerOn | FeedbackOn | FeedbackPerOn | LowerPointOut               | UpperPointOut               |
|-------------|------------|---------------|-----------------------------|-----------------------------|
| FALSE       | FALSE      | FALSE         | nicht einstellbar (0.0 %)   | nicht einstellbar (100.0 %) |
| FALSE       | TRUE       | FALSE         | -100.0 % oder 0.0 %         | 0.0 % oder +100.0 %         |
| FALSE       | TRUE       | TRUE          | -100.0 % oder 0.0 %         | 0.0 % oder +100.0 %         |
| TRUE        | FALSE      | FALSE         | nicht einstellbar (100.0 %) | nicht einstellbar (100.0 %) |
| TRUE        | TRUE       | FALSE         | -100.0 % oder 0.0 %         | 0.0 % oder +100.0 %         |
| TRUE        | TRUE       | TRUE          | -100.0 % oder 0.0 %         | 0.0 % oder +100.0 %         |

Wenn OutputPerOn = FALSE und FeedbackOn = FALSE, können Sie den Ausgangswert nicht begrenzen. Die Digitalausgänge entweder bei Actuator\_H = TRUE oder Actuator\_L = TRUE oder nach einer Verfahrzeit von 110% der Motorstellzeit zurück gesetzt.

Der Ausgangswert beträgt 27648 bei 100 % und -27648 bei -100 %. PID\_3Step muss das Ventil vollständig schließen können. Daher muss die Null in den Ausgangswertgrenzen enthalten sein.

### Ersatzausgangswert

PID\_3Step kann im Fehlerfall einen Ersatzausgangswert ausgeben und das Stellglied in eine sichere Position stellen, die Sie an der Variablen SavePosition vorgeben. Der Ersatzausgangswert muss innerhalb der Ausgangswertgrenzen liegen.

### Gültigkeit der Signale überwachen

Die Werte der folgenden Parameter werden auf Gültigkeit überwacht:

- Setpoint
- Input
- Input\_PER
- Feedback

- Feedback\_PER
- Output

### Abtastzeit PID\_3Step überwachen

Die Abtastzeit entspricht im Idealfall der Zykluszeit des aufrufenden OB. Die Anweisung PID\_3Step misst jeweils den Zeitabstand zwischen zwei Aufrufen. Das ist die aktuelle Abtastzeit. Bei jedem Wechsel der Betriebsart und bei Erstanlauf wird der Mittelwert der ersten 10 Abtastzeiten gebildet. Wenn die aktuelle Abtastzeit zu stark von diesem Mittelwert abweicht, tritt ein Fehler auf (ErrorBits = 0800 hex).

Folgende Bedingungen versetzen PID\_3Step während der Optimierung in die Betriebsart "Inaktiv":

- Neuer Mittelwert  $\geq 1,1 \times$  alter Mittelwert
- Neuer Mittelwert  $\leq 0,9 \times$  alter Mittelwert

Folgende Bedingungen versetzen PID\_3Step bei Automatikbetrieb in die Betriebsart "Inaktiv":

- Neuer Mittelwert  $\geq 1,5 \times$  alter Mittelwert
- Neuer Mittelwert  $\leq 0,5 \times$  alter Mittelwert

### Abtastzeit des PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID\_3Step werden bei jedem Aufruf durchgeführt.

### Motorstellzeit messen

Die Motorstellzeit ist die Zeit in Sekunden, die der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen. Das Stellglied wird maximal 110% der Motorstellzeit in eine Richtung bewegt. PID\_3Step benötigt die Motorstellzeit so genau wie möglich, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren. Die Motorstellzeit können Sie während der Inbetriebnahme messen. Die Ausgangswertgrenzen werden während der Motorstellzeitmessung nicht berücksichtigt. Das Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

### Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn. Für Kühlungen und Abflussregelungen kann es notwendig sein, den Regelsinn zu invertieren. PID\_3Step arbeitet nicht mit negativer Proportionalverstärkung. Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts. Der Regelsinn wird auch während Erst- und Nachoptimierung berücksichtigt.

Siehe auch

PID\_3Step V1 konfigurieren (Seite 4111)

Eingangsparameter PID\_3Step

Tabelle 9-87

| Parameter    | Datentyp | Vorbelegung | Beschreibung   |
|--------------|----------|-------------|--|
| Setpoint     | REAL     | 0.0         | Sollwert des PID-Reglers im Automatikbetrieb   |
| Input        | REAL     | 0.0         | Eine Variable des Anwenderprogramms wird als Quelle des Istwerts verwendet.<br>Wenn Sie den Parameter Input verwenden, muss Config.InputPerOn = FALSE sein.  |
| Input_PER    | WORD     | W#16#0      | Ein Analogeingang wird als Quelle des Istwerts verwendet.<br>Wenn Sie den Parameter Input_PER verwenden, muss Config.InputPerOn = TRUE sein.   |
| Actuator_H   | BOOL     | FALSE       | Digitale Stellungsrückmeldung des Ventils für den Oberen Anschlag<br>Wenn Actuator_H = TRUE ist die Stellung des Ventils am oberen Anschlag und das Ventil wird nicht weiter in diese Richtung bewegt.   |
| Actuator_L   | BOOL     | FALSE       | Digitale Stellungsrückmeldung des Ventils für den Unteren Anschlag<br>Wenn Actuator_L = TRUE ist die Stellung des Ventils am unteren Anschlag und das Ventil wird nicht weiter in diese Richtung bewegt.   |
| Feedback     | REAL     | 0.0         | Stellungsrückmeldung des Ventils<br>Wenn Sie den Parameter Feedback verwenden, muss Config.FeedbackPerOn = FALSE sein.   |
| Feedback_PER | WORD     | W#16#0      | Analoge Stellungsrückmeldung eines Ventils<br>Wenn Sie den Parameter Feedback_PER verwenden, muss Config.FeedbackPerOn = TRUE sein.<br>Feedback_PER wird skaliert anhand der Variablen: <ul style="list-style-type: none"> <li>• Config.FeedbackScaling.LowerPointIn</li> <li>• Config.FeedbackScaling.UpperPointIn</li> <li>• Config.FeedbackScaling.LowerPointOut</li> <li>• Config.FeedbackScaling.UpperPointOut</li> </ul> |

| Parameter    | Datentyp | Vorbelegung | Beschreibung   |
|--------------|----------|-------------|--|
| ManualEnable | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>• Flanke FALSE -&gt; TRUE wählt die Betriebsart "Handbetrieb", State = 4, Retain.Mode bleibt unverändert.</li> <li>• Flanke TRUE -&gt; FALSE wählt die zuletzt aktive Betriebsart</li> </ul> <p>Während ManualEnable = TRUE wirkt sich eine Änderung von Retain.Mode nicht aus. Erst bei der Flanke TRUE -&gt; FALSE an ManualEnable wird die Änderung von Retain.Mode berücksichtigt.</p> <p><b>PID_3Step V1.1</b> Wenn beim Start der CPU ManualEnable = TRUE, startet PID_3Step im Handbetrieb. Eine steigende Flanke (FALSE &gt; TRUE) an ManualEnable ist nicht notwendig.</p> <p><b>PID_3Step V1.0</b></p> <p>Beim Start der CPU schaltet PID_3Step nur bei einer steigenden Flanke (FALSE-&gt;TRUE) an ManualEnable in den Handbetrieb. Ohne steigende Flanke startet PID_3Step in der letzten Betriebsart bei der ManualEnable FALSE war.</p> |
| ManualValue  | REAL     | 0.0         | Im Handbetrieb wird die absolute Position des Ventils vorgegeben. ManualValue wird nur ausgewertet, wenn Sie OutputPer verwenden <b>oder</b> eine Stellungsrückmeldung verfügbar ist.  |
| Manual_UP    | BOOL     | FALSE       | Im Handbetrieb öffnet jede steigende Flanke das Ventil um 5 % des gesamten Stellbereichs oder für die Mindeststellzeit des Motors. Manual_UP wird nur ausgewertet, wenn Sie weder Output_PER verwenden noch eine Stellungsrückmeldung verfügbar ist.   |
| Manual_DN    | BOOL     | FALSE       | Im Handbetrieb schließt jede steigende Flanke das Ventil um 5 % des gesamten Stellbereichs oder für die Mindeststellzeit des Motors. Manual_DN wird nur ausgewertet, wenn Sie weder Output_PER verwenden noch eine Stellungsrückmeldung verfügbar ist.   |
| Reset        | BOOL     | FALSE       | <p>Führt einen Neustart des Reglers durch.</p> <ul style="list-style-type: none"> <li>• Flanke FALSE -&gt; TRUE <ul style="list-style-type: none"> <li>- Wechsel in Betriebsart "Inaktiv"</li> <li>- Zwischenwerte der Regelung werden zurückgesetzt (PID-Parameter bleiben erhalten)</li> </ul> </li> <li>• Flanke TRUE -&gt; FALSE <ul style="list-style-type: none"> <li>- Wechsel in letzte aktive Betriebsart</li> </ul> </li> </ul>  |

## Ausgangsparameter PID\_3Step

Tabelle 9-88

| Parameter      | Datentyp | Vorbelegung | Beschreibung  |
|----------------|----------|-------------|---|
| ScaledInput    | REAL     | 0.0         | Skalierter Istwert  |
| ScaledFeedback | REAL     | 0.0         | <p>Skalierte Stellungsrückmeldung</p> <p>Bei einem Stellglied ohne Stellungsrückmeldung zeigt ScaledFeedback die Position des Stellglieds sehr ungenau an. Dann darf ScaledFeedback nur zur groben Abschätzung der aktuellen Position genutzt werden.</p> |

| Parameter       | Datentyp | Vorbelegung | Beschreibung  |
|-----------------|----------|-------------|---|
| Output_UP       | BOOL     | FALSE       | Digitaler Ausgangswert zum Öffnen des Ventils<br>Wenn Config.OutputPerOn = FALSE, wird der Parameter Output_UP verwendet.   |
| Output_DN       | BOOL     | FALSE       | Digitaler Ausgangswert zum Schließen des Ventils<br>Wenn Config.OutputPerOn = FALSE, wird der Parameter Output_DN verwendet.  |
| Output_PER      | WORD     | W#16#0      | Analoger Ausgangswert<br>Wenn Config.OutputPerOn = TRUE, wird Output_PER verwendet.   |
| SetpointLimit_H | BOOL     | FALSE       | Wenn SetpointLimit_H = TRUE, ist die absolute Obergrenze des Sollwerts erreicht. In der CPU wird der Sollwert auf die konfigurierte absolute Obergrenze des Sollwerts begrenzt. Als Obergrenze Sollwert wird die konfigurierte absolute Obergrenze Istwert vorbelegt.<br>Wenn Sie Config.SetpointUpperLimit auf einen Wert innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.  |
| SetpointLimit_L | BOOL     | FALSE       | Wenn SetpointLimit_L = TRUE, ist die absolute Untergrenze des Sollwerts erreicht. In der CPU wird der Sollwert auf die konfigurierte absolute Untergrenze des Sollwerts begrenzt. Als Untergrenze Sollwert ist die konfigurierte absolute Untergrenze Istwert vorbelegt.<br>Wenn Sie Config.SetpointLowerLimit auf einen Wert innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.  |
| InputWarning_H  | BOOL     | FALSE       | Wenn InputWarning_H = TRUE, ist die obere Warngrenze des Istwerts erreicht oder überschritten.  |
| InputWarning_L  | BOOL     | FALSE       | Wenn InputWarning_L = TRUE, ist die untere Warngrenze des Istwerts erreicht oder unterschritten.  |
| State           | INT      | 0           | Der Parameter State (Seite 2768) zeigt die aktuelle Betriebsart des PID-Reglers. Sie ändern die Betriebsart mit der Variable Retain.Mode. <ul style="list-style-type: none"> <li>• State = 0: Inaktiv</li> <li>• State = 1: Erstopptimierung</li> <li>• State = 2: Nachoptimierung</li> <li>• State = 3: Automatikbetrieb</li> <li>• State = 4: Handbetrieb</li> <li>• State = 5: Ersatzausgangswert anfahren</li> <li>• State = 6: Stellzeitmessung</li> <li>• State = 7: Fehlerüberwachung</li> <li>• State = 8: Ersatzausgangswert anfahren mit Fehlerüberwachung</li> </ul> |
| Error           | BOOL     | FALSE       | Wenn Error = TRUE, liegt mindestens eine Fehlermeldungen vor.   |
| ErrorBits       | DWORD    | DW#16#0     | Der Parameter ErrorBits (Seite 2776) zeigt die Fehlermeldungen.   |

**Siehe auch**

Parameter State und Retain.Mode (Seite 2768)

Parameter ErrorBits (Seite 2776)

**Statische Variablen PID\_3Step**

Nicht aufgeführte Variablen dürfen Sie nicht verändern. Diese werden nur intern verwendet.

Tabelle 9-89

| Variable            | Datentyp | Vorbelegung | Beschreibung   |
|---------------------|----------|-------------|--|
| ActivateRecoverMode | BOOL     | TRUE        | Die Variable ActivateRecoverMode (Seite 2778) bestimmt das Verhalten im Fehlerfall.  |
| RunModeByStartup    | BOOL     | TRUE        | Nach CPU Neustart letzte Betriebsart aktivieren<br>Wenn RunModeByStartup = TRUE, kehrt der Regler nach CPU-Anlauf in die zuletzt aktive Betriebsart zurück.<br>Wenn RunModeByStartup = FALSE, bleibt der Regler nach CPU-Anlauf inaktiv.   |
| PhysicalUnit        | INT      | 0           | Physikalische Einheit des Ist- und Sollwerts, z. B. °C oder °F.  |
| PhysicalQuantity    | INT      | 0           | Physikalische Größe des Ist- und Sollwerts, z. B. Temperatur   |
| ErrorBehaviour      | INT      | 0           | Wenn ErrorBehaviour = 0, bleibt das Ventil im Fehlerfall in der aktuellen Position und der Regler wechselt direkt in die Betriebsart "Inaktiv" oder "Fehlerüberwachung".<br>Wenn ErrorBehaviour = 1, wird im Fehlerfall das Stellglied auf den Ersatzausgangswert gefahren und erst dann in die Betriebsart "Inaktiv" oder "Fehlerüberwachung" gewechselt.<br>Wenn die folgenden Fehler auftreten, kann das Ventil nicht mehr auf einen konfigurierten Ersatzausgangswert gefahren werden. <ul style="list-style-type: none"> <li>• 2000h: Ungültiger Wert am Parameter Feedback_PER.</li> <li>• 4000h: Ungültiger Wert am Parameter Feedback.</li> <li>• 8000h: Fehler bei der digitalen Stellungsrückmeldung.</li> </ul> |
| Warning             | DWORD    | DW#16#0     | Die Variable Warning (Seite 2768) zeigt die Warnungen seit Reset oder dem letzten Wechsel der Betriebsart.<br>Zyklische Warnungen (z.B. Istwertwarnung) werden solange angezeigt wie die Ursache der Warnung anliegt. Fällt der Grund weg, werden sie automatisch gelöscht. Nichtzyklische Warnungen (z.B. Wendepunkt nicht gefunden) bleiben bestehen und werden wie Fehler gelöscht.   |
| SavePosition        | REAL     | 0.0         | Ersatzausgangswert<br>Wenn ErrorBehaviour = 1, wird das Stellglied im Fehlerfall in eine für die Anlage sichere Position gefahren und erst dann in die Betriebsart "Inaktiv" gewechselt.   |
| CurrentSetpoint     | REAL     | 0.0         | Aktuell aktiver Sollwert. Dieser Wert wird beim Start der Optimierung eingefroren.   |
| Progress            | REAL     | 0.0         | Fortschritt der Optimierung in Prozent (0.0 - 100.0)   |
| Config.InputPerOn   | BOOL     | TRUE        | Wenn InputPerOn = TRUE, wird der Parameter Input_PER verwendet. Wenn InputPerOn = FALSE, wird der Parameter Input verwendet.   |

| Variable                 | Datentyp | Vorbelegung       | Beschreibung  |
|--------------------------|----------|-------------------|---|
| Config.OutputPerOn       | BOOL     | FALSE             | Wenn OutputPerOn = TRUE, wird der Parameter Output_PER verwendet. Wenn OutputPerOn = FALSE, werden die Parameter Ouput_UP und Output_DN verwendet.  |
| Config.LoadBackUp        | BOOL     | FALSE             | Wenn LoadBackUp = TRUE, wird der letzte Satz PID-Parameter wieder geladen. Der Satz wurde vor der letzten Optimierung gespeichert. LoadBackUp wird automatisch wieder auf FALSE gesetzt.  |
| Config.InvertControl     | BOOL     | FALSE             | Invertieren des Regelsinns<br>Wenn InvertControl = TRUE, bewirkt eine steigende Regeldifferenz eine Verringerung des Ausgangswerts.   |
| Config.FeedbackOn        | BOOL     | FALSE             | Wenn FeedbackOn = FALSE, wird eine Stellungsrückmeldung simuliert.<br>Wenn FeedbackOn = TRUE, wird generell die Stellungsrückmeldung aktiviert.   |
| Config.FeedbackPerOn     | BOOL     | FALSE             | FeedbackPerOn ist nur wirksam, wenn FeedbackOn = TRUE.<br>Wenn FeedbackPerOn = TRUE, wird der Analogeingang für die Stellungsrückmeldung verwendet (Parameter Feedback_PER).<br>Wenn FeedbackPerOn = FALSE, wird der Parameter Feedback für die Stellungsrückmeldung verwendet.   |
| Config.ActuatorEndStopOn | BOOL     | FALSE             | Wenn ActuatorEndStopOn = TRUE, wird die digitale Stellungsrückmeldung Actuator_L und Actuator_H berücksichtigt.   |
| Config.InputUpperLimit   | REAL     | 120.0             | Obergrenze des Istwerts<br>Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird kein Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID_3Step verhält sich wie unter Verhalten im Fehlerfall konfiguriert.<br>InputUpperLimit > InputLowerLimit                     |
| Config.InputLowerLimit   | REAL     | 0.0               | Untergrenze des Istwerts<br>InputLowerLimit < InputUpperLimit   |
| Config.InputUpperWarning | REAL     | +3.40282<br>2e+38 | Obere Warngrenze des Istwerts<br>Wenn Sie InputUpperWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Obergrenze Istwert als Obere Warngrenze verwendet.<br>Wenn Sie InputUpperWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obere Warngrenze verwendet.<br>InputUpperWarning > InputLowerWarning<br>InputUpperWarning ≤ InputUpperLimit     |
| Config.InputLowerWarning | REAL     | -3.40282<br>2e+38 | Untere Warngrenze des Istwerts<br>Wenn Sie InputLowerWarning außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Untergrenze Istwert als Untere Warngrenze verwendet.<br>Wenn Sie InputLowerWarning innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untere Warngrenze verwendet.<br>InputLowerWarning < InputUpperWarning<br>InputLowerWarning ≥ InputLowerLimit |



| Variable                          | Datentyp | Vorbelegung           | Beschreibung  |
|-----------------------------------|----------|-----------------------|---|
| Config.OutputUpperLimit           | REAL     | 100.0                 | Obergrenze des Ausgangswerts<br>Details siehe OutputLowerLimit  |
| Config.OutputLowerLimit           | REAL     | 0.0                   | Untergrenze des Ausgangswerts<br>Wenn OutputPerOn = TRUE oder FeedbackOn = TRUE, gilt der Wertebereich von -100 bis +100 % einschließlich Null. Bei -100 % beträgt Output = -27648; bei +100 % Output = 27648<br>Wenn OutputPerOn = FALSE, gilt der Wertebereich von 0 bis 100 %. Bei 0 % ist das Ventil vollständig geschlossen, bei 100 % vollständig geöffnet. |
| Config.SetpointUpperLimit         | REAL     | +3.40282<br>2e+38     | Obergrenze des Sollwerts<br>Wenn Sie SetpointUpperLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertobergrenze als Obergrenze Sollwert vorbelegt.<br>Wenn Sie SetpointUpperLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Obergrenze Sollwert verwendet.   |
| Config.SetpointLowerLimit         | REAL     | -<br>3.402822<br>e+38 | Untergrenze des Sollwerts<br>Wenn Sie SetpointLowerLimit außerhalb der Istwertgrenzen konfigurieren, wird die konfigurierte absolute Istwertuntergrenze als Untergrenze Sollwert vorbelegt.<br>Wenn Sie SetpointLowerLimit innerhalb der Istwertgrenzen konfigurieren, wird dieser Wert als Untergrenze Sollwert verwendet.                                       |
| Config.MinimumOnTime              | REAL     | 0.0                   | Minimale Einschaltzeit<br>Zeit in Sekunden, die der Stellantrieb mindestens eingeschaltet sein muss.  |
| Config.MinimumOffTime             | REAL     | 0.0                   | Minimale Ausschaltzeit<br>Zeit in Sekunden, die der Stellantrieb mindestens ausgeschaltet sein muss.  |
| Config.TransitTime                | REAL     | 30.0                  | Motorstellzeit<br>Zeit in Sekunden, die der Stellantrieb benötigt, um das Ventil vom geschlossenen in den geöffneten Zustand zu bewegen.  |
| Config.InputScaling.UpperPointIn  | REAL     | 27648.0               | Skalierung Input_PER Oben<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.  |
| Config.InputScaling.LowerPointIn  | REAL     | 0.0                   | Skalierung Input_PER Unten<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.   |
| Config.InputScaling.UpperPointOut | REAL     | 100.0                 | Skalierter oberer Istwert<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.  |
| Config.InputScaling.LowerPointOut | REAL     | 0.0                   | Skalierter unterer Istwert<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur InputScaling wird Input_PER in Prozent umgerechnet.   |

| Variable                             | Datentyp | Vorbelegung | Beschreibung  |
|--------------------------------------|----------|-------------|---|
| Config.FeedbackScaling.UpperPointIn  | REAL     | 27648.0     | Skalierung Feedback_PER Oben<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.   |
| Config.FeedbackScaling.LowerPointIn  | REAL     | 0.0         | Skalierung Feedback_PER Unten<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.  |
| Config.FeedbackScaling.UpperPointOut | REAL     | 100.0       | Oberer Anschlag<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.  |
| Config.FeedbackScaling.LowerPointOut | REAL     | 0.0         | Unterer Anschlag<br>Anhand der zwei Wertepaare UpperPointOut, UpperPointIn und LowerPointOut, LowerPointIn der Struktur FeedbackScaling wird Feedback_PER in Prozent umgerechnet.   |
| GetTransitTime.InvertDirection       | BOOL     | FALSE       | Wenn InvertDirection = FALSE, wird das Ventil zum Ermitteln der Stellzeit vollständig geöffnet, geschlossen und wieder geöffnet.<br>Wenn InvertDirection = TRUE, wird das Ventil vollständig geschlossen, geöffnet und wieder geschlossen.  |
| GetTransitTime.SelectFeedback        | BOOL     | FALSE       | Wenn SelectFeedback = TRUE, wird Feedback_PER oder Feedback bei der Stellzeitmessung berücksichtigt.<br>Wenn SelectFeedback = FALSE, wird Actuator_H und Actuator_L bei der Stellzeitmessung berücksichtigt.  |
| GetTransitTime.Start                 | BOOL     | FALSE       | Wenn Start = TRUE, wird die Stellzeitmessung gestartet.   |
| GetTransitTime.State                 | INT      | 0           | Aktuelle Phase der Stellzeitmessung <ul style="list-style-type: none"> <li>• State = 0: Inaktiv</li> <li>• State = 1: Ventil vollständig öffnen</li> <li>• State = 2: Ventil vollständig schließen</li> <li>• State = 3: Ventil auf Zielstellung (NewOutput) stellen</li> <li>• State = 4: Stellzeitmessung erfolgreich beendet</li> <li>• State = 5: Stellzeitmessung abgebrochen</li> </ul> |
| GetTransitTime.NewOutput             | REAL     | 0.0         | Zielstellung für die Stellzeitmessung mit Stellungsrückmeldung<br>Die Zielstellung muss innerhalb "Oberer Anschlag" und "Unterer Anschlag" liegen. Die Differenz zwischen NewOutput und ScaledFeedback muss mindestens 50 % des zulässigen Stellbereichs sein.  |
| CycleTime.StartEstimation            | BOOL     | TRUE        | Wenn StartEstimation = TRUE, wird die Messung der Abtastzeit PID_3Step gestartet. Nach Abschluss der Messung wird CycleTime.StartEstimation = FALSE.  |
| CycleTime.EnEstimation               | BOOL     | TRUE        | Wenn EnEstimation = TRUE, wird die Abtastzeit PID_3Step berechnet.  |

| Variable                           | Datentyp | Vorbelegung | Beschreibung   |
|------------------------------------|----------|-------------|--|
| CycleTime.EnMonitoring             | BOOL     | TRUE        | Wenn EnMonitoring = TRUE, wird die Abtastzeit PID_3Step überwacht. Wenn PID_3Step nicht innerhalb der Abtastzeit ausgeführt werden kann, wird der Fehler 0800h ausgegeben und die Betriebsart gewechselt. In welche Betriebsart gewechselt wird, hängt ab von ActivateRecoverMode und ErrorBehaviour.<br>Wenn EnMonitoring = FALSE, wird die Abtastzeit PID_3Step nicht überwacht, der Fehler 0800h nicht ausgegeben und die Betriebsart nicht gewechselt. |
| CycleTime.Value                    | REAL     | 0.1         | Abtastzeit PID_3Step in Sekunden<br>CycleTime.Value wird automatisch ermittelt und entspricht normalerweise der Zykluszeit des aufrufenden OB.   |
| CtrlParamsBackUp.SetByUser         | BOOL     | FALSE       | Gespeicherter Wert von Retain.CtrlParams.SetByUser<br>Werte aus der Struktur CtrlParamsBackUp können mit Config.LoadBackUp = TRUE wieder geladen werden.   |
| CtrlParamsBackUp.Gain              | REAL     | 1.0         | Gespeicherte Proportionalverstärkung   |
| CtrlParamsBackUp.Ti                | REAL     | 20.0        | Gespeicherte Integrationszeit  |
| CtrlParamsBackUp.Td                | REAL     | 0.0         | Gespeicherte Differenzierzeit  |
| CtrlParamsBackUp.TdFiltRatio       | REAL     | 0.0         | Gespeicherter Koeffizient für den Differenzierverszug  |
| CtrlParamsBackUp.PWeighting        | REAL     | 0.0         | Gespeicherte Gewichtung des P-Anteils  |
| CtrlParamsBackUp.DWeighting        | REAL     | 0.0         | Gespeicherte Gewichtung des D-Anteils  |
| CtrlParamsBackUp.Cycle             | REAL     | 1.0         | Gespeicherte Abtastzeit PID-Algorithmus  |
| CtrlParamsBackUp.InputDeadBand     | REAL     | 0.0         | Gespeicherte Totzonenbreite der Regeldifferenz   |
| PIDSelfTune.SUT.CalculateSUTParams | BOOL     | FALSE       | Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn CalculateSUTParams = TRUE, werden anhand dieser Eigenschaften die PID-Parameter neu berechnet. Die PID-Parameter werden nach der Methode berechnet, die in TuneRuleSUT eingestellt ist. CalculateSUTParams wird nach der Berechnung auf FALSE gesetzt.   |
| PIDSelfTune.SUT.TuneRuleSUT        | INT      | 1           | Parameter während Erstopptimierung berechnen nach Methode: <ul style="list-style-type: none"> <li>• TuneRuleSUT = 0: PID schnell I</li> <li>• TuneRuleSUT = 1: PID langsam I</li> <li>• TuneRuleSUT = 2: Chien, Hrones, Reswick PID</li> <li>• TuneRuleSUT = 3: Chien, Hrones, Reswick PI</li> <li>• TuneRuleSUT = 4: PID schnell II</li> <li>• TuneRuleSUT = 5: PID langsam II</li> </ul>   |
| PIDSelfTune.SUT.State              | INT      | 0           | Die Variable SUT.State zeigt die aktuelle Phase der Erstopptimierung:  |

| Variable                           | Datentyp | Vorbelegung | Beschreibung  |
|------------------------------------|----------|-------------|---|
| PIDSelfTune.TIR.RunIn              | BOOL     | FALSE       | <ul style="list-style-type: none"> <li>RunIn = FALSE<br/>Wenn die Nachoptimierung aus der Betriebsart Inaktiv oder Handbetrieb gestartet wird, wird eine Erstopptimierung gestartet. Wenn die Nachoptimierung aus dem Automatikbetrieb gestartet wird, wird mit den vorhandenen PID-Parametern auf den Sollwert geregelt. Erst dann startet die Nachoptimierung. Ist die Erstopptimierung nicht möglich, wechselt PID_3Step in die Betriebsart "Inaktiv".</li> <li>RunIn = TRUE<br/>Die Erstopptimierung wird übersprungen. PID_3Step versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Erst dann startet die Nachoptimierung. RunIn wird nach der Nachoptimierung auf FALSE gesetzt.</li> </ul> |
| PIDSelfTune.TIR.CalculateTIRParams | BOOL     | FALSE       | Die Eigenschaften der Regelstrecke werden bei der Optimierung gespeichert. Wenn CalculateTIRParams = TRUE, werden anhand dieser Eigenschaften die PID-Parameter neu berechnet. Die PID-Parameter werden nach der Methode berechnet, die in TuneRuleTIR eingestellt ist. CalculateTIRParams wird nach der Berechnung auf FALSE gesetzt.  |
| PIDSelfTune.TIR.TuneRuleTIR        | INT      | 0           | Parameter während Nachoptimierung berechnen nach Methode: <ul style="list-style-type: none"> <li>TuneRuleTIR = 0: PID automatisch</li> <li>TuneRuleTIR = 1: PID schnell</li> <li>TuneRuleTIR = 2: PID langsam</li> <li>TuneRuleTIR = 3: Ziegler-Nichols PID</li> <li>TuneRuleTIR = 4: Ziegler-Nichols PI</li> <li>TuneRuleTIR = 5: Ziegler-Nichols P</li> </ul>   |
| PIDSelfTune.TIR.State              | INT      | 0           | Die Variable TIR.State zeigt die aktuelle Phase der "Nachoptimierung":  |
| Retain.Mode                        | INT      | 0           | Wenn sich der Wert von Retain.Mode ändert, wird in eine andere Betriebsart gewechselt.<br>Folgende Betriebsart wird aktiviert bei Änderung von Mode zu: <ul style="list-style-type: none"> <li>Mode = 0: Inaktiv</li> <li>Mode = 1: Erstopptimierung</li> <li>Mode = 2: Nachoptimierung</li> <li>Mode = 3: Automatikbetrieb</li> <li>Mode = 4: Handbetrieb</li> <li>Mode = 5: Ersatzausgangswert anfahren</li> <li>Mode = 6: Stellzeitmessung</li> <li>Mode = 7: Fehlerüberwachung</li> <li>Mode = 8: Ersatzausgangswert anfahren mit Fehlerüberwachung</li> </ul> Mode ist remanent.   |

| Variable                        | Datentyp | Vorbelegung | Beschreibung  |
|---------------------------------|----------|-------------|---|
| Retain.CtrlParams.SetByUser     | BOOL     | FALSE       | Wenn SetByUser = FALSE, sind die PID-Parameter automatisch ermittelt und PID_3Step arbeitet mit einer Totzone am Ausgangswert. Die Totzonenbreite wird während der Optimierung anhand der Standardabweichung des Ausgangswerts berechnet und in Retain.CtrlParams.OutputDeadBand gespeichert.<br>Wenn SetByUser = TRUE, sind die PID-Parameter manuell eingegeben und PID_3 Step arbeitet ohne Totzone am Ausgangswert. Retain.CtrlParams.OutputDeadBand = 0.0<br>SetByUser ist remanent. |
| Retain.CtrlParams.Gain          | REAL     | 1.0         | Aktive Proportionalverstärkung<br>Gain ist remanent.  |
| Retain.CtrlParams.Ti            | REAL     | 20.0        | <ul style="list-style-type: none"> <li>Ti &gt; 0.0: Aktive Integrationszeit</li> <li>Ti = 0.0: I-Anteil ist ausgeschaltet</li> </ul> Ti ist remanent.   |
| Retain.CtrlParams.Td            | REAL     | 0.0         | <ul style="list-style-type: none"> <li>Td &gt; 0.0: Aktive Differenzierzeit</li> <li>Td = 0.0: D-Anteil ist ausgeschaltet</li> </ul> Td ist remanent.   |
| Retain.CtrlParams.TdFiltRatio   | REAL     | 0.0         | Aktiver Koeffizient für den Differenzierverszug<br>TdFiltRatio ist remanent.  |
| Retain.CtrlParams.PWeighting    | REAL     | 0.0         | Aktive Gewichtung des P-Anteils<br>PWeighting ist remanent.   |
| Retain.CtrlParams.DWeighting    | REAL     | 0.0         | Aktive Gewichtung des D- Anteils<br>DWeighting ist remanent.  |
| Retain.CtrlParams.Cycle         | REAL     | 1.0         | Aktive Abtastzeit PID-Algorithmus in Sekunden, die zu einem ganzzahligen Vielfachen der Zykluszeit des aufrufenden OB aufgerundet wird.<br>Cycle ist remanent.  |
| Retain.CtrlParams.InputDeadBand | REAL     | 0.0         | Totzonenbreite der Regeldifferenz<br>InputDeadBand ist remanent.  |

---

### Hinweis

Verändern Sie die hier aufgelisteten Variablen in der Betriebsart "Inaktiv", um ein Fehlverhalten des PID-Reglers zu vermeiden. Die Betriebsart "Inaktiv" erzwingen Sie durch den Wert "0" der Variable "Retain.Mode".

---

### Siehe auch

Parameter State und Retain.Mode (Seite 2768)

Variable ActivateRecoverMode (Seite 2778)

Technologieobjekte in Gerät laden (Seite 4056)

## Parameter State und Retain.Mode

### Zusammenhang der Parameter

Der Parameter State zeigt die aktuelle Betriebsart des PID-Reglers. Sie können den Parameter State nicht ändern.

Um die Betriebsart zu wechseln, müssen Sie die Variable Retain.Mode ändern. Das gilt auch, wenn in Retain.Mode bereits der Wert für die neue Betriebsart steht. Setzen Sie dann z. B. erst Retain.Mode = 0 und anschließend Retain.Mode = 3. Wenn die aktuelle Betriebsart des Reglers diesen Wechsel erlaubt, wird State auf den Wert von Retain.Mode gesetzt.

Wenn PID\_3Step automatisch die Betriebsart wechselt gilt: State != Retain.Mode.

Beispiele:

- Nach erfolgreicher Erstopptimierung  
State = 3 und Retain.Mode = 1
- Im Fehlerfall  
State = 0 und Retain.Mode bleibt auf bisherigem Wert, z. B. Retain.Mode = 3
- ManualEnalbe = TRUE  
State = 4 und Retain.Mode bleibt auf bisherigem Wert, z. B. Retain.Mode = 3

---

#### Hinweis

Sie wollen z. B. eine erfolgreiche Nachoptimierung wiederholen ohne den Automatikbetrieb mit Mode = 0 zu beenden.

Wenn Sie Retain.Mode für einen Takt auf einen ungültigen Wert setzen, z. B. 9999, hat das keinen Einfluss auf State. Im nächsten Takt setzen Sie Mode = 2. So können Sie eine Änderung an Retain.Mode erzeugen, ohne erst in die Betriebsart "Inaktiv" zu wechseln.

---

## Bedeutung der Werte

| State / Retain.Mode | Beschreibung  |
|---------------------|---|
| 0                   | <p>Inaktiv</p> <p>Der Regler ist ausgeschaltet und verändert die Position des Ventils nicht mehr.</p>   |
| 1                   | <p>Erstoptimierung</p> <p>Die Erstoptimierung ermittelt die Prozessantwort auf einen Impuls des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet.</p> <p>Voraussetzungen für die Erstoptimierung:</p> <ul style="list-style-type: none"> <li>• State = 0 oder State = 4</li> <li>• ManualEnable = FALSE</li> <li>• Die Motorstellzeit ist konfiguriert oder gemessen.</li> <li>• Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen.</li> </ul> <p>Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen.</p> <p>Bevor die PID-Parameter neu berechnet werden, werden sie gesichert und können mit Config.LoadBackUp reaktiviert werden. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren.</p> <p>Nach erfolgreicher Erstoptimierung wird in den Automatikbetrieb geschaltet; nach erfolgloser Erstoptimierung in die Betriebsart "Inaktiv".</p> <p>Die Phase der Erstoptimierung wird mit Variable SUT.State angezeigt.</p> |

| State / Retain.Mode | Beschreibung   |
|---------------------|--|
| 2                   | <p>Nachoptimierung</p> <p>Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter optimiert. Die Unterschiede zwischen der Prozessantwort während Erst- und Nachoptimierung werden analysiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstopptimierung.</p> <p>PID_3Step versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwertes. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwertes beeinflusst.</p> <p>Vor der Nachoptimierung werden die PID-Parameter gesichert. Sie können mit Config.LoadBackUp reaktiviert werden. Der Sollwert wird in der Variablen CurrentSetpoint eingefroren.</p> <p>Voraussetzungen für die Nachoptimierung:</p> <ul style="list-style-type: none"> <li>• Die Motorstellzeit ist konfiguriert oder gemessen.</li> <li>• Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen.</li> <li>• ManualEnable = FALSE</li> <li>• Betriebsart Automatikbetrieb (State = 3), Inaktiv (State = 0) oder Handbetrieb (State = 4)</li> </ul> <p>Die Nachoptimierung verläuft folgendermaßen beim Start aus:</p> <ul style="list-style-type: none"> <li>• Automatikbetrieb (State = 3)<br/>Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.<br/>PID_3Step regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.</li> <li>• Inaktiv (State = 0) oder Handbetrieb (State = 4)<br/>Zuerst wird immer eine Erstopptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind.<br/>Wenn PIDSelfTune.TIR.RunIn = TRUE, wird die Erstopptimierung übersprungen und versucht den Sollwert mit minimalem oder maximalen Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen. Die Nachoptimierung startet dann automatisch.</li> </ul> <p>Nach erfolgreicher Nachoptimierung wechselt der Regler in den Automatikbetrieb; bei erfolgloser Nachoptimierung in die Betriebsart "Inaktiv".</p> <p>Die Phase der Nachoptimierung wird mit Variable TIR.State angezeigt.</p> |
| 3                   | <p>Automatikbetrieb</p> <p>Im Automatikbetrieb regelt PID_3Step die Regelstrecke entsprechend der vorgegebenen Parameter aus. Wenn eine der folgenden Voraussetzungen erfüllt ist, wird in den Automatikbetrieb gewechselt:</p> <ul style="list-style-type: none"> <li>• Erstopptimierung erfolgreich abgeschlossen</li> <li>• Nachoptimierung erfolgreich abgeschlossen</li> <li>• Ändern der Variable Retain.Mode auf den Wert 3.</li> </ul> <p>Wenn die CPU eingeschaltet wird oder von Stopp in RUN wechselt, startet PID_3Step in der zuletzt aktiven Betriebsart. Um PID_3Step in der Betriebsart "Inaktiv" zu belassen, setzen Sie RunModeByStartup = FALSE.</p> <p>Im Automatikbetrieb wird die Variable ActivateRecoverMode berücksichtigt.</p>   |

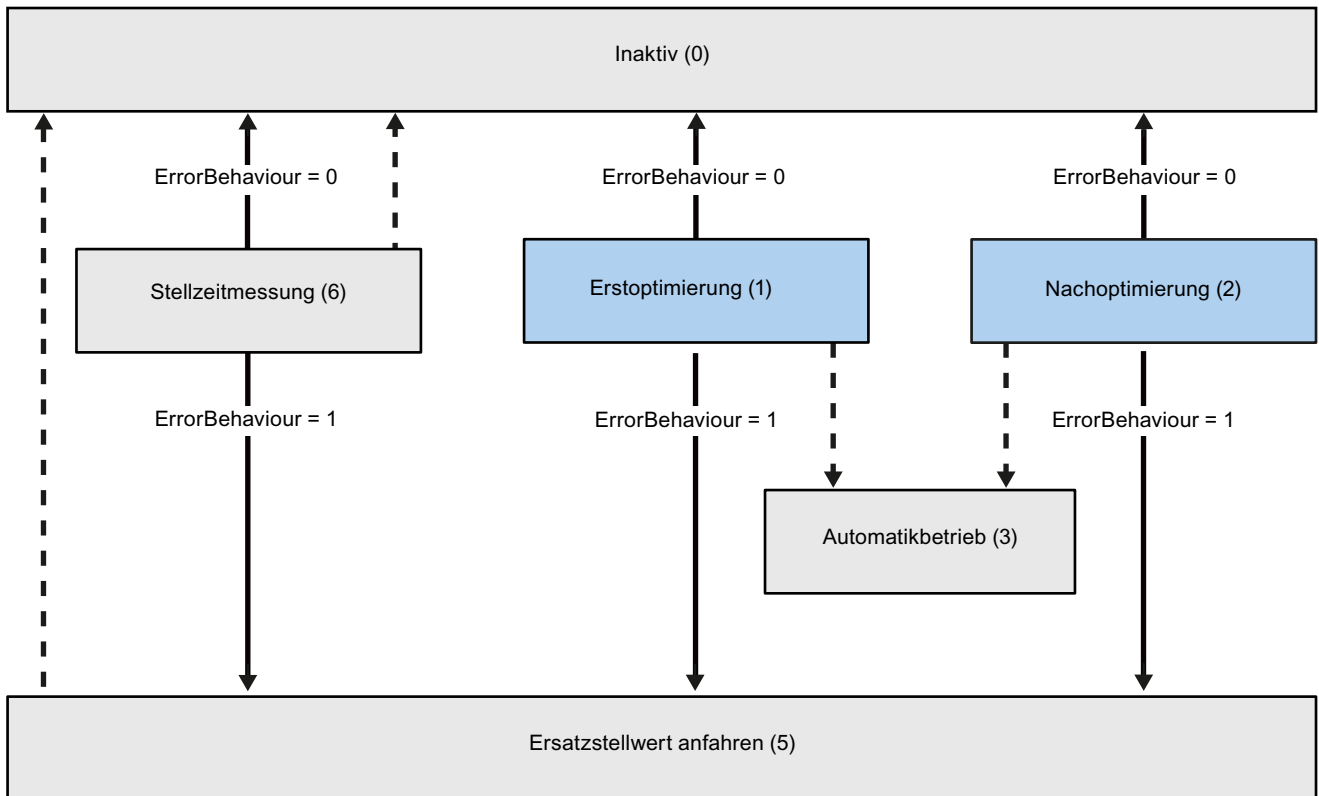


| State / Retain.Mode | Beschreibung   |
|---------------------|--|
| 4                   | <p>Handbetrieb</p> <p>Im Handbetrieb geben Sie manuelle Ausgangswerte an den Parametern Manual_UP und Manual_DN oder ManualValue vor. Ob das Stellglied im Fehlerfall auf den Ausgangswert verfahren werden kann, ist beim Parameter ErrorBits beschrieben.</p> <p>Diese Betriebsart wird aktiviert, wenn Retain.Mode = 4 oder bei steigender Flanke an ManualEnable.</p> <p>Wenn ManualEnable TRUE wird, ändert sich nur State. Retain.Mode bleibt auf dem bisherigen Wert. Bei einer fallenden Flanke an ManualEnable kehrt PID_3Step in die vorhergehende Betriebsart zurück.</p> <p>Der Wechsel in den Automatikbetrieb erfolgt stoßfrei.</p> <p><b>PID_3Step V1.1</b></p> <p>Im Fehlerfall ist der Handbetrieb immer möglich.</p> <p><b>PID_3Step V1.0</b></p> <p>Im Fehlerfall ist der Handbetrieb abhängig von der Variablen ActivateRecoverMode.</p> |
| 5                   | <p>Ersatzausgangswert anfahren</p> <p>Diese Betriebsart wird im Fehlerfall oder bei Reset = TRUE aktiviert, wenn Errorbehaviour = 1 und ActivateRecoverMode = FALSE.</p> <p>PID_3Step fährt das Stellglied auf den Ersatzausgangswert und wechselt anschließend in die Betriebsart "Inaktiv".</p>  |
| 6                   | <p>Stellzeitmessung</p> <p>Es wird die Zeit ermittelt, die der Motor benötigt, um das Ventil aus dem geschlossenen Zustand vollständig zu öffnen.</p> <p>Diese Betriebsart wird aktiviert, wenn GetTransitTime.Start = TRUE gesetzt wird.</p> <p>Wenn zur Stellzeitmessung Anschlagsignale verwendet werden, wird das Ventil von der aktuellen Position aus vollständig geöffnet, vollständig geschlossen und erneut vollständig geöffnet. Wenn GetTransitTime.InvertDirection = TRUE, wird dieses Verhalten umgekehrt.</p> <p>Wenn zur Stellzeitmessung eine Stellungsrückmeldung verwendet wird, wird das Stellglied von der aktuellen Stellung in eine Zielstellung gefahren.</p> <p>Die Ausgangswertgrenzen werden während der Stellzeitmessung nicht berücksichtigt. Das Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.</p>     |

| State / Retain.Mode | Beschreibung   |
|---------------------|--|
| 7                   | <p>Fehlerüberwachung</p> <p>Der Regelalgorithmus ist ausgeschaltet und verändert die Position des Ventils nicht mehr. Diese Betriebsart wird im Fehlerfall statt der Betriebsart "Inaktiv" aktiviert.</p> <p>Alle folgenden Bedingungen müssen erfüllt sein:</p> <ul style="list-style-type: none"> <li>• Mode = 3 (Automatikbetrieb)</li> <li>• Errorbehaviour = 0</li> <li>• ActivateRecoverMode = TRUE</li> <li>• Einer oder mehrere Fehler sind auftreten, bei denen ActivateRecoverMode (Seite 2778) wirkt.</li> </ul> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.</p>   |
| 8                   | <p>Ersatzausgangswert anfahren mit Fehlerüberwachung</p> <p>Diese Betriebsart wird im Fehlerfall statt der Betriebsart "Ersatzausgangswert anfahren" aktiviert. PID_3Step fährt das Stellglied auf den Ersatzausgangswert und wechselt anschließend in die Betriebsart "Fehlerüberwachung".</p> <p>Alle folgenden Bedingungen müssen erfüllt sein:</p> <ul style="list-style-type: none"> <li>• Mode = 3 (Automatikbetrieb)</li> <li>• Errorbehaviour = 1</li> <li>• ActivateRecoverMode = TRUE</li> <li>• Einer oder mehrere Fehler sind auftreten, bei denen ActivateRecoverMode (Seite 2778) wirkt.</li> </ul> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.</p> |

### Automatische Betriebsartwechsel während der Inbetriebnahme

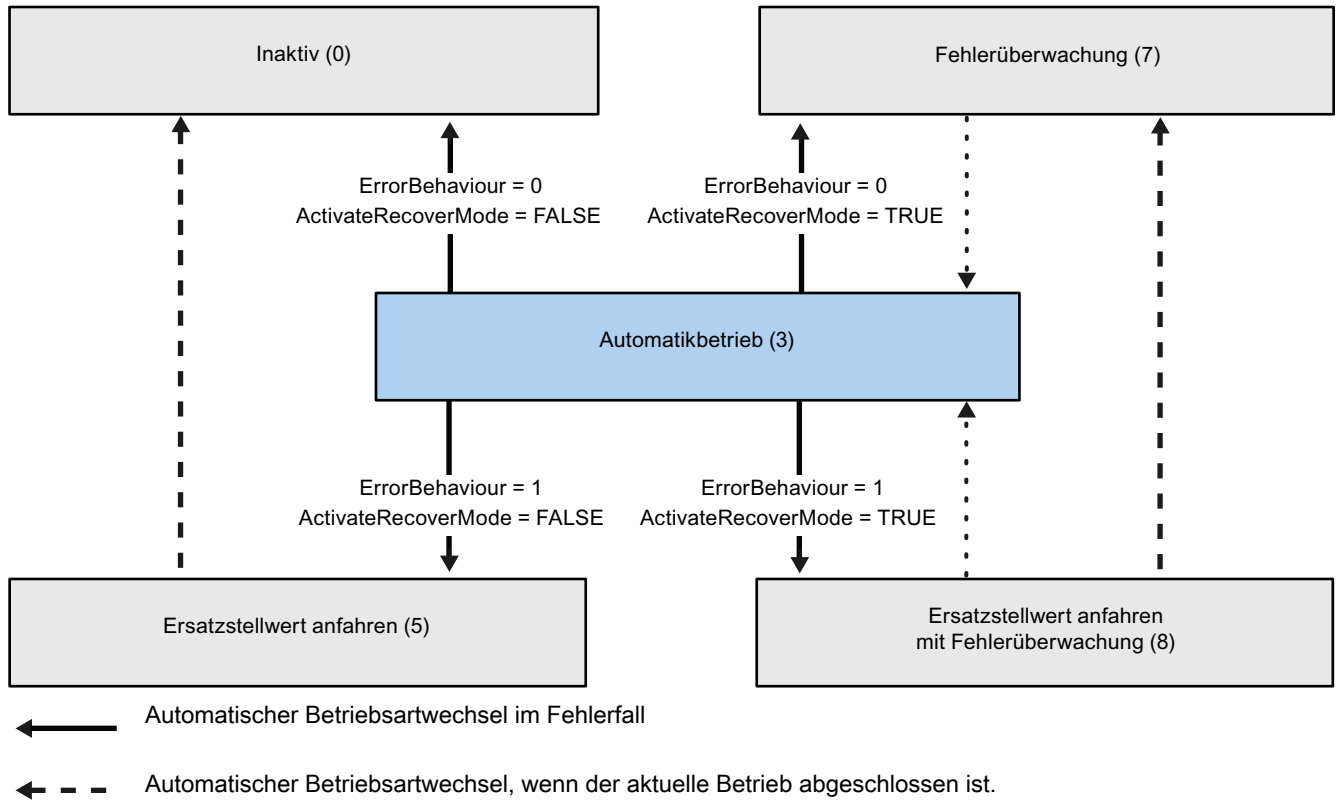
Im Fehlerfall wechselt PID\_3Step automatisch die Betriebsart. Das folgende Diagramm zeigt den Einfluss von ErrorBehaviour auf den Betriebsartwechsel aus den Betriebsarten Stellzeitmessung, Erstopptimierung und Nachoptimierung.



- ← Automatischer Betriebsartwechsel im Fehlerfall
- ← - - - Automatischer Betriebsartwechsel, wenn der aktuelle Betrieb abgeschlossen ist.

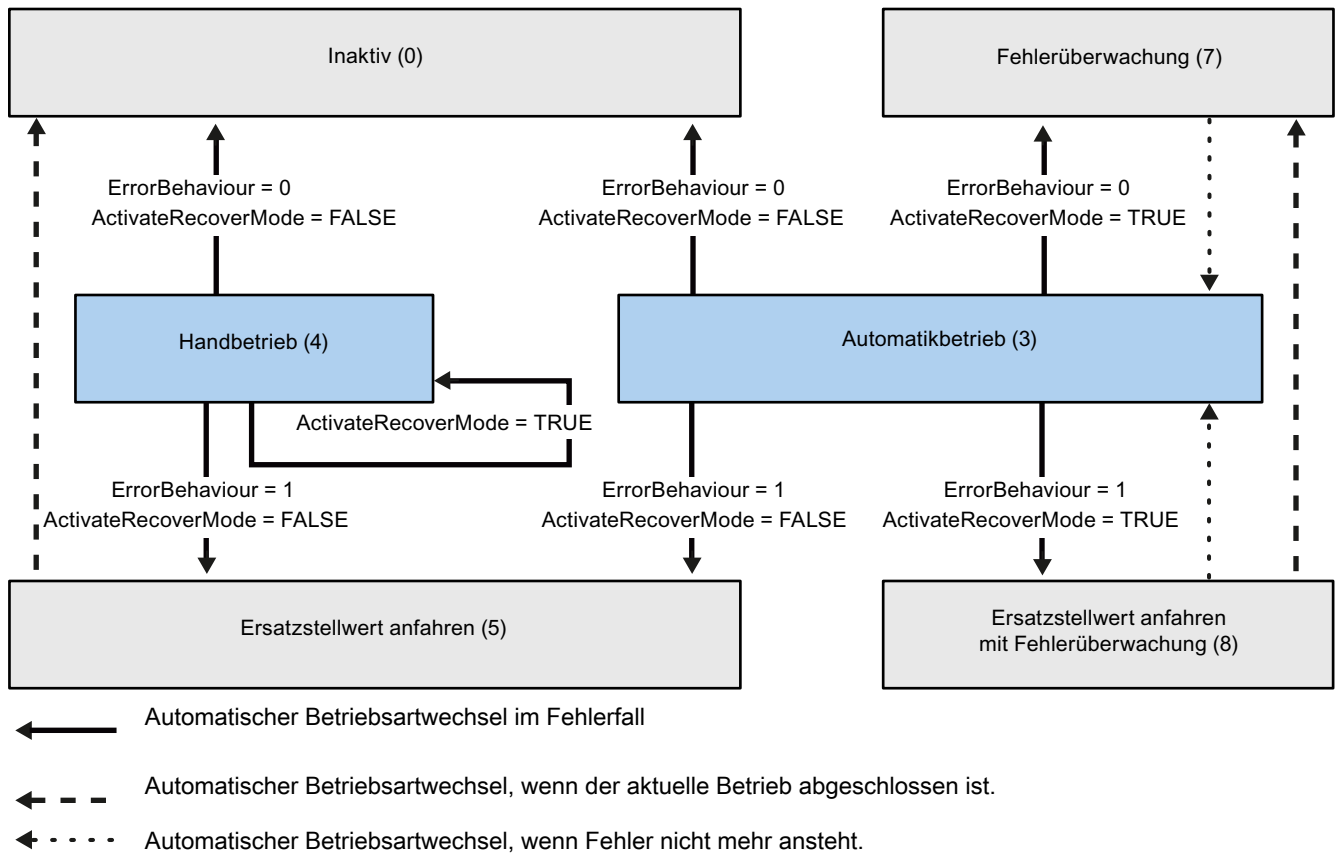
### Automatische Betriebsartwechsel im Automatikbetrieb (PID\_3Step V1.1)

Im Fehlerfall wechselt PID\_3Step automatisch die Betriebsart. Das folgende Diagramm zeigt den Einfluss von ErrorBehaviour und ActivateRecoverMode auf diesen Betriebsartwechsel.



### Automatische Betriebsartwechsel im Automatik- und Handbetrieb (PID\_3Step V1.0)

Im Fehlerfall wechselt PID\_3Step automatisch die Betriebsart. Das folgende Diagramm zeigt den Einfluss von ErrorBehaviour und ActivateRecoverMode auf diesen Betriebsartwechsel.



#### Siehe auch

Variable ActivateRecoverMode (Seite 2778)

Parameter ErrorBits (Seite 2776)

**Parameter ErrorBits**

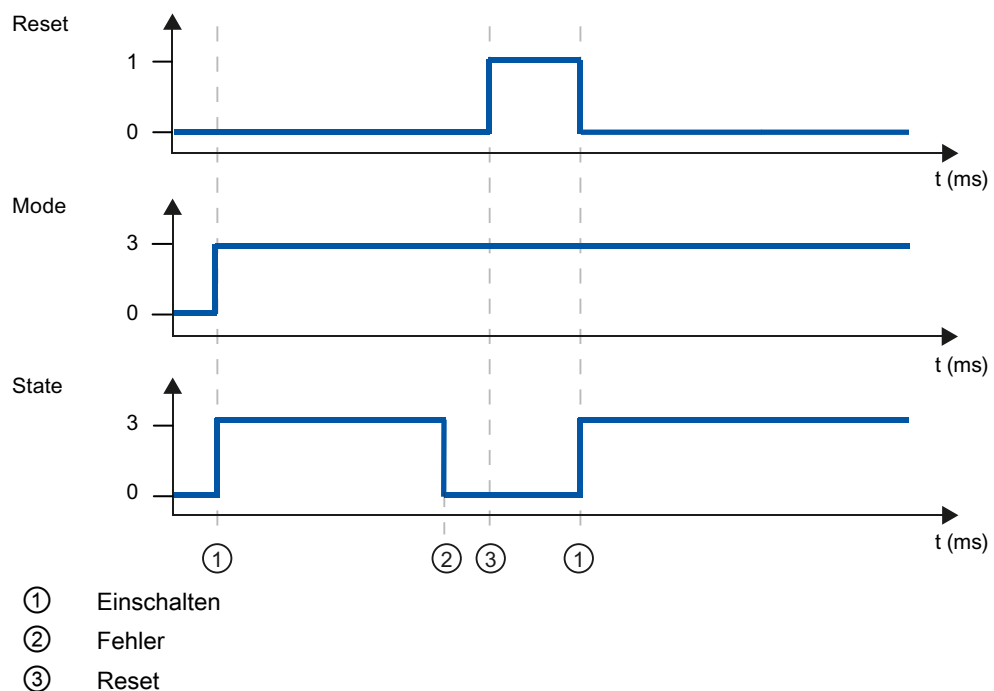
Stehen gleichzeitig mehrere Fehler an, so werden die Werte der Errorcodes binär addiert angezeigt. Wird z. B. der Errorcode 0003 angezeigt, so stehen gleichzeitig die Fehler 0001 und 0002 an.

| <b>ErrorBits<br/>(DW#16#...)</b> | <b>Beschreibung</b>  |
|----------------------------------|--|
| 0000                             | Es steht kein Fehler an.   |
| 0001                             | Der Parameter "Input" ist außerhalb der Istwertgrenzen. <ul style="list-style-type: none"> <li>• Input &gt; Config.InputUpperLimit oder</li> <li>• Input &lt; Config.InputLowerLimit</li> </ul> Wenn ActivateRecoverMode = TRUE und ErrorBehaviour = 1, wird das Stellglied auf den Ersatzausgangswert verfahren. Wenn ActivateRecoverMode = TRUE und ErrorBehaviour = 0, bleibt das Stellglied in der aktuellen Position stehen. Wenn ActivateRecoverMode = FALSE, bleibt das Stellglied in der aktuellen Position stehen.<br>PID_3Step V1.1<br>Im Handbetrieb können Sie das Stellglied verfahren.<br>PID_3Step V1.0<br>Handbetrieb ist in diesem Zustand nicht möglich. Erst wenn der Fehler behoben ist, können Sie das Stellglied wieder verfahren. |
| 0002                             | Ungültiger Wert am Parameter "Input_PER". Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.  |
| 0004                             | Fehler während der Nachoptimierung. Die Schwingung des Istwerts konnte nicht aufrecht erhalten werden.   |
| 0020                             | Die Ersoptimierung ist im Automatikbetrieb und während der Nachoptimierung nicht erlaubt.  |
| 0080                             | Fehler bei der Ersoptimierung. Die Ausgangswertgrenzen sind nicht korrekt konfiguriert. Überprüfen Sie, ob die Grenzen des Ausgangswerts korrekt konfiguriert sind und zum Regelsinn passen.   |
| 0100                             | Fehler während der Nachoptimierung führte zu ungültigen Parametern.  |
| 0200                             | Ungültiger Wert am Parameter "Input": Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.   |
| 0400                             | Berechnung des Ausgangswerts fehlgeschlagen. Überprüfen Sie die PID-Parameter.   |
| 0800                             | Abtastzeitfehler: PID_3Step wird nicht innerhalb der Abtastzeit des Weckalarm-OBs aufgerufen. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.  |
| 1000                             | Ungültiger Wert am Parameter "Setpoint": Wert hat kein gültiges Zahlenformat. Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.  |
| 2000                             | Ungültiger Wert am Parameter Feedback_PER. Überprüfen Sie, ob am Analogeingang ein Fehler ansteht. Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Handbetrieb ist in diesem Zustand nicht möglich. Um das Stellglied aus diesem Zustand verfahren zu können, müssen Sie die Stellungsrückmeldung deaktivieren (Config.FeedbackOn = FALSE). Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.   |

| ErrorBits<br>(DW#16#...) | Beschreibung  |
|--------------------------|---|
| 4000                     | Ungültiger Wert am Parameter Feedback. Wert hat kein gültiges Zahlenformat.<br>Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Handbetrieb ist in diesem Zustand nicht möglich. Um das Stellglied aus diesem Zustand verfahren zu können, müssen Sie die Stellungsrückmeldung deaktivieren (Config. FeedbackOn = FALSE).<br>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb.                                |
| 8000                     | Fehler bei der digitalen Stellungsrückmeldung. Actuator_H = TRUE und Actuator_L = TRUE.<br>Das Stellglied kann nicht auf den Ersatzausgangswert verfahren werden und bleibt in der aktuellen Position stehen. Handbetrieb ist in diesem Zustand nicht möglich.<br>Um das Stellglied aus diesem Zustand verfahren zu können, müssen die "Anschlagsignale Stellglied" deaktiviert werden (Config.ActuatorEndStopOn = FALSE).<br>Wenn vor Auftreten des Fehlers der Automatikbetrieb aktiv war, ActivateRecoverMode = TRUE und der Fehler nicht mehr ansteht, wechselt PID_3Step wieder in den Automatikbetrieb. |

### Parameter Reset

Durch eine steigende Flanke an Reset werden Fehler und Warnungen zurückgesetzt und der I-Anteil gelöscht. Durch eine fallende Flanke an Reset wird in die zuletzt aktive Betriebsart gewechselt.



**Variable ActivateRecoverMode**

Der Einfluss der Variablen ActivateRecoverMode ist abhängig von der Version der Anweisung PID\_3Step.

**Verhalten in Version 1.1**

Die Variable ActivateRecoverMode bestimmt im Automatikbetrieb das Verhalten im Fehlerfall. Während der Erstopptimierung, Nachoptimierung und Stellzeitmessung wirkt ActivateRecoverMode nicht.

| ActivateRecover Mode | Beschreibung  |
|----------------------|---|
| FALSE                | Im Fehlerfall wechselt PID_3Step in die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren". Der Regler wird erst durch Reset oder eine Änderung an Retain.Mode aktiviert.   |
| TRUE                 | <p>Wenn im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache.</p> <p>Wenn einer oder mehrere der folgenden Fehler auftreten, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung":</p> <ul style="list-style-type: none"> <li>• 0002h: Ungültiger Wert am Parameter Input_PER.</li> <li>• 0200h: Ungültiger Wert am Parameter Input.</li> <li>• 0800h: Abtastzeitfehler</li> <li>• 1000h: Ungültiger Wert am Parameter Setpoint.</li> <li>• 2000h: Ungültiger Wert am Parameter Feedback_PER.</li> <li>• 4000h: Ungültiger Wert am Parameter Feedback.</li> <li>• 8000h: Fehler bei der digitalen Stellungsrückmeldung.</li> </ul> <p>Bei den Fehlern 2000h, 4000h und 8000h kann PID_3Step den konfigurierten Ersatzausgangswert <b>nicht</b> anfahren.</p> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.</p> |



## Verhalten in Version 1.0

Die Variable ActivateRecoverMode bestimmt im Automatik- und Handbetrieb das Verhalten im Fehlerfall. Während der Erstopptimierung, Nachoptimierung und Stellzeitmessung wirkt ActivateRecoverMode nicht.

| ActivateRecover Mode | Beschreibung  |
|----------------------|---|
| FALSE                | Im Fehlerfall wechselt PID_3Step in die Betriebsart "Inaktiv" oder "Ersatzausgangswert anfahren". Der Regler wird erst durch Reset oder eine Änderung an Retain.Mode aktiviert.   |
| TRUE                 | <p><b>Fehler im Automatikbetrieb</b></p> <p>Wenn im Automatikbetrieb häufig Fehler auftreten, wird durch diese Einstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter ErrorBits und beheben Sie die Fehlerursache. Wenn einer oder mehrere der folgenden Fehler auftreten, wechselt PID_3Step in die Betriebsart "Ersatzausgangswert anfahren mit Fehlerüberwachung" oder "Fehlerüberwachung":</p> <ul style="list-style-type: none"> <li>• 0002h: Ungültiger Wert am Parameter Input_PER.</li> <li>• 0200h: Ungültiger Wert am Parameter Input.</li> <li>• 0800h: Abtastzeitfehler</li> <li>• 1000h: Ungültiger Wert am Parameter Setpoint.</li> <li>• 2000h: Ungültiger Wert am Parameter Feedback_PER.</li> <li>• 4000h: Ungültiger Wert am Parameter Feedback.</li> <li>• 8000h: Fehler bei der digitalen Stellungsrückmeldung.</li> </ul> <p>Bei den Fehlern 2000h, 4000h und 8000h kann PID_3Step den konfigurierten Ersatzausgangswert <b>nicht</b> anfahren.</p> <p>Sobald die Fehler nicht mehr anstehen, wechselt PID_3Step wieder in den Automatikbetrieb.</p> <p><b>Fehler im Handbetrieb</b></p> <p>Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID_3Step im Handbetrieb:</p> <ul style="list-style-type: none"> <li>• 0002h: Ungültiger Wert am Parameter Input_PER.</li> <li>• 0200h: Ungültiger Wert am Parameter Input.</li> <li>• 0800h: Abtastzeitfehler</li> <li>• 1000h: Ungültiger Wert am Parameter Setpoint.</li> <li>• 2000h: Ungültiger Wert am Parameter Feedback_PER.</li> <li>• 4000h: Ungültiger Wert am Parameter Feedback.</li> <li>• 8000h: Fehler bei der digitalen Stellungsrückmeldung.</li> </ul> <p>Bei den Fehlern 2000h, 4000h und 8000h können Sie das Ventil <b>nicht</b> in eine geeignete Position fahren.</p> |

## Siehe auch

Statische Variablen PID\_3Step (Seite 2761)

Parameter State und Retain.Mode (Seite 2768)

### Variable Warning

Stehen gleichzeitig mehrere Warnungen an, so werden die Werte der Warnungen binär addiert angezeigt. Wird z. B. die Warnung 0003 angezeigt, so stehen gleichzeitig die Warnungen 0001 und 0002 an.

| Warning<br>(DW#16#...) | Beschreibung  |
|------------------------|---|
| 0000                   | Es steht keine Warnung an.  |
| 0001                   | Während der Erstopoptimierung wurde der Wendepunkt nicht gefunden.  |
| 0002                   | Während der Nachoptimierung war die Oszillation verstärkt.  |
| 0004                   | Der Sollwert wurde begrenzt auf die eingestellten Grenzen.  |
| 0008                   | Für die gewählte Berechnungsmethode wurden nicht alle notwendigen Eigenschaften der Regelstrecke bestimmt. Ersatzweise wurden die PID-Parameter mit der Methode TuneRuleTIR = 3 berechnet.  |
| 0010                   | Die Betriebsart konnte nicht geändert werden, da ManualEnable = TRUE.   |
| 0020                   | Die Abtastzeit des PID-Algorithmus wird durch die Zykluszeit des aufrufenden OB begrenzt. Um bessere Ergebnisse zu erzielen, verwenden Sie kürzere Zykluszeiten des OB.   |
| 0040                   | Der Istwert hat eine seiner Warngrenze überschritten.   |
| 0080                   | Ungültiger Wert an Retain.Mode. Die Betriebsart wird nicht gewechselt.  |
| 0100                   | Der Handwert wurde begrenzt auf die Grenzen des Reglerausgangs.   |
| 0200                   | Die verwendete Regel zur Optimierung führt zu keinem richtigen Ergebnis oder wird nicht unterstützt.  |
| 0400                   | Bei Stellzeitmessung Methode ausgewählt nicht zum Stellglied passt.<br>Die Stellzeit kann nicht gemessen werden, da die Einstellungen des Stellglieds nicht zur gewählten Messmethode passen.   |
| 0800                   | Bei der Stellzeitmessung ist die Differenz zwischen der aktuellen Position und dem neuen Ausgangswert zu klein. Dies kann zu falschen Ergebnissen führen. Die Differenz zwischen aktuellem Ausgangswert und neuem Ausgangswert muss mindestens 50% des gesamten Stellbereichs betragen. |
| 1000                   | Der Ersatzausgangswert kann nicht erreicht werden, da er außerhalb der Ausgangswertgrenzen liegt.   |

Folgende Warnungen werden gelöscht, sobald die Ursache behoben ist:

- 0004
- 0020
- 0040
- 0100

Alle anderen Warnungen werden bei steigender Flanke an Reset gelöscht.

### Variable SUT.State

| SUT.State | Name        | Beschreibung                                      |
|-----------|-------------|---|
| 0         | SUT_INIT    | Erstopoptimierung initialisieren                  |
| 50        | SUT_TPDN    | Startposition ohne Stellungsrückmeldung ermitteln |
| 100       | SUT_STDABW  | Standardabweichung berechnen                      |
| 200       | SUT_GET_POI | Wendepunkt ermitteln                              |

| SUT.State | Name           | Beschreibung                      |
|-----------|----------------|-----------------------------------|
| 300       | SUT_GET_RISETM | Anregelzeit ermitteln             |
| 9900      | SUT_IO         | Erstoptimierung erfolgreich       |
| 1         | SUT_NIO        | Erstoptimierung nicht erfolgreich |

### Variable TIR.State

| TIR.State | Name          | Beschreibung  |
|-----------|---------------|---|
| -100      | TIR_FIRST_SUT | Die Nachoptimierung ist nicht möglich. Es wird erst eine Erstoptimierung durchgeführt.                |
| 0         | TIR_INIT      | Nachoptimierung initialisieren  |
| 200       | TIR_STDABW    | Standardabweichung berechnen  |
| 300       | TIR_RUN_IN    | Versuchen Sollwert mit dem maximalen oder minimalen Ausgangswert zu erreichen                         |
| 400       | TIR_CTRLN     | Versuchen Sollwert mit vorhandenen PID-Parametern zu erreichen (wenn Erstoptimierung erfolgreich war) |
| 500       | TIR_OSZIL     | Oszillation ermitteln und Parameter berechnen   |
| 9900      | TIR_IO        | Nachoptimierung erfolgreich   |
| 1         | TIR_NIO       | Nachoptimierung nicht erfolgreich   |

## 9.7.5 Kommunikation

### 9.7.5.1 Kommunikationsprozessor

SIMATIC NET-CM/CP

S7-1500-CM/CP

Industrial Ethernet

Anweisungen für FTP-Dienste

FTP\_CMD für FTP-Dienste

Übersicht FTP\_CMD

#### Bedeutung

Mit der Anweisung FTP\_CMD können Sie FTP-Verbindungen aufbauen und Dateien von und zu einem FTP-Server übertragen.

Der Datentransfer ist über FTP oder FTPS (gesicherte SSL-Verbindungen) möglich.

---

#### Hinweis

##### **FTPS: Zertifikate abgleichen**

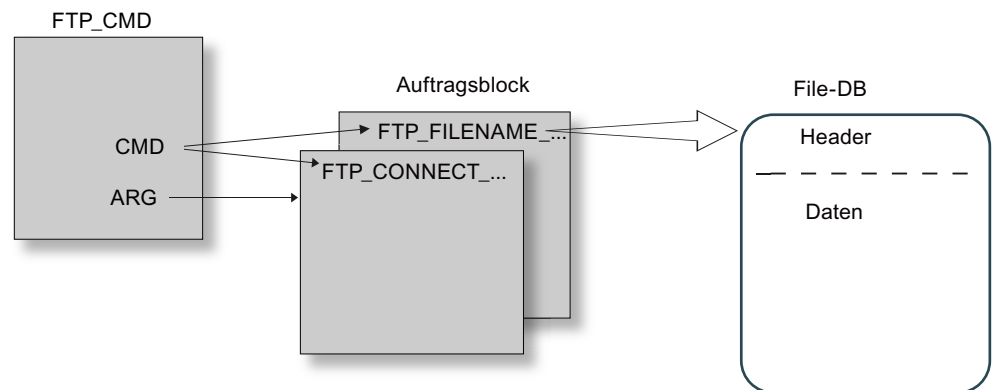
FTPS erfordert den Abgleich der Zertifikate zwischen FTP-Server und FTP-Client. Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich. Importieren Sie im Zertifikatsmanager das Zertifikat des FTP-Servers als vertrauenswürdige Zertifikat.

---

#### Funktionsweise

Die Anweisung FTP\_CMD verweist auf einen Auftragsblock (ARG), in dem das FTP-Kommando spezifiziert wird. Je nach Typ des FTP-Kommandos (CMD) verwendet dieser Auftragsblock unterschiedliche Datenstrukturen zur Parametrierung. Für diese unterschiedlichen Strukturen stehen jeweils passende Datentypen (UDTs) zur Verfügung.

Die folgende Darstellung zeigt die Aufrufstruktur:



## Auftragsblöcke

Für die Auftragsblöcke werden folgende Datenstrukturen verwendet:

- **Verbindungsaufbau**  
Für den Verbindungsaufbau stehen unterschiedliche Datenstrukturen für folgende Zugriffsarten zur Verfügung:
  - UDT FTP\_CONNECT\_IPV4: Verbindungsaufbau mit IP-Adressen gemäß IPv4
  - UDT FTP\_CONNECT\_IPV6: Verbindungsaufbau mit IP-Adressen gemäß IPv6
  - UDT FTP\_CONNECT\_NAME: Verbindungsaufbau mit Server-Namen (DNS)
- **Datentransfer**  
Für den Datentransfer stehen zwei unterschiedliche Datenstrukturen zur Verfügung:
  - UDT FTP\_FILENAME: Datenstruktur für den Zugriff auf eine vollständige Datei
  - UDT FTP\_FILENAME\_PART: Datenstruktur für den lesenden Zugriff auf einen Datenbereich

## Datenübertragung im File\_DB

Der Datentransfer erfolgt über Datenbausteine, die einen Header für Auftragsdaten sowie den Bereich für die Nutzdaten enthalten. Der Datenbaustein wird im Auftragspuffer angegeben.

## Voraussetzungen in der CPU-Projektierung

Verwenden Sie folgende Einstellungen um den FTP-Zugriff zu ermöglichen:

- In den Projektierdaten der CPU unter "Eigenschaften > Allgemein > Schutz": Deaktivieren Sie die Option "PUT/GET-Kommunikation deaktivieren".
- Bei allen als File-DB verwendeten Datenbausteinen deaktivieren Sie das Attribut "Optimierter Bausteinzugriff"

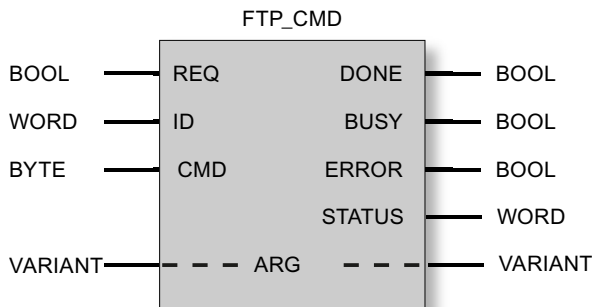
### Gültigkeit

Die Anweisung FTP\_CMD kann mit folgenden Baugruppentypen eingesetzt werden:

- CP 1543-1

### Aufrufschnittstelle

Aufrufschnittstelle in FUP-Darstellung



### Siehe auch

Ausgangsparameter und Statusinformationen FTP\_CMD (Seite 2792)

Eingangsparameter FTP\_CMD (Seite 2784)

Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Client-Betrieb (Seite 2795)

### Eingangsparameter FTP\_CMD

#### Erläuterung der Eingangsparameter

Sie versorgen die Anweisung FTP\_CMD mit folgenden Eingangsparametern:

Tabelle 9-90 Formalparameter der Anweisung FTP\_CMD - Eingangsparameter

| Parameter | Deklaration | Datentyp | Speicherbereich | Bedeutung / Bemerkung   |
|-----------|-------------|----------|-----------------|---|
| REQ       | Input       | BOOL     | E, A, M, D, L   | Startet den Sendeauftrag bei einer steigenden Flanke.   |
| ID        | INPUT       | INT      |                 | Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung. |

| Parameter | Deklaration | Datentyp | Speicherbereich | Bedeutung / Bemerkung  |
|-----------|-------------|----------|-----------------|--|
| CMD       | INPUT       | BYTE     |                 | <p>FTP-Kommando, das beim Aufruf der Anweisung ausgeführt werden soll. Wertebereiche für die FTP-Kommandotypen finden Sie im Anschluss an die Tabelle.</p> <p>Hinweis:<br/>Das hier angegebene FTP-Kommando muss identisch im Auftragsblock (Parameter ARG) angegeben werden.<br/>Wenn ein Kommando nicht von der CP-Firmware unterstützt wird, dann wird eine Fehlermeldung mit STATUS = 8F6B<sub>H</sub> ausgegeben.</p> |
| ARG       | INPUT       | VARIANT  |                 | <p>Auftragsblock</p> <p>Verweist auf den Datenbereich mit den zum FTP-Kommando passenden Ausführungsparametern.</p> <p>Abhängig vom FTP-Kommando werden spezifische Datentypen (UDT) verwendet. Die UDTs werden nachfolgend angegeben.</p> <p>Hinweis:<br/>Datentyp ANY nicht verwenden<br/>Für den hier anzugebenden Zeiger ist der Datentyp ANY <b>nicht</b> zulässig!</p>   |

### FTP-Kommandos im Parameter "CMD"

Entnehmen Sie der folgenden Tabelle, welche Bedeutung die Kommandos des Parameters "CMD" haben und welche UDTs Sie zur Versorgung der Auftragsblöcke verwenden.

Tabelle 9-91 Kommandotypen

| CMD (Kommandotyp) | Relevante Auftragsblöcke / UDT                                    | Bedeutung / Handhabung  |
|-------------------|---|---|
| 0 (NOOP)          | -   | <p>Der aufgerufene FC führt keine Aktionen aus. Die Statusanzeigen werden bei dieser Parameterversorgung wie folgt gesetzt:</p> <ul style="list-style-type: none"> <li>• DONE=1; ERROR=0; STATUS=0</li> </ul>   |
| 1 (CONNECT)       | <p>FTP_CONNECT_IPV4<br/>FTP_CONNECT_IPV6<br/>FTP_CONNECT_NAME</p> | <p>FTP-Verbindungsaufbau</p> <p>Mit diesem Kommando baut der FTP-Client eine FTP-Verbindung zu einem FTP-Server auf (Port 21).</p> <p>Die Verbindung steht unter der hier zugewiesenen Verbindungs-ID für alle weiteren FTP-Kommandos zur Verfügung. Daten werden dann mit dem für diesen Benutzer angegebenen FTP-Server ausgetauscht.</p> |
| 2 (STORE)         | FTP_FILENAME  | <p>Mit diesem Funktionsaufruf wird ein Datenbaustein (File-DB) vom FTP-Client (S7-CPU) zum FTP-Server übertragen.</p> <p>Achtung: Falls die Datei (File-DB) auf dem FTP-Server schon vorhanden ist, wird diese überschrieben.</p>   |

| CMD (Kommandotyp) | Relevante Auftragsblöcke / UDT | Bedeutung / Handhabung  |
|-------------------|--------------------------------|---|
| 3 (RETRIEVE)      | FTP_FILENAME                   | Mit diesem Funktionsaufruf wird eine Datei vom FTP-Server zum FTP-Client (S7-CPU) übertragen.<br>Achtung: Falls der Datenbaustein (File-DB) beim FTP-Client schon eine Datei enthält, wird diese überschrieben.   |
| 4 (DELETE)        | FTP_FILENAME                   | Mit diesem Funktionsaufruf löschen Sie eine Datei auf dem FTP-Server.   |
| 5 (QUIT)          | Keine                          | Mit diesem Funktionsaufruf bauen Sie die über die ID benannte FTP-Verbindung ab.  |
| 6 (APPEND)        | FTP_FILENAME                   | Ähnlich wie "STORE" speichert das Kommando "APPEND" (anhängen) eine Datei auf dem FTP-Server. Bei "APPEND" wird die Datei auf dem FTP-Server aber nicht überschrieben, sondern der neue zu speichernde Inhalt wird an die Datei angehängt.<br>Falls die Datei auf dem FTP-Server nicht vorhanden ist, wird sie angelegt.  |
| 7 (RETR_PART)     | FTP_FILENAME_PART              | Mit dem Kommando "RETR_PART" (Teillänge lesen) können Sie einen Ausschnitt einer Datei vom FTP-Server anfordern.<br>Bei sehr großen Dateien können Sie damit das Lesen auf den Teil beschränken, den Sie gerade benötigen.<br>Dazu müssen Sie die Struktur dieser Datei kennen.<br>Geben Sie den gewünschten Ausschnitt der Datei mithilfe der zwei Parameter "OFFSET" und "LEN" am FB 40 an. |

**Siehe auch**

Übersicht FTP\_CMD (Seite 2782)

**Auftragsblöcke für FTP\_CMD**

**Bedeutung**

Sie versorgen die Anweisung FTP\_CMD über den Parameter ARG mit einem Auftragsblock. Die Struktur ist abhängig vom FTP-Kommandotyp. Indem Sie die vorgegebenen Datentypen (UDT) verwenden, erkennt die Anweisung den Typ des Auftragsblocks. Nachfolgend werden die jeweiligen Datentypen (UDT) für folgende Auftragsblöcke angegeben:

- FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4
- FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6
- FTP-Verbindungsaufbau mit Server-Name
- Schreib- und Lesezugriff sowie übrige FTP-Kommandos
- FTP-Kommando RETR\_PART



### Auftragsblock für FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4

Für den FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv4 wird die folgende Datenstruktur verwendet.

Tabelle 9-92 UDT FTP\_CONNECT\_IPV4

| Parameter         | Typ        | Wertebereich        | Bedeutung / Bemerkung  |
|-------------------|------------|---------------------|--|
| LADDR             | WORD       |                     | Baugruppen-Anfangsadresse<br>Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP.<br>Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter:<br>"Eigenschaften>Adressen>Eingänge"  |
| ID                | INT        | 1, 2...64           | Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.  |
| ConnectionType    | BYTE       | 0                   | Verbindungstyp "FTP"   |
| ActiveEstablished | BOOL       | TRUE                |  |
| FTPcmd            | BYTE       | 1                   | FTP-Kommando "CONNECT"<br>FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche finden Sie in Tabelle 9-91 Kommandotypen (Seite 2785)<br>Hinweis:<br>Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden. |
| CertIndex         | BYTE       | 0 = FTP<br>1 = FTPS | Wählen Sie hier zwischen den Protokolltypen FTP oder FTPS.<br>Hinweis für FTPS:<br>Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.                             |
| UserName          | STRING[32] | 'benutzer'          | Benutzername für das Login auf dem FTP-Server  |
| Password          | STRING[32] | 'passwort'          | Passwort für das Login auf dem FTP-Server  |
| FTPserverIP_addr  | IPv4       | '142.11.25.135'     | IP-Adresse des FTP-Servers   |

### Auftragsblock für FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6

Für den FTP-Verbindungsaufbau mit IP-Adresse gemäß IPv6 wird die folgende Datenstruktur verwendet.

Tabelle 9-93 UDT FTP\_CONNECT\_IPV6

| Parameter         | Typ                              | Wertebereich        | Bedeutung / Bemerkung  |
|-------------------|----------------------------------|---------------------|--|
| LADDR             | WORD                             |                     | Baugruppen-Anfangsadresse<br>Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP.<br>Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter:<br>"Eigenschaften>Adressen>Eingänge"  |
| ID                | INT                              | 1, 2...64           | Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.  |
| ConnectionType    | BYTE                             | 0                   | Verbindungstyp "FTP"   |
| ActiveEstablished | BOOL                             | TRUE                |  |
| FTPcmd            | BYTE                             | 1                   | FTP-Kommando "CONNECT"<br>FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche finden Sie in Tabelle 9-91 Kommandotypen (Seite 2785)<br>Hinweis:<br>Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden. |
| CertIndex         | BYTE                             | 0 = FTP<br>1 = FTPS | Wählen Sie hier zwischen den Protokolltypen FTP oder FTPS.<br>Hinweis für FTPS:<br>Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.                             |
| UserName          | STRING[32]                       | 'benutzer'          | Benutzername für das Login auf dem FTP-Server  |
| Password          | STRING[32]                       | 'passwort'          | Passwort für das Login auf dem FTP-Server  |
| FTPserverIP_addr  | IPv6<br>array [1..16] of<br>Byte |                     | IP-Adresse des FTP-Servers   |

### Auftragsblock für FTP-Verbindungsaufbau mit Server-Name

Für den FTP-Verbindungsaufbau mit Angabe eines Server-Namens wird die folgende Datenstruktur verwendet. Der Server-Name wird über DNS einer IP-Adresse zugewiesen.

Tabelle 9-94 UDT FTP\_CONNECT\_NAME

| Parameter         | Typ         | Wertebereich        | Bedeutung / Bemerkung  |
|-------------------|-------------|---------------------|--|
| LADDR             | WORD        |                     | Baugruppen-Anfangsadresse<br>Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP.<br>Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter:<br>"Eigenschaften>Adressen>Eingänge"  |
| ID                | INT         | 1, 2...64           | Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.  |
| ConnectionType    | BYTE        | 0                   | Verbindungstyp "FTP"   |
| ActiveEstablished | BOOL        | TRUE                |  |
| FTPcmd            | BYTE        | 1                   | FTP-Kommando "CONNECT"<br>FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche finden Sie in Tabelle 9-91 Kommandotypen (Seite 2785)<br>Hinweis:<br>Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden. |
| CertIndex         | BYTE        | 0 = FTP<br>1 = FTPS | Wählen Sie hier zwischen den Protokolltypen FTP oder FTPS.<br>Hinweis für FTPS:<br>Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.                             |
| UserName          | STRING[32]  | 'benutzer'          | Benutzername für das Login auf dem FTP-Server  |
| Password          | STRING[32]  | 'passwort'          | Passwort für das Login auf dem FTP-Server  |
| FTPserverName     | STRING[254] |                     | IP-Adresse des FTP-Servers   |

### Auftragsblock für Schreib- und Lesezugriff sowie übrige FTP-Kommandos

Für die FTP-Kommandos store, retrieve, delete und append wird die folgende Datenstruktur verwendet.

Tabelle 9-95 UDT FTP\_FILENAME

| Parameter         | Typ                            | Wertebereich        | Bedeutung / Bemerkung   |
|-------------------|--------------------------------|---------------------|---|
| LADDR             | WORD                           |                     | Baugruppen-Anfangsadresse<br>Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP.<br>Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter:<br>"Eigenschaften>Adressen>Eingänge"                                     |
| ID                | INT                            | 1, 2...64           | Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.   |
| ConnectionType    | BYTE                           | 0                   | Verbindungstyp "FTP"  |
| ActiveEstablished | BOOL                           | TRUE                |   |
| FTPcmd            | BYTE                           | 2, 3, 4, 6          | FTP-Kommando "STORE / RETRIEVE / DELETE / APPEND"<br>FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. Wertebereiche finden Sie in Tabelle Tabelle 9-91 Kommandotypen (Seite 2785)<br>Hinweis:<br>Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden. |
| CertIndex         | BYTE                           | 0 = FTP<br>1 = FTPS | Wählen Sie hier zwischen den Protokolltypen FTP oder FTPS.<br>Hinweis für FTPS:<br>Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.  |
| DataBlockNumber   | UINT                           |                     | Der hier angegebene Datenbaustein enthält den zu lesenden / zu schreibenden File-DB.  |
| LenFilename       | UINT                           | 0...1000            | Gesamtlänge des Dateinamens.  |
| Filename          | ARRAY[0..3] OF<br>STRING[ 220] |                     | Dateiname der Ziel- bzw. Quelldatei.  |

## Auftragsblock für das FTP-Kommando RETR\_PART

Für das FTP-Kommando RETR\_PART wird die folgende Datenstruktur verwendet.

Tabelle 9-96 UDT FTP\_FILENAME\_PART

| Parameter         | Typ   | Wertebereich        | Bedeutung / Bemerkung   |
|-------------------|-------|---------------------|---|
| LADDR             | WORD  |                     | Baugruppen-Anfangsadresse<br>Beim Aufruf eine Anweisung übergeben Sie im Parameter LADDR die Baugruppen-Anfangsadresse des CP.<br>Die Baugruppen-Anfangsadresse des CP können Sie in der Projektierung des CP entnehmen unter:<br>"Eigenschaften>Adressen>Eingänge"   |
| ID                | INT   | 1, 2...64           | Die FTP-Aufträge werden über FTP-Verbindungen abgewickelt. Der Parameter identifiziert die genutzte Verbindung.   |
| ConnectionType    | BYTE  | 0                   | Verbindungstyp "FTP"  |
| ActiveEstablished | BOOL  | TRUE                |   |
| FTPcmd            | BYTE  | 7                   | FTP-Kommando "RETR_PART"<br>FTP-Kommando, das beim Aufruf der Anweisung ausgeführt wird. I. Wertebereiche finden Sie in Tabelle Tabelle 9-91 Kommandotypen (Seite 2785)<br>Hinweis:<br>Das hier angegebene FTP-Kommando muss identisch im Eingangsparameter CMD angegeben werden.   |
| CertIndex         | BYTE  | 0 = FTP<br>1 = FTPS | Wählen Sie hier zwischen den Protokolltypen FTP oder FTPS.<br>Hinweis für FTPS:<br>Wenn der FTP-Server außerhalb des STEP 7-Projekts des FTP-Client konfiguriert wird, dann ist der Import des Zertifikates vom FTP-Server erforderlich.  |
| Offset            | DWORD |                     | Offset in Byte, ab dem die Datei gelesen werden soll.   |
| Length            | DWORD |                     | Teillänge in Byte, die ab dem in "OFFSET" angegebenen Wert gelesen werden.<br>Besonderheiten: <ul style="list-style-type: none"> <li>• Bei Angabe von "DW#16#FFFFFFFF" wird der verfügbare Rest der Datei gelesen.<br/>Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt.</li> <li>• Wenn OFFSET &gt; Länge der Original-Datei:<br/>Länge der Ziel-Datei (ACT_LENGTH im File-DB): 0 Byte in der CPU.<br/>Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt.</li> <li>• Wenn OFFSET + LEN &gt; Länge der Original-Datei (und LEN ≠ 0xFFFFFFFF):<br/>Länge der Ziel-Datei (ACT_LENGTH im File-DB):<br/>Verfügbare Bytes ab "OFFSET".</li> </ul> Ergebnis OK (DONE = 1, STATUS = 0), falls kein sonstiger Fehler auftritt. |

| Parameter       | Typ                        | Wertebereich | Bedeutung / Bemerkung  |
|-----------------|----------------------------|--------------|--|
| DataBlockNumber | UINT                       |              | Der hier angegebene Datenbaustein enthält den zu lesenden / zu schreibenden File-DB. |
| LenFilename     | UINT                       | 0...1000     | Gesamtlänge des Dateinamens.   |
| Filename        | ARRAY[0..3] OF STRING[220] |              | Dateiname der Ziel- bzw. Quelldatei.   |

**Siehe auch**

Übersicht FTP\_CMD (Seite 2782)

Eingangsparameter FTP\_CMD (Seite 2784)

**Ausgangsparameter und Statusinformationen FTP\_CMD**

**Parameter BUSY, DONE und ERROR**

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "FTP\_CMD" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| BUSY | DONE | ERROR | Beschreibung  |
|------|------|-------|---|
| 1    | -    | -     | Der Auftrag wird bearbeitet.  |
| 0    | 1    | 0     | Der Auftrag wurde erfolgreich durchgeführt.   |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben. |
| 0    | 0    | 0     | Kein neuer Auftrag wurde zugewiesen.  |

**Statusanzeigen auswerten**

**Hinweis**

Beachten Sie für die Einträge mit der Codierung 8FxxH unter STATUS auch die Angaben im Referenzhandbuch STEP 7 Standard und Systemfunktionen. Sie finden dort Hinweise im Kapitel "Fehlerauswertung mit dem Ausgangsparameter RET\_VAL".

Tabelle 9-97 FTP\_CMD: Bedeutung des Parameters STATUS in Zusammenhang mit DONE und ERROR

| DONE | ERROR | STATUS            | Bedeutung   |
|------|-------|-------------------|---|
| 0    | 0     | 0000 <sub>H</sub> | Es ist kein Auftrag in Bearbeitung.   |
| 1    | 0     | 0000 <sub>H</sub> | Der Auftrag ist fertig ohne Fehler.   |
| 0    | 0     | 7001 <sub>H</sub> | Der Auftrag wurde erstmalig angestoßen (BUSY=1)   |
| 0    | 0     | 7002 <sub>H</sub> | Der Auftrag läuft noch (BUSY=1)   |
| 0    | 0     | 80C4 <sub>H</sub> | Kommunikationsfehler (tritt temporär auf; daher ist Wiederholung im Anwenderprogramm sinnvoll.)   |
| 0    | 0     | 8183 <sub>H</sub> | Die Projektierung entspricht nicht den Auftragsparametern.  |
| 0    | 1     | 8401 <sub>H</sub> | unbekannter Fehler<br>mögliche Ursachen: <ul style="list-style-type: none"> <li>• Auf der Verbindung wurde ein Timeout erkannt.</li> <li>• Der FTP-Server hat die Verbindung abgebrochen.</li> </ul> Abhilfe: <ul style="list-style-type: none"> <li>• QUIT und CONNECT-Kommandos erneut senden um die Verbindung wieder einzurichten.</li> </ul> |
| 0    | 1     | 8402 <sub>H</sub> | Verbindung befindet sich in fehlerhaftem Zustand.<br>Das Timeout der Verbindung kann überschritten sein oder der FTP-Server kann die Verbindung abgebaut haben.<br>Abhilfe:<br>Senden Sie erneut die Kommandos QUIT und CONNECT um die Verbindung wieder aufzubauen.  |
| 0    | 1     | 8403 <sub>H</sub> | Login ist fehlgeschlagen.   |
| 0    | 1     | 8404 <sub>H</sub> | FTP-Server ist nicht erreichbar.  |
| 0    | 1     | 8405 <sub>H</sub> | Übertragung ist fehlgeschlagen.   |
| 0    | 1     | 8406 <sub>H</sub> | Timeout bei aktuellem Vorgang   |
| 0    | 1     | 8407 <sub>H</sub> | Datei im FTP-Server wurde nicht gefunden.   |
| 0    | 1     | 8408 <sub>H</sub> | Übertragung nicht möglich.  |
| 0    | 1     | 8409 <sub>H</sub> | Datei konnte nicht geholt werden.   |
| 0    | 1     | 8410 <sub>H</sub> | Das Setzen des TCP Port für die Datenverbindung ist fehlgeschlagen.   |
| 0    | 1     | 8411 <sub>H</sub> | Offset-Angabe passt nicht.  |
| 0    | 1     | 8412 <sub>H</sub> | Fehler beim Wechsel der Verzeichnisangabe   |
| 0    | 1     | 8413 <sub>H</sub> | Fehler beim Daten Empfangen   |
| 0    | 1     | 8414 <sub>H</sub> | Fehler beim Daten Senden.   |
| 0    | 1     | 8415 <sub>H</sub> | CMD-Angabe (Kommandotyp) wurde vom Client abgewiesen.   |
| 0    | 1     | 8416 <sub>H</sub> | Verbindung wurde vom FTP-Server geschlossen.  |
| 0    | 1     | 8418 <sub>H</sub> | Fehler in den Nutzdaten.<br>Mögliche Ursachen: <ul style="list-style-type: none"> <li>• Dateiname ist leer.</li> <li>• Datenlänge ist "0"</li> <li>• usw.</li> </ul>  |
| 0    | 1     | 8419 <sub>H</sub> | Es ist keine Socket Ressource zum Öffnen einer Datenverbindung vorhanden.   |
| 0    | 1     | 8420 <sub>H</sub> | Es ist keine Socket Ressource zum Öffnen einer Controlverbindung vorhanden.   |

| DONE | ERROR | STATUS            | Bedeutung   |
|------|-------|-------------------|---|
| 0    | 1     | 8421 <sub>H</sub> | Fehler beim Öffnen des zu lesenden File-DB.   |
| 0    | 1     | 8422 <sub>H</sub> | Fehler beim Öffnen des zu beschreibenden File-DB.   |
| 0    | 1     | 8423 <sub>H</sub> | Der Verbindungsaufbau zum FTP Server ist fehlgeschlagen.  |
| 0    | 1     | 8424 <sub>H</sub> | interner Fehler   |
| 0    | 1     | 8425 <sub>H</sub> | Formatfehler beim Domain-Namen  |
| 0    | 1     | 8426 <sub>H</sub> | Es stehen zu viele DNS-Anfragen an.   |
| 0    | 1     | 8427 <sub>H</sub> | Der angegebene DNS-Server konnte den spezifizierten Domain-Namen nicht zuordnen.  |
| 0    | 1     | 8428 <sub>H</sub> | Es ist keine Verbindungs-Ressource verfügbar.   |
| 0    | 1     | 8429 <sub>H</sub> | Unbekannte Kanal-ID   |
| 0    | 1     | 8430 <sub>H</sub> | Der File-DB ist zu kurz.  |
| 0    | 1     | 8431 <sub>H</sub> | Fehler beim Schreiben in den File-DB.   |
| 0    | 1     | 8432 <sub>H</sub> | Fehler beim Lesen aus dem File-DB.  |
| 0    | 1     | 8433 <sub>H</sub> | Fehler beim Zugriff auf den File-DB.  |
| 0    | 1     | 8434 <sub>H</sub> | Aktion wurde abgebrochen.   |
| 0    | 1     | 8435 <sub>H</sub> | Kanal wird zurückgesetzt.   |
| 0    | 1     | 8436 <sub>H</sub> | Unerwartete Server Antwort  |
| 0    | 1     | 8437 <sub>H</sub> | Zertifikat konnte nicht verifiziert werden.   |
| 0    | 1     | 8438 <sub>H</sub> | Unbekannter Fehler aufgetreten  |
| 0    | 1     | 8439 <sub>H</sub> | Das FTP-Kommando führt zu einem Fehler. Die Ursache ist beim FTP-Server zu suchen (REST-Kommando).  |
| 0    | 1     | 8440 <sub>H</sub> | Der FTP-Server unterstützt das geforderte SSL-Protokoll nicht.  |
| 0    | 1     | 8446 <sub>H</sub> | Nachdem das FTP-Passwort an den FTP-Server gesendet wurde, wurde ein unerwarteter Code vom FTP-Server zurückgegeben.  |
| 0    | 1     | 8451 <sub>H</sub> | Beim Versuch, den Übertragungsmodus von binär auf ASCII umzustellen, wurde ein Fehler gemeldet.   |
| 0    | 1     | 8455 <sub>H</sub> | Eine Speicheranforderung im CM/CP ist fehlgeschlagen.   |
| 0    | 1     | 8460 <sub>H</sub> | Bei der SSL/TLS-Abwicklung ist ein Problem aufgetreten.   |
| 0    | 1     | 8469 <sub>H</sub> | Schnittstellen-Fehler<br>Die angegebene Ausgangs-Schnittstelle konnte nicht genutzt werden.<br>Abhilfe:<br>Stellen Sie ein, welche Schnittstelle für ausgehende Verbindungen genutzt werden soll. |
| 0    | 1     | 8475 <sub>H</sub> | Das SSL-Zertifikat oder der SSH md5 fingerprint wurde als nicht vertrauenswürdig erachtet.  |
| 0    | 1     | 8476 <sub>H</sub> | Vom FTP-Server wurde nichts empfangen. Im aktuellen Zustand muss von einem fehlerhaften Verhalten ausgegangen werden.   |
| 0    | 1     | 8477 <sub>H</sub> | Die angegebene "Crypto engine" (kryptografisches Modul) wurde nicht gefunden.   |
| 0    | 1     | 8478 <sub>H</sub> | Der Vorgang schlug fehl, die gewählte SSL-"Crypto engine" (kryptografisches Modul) als Standard zu setzen.  |
| 0    | 1     | 8480 <sub>H</sub> | Es ist ein Problem mit dem Zertifikat des FTP-Client aufgetreten.   |
| 0    | 1     | 8481 <sub>H</sub> | Die angegebene Ziffer konnte nicht verwendet werden.  |
| 0    | 1     | 8482 <sub>H</sub> | Der FTP-Server verwendet eine Codierung die nicht unterstützt wird.   |
| 0    | 1     | 8484 <sub>H</sub> | Die maximale Dateigröße wurde überschritten.  |



| DONE | ERROR | STATUS            | Bedeutung  |
|------|-------|-------------------|--|
| 0    | 1     | 8485 <sub>H</sub> | Der File-DB wurde während der Sendebearbeitung verändert oder der File-DB ist nicht korrekt aufgebaut.   |
| 0    | 1     | 8489 <sub>H</sub> | Daten konnten nicht gesendet werden. Es ist nicht genügend Speicherkapazität für den Vorgang auf dem FTP-Server verfügbar.   |
| 0    | 1     | 8492 <sub>H</sub> | Die Datei existiert bereits. Die Datei wird nicht überschrieben.   |
| 0    | 1     | 8496 <sub>H</sub> | Beim Lesen des SSL CA-Zertifikats ist ein Problem aufgetreten.   |
| 0    | 1     | 8497 <sub>H</sub> | Ein unerwarteter Fehler ist bei der SSH-Sitzung aufgetreten.   |
| 0    | 1     | 8498 <sub>H</sub> | Es war nicht möglich, die SSL-Verbindung abzubauen.  |
| 0    | 1     | 8499 <sub>H</sub> | Der Socket ist nicht bereit zum Senden/Empfangen. Warten Sie, bis die Bereitschaft besteht und versuchen Sie es erneut.  |
| 0    | 1     | 8501 <sub>H</sub> | Die Überprüfung des SSL-Zertifikates vom FTP-Server ist fehlgeschlagen.  |
| 0    | 1     | 8507 <sub>H</sub> | Während der aktiven FTP-Sitzung ist im Rahmen des Verbindungsaufbaus beim Warten auf den FTP-Server ein timeout aufgetreten.   |
| 0    | 1     | 8F55 <sub>H</sub> | Header-Status-Bit: Locked  |
| 0    | 1     | 8F56 <sub>H</sub> | Das NEW-Bit im File-DB-Header wurde nicht zurückgesetzt  |
| 0    | 1     | 8F6B <sub>H</sub> | Mögliche Ursachen: <ul style="list-style-type: none"> <li>• Falscher Wert für den Parameter CMD<br/>Werte von 0 bis 15 sind zugelassen.</li> <li>• Ein Kommando des FB 40 wird nicht unterstützt.<br/>Mögliche Ursache: Falsche Firmware des CP<br/>Abhilfe: Firmware-Update (bei älteren CPs statt des FB 40 die Funktionen FC 40...FC 44 benutzen.)</li> </ul> |
| 0    | 1     | 8F7F <sub>H</sub> | Interner Fehler; beispielsweise unzulässige ANY-Referenz   |

## Siehe auch

Übersicht FTP\_CMD (Seite 2782)

## Aufbau und Verwendung des File-DB - FTP-Client-Betrieb

### Aufbau der Datenbausteine (File-DB) für FTP-Dienste - FTP-Client-Betrieb

#### Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten:

- Abschnitt 1: File-DB-Header (besitzt feste Struktur mit einer Länge von 20 Byte)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)

### Voraussetzungen in der CPU-Projektierung

Verwenden Sie folgende Einstellungen um den FTP-Zugriff zu ermöglichen:

- In den Projektierdaten der CPU unter "Eigenschaften > Allgemein > Schutz": Deaktivieren Sie die Option "PUT/GET-Kommunikation deaktivieren".
- Bei allen als File-DB verwendeten Datenbausteinen deaktivieren Sie das Attribut "Optimierter Bausteinzugriff"

### File-DB-Header für FTP-Client-Betrieb

Anmerkung: Der hier beschriebene File-DB Header ist weitgehend identisch zu dem für den Server-Betrieb beschriebenen File-DB-Header. Die Unterschiede beziehen sich auf die Parameter:

- WRITE\_ACCESS
- FTP\_REPLY\_CODE

| Parameter | Typ  | Wert / Bedeutung   | Versorgung   |
|-----------|------|--|--|
| EXIST     | BOOL | <p>Das EXIST-Bit zeigt an, ob der Nutzdatenbereich gültige Daten enthält.</p> <p>Das FTP-Kommando retrieve bearbeitet den Auftrag nur, wenn EXIST=1.</p> <ul style="list-style-type: none"> <li>• 0:<br/>Der File-DB enthält keine gültigen Nutzdaten ("Datei existiert nicht").</li> <li>• 1:<br/>Der File-DB enthält gültige Nutzdaten ("Datei existiert").</li> </ul> | <p>Das FTP-Kommando dele setzt EXIST=0;</p> <p>Das FTP-Kommando store setzt EXIST=1;</p>   |
| LOCKED    | BOOL | <p>Das LOCKED-Bit dient zum Zugriffsschutz für den File-DB.</p> <ul style="list-style-type: none"> <li>• 0:<br/>Auf den File-DB kann zugegriffen werden.</li> <li>• 1:<br/>Der File-DB ist gesperrt.</li> </ul>  | <p>Die FTP-Kommandos store und retr setzen während der Bearbeitung LOCKED=1.</p> <p>Für einen Schreibvorgang aus dem Anwenderprogramm gilt:</p> <p>Das Anwenderprogramm in der S7-CPU kann zur Konsistenzsicherung während eines Schreibzugriffes LOCKED setzen bzw. zurücksetzen.</p> <p>Empfehlung zur Vorgehensweise im Anwenderprogramm:</p> <ol style="list-style-type: none"> <li>1. LOCKED-Bit prüfen;<br/>wenn =0</li> <li>2. WRITEACCESS-Bit=0 setzen</li> <li>3. LOCKED-Bit prüfen;<br/>wenn =0</li> <li>4. LOCKED-Bit=1 setzen</li> <li>5. Daten schreiben</li> <li>6. LOCKED-Bit=0 setzen</li> </ol> |

| Parameter      | Typ           | Wert / Bedeutung   | Versorgung   |
|----------------|---------------|--|--|
| NEW            | BOOL          | Das NEW-Bit informiert, ob Daten seit dem letzten Lesevorgang verändert wurden. <ul style="list-style-type: none"> <li>0: Inhalt des File-DB ist unverändert seit letztem Schreibvorgang. Das Anwenderprogramm der S7-CPU hat die letzte Änderung registriert.</li> <li>1: Das Anwenderprogramm der S7-CPU hat den letzten Schreibvorgang noch nicht registriert.</li> </ul> | FTP-Kommando store setzt nach der Bearbeitung NEW=1<br>Das Anwenderprogramm in der S7-CPU muß nach dem Lesen der Daten NEW=0 setzen, um ein erneutes Kommando retr zu ermöglichen.   |
| WRITE_ACCESS   | BOOL          | 0:<br>Das Anwenderprogramm (FTP-Client Bausteine) hat Schreibrecht für die File-DBs in der S7-CPU.<br>1:<br>Das Anwenderprogramm (FTP-Client Bausteine) hat kein Schreibrecht für die File-DBs in der S7-CPU.  | Das Bit wird bei der DB-Projektierung auf einen Initialisierungswert gesetzt.<br>Empfehlung:<br>Das Bit sollte nach Möglichkeit unverändert bleiben! In besonderen Fällen ist eine Anpassung im laufenden Betrieb möglich.   |
| ACT_LENGTH     | DINT          | Aktuelle Länge des Nutzdatenbereiches.<br>Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.  | Die aktuelle Länge wird nach einem Schreibvorgang aktualisiert.  |
| MAX_LENGTH     | DINT          | Maximale Länge des Nutzdatenbereiches (Länge des gesamten DB abzüglich 20 Byte Header).  | Die maximale Länge sollte bei der DB-Projektierung festgelegt werden.<br>Der Wert kann auch im laufenden Betrieb vom Anwenderprogramm geändert werden.   |
| FTP_REPLY_CODE | INT           | Vorzeichenlose Zahl (16 Bit), die den letzten Reply-Code von FTP als Binärwert enthält.<br>Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.   | Wird vom FTP-Client bei der FTP-Kommandobearbeitung aktualisiert.  |
| DATE_TIME      | DATE_AND_TIME | Datum und Zeit der letzten Änderung des Files.<br>Der Inhalt dieses Feldes ist nur dann gültig, wenn EXIST = 1.  | Das aktuelle Datum wird nach einem Schreibvorgang aktualisiert.<br>Wird die Funktion "Uhrzeitweiterleitung" genutzt, entspricht der Eintrag der weitergeleiteten Zeit.<br>Wird die Funktion "Uhrzeitweiterleitung" nicht genutzt, wird eine relative Zeit eingetragen. Bezug ist der Anlaufzeitpunkt des CP (Initialisierungswert ist 1.1.1994 0.0 Uhr). |

## Siehe auch

Datenbaustein FILE\_DB\_HEADER als Vorlage - FTP-Client-Betrieb (Seite 2798)

## Datenbaustein FILE\_DB\_HEADER als Vorlage - FTP-Client-Betrieb

### Bedeutung

Für das Anlegen eines File-DB-Header ist der Datentyp FILE\_DB\_HEADER vordefiniert.

### Funktionsweise

Für die Übertragung von Daten mittels FTP legen Sie in der CPU Ihrer S7-Station Datenbausteine (File-DBs) an. Diese Datenbausteine müssen einer bestimmten Struktur genügen, damit sie von den FTP-Diensten als übertragbare Dateien hantiert werden können. Sie bestehen aus folgenden Abschnitten:

- Abschnitt 1: File-DB-Header (besitzt feste Struktur mit einer Länge von 20 Byte)
- Abschnitt 2: Nutzdaten (besitzt variable Länge und Struktur)

### Gehen Sie wie folgt vor:

1. Legen Sie bei der PLC-CPU, in der Sie das Anwenderprogramm mit den FTP-Anweisungen erstellen, einen Datenbaustein vom Typ "Global-DB" an.
2. Selektieren Sie die Zeile, die Sie als Startzeile für den File-DB verwenden werden.
3. Wählen Sie über die Klappliste in der Spalte "Datentyp" ein Strukturelement vom Typ FILE\_DB\_HEADER aus.  
Ergebnis: Es wird eine Datenstruktur mit der für den File-DB benötigte Header-Struktur angelegt.
4. Wählen Sie die Eigenschaften des neu angelegte Datenbausteins und deaktivieren Sie das Attribut "Optimierter Bausteinzugriff".

---

### Hinweis

#### Funktion "Neuen Baustein hinzufügen" - Typauswahl

Beim Anlegen neuer Datenbausteine wird unter dem Eintrag "Typ" in der Klappliste auch der Bausteintyp "FILE\_DB\_HEADER" angeboten. Verwenden Sie diese Auswahlmöglichkeit nicht! Der so erzeugte Datenbaustein enthält lediglich die Header-Struktur und kann nicht um den benötigten Bereich zur Ablage von Nutzdaten erweitert werden.

---

## Datenbaustein FILE\_DB\_HEADER - Beispiel und Vorlage für den File-DB-Header

In der Deklarationsansicht erkennen Sie folgende Struktur:

| Adresse | Name           | Typ           | Anfangswert         | Kommentar   |
|---------|----------------|---------------|---------------------|---|
| 0.0     |                | STRUCT        |                     |   |
| +0.0    | bit08          | BOOL          | FALSE               | reserviert  |
| +0.1    | bit09          | BOOL          | FALSE               | reserviert  |
| +0.2    | bit10          | BOOL          | FALSE               | reserviert  |
| +0.3    | bit11          | BOOL          | FALSE               | reserviert  |
| +0.4    | bit12          | BOOL          | FALSE               | reserviert  |
| +0.5    | bit13          | BOOL          | FALSE               | reserviert  |
| +0.6    | bit14          | BOOL          | FALSE               | reserviert  |
| +0.7    | bit15          | BOOL          | FALSE               | reserviert  |
| +1.0    | EXIST          | BOOL          | FALSE               | wenn TRUE:<br>File-DB enthält gültige Daten   |
| +1.1    | LOCKED         | BOOL          | FALSE               | wenn TRUE:<br>Der File-DB ist blockiert<br>aufgrund einer inhaltlichen<br>Änderung.     |
| +1.2    | NEW            | BOOL          | FALSE               | wenn TRUE:<br>Der Inhalt des File-DB ist neu<br>und darf nicht überschrieben<br>werden. |
| +1.3    | WRITE_ACCESS   | BOOL          | FALSE               | wenn TRUE:<br>Der FTP-Server hat Schreib-<br>Zugriff.                                   |
| +1.4    | bit04          | BOOL          | FALSE               | reserviert  |
| +1.5    | bit05          | BOOL          | FALSE               | reserviert  |
| +1.6    | bit06          | BOOL          | FALSE               | reserviert  |
| +1.7    | bit07          | BOOL          | FALSE               | reserviert  |
| +2.0    | ACT_LENGTH     | DINT          | L#0                 | Aktuelle Länge des Inhalts in<br>Byte (nicht eingeschlossen 20<br>Byte für den Header)  |
| +6.0    | MAX_LENGTH     | DINT          | L#0                 | Maximale Länge des Inhalts in<br>Byte (nicht eingeschlossen 20<br>Byte für den Header)  |
| +10.0   | FTP_REPLY_CODE | INT           | 0                   | Letzte Antwortinformation vom<br>Remote-FTP-Server.                                     |
| +12.0   | DATE_TIME      | DATE_AND_TIME | DT#00-1-1-0:0:0.000 | Datum und Zeitangabe für die<br>letzte Änderung am Inhalt des<br>File-DB.               |
| =20.0   |                | END_STRUCT    |                     |   |

### Unterschiede bei den Betriebsarten

File-DB-Header für FTP-Client-Betrieb

Der hier beschriebene File-DB-Header ist in den Betriebsarten für FTP-Client-Betrieb und FTP-Server-Betrieb weitgehend identisch. Die Unterschiede beziehen sich auf die Parameter

- WRITE\_ACCESS
- FTP\_REPLY\_CODE

## S7-1200-CM/CP

### Telecontrol

### Telecontrol-Anweisungen

### TC\_CON: Verbindung über das GSM-Netz aufbauen

#### Bedeutung

Die Anweisung TC\_CON ermöglicht einer S7-1200 mit CP 1242-7 den Aufbau einer Verbindungen der folgenden Typen:

- ISO-ON-TCP  
Verbindungspartner ist ein CP 1242-7.  
ISO-ON-TCP-Verbindungen werden nur in der Betriebsart "GPRS direkt" verwendet.
- UDP  
Der Verbindungspartner ist beliebig.
- SMS  
Verbindungspartner ist ein SMS-Client.
- Telecontrol-Verbindung  
Verbindungspartner ist entweder ein Telecontrol-Server oder eine andere Station, die über den Telecontrol-Server erreichbar ist.

Ein TC\_CON baut genau eine Verbindung auf. Abhängig von der Betriebsart des CP 1242-7 und dem verwendeten Protokoll werden pro CP maximal 3 bis 5 parallele Verbindungen mit eindeutigen IDs (siehe unten) unterstützt. Die maximal mögliche Anzahl paralleler Verbindungen entnehmen Sie den Leistungsdaten des CP.

Der Parameter CONNECT verwendet zur Verbindungsbeschreibung einen Datenbaustein (DB) mit der Struktur eines Systemdatentyps (SDT).

Der gewünschte Verbindungstyp wird über einen jeweils verbindungsspezifischen SDT "TCON\_..." (siehe unten) definiert. Für jeden der oben genannten Verbindungstypen muss einer der folgenden SDTs parametrieren werden:

- TCON\_IP\_RFC für ISO-ON-TCP-Verbindungen
- TCON\_IP\_V4 für UDP-Verbindungen
- TCON\_PHONE für SMS-Verbindungen
- TCON\_WDC für Telecontrol-Verbindungen

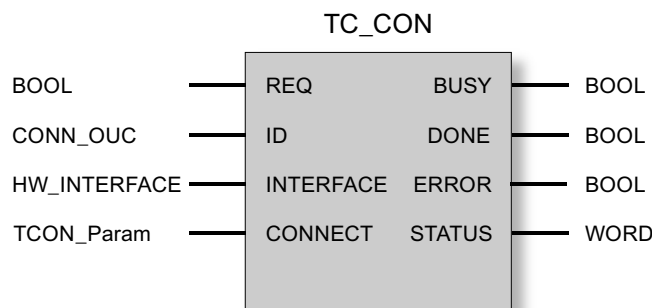
Im Parameter "ActiveEstablished" dieser SDTs wird unter anderem auch festgelegt, ob es sich um einen aktiven oder passiven Verbindungsaufbau handelt.

Zur Parametrierung dieser SDTs siehe TCON\_...: SDTs für den Telecontrol-Verbindungsaufbau (Seite 2816).

Der Parameter ID referenziert die GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Diese muss aus STEP 7 übernommen werden.

### Aufrufschnittstelle in FUP-Darstellung



### Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC\_CON.

| Parameter | Deklaration | Datentyp         | Wertebereich          | Beschreibung  |
|-----------|-------------|------------------|-----------------------|---|
| REQ       | INPUT       | BOOL             | 0, 1                  | Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert.<br>Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht. |
| ID        | INPUT       | CONN_OUC (WORD)  | 1...07FF <sub>n</sub> | Referenz auf die jeweilige Verbindung. Die ID wird vergeben.<br>Der Wert von ID wird auch vom jeweiligen Systemdatentyp (SDT) des Parameters CONNECT benötigt.  |
| INTERFACE | INPUT       | HW_INTERFA<br>CE |                       | Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")   |

| Parameter | Deklaration | Datentyp   | Wertebereich   | Beschreibung   |
|-----------|-------------|------------|--|--|
| CONNECT   | INOUT       | TCON_Param | Siehe auch "TCON_...: SDTs für den Telecontrol-Verbindungsaufbau"  | Referenz auf einen Datenbaustein für den Verbindungsaufbau.<br>Die SDTs vom Typ TCON_IP_RFC, TCON_IP_V4, TCON_PHONE oder TCON_WDC geben die Struktur des passenden Datenbausteins für die jeweilige Verbindung vor.<br>Beachten Sie bei den SDTs den Parameter "ActiveEstablished" (aktiver / passiver Verbindungsaufbau). |
| ENO       | OUTPUT      | BOOL       | 0: Fehler<br>1: Fehlerfrei   | Freigabeausgang<br>Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.   |
| BUSY      | OUTPUT      | BOOL       | 0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen<br>1: Bearbeitung der Anweisung läuft | Anzeige des Bearbeitungs-Status der Anweisung  |
| DONE      | OUTPUT      | BOOL       | 0: -<br>1: Bearbeitung der Anweisung erfolgreich beendet   | Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.<br>Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.  |
| ERROR     | OUTPUT      | BOOL       | 0: -<br>1: Fehler  | Fehleranzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.  |
| STATUS    | OUTPUT      | WORD       |  | Statusanzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.   |

### Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

| BUSY | DONE | ERROR | Bedeutung                   |
|------|------|-------|-----------------------------|
| 0    | 0    | 0     | Kein Auftrag in Bearbeitung |

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

Beim Aufruf bleibt die Anweisung für einige Sekunden im Zustand BUSY = 1. In folgenden Fällen kann der Zustand BUSY = 1 länger andauern:

- Bei aktiven ISO-ON-TCP-Verbindungen, wenn der Partner nicht erreichbar ist.
- Bei passiven Verbindungen, wenn kein Telegramm empfangen wird.



## Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

| DONE | ERROR | STATUS            | Bedeutung   |
|------|-------|-------------------|---|
| 1    | 0     | 0000 <sub>H</sub> | Auftrag fehlerfrei ausgeführt   |
| 0    | 0     | 7000 <sub>H</sub> | Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)  |
| 0    | 0     | 7001 <sub>H</sub> | Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)  |
| 0    | 0     | 7002 <sub>H</sub> | Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)  |
| 0    | 1     | 8086 <sub>H</sub> | Nicht erlaubter Wert für ID   |
| 0    | 1     | 8087 <sub>H</sub> | Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich   |
| 0    | 1     | 80E3 <sub>H</sub> | Die ID wird bereits für eine andere Verbindung verwendet.   |
| 0    | 1     | 80E6 <sub>H</sub> | Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)   |
| 0    | 1     | 80E8 <sub>H</sub> | Entfernter Partner nicht erreichbar. Prüfen Sie die Verbindungsparameter.<br>In der Betriebsart "GPRS direkt" wird die Meldung ausgegeben, wenn der Partner erreichbar ist, aber keine Verbindungsanfrage entgegennimmt.  |
| 0    | 1     | 80EB <sub>H</sub> | Anfrage vorübergehend zurückgewiesen (TC_CON wurde bereits mit der selben Zieladresse aufgerufen.)  |
| 0    | 1     | 80EC <sub>H</sub> | Öffnen des Listener Port konnte nicht durchgeführt werden:<br>Prüfen Sie die Verbindungsparameter.  |
| 0    | 1     | 80F2 <sub>H</sub> | Der CP befindet sich in der falschen Betriebsart: <ul style="list-style-type: none"> <li>• Telecontrol-Verbindungen sind nur in der Betriebsart "Telecontrol" erlaubt.</li> <li>• ISO-ON-TCP-Verbindungen sind nur in der Betriebsart "GPRS direkt" erlaubt.</li> </ul>   |
| 0    | 1     | 80F3 <sub>H</sub> | Kein freier Verbindungsendpunkt zum Senden von Daten: <ul style="list-style-type: none"> <li>• Nutzen Sie weniger Verbindungen oder</li> <li>• Nutzen Sie weniger passive Verbindungen oder</li> <li>• Schalten Sie NTP aus.</li> </ul> Beachten Sie die maximale Anzahl paralleler Verbindungen des CP 1242-7. |
| 0    | 1     | 80F4 <sub>H</sub> | Verbindungsendpunkt kann nicht erzeugt werden:<br>Wiederholen Sie den Aufruf. Prüfen Sie ggf. die Verbindungsparameter.   |
| 0    | 1     | 80F6 <sub>H</sub> | Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert)<br>Prüfen Sie die Projektierung des SDT "TC_CON...".  |

## TC\_DISCON: Verbindung über das GSM-Netz abbauen

### Bedeutung

Die Anweisung TC\_DISCON bewirkt bei einer S7-1200 mit CP 1242-7 den Abbau einer ISO-ON-TCP-, UDP, SMS- oder Telecontrol-Verbindung, die mit der Anweisung TC\_CON aufgebaut wurde.

Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC\_CON.

TC\_DISCON baut die Verbindung zum Telecontrol-Server nur logisch ab. Wenn die Verbindung zum Telecontrol-Server physikalisch abgebaut werden soll, dann projektieren Sie

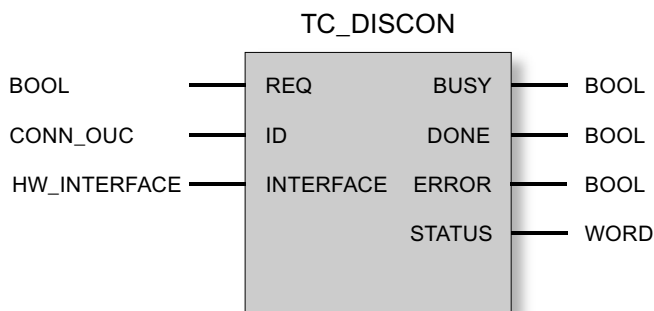
die Verbindung in STEP 7 unter der Parametergruppe "Telecontrol-Server" als "Temporäre Verbindung".

Auf TCP/IP-Ebene bleibt die Verbindung erhalten. Temporäre Stationen bauen die Verbindung nach dem Senden der Daten automatisch ab.

Der Parameter ID referenziert die GPRS-Verbindung. Die ID muss innerhalb der CPU eindeutig sein und gleich derjenigen ID sein, die bei TC\_CON verwendet wird.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC\_CON für INTERFACE verwendet wird.

### Aufrufschnittstelle in FUP-Darstellung



### Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC\_DISCON

| Parameter | Deklaration | Datentyp        | Wertebereich   | Beschreibung  |
|-----------|-------------|-----------------|--|---|
| REQ       | INPUT       | BOOL            | 0, 1   | Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert.<br>Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht. |
| ID        | INPUT       | CONN_OUC (WORD) | 1...07FF <sub>h</sub>  | Referenz auf die jeweilige Verbindung   |
| INTERFACE | INPUT       | HW_INTERFACE    |  | Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")   |
| ENO       | OUTPUT      | BOOL            | 0: Fehler<br>1: Fehlerfrei   | Freigabeausgang<br>Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.  |
| BUSY      | OUTPUT      | BOOL            | 0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen<br>1: Bearbeitung der Anweisung läuft | Anzeige des Bearbeitungs-Status der Anweisung   |

| Parameter | Deklaration | Datentyp | Wertebereich   | Beschreibung  |
|-----------|-------------|----------|--|---|
| DONE      | OUTPUT      | BOOL     | 0: -<br>1: Bearbeitung der Anweisung erfolgreich beendet | Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.<br>Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung. |
| ERROR     | OUTPUT      | BOOL     | 0: -<br>1: Fehler  | Fehleranzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.   |
| STATUS    | OUTPUT      | WORD     |  | Statusanzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.  |

### Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

| BUSY | DONE | ERROR | Bedeutung                                  |
|------|------|-------|--|
| 0    | 0    | 0     | Die Anweisung wurde noch nicht aufgerufen. |

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

#### Hinweis

Beim Aufruf bleibt die Anweisung für einige Sekunden im Zustand BUSY = 1.

### Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

| DONE | ERROR | STATUS            | Bedeutung  |
|------|-------|-------------------|--|
| 1    | 0     | 0000 <sub>H</sub> | Auftrag fehlerfrei ausgeführt  |
| 0    | 0     | 7000 <sub>H</sub> | Keine Auftragsbearbeitung aktiv (Erstaufwurf der Anweisung)                    |
| 0    | 0     | 7001 <sub>H</sub> | Auftragsbearbeitung gestartet (Erstaufwurf der Anweisung)                      |
| 0    | 0     | 7002 <sub>H</sub> | Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1) |
| 0    | 1     | 8086 <sub>H</sub> | Nicht erlaubter Wert für ID  |
| 0    | 1     | 80E4 <sub>H</sub> | Unbekannte ID: Keine Verbindung mit dieser ID ist durch TC_CON aufgebaut.      |
| 0    | 1     | 80E6 <sub>H</sub> | Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)            |

| DONE | ERROR | STATUS            | Bedeutung   |
|------|-------|-------------------|---|
| 0    | 1     | 80F5 <sub>H</sub> | Ungültiger Verbindungsendpunkt: <ul style="list-style-type: none"> <li>• Verbindungsaufbau durch TC_CON fehlgeschlagen oder</li> <li>• Verbindung vom entfernten Partner abgebaut.</li> </ul> |
| 0    | 1     | 80F6 <sub>H</sub> | Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert)<br>Prüfen Sie die Projektierung des SDT "TC_CON...".                        |

## TC\_SEND: Daten über das GSM-Netz senden

### Bedeutung

Die Anweisung TC\_SEND ermöglicht das Senden von Daten über programmierte Verbindungen der folgenden Typen:

- ISO-ON-TCP-Verbindungen
- UDP-Verbindungen
- SMS-Verbindungen  
Das Versenden von SMS wird nur unterstützt, wenn dies in der STEP 7-Projektierung des CP eingerichtet wurde.
- Telecontrol-Verbindungen

---

#### Hinweis

##### SMS an mehrere Empfänger senden

Wenn Sie eine identische SMS an mehrere Empfänger senden wollen, dann müssen Sie für jeden Empfänger eine Verbindung aufbauen.

---

Weitere Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC\_CON.

Der Parameter ID referenziert die GPRS-Verbindung. Der Wert von ID muss dem bei TC\_CON verwendeten Wert für ID entsprechen.

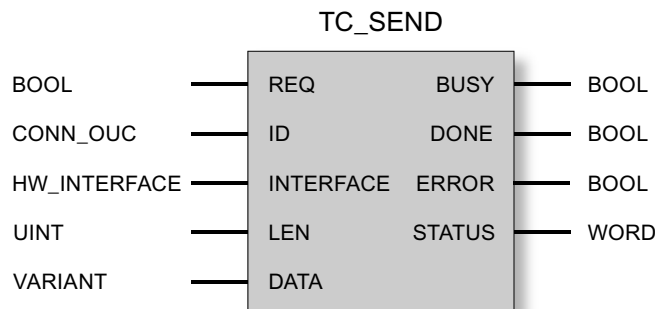
Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC\_CON für INTERFACE verwendet wird.

Die Anzahl zu sendender Daten wird mit dem Parameter LEN angegeben.

Die Größe des in DATA angegebenen Datenbereichs muss mindestens so groß sein wie die unter LEN projektierte Byte-Anzahl. Erlaubte Datentypen in dem unter DATA angegebenen Datenbereich sind alle außer BOOL und ARRAY of BOOL.

Die Zieladresse (Verbindungspartner) für die zu sendenden Daten wird in der Anweisung TC\_CON projektiert.

### Aufrufschnittstelle in FUP-Darstellung



### Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC\_SEND.

| Parameter | Deklaration | Datentyp           | Wertebereich   | Beschreibung  |
|-----------|-------------|--------------------|--|---|
| REQ       | INPUT       | BOOL               | 0, 1   | Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert.<br>Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht. |
| ID        | INPUT       | CONN_OUC<br>(WORD) | 1...07FF <sub>n</sub>  | Referenz auf die jeweilige Verbindung   |
| INTERFACE | INPUT       | HW_INTERFA<br>CE   |  | Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")   |
| LEN       | INPUT       | UINT               | 1...2048   | Anzahl der Bytes der zu sendenden Daten, maximal 2048. Der Wert sollte der Größe des Bereichs von DATA entsprechen.   |
| DATA      | INOUT       | VARIANT            |  | Adressverweis auf den zu übertragenden Datenbereich der CPU *   |
| ENO       | OUTPUT      | BOOL               | 0: Fehler<br>1: Fehlerfrei   | Freigabeausgang<br>Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.  |
| BUSY      | OUTPUT      | BOOL               | 0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen<br>1: Bearbeitung der Anweisung läuft | Anzeige des Bearbeitungs-Status der Anweisung   |
| DONE      | OUTPUT      | BOOL               | 0: -<br>1: Bearbeitung der Anweisung erfolgreich beendet   | Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde. **<br>Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung.                  |

| Parameter | Deklaration | Datentyp | Wertebereich      | Beschreibung  |
|-----------|-------------|----------|-------------------|---|
| ERROR     | OUTPUT      | BOOL     | 0: -<br>1: Fehler | Fehleranzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung. |
| STATUS    | OUTPUT      | WORD     |                   | Statusanzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.  |

\* Zu Besonderheiten des Parameters DATA für SMS-Texte siehe nachfolgender Abschnitt.

\*\* Nach dem Senden eines Telegramms setzt TC\_SEND DONE = 1. Beachten Sie folgendes Verhalten:

Der Ausfall einer ISO-on-TCP-Verbindungen wird beim Sender erst nach 1 bis 2 Minuten festgestellt. Die übertragenen Daten können verloren sein, obwohl TC\_SEND beim Sender DONE = 1 gesetzt hat.

Wenn eine ISO-on-TCP-Verbindung nach dem Empfang eines Telegramms abgebrochen wird, bevor TC\_RECV gestartet wurde, dann können die übertragenen Daten verloren sein, auch wenn TC\_SEND beim Sender DONE = 1 gesetzt hat.

### Projektierung von SMS-Texten mit dem Parameter DATA

Die Anweisung sendet diejenigen Daten als SMS-Text, auf welche der Zeiger vom Typ VARIANT des Parameters DATA referenziert.

Wenn von DATA auf einen Operanden des Datentyps STRING für SMS-Texte referenziert wird, dann werden die ersten zwei Bytes mit Längenangaben des Strings mit übertragen.

Eine Möglichkeit für die korrekte Textdarstellung von zu sendenden SMS ist die Umwandlung des Text-Strings in ein Array of BYTE oder Array of CHAR mithilfe der Konvertierungsfunktion Strg\_TO\_Chars. Strg\_TO\_Chars wird am Parameter EN mit dem Ausgangsparameter ENO von TC\_SEND verknüpft.

Für SMS-Texte unterstützt der CP nicht alle Sonderzeichen, bspw. keine Umlaute. Es gilt die Spezifikation GSM 03.38. Zusätzliche Einschränkungen können durch den GSM-Netzbetreiber bestehen.

### Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

| BUSY | DONE | ERROR | Bedeutung                   |
|------|------|-------|-----------------------------|
| 0    | 0    | 0     | Kein Auftrag in Bearbeitung |

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

## Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

| DONE | ERROR | STATUS *          | Bedeutung  |
|------|-------|-------------------|--|
| 1    | 0     | 0000 <sub>H</sub> | Auftrag fehlerfrei ausgeführt  |
| 0    | 0     | 7000 <sub>H</sub> | Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)   |
| 0    | 0     | 7001 <sub>H</sub> | Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)   |
| 0    | 0     | 7002 <sub>H</sub> | Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)   |
| 0    | 1     | 8086 <sub>H</sub> | Nicht erlaubter Wert für ID  |
| 0    | 1     | 80E0 <sub>H</sub> | <ul style="list-style-type: none"> <li>Die Längenangabe unter LEN ist größer als der Bereich der unter DATA zu übertragenden Daten.</li> <li>oder</li> <li>Interner Fehler<br/>Wenn Sie Telegramme direkt an den Telecontrol-Server schicken (Betriebsart "Telecontrol"), dann stellen Sie sicher, dass die Sendezykluszeit <math>\geq 1</math> Sekunde ist.</li> </ul>  |
| 0    | 1     | 80E1 <sub>H</sub> | Zeitüberschreitung: <ul style="list-style-type: none"> <li>Erhöhen Sie in der Projektierung des CP 1242-7 den Wert der "Verbindungsüberwachungszeit" oder</li> <li>Überprüfen Sie den Verbindungspartner.</li> </ul>   |
| 0    | 1     | 80E4 <sub>H</sub> | Unbekannte ID:<br>Rufen Sie zuerst TC_CON auf.   |
| 0    | 1     | 80E6 <sub>H</sub> | Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)  |
| 0    | 1     | 80E7 <sub>H</sub> | Zu sendende Daten nicht vollständig übertragen:<br>Wiederholen Sie den Auftrag.  |
| 0    | 1     | 80E8 <sub>H</sub> | Entfernter Partner nicht erreichbar. Prüfen Sie die Verbindungsparameter.<br>In der Betriebsart "GPRS direkt" wird die Meldung ausgegeben, wenn der Partner erreichbar ist, aber keine Verbindungsanfrage entgegennimmt.   |
| 0    | 1     | 80E9 <sub>H</sub> | Verbindungsabbau durch entfernten Partner:<br>Überprüfen Sie den Verbindungspartner. Bauen Sie ggf. die Verbindung mit TC_DISCON ab und mit TC_CON wieder neu auf.   |
| 0    | 1     | 80EA <sub>H</sub> | Fehlermeldung vom entfernten Partner: <ul style="list-style-type: none"> <li>Überprüfen Sie den Verbindungspartner. Aktivieren Sie beim Kommunikationspartner die Anweisung "TC_RECV".</li> <li>Bauen Sie ggf. die Verbindung mit TC_DISCON ab und mit TC_CON wieder neu auf.</li> </ul>   |
| 0    | 1     | 80EF <sub>H</sub> | Senden der SMS konnte nicht durchgeführt werden: <ul style="list-style-type: none"> <li>Überprüfen Sie, ob die Zieladresse (Rufnummer des Zielteilnehmers) existiert.</li> <li>Überprüfen Sie, ob die gesteckte SIM-Karte das Senden von SMS unterstützt.</li> <li>Stellen Sie sicher, dass beim Anlegen des Datenbausteins TCON_PHONE die Option "Standard" für den Bausteinzugriff gewählt wurde.</li> </ul> |
| 0    | 1     | 80F1 <sub>H</sub> | Das Senden von SMS ist in der STEP 7-Projektierung des CP nicht aktiviert:<br>Aktivieren Sie in der Projektierung des CP die Option "SMS ermöglichen".   |
| 0    | 1     | 80F4 <sub>H</sub> | Verbindungsendpunkt kann nicht erzeugt werden:<br>Überprüfen Sie den Verbindungspartner.   |

| DONE | ERROR | STATUS *          | Bedeutung   |
|------|-------|-------------------|---|
| 0    | 1     | 80F5 <sub>H</sub> | Ungültiger Verbindungsendpunkt: <ul style="list-style-type: none"> <li>• Verbindungsaufbau durch TC_CON fehlgeschlagen. oder</li> <li>• Verbindung vom entfernten Partner abgebaut: Rufen Sie TC_DISCON auf.</li> </ul> |
| 0    | 1     | 80F6 <sub>H</sub> | Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert):<br>Prüfen Sie die Projektierung des SDT "TC_CON...".   |

\* Weitere Status, die hier nicht aufgeführt sind, finden Sie bei den Statusanzeigen der Anweisungen "RDREC" bzw. "WRREC" unter den mittleren beiden Status-Bytes (STATUS[2], STATUS[3]).

## TC\_RECV: Daten über das GSM-Netz empfangen

### Bedeutung

Die Anweisung TC\_RECV ermöglicht das Empfangen von Daten über programmierte Verbindungen der folgenden Typen:

- ISO-ON-TCP-Verbindungen
- SMS-Verbindungen  
Zum Empfangen von SMS muss die Rufnummer des Absenders in der STEP 7-Projektierung des empfangenden CP projektiert sein (Autorisierte Rufnummern). Der Absender muss die CLIP-Funktion unterstützen.  
Die Rufnummer des Verbindungspartners muss im SDT "TCON\_PHONE" eingetragen sein. Weck-SMS werden ausgefiltert.
- Telecontrol-Verbindungen

---

#### Hinweis

##### SMS von verschiedenen Sendern empfangen

Wenn Sie SMS von verschiedenen Sendern empfangen wollen, dann stehen Ihnen hierfür zwei Alternativen zur Wahl:

- Sie müssen mehrere Verbindungen (TC\_CON, TC\_RECV, TC\_DISCON) projektieren. oder
  - Sie dürfen bei nur einer einzigen projektierten Verbindung in dem erforderlichen Datenbaustein "TCON\_PHONE" im Parameter "PhoneNumber" keine Rufnummer eingeben. Dies wird beim Empfangen von Nachrichten als Platzhalter für alle autorisierten Verbindungspartner interpretiert.
- 

Weitere Details zu den Verbindungstypen finden Sie bei der Beschreibung der Anweisung TC\_CON.

Der Parameter ID referenziert die GPRS-Verbindung. Der Wert von ID muss dem bei TC\_CON verwendeten Wert für ID entsprechen.

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP. Der Wert muss gleich demjenigen sein, der bei TC\_CON für INTERFACE verwendet wird.

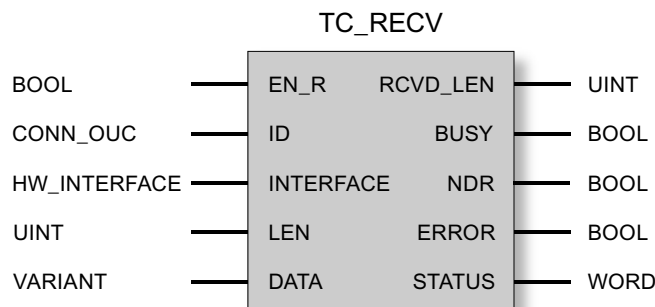
Die maximale Anzahl der Empfangsdaten wird mit dem Parameter LEN angegeben.



Die Größe des in DATA angegebenen Datenbereichs muss mindestens so groß sein wie die unter LEN projizierte Byte-Anzahl. Erlaubte Datentypen in dem unter DATA angegebenen Datenbereich sind alle außer BOOL und ARRAY of BOOL. Die empfangenen Daten werden so interpretiert, als hätte der sendende Partner die gleichen Datentypen verwendet.

Der für die Verbindungsbeschreibung des TC\_RECV verwendete DB (Systemdatentyp) muss sich von einem für TC\_SEND verwendeten DB unterscheiden.

### Aufrufschnittstelle in FUP-Darstellung



### Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC\_RECV

| Parameter | Deklaration | Datentyp         | Wertebereich   | Beschreibung  |
|-----------|-------------|------------------|--|---|
| EN_R      | INPUT       | BOOL             | 0: Datenempfang gesperrt<br>1: Datenempfang ermöglicht | Ermöglichen / Sperren des Empfangs von Daten.<br>Nach dem Setzen von 1 auf 0 empfängt der Programmbaustein noch einmal Daten.     |
| ID        | INPUT       | CONN_OUC (WORD)  | 1...07FF <sub>n</sub>                                  | Referenz auf die jeweilige Verbindung   |
| INTERFACE | INPUT       | HW_INTERFA<br>CE |  | Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung") |
| LEN       | INPUT       | UINT             | 1...2048   | (Mindest-) Anzahl der Bytes der zu empfangenden Daten, maximal 2048   |
| DATA      | INOUT       | VARIANT          |  | Adressverweis auf Empfangs-Datenbereich der CPU *   |
| ENO       | OUTPUT      | BOOL             | 0: Fehler<br>1: Fehlerfrei                             | Freigabeausgang<br>Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.  |
| RCVD_LEN  | OUTPUT      | UINT             |  | Byte-Anzahl der empfangenen Daten   |

| Parameter | Deklaration | Datentyp | Wertebereich   | Beschreibung  |
|-----------|-------------|----------|--|---|
| BUSY      | OUTPUT      | BOOL     | 0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen<br>1: Bearbeitung der Anweisung läuft | Anzeige des Bearbeitungs-Status der Anweisung   |
| DONE      | OUTPUT      | BOOL     | 0: -<br>1: Bearbeitung der Anweisung erfolgreich beendet   | Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.<br>Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung. |
| ERROR     | OUTPUT      | BOOL     | 0: -<br>1: Fehler  | Fehleranzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.   |
| STATUS    | OUTPUT      | WORD     |  | Statusanzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.  |

\* Zu Besonderheiten des Parameters DATA für SMS-Texte siehe nachfolgender Abschnitt.

### Projektierung von SMS-Texten mit dem Parameter DATA

Die Anweisung referenziert den empfangenen SMS-Text mit dem Zeiger vom Typ VARIANT des Parameters DATA auf den Datenbereich der CPU.

Wenn von DATA auf einen Operanden des Datentyps STRING für den SMS-Text referenziert wird, dann werden die ersten zwei Bytes des SMS-Texts als Längenangaben des Datentyps STRING und nicht als SMS-Text interpretiert.

Eine Möglichkeit für die korrekte Textdarstellung von zu empfangenden SMS ist die Umwandlung eines Array of BYTE oder Array of CHAR in einen Text-String mithilfe der Konvertierungsfunktion Chars\_TO\_Strg. Chars\_TO\_Strg wird am Parameter EN mit dem Ausgangsparameter ENO von TC\_RECV verknüpft.

Für SMS-Texte unterstützt der CP nicht alle Sonderzeichen, bspw. keine Umlaute. Es gilt die Spezifikation GSM 03.38. Zusätzliche Einschränkungen können durch den GSM-Netzbetreiber bestehen.

### Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

| BUSY | DONE | ERROR | Bedeutung                   |
|------|------|-------|-----------------------------|
| 0    | 0    | 0     | Kein Auftrag in Bearbeitung |

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

## Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

| DONE | ERROR | STATUS *          | Bedeutung   |
|------|-------|-------------------|---|
| 1    | 0     | 0000 <sub>H</sub> | Auftrag fehlerfrei ausgeführt   |
| 0    | 0     | 7000 <sub>H</sub> | Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)  |
| 0    | 0     | 7001 <sub>H</sub> | Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)  |
| 0    | 0     | 7002 <sub>H</sub> | Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)  |
| 0    | 1     | 80A3 <sub>H</sub> | <ul style="list-style-type: none"> <li>• Es wird versucht, eine vorhandene Verbindung erneut aufzubauen.</li> <li>• Es wird versucht, eine nicht vorhandene Verbindung zu beenden.</li> </ul>                               |
| 0    | 1     | 80E0 <sub>H</sub> | <ul style="list-style-type: none"> <li>• Die Größe der Daten, die für den unter DATA angegebenen Bereich empfangen wird, ist größer als die Längenangabe unter LEN.<br/>oder</li> <li>• Interner Fehler</li> </ul>          |
| 0    | 1     | 8086 <sub>H</sub> | Nicht erlaubter Wert für ID   |
| 0    | 1     | 80E4 <sub>H</sub> | Unbekannte ID:<br>Rufen Sie zuerst TC_CON auf.  |
| 0    | 1     | 80E6 <sub>H</sub> | Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)   |
| 0    | 1     | 80F5 <sub>H</sub> | Ungültiger Verbindungsendpunkt: <ul style="list-style-type: none"> <li>• Verbindungsaufbau durch TC_CON fehlgeschlagen.<br/>oder</li> <li>• Verbindung vom entfernten Partner abgebaut: Rufen Sie TC_DISCON auf.</li> </ul> |
| 0    | 1     | 80F6 <sub>H</sub> | Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert)<br>Prüfen Sie die Projektierung des SDT "TC_CON...".  |

\* Weitere Status, die hier nicht aufgeführt sind, finden Sie bei den Statusanzeigen der Anweisungen "RDREC" bzw. "WRREC" unter den mittleren beiden Status-Bytes (STATUS[2], STATUS[3]).

## TC\_CONFIG: Projektierungsdaten auf CP übertragen

### Bedeutung

Mit der Anweisung TC\_CONFIG können die in STEP 7 projektierten Parameter eines CP 1242-7 geändert werden. Die projektierten Werte werden nicht remanent überschrieben. Die überschriebenen Werte bleiben gültig bis zum erneuten Aufruf von TC\_CONFIG oder bis zum nächsten Anlauf der Station (Kaltstart durch Spannung AUS → EIN).

Wenn die STEP 7-Projektierungsdaten des CP dauerhaft geändert werden sollen, dann muss die Anweisung nach jedem Anlauf der Station (Kaltstart) neu aufgerufen werden oder ein geändertes Projekt muss in die Station geladen werden.

Der Parameter CONFIG zeigt auf den Speicherbereich mit den Projektierungsdaten. Die Projektierungsdaten werden in einem Datenbaustein (DB) gespeichert. Die Struktur des DB wird durch den Systemdatentyp (SDT) IF\_CONF vorgegeben.

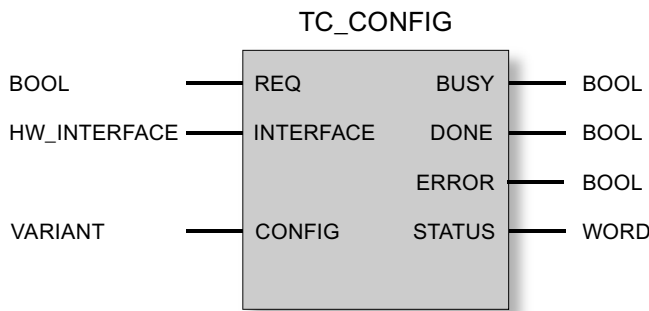
Diejenigen Projektierungsdaten, die im CP geändert werden sollen, werden im IF\_CONF als Blöcke "IF\_CONF\_..." für die einzelnen Parameter nach Bedarf zusammengestellt.

Parameter, die sich durch die Anweisung nicht ändern sollten, werden im IF\_CONF nicht eingetragen. Sie behalten dann den in STEP 7 projektierten Wert.

Details zur Parametrierung von IF\_CONF enthält der Abschnitt IF\_CONF: SDT für Telecontrol-Projektierungsdaten (Seite 2821).

Der Parameter INTERFACE referenziert die GPRS-Schnittstelle des gewünschten lokalen CP.

### Aufrufchnittstelle in FUP-Darstellung



### Erläuterung der Formalparameter

Die folgende Tabelle erläutert die Formalparameter der Anweisung TC\_CONFIG

| Parameter | Deklaration | Datentyp            | Wertebereich   | Beschreibung  |
|-----------|-------------|---------------------|--|---|
| REQ       | INPUT       | BOOL                | 0, 1   | Bei steigender Flanke wird die Bearbeitung der Anweisung gestartet und die Statusanzeigen initialisiert.<br>Aktualisierung der Statusanzeigen DONE, ERROR und STATUS, wenn keine positive Flanke ansteht. |
| INTERFACE | INPUT       | HW_INTERFACE (WORD) |  | Referenz auf die Schnittstelle des lokalen CP 1242-7  |
| CONFIG    | INOUT       | VARIANT             | Siehe auch "IF_CONF: SDT für Telecontrol-Projektierungsdaten"  | Referenz auf den Speicherbereich mit der Zusammenstellung der zu ändernden Projektierungsdaten  |
| ENO       | OUTPUT      | BOOL                | 0: Fehler<br>1: Fehlerfrei   | Freigabeausgang<br>Bei Auftreten eines Laufzeitfehlers der Anweisung wird ENO = 0 gesetzt.  |
| BUSY      | OUTPUT      | BOOL                | 0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen<br>1: Bearbeitung der Anweisung läuft | Anzeige des Bearbeitungs-Status der Anweisung   |

| Parameter | Deklaration | Datentyp | Wertebereich   | Beschreibung  |
|-----------|-------------|----------|--|---|
| DONE      | OUTPUT      | BOOL     | 0: -<br>1: Bearbeitung der Anweisung erfolgreich beendet | Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.<br>Zur Bedeutung im Zusammenhang mit den Parametern ERROR und STATUS siehe unter Anzeigen der Anweisung. |
| ERROR     | OUTPUT      | BOOL     | 0: -<br>1: Fehler  | Fehleranzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und STATUS siehe unter Anzeigen der Anweisung.   |
| STATUS    | OUTPUT      | WORD     |  | Statusanzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.  |

### Die Anzeigen BUSY, DONE und ERROR

Die Anzeigen von DONE und ERROR sind nur relevant bei BUSY = 0.

| BUSY | DONE | ERROR | Bedeutung                   |
|------|------|-------|-----------------------------|
| 0    | 0    | 0     | Kein Auftrag in Bearbeitung |

Alle weiteren Anzeigenkombinationen von DONE und ERROR finden Sie in der nachfolgenden Tabelle.

### Die Anzeigen DONE, ERROR und STATUS

Die folgende Tabelle informiert über die vom Anwenderprogramm auszuwertende Anzeige, gebildet aus DONE, ERROR und STATUS.

| DONE | ERROR | STATUS            | Bedeutung  |
|------|-------|-------------------|--|
| 1    | 0     | 0000 <sub>H</sub> | Auftrag fehlerfrei ausgeführt  |
| 0    | 0     | 7000 <sub>H</sub> | Keine Auftragsbearbeitung aktiv (Erstaufruf der Anweisung)   |
| 0    | 0     | 7001 <sub>H</sub> | Auftragsbearbeitung gestartet (Erstaufruf der Anweisung)   |
| 0    | 0     | 7002 <sub>H</sub> | Auftragsbearbeitung läuft bereits (erneuter Aufruf der Anweisung bei BUSY = 1)   |
| 0    | 1     | 80E6 <sub>H</sub> | Keine Anfrage in Bearbeitung (Aufruf der Anweisung nicht gestartet)  |
| 0    | 1     | 80EB <sub>H</sub> | Anfrage vorübergehend zurückgewiesen (der CP wird momentan von STEP 7 konfiguriert.)   |
| 0    | 1     | 80F6 <sub>H</sub> | Formatfehler eines Parameters im aufgerufenen Datenbaustein (falsche Länge, falsches Format oder ungültiger Wert)<br>Prüfen Sie den SDT "IF_CONF". |
| 0    | 1     | 80F7 <sub>H</sub> | Falsche ID in den Parameterblöcken der Projektierungsdaten:<br>Prüfen Sie den SDT "IF_CONF".   |

## Weitere Fehlermeldungen

### Weitere Fehlermeldungen

Die nachfolgenden Fehlermeldungen dienen nur zu Diagnosezwecken. Weitergehende Informationen kann Ihnen die Siemens-Hotline geben.

| DONE | ERROR | STATUS            | Bedeutung  |
|------|-------|-------------------|--|
| 0    | 1     | 80E0 <sub>H</sub> | Interner Fehler<br>Beachten Sie auch die mögliche Bedeutung bei den Anweisungen TC_SEND und TC_RECV. |

## TCON\_...: SDTs für den Telecontrol-Verbindungsaufbau

### Systemdatentypen TCON\_... für die Anweisung TC\_CON

Um den Aufbau einer Telecontrol-Verbindung mithilfe der Anweisung TC\_CON zu projektieren, wird der Parameter CONNECT der Anweisung für die Verbindungsbeschreibung verwendet.

Die Verbindungsbeschreibung ist durch die Struktur eines Systemdatentyps (SDT) vorgegeben. Die Struktur des jeweiligen SDT enthält die notwendigen Parameter, die zum Aufbau der Verbindung mit dem entfernten Kommunikationspartner erforderlich sind.

Für unterschiedliche Verbindungstypen, die vom jeweiligen entfernten Kommunikationspartner abhängen, werden folgende SDTs verwendet:

- TCON\_IP\_RFC für ISO-on-TCP-Verbindungen zu IPv4-Stationen mit CP 1242-7
- TCON\_IP\_V4 für UDP-Verbindungen zu IPv4-Stationen (nur Senden)
- TCON\_PHONE für Verbindungen zu SMS-Clients
- TCON\_WDC für Verbindungen zu Telecontrol-Servern oder Stationen, die über den Telecontrol-Server erreichbar sind.

Die Parametrierung der Verbindungsbeschreibung wird in einem Datenbaustein des gleichen Typs wie der SDT vorgenommen.

### Anlegen eines DB vom Typ TCON\_...

Die Datentypen der jeweiligen DBs müssen Sie über die Tastatur eintippen. Sie werden nicht in der Auswahlliste angezeigt. Groß-/Kleinschreibung spielt bei der Eingabe der Datentypen keine Rolle.

Gehen Sie zum Anlegen eines TCON\_...-DB folgendermaßen vor:

1. Legen Sie einen Datenbaustein vom Typ "Global-DB" mit Bausteinzugriff "Standard" an.
2. Legen Sie in der Tabelle der Parameterkonfiguration des DB einen SDT an, indem Sie den Namen vergeben und in die Zelle des Datentyps den gewünschten Typ (bspw. "TCON\_IP\_RFC") eintippen.  
Der SDT mit seinen Parametern (siehe unten) wird angelegt.
3. Projektieren Sie die Parameter, die nachfolgend für jeden SDT-Typ beschrieben sind.

Reservierte Bits werden nicht angezeigt.

### Systemdatentyp TCON\_IP\_RFC für Verbindungen zu IPv4-Stationen

Dieser Verbindungstyp wird nur bei ISO-on-TCP-Verbindungen zu Kommunikationspartnern mit fester IP-Adresse unterstützt. Der CP muss für die Betriebsart "GPRS direkt" projiziert sein.

Tabelle 9-98 Parameter von TCON\_IP\_RFC

| Byte      | Parameter         | Datentyp               | Anfangswert           | Beschreibung   |
|-----------|-------------------|------------------------|-----------------------|--|
| 0 ... 1   | InterfaceID       | HW_ANY                 |                       | Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")  |
| 2 ... 3   | ID                | CONN_OUC               | 1...07FF <sub>h</sub> | Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein.<br>Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.   |
| 4         | ConnectionType    | BYTE                   | W#16#0C               | Protokoll-Variante 12 (C <sub>h</sub> ): ISO-on-TCP-Verbindung   |
| 5         | ActiveEstablished | BOOL                   |                       | Kennung für die Art des Verbindungsaufbaus: <ul style="list-style-type: none"> <li>• 0: Passiver Verbindungsaufbau</li> <li>• 1: Aktiver Verbindungsaufbau</li> </ul>  |
| 6 ... 7   | -                 | -                      | -                     | - reserviert -   |
| 8 ... 11  | RemoteAddress     | IP_V4                  |                       | IP-Adresse der Verbindungspartner  |
|           | ADDR              | Array [1...4] of Byte  |                       | IP-Adresse des jeweiligen Verbindungspartners  |
| 12 ... 13 | RemoteTSelector   | TSelector              |                       | Entfernter T-Selektor  |
|           | TSelLen           | UINT                   |                       | Länge des entfernten T-Selektors "RemoteTSelector"   |
| 14 ... 45 | TSel              | Array [1...32] of Byte | beliebig              | Entfernter Transport-Selektor der Verbindung <ul style="list-style-type: none"> <li>• Bei "ActiveEstablished" = 1:<br/>Der T-Selektor des lokalen Partners muss bei aktivem Verbindungsaufbau gleich dem T-Selektor des Verbindungspartners sein (passiver Verbindungsaufbau beim entfernten Partner).</li> <li>• Bei "ActiveEstablished" = 0 entsprechend (Passiver Verbindungsaufbau lokal, aktiver Verbindungsaufbau entfernt)</li> </ul> |
| 46 ... 47 | LocalTSelector    | TSelector              |                       | Lokaler T-Selektor   |

| Byte         | Parameter | Datentyp                  | Anfangswert | Beschreibung  |
|--------------|-----------|---------------------------|-------------|---|
|              | TSelLen   | UINT                      |             | Länge des lokalen T-Selektors "LOCAL_TSel"  |
| 48 ...<br>79 | TSel      | Array [1...32]<br>of Byte | beliebig    | Lokaler Transport-Selektor der Verbindung <ul style="list-style-type: none"> <li>Bei "ActiveEstablished" = 1:<br/>Der T-Selektor des lokalen Partners muss bei aktivem Verbindungsaufbau gleich dem T-Selektor des Verbindungspartners sein (passiver Verbindungsaufbau beim entfernten Partner).</li> <li>Bei "ActiveEstablished" = 0 entsprechend (Passiver Verbindungsaufbau lokal, aktiver Verbindungsaufbau entfernt)</li> </ul> |

### Systemdatentyp TCON\_IP\_V4 für Verbindungen zu IPv4-Stationen

Dieser Verbindungstyp wird nur zum Senden über UDP-Verbindungen zu Kommunikationspartnern mit fester IP-Adresse unterstützt.

Wenn empfangen werden soll, dann muss ActiveEstablished = 0 gesetzt werden.

Tabelle 9-99 Parameter von TCON\_IP\_V4

| Byte         | Parameter         | Datentyp                 | Anfangswert           | Beschreibung  |
|--------------|-------------------|--------------------------|-----------------------|---|
| 0 ... 1      | InterfaceID       | HW_ANY                   |                       | Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")   |
| 2 ... 3      | ID                | CONN_OUC                 | 1...07FF <sub>h</sub> | Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein.<br><br>Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.  |
| 4            | ConnectionType    | BYTE                     | W#16#0B               | Protokoll-Variante 11 (B <sub>h</sub> ): UDP-Verbindung   |
| 5            | ActiveEstablished | BOOL                     |                       | Kennung für die Art des Verbindungsaufbaus: <ul style="list-style-type: none"> <li>0: Passiver Verbindungsaufbau<br/>Einstellung zum Senden und Empfangen von Daten.</li> <li>1: Aktiver Verbindungsaufbau<br/>Einstellung nur zum Senden von Daten.</li> </ul> |
| 6 ... 7      | -                 | -                        | -                     | - reserviert -  |
| 8 ... 11     | RemoteAddress     | IP_V4                    |                       | IP-Adresse der Verbindungspartner   |
|              | ADDR              | Array [1...4] of<br>Byte |                       | IP-Adresse des jeweiligen Verbindungspartners   |
| 12 ...<br>13 | RemotePort        | UINT                     | 1...65535             | IP-Port des Verbindungspartners<br>Nicht relevant, wenn ActiveEstablished = 0   |
| 14 ...<br>15 | LocalPort         | UINT                     | 1...65535             | Lokaler IP-Port ("0" ist nicht erlaubt.)<br>Nicht relevant, wenn ActiveEstablished = 1  |



## Systemdatentyp TCON\_PHONE für SMS-Verbindungen

### Hinweis

#### Autorisierte Rufnummern

Voraussetzung dafür, dass der CP eine SMS akzeptiert, ist die Autorisierung des sendenden Kommunikationspartners mithilfe seiner Rufnummer. Diese Rufnummern werden in STEP 7 beim CP in der Liste "Autorisierte Rufnummern" projiziert.

#### SMS-Text

- Auf programmierte SMS-Texte für zu sendende SMS wird über den Parameter DATA der Anweisung TC\_SEND zugegriffen.
- Der Text einer empfangenen SMS wird dem Adressbereich der CPU über den Parameter DATA der Anweisung TC\_RECV zugewiesen.

Tabelle 9-100 Parameter von TCON\_PHONE

| Byte     | Parameter         | Datentyp   | Anfangswert           | Beschreibung  |
|----------|-------------------|------------|-----------------------|---|
| 0 ... 1  | InterfaceID       | HW_ANY     |                       | Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")   |
| 2 ... 3  | ID                | CONN_OUC   | 1...07FF <sub>h</sub> | Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein.<br><br>Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.  |
| 4        | ConnectionType    | BYTE       | W#16#0E               | Protokoll-Variante 14 (E <sub>h</sub> ): SMS-Verbindung   |
| 5        | ActiveEstablished | BOOL       |                       | Kennung für die Art des Verbindungsaufbaus (nicht relevant für den CP 1242-7): <ul style="list-style-type: none"> <li>• 0: Passiver Verbindungsaufbau (hier nicht relevant)</li> <li>• 1: Aktiver Verbindungsaufbau</li> </ul>  |
| 6...7    | -                 | -          | -                     | - reserviert -  |
| 8 ... 31 | PhoneNumber       | STRING[22] |                       | Rufnummer des Verbindungspartners<br>Erlaubte Werte: Plus-Zeichen (+) und Ziffern<br>Achten Sie auf die genaue Zeichenfolge der vom Netzbetreiber zugewiesenen Ländervorwahl der betreffenden Rufnummer ("+"-Zeichen oder Nullen).<br>Ohne Eintrag des Parameters PhoneNumber wird kein Verbindungspartner spezifiziert und SMS können von allen autorisierten Verbindungspartnern empfangen werden.<br>Beachten Sie beim Anlauf: Ohne Eintrag liefert TC_RECV zunächst die älteste empfangene SMS. |

### Systemdatentyp TCON\_WDC für Verbindungen zu Telecontrol-Servern oder entfernten Stationen

Sie können die Verbindung mit dem Telecontrol-Server, welcher der S7-1200 zugeordnet ist, oder zu einer entfernten Station, die über den Telecontrol-Server erreichbar ist, mit dem TCON\_WDC projektieren. Die Adressdaten des Telecontrol-Servers, der dem CP zugeordnet ist, finden Sie in STEP 7 im Register "Telecontrol-Schnittstelle > Betriebsart" des CP. Der Telecontrol-Server oder eine entfernte Station wird über den Host-Namen oder die IP-Adresse adressiert.

Im Parameter "RemoteWdcAddress" des TCON\_WDC wird die Zugangs-ID des Verbindungspartners angegeben.

Tabelle 9-101 Parameter von TCON\_WDC

| Byte     | Parameter         | Datentyp | Anfangswert           | Beschreibung   |
|----------|-------------------|----------|-----------------------|--|
| 0 ... 1  | InterfaceID       | HW_ANY   |                       | Referenz auf die Schnittstelle des lokalen CP 1242-7 (siehe STEP 7 > CP-Projektierung > Telecontrol-Schnittstelle > "HW-Kennung")  |
| 2 ... 3  | ID                | CONN_OUC | 1...07FF <sub>h</sub> | Referenz auf die lokale GPRS-Verbindung. Die ID wird vergeben und muss innerhalb der CPU eindeutig sein.<br><br>Hier ist der gleiche Wert wie derjenige des Parameters ID der Anweisung TC_CON zu verwenden.   |
| 4        | ConnectionType    | BYTE     | W#16#0F               | Protokoll-Variante 15 (F <sub>h</sub> ): Telecontrol-Verbindung über IP-Adresse  |
| 5        | ActiveEstablished | BOOL     |                       | Kennung für die Art des Verbindungsaufbaus: <ul style="list-style-type: none"> <li>• 0: Passiver Verbindungsaufbau</li> <li>• 1: Aktiver Verbindungsaufbau</li> </ul>  |
| 6 ... 7  | -                 | -        | -                     | - reserviert -   |
| 8 ... 11 | RemoteWdcAddress  | DWORD    |                       | Angabe der Zugangs-ID (hex). Die Zugangs-ID ist abhängig vom Verbindungspartner. <ul style="list-style-type: none"> <li>• Verbindung zu einem entfernten CP:<br/>Die Zugangs-ID wird gebildet aus:                             <ul style="list-style-type: none"> <li>- STEP 7-Projektnummer</li> <li>- Stationsnummer</li> <li>- Steckplatz</li> </ul>                             Wenn die entfernte Station mehrere GPRS-CPs hat und der Weg nicht festgelegt werden soll, dann muss das letzte Byte für den Steckplatz gleich 0 gesetzt werden.                         </li> <li>Die Zugangs-ID finden Sie im STEP 7-Projekt unter der Parametergruppe "CP-Authentifizierung" des CP.</li> <li>• Verbindung zum Telecontrol-Server:<br/>Zugangs-ID = 0</li> <li>• Um nur in das Prozessabbild des CP zu schreiben:<br/>Zugangs-ID = DW#16#FEEDDADA</li> </ul> |

## IF\_CONF: SDT für Telecontrol-Projektierungsdaten

### Aufbau des Systemdatentyps IF\_CONF für die Anweisung TC\_CONFIG

Der Parameter CONFIG der Anweisung TC\_CONFIG referenziert den Speicherbereich mit den zu ändernden Projektierungsdaten des CP 1242-7. Die in einem Datenbaustein abgelegten Projektierungsdaten werden als Struktur vom Systemdatentyp (SDT) IF\_CONF beschrieben.

IF\_CONF setzt sich aus einem Header und nachfolgenden Blöcken zusammen, die den Parametern oder Parameterbereichen des CP in den Geräteeigenschaften des STEP 7-Projekts entsprechen.

Die zu ändernden Projektierungsdaten des CP werden als IF\_CONF-Blöcke zusammengestellt. Nicht zu ändernde Parameter werden in der IF\_CONF-Struktur nicht berücksichtigt und bleiben so, wie sie im STEP 7-Projekt konfiguriert wurden.

### Anlegen des DB und der IF\_CONF-Strukturen

Die Parameter des CP können Sie innerhalb des IF\_CONF-DB in einer oder in mehreren Strukturen mit jeweils einem oder mehreren Blöcken anlegen.

Die Datentypen der jeweiligen Blöcke müssen Sie über die Tastatur eintippen. Sie werden nicht in der Auswahlliste angezeigt. Groß-/Kleinschreibung spielt bei der Eingabe der Datentypen keine Rolle.

Gehen Sie zum Anlegen von IF\_CONF folgendermaßen vor:

1. Legen Sie einen Datenbaustein vom Typ "Global-DB" mit Bausteinzugriff "Standard" an.
2. Legen Sie in der Tabelle der Parameterkonfiguration des DB eine Struktur an (Datentyp "Struct").  
Den Name können Sie frei festlegen.
3. Fügen Sie unter dieser Struktur einen Header ein, indem Sie den Namen des Headers vergeben und in die Zelle des Datentyps "IF\_CONF\_Header" eintippen.  
Der Header der Struktur mit seinen drei Parametern (siehe unten) wird angelegt.
4. Legen Sie einen Block für den ersten zu ändernden Parameter an, indem Sie den gewünschten Datentyp (bspw. "IF\_CONF\_APN") in die Zelle des Datentyps eintippen.
5. Wiederholen Sie den letzten Schritt für all diejenigen Parameter, die Sie mithilfe der Anweisung TC\_CONFIG im CP ändern wollen.
6. Aktualisieren Sie abschließend im Header die Blockanzahl im Parameter "subfieldCnt".

### Header von IF\_CONF

Tabelle 9-102 IF\_CONF\_Header

| Byte    | Parameter   | Datentyp | Anfangswert | Beschreibung  |
|---------|-------------|----------|-------------|---|
| 0 ... 1 | fieldType   | UINT     |             | Blocktyp: Muss immer 0 sein.                        |
| 2 ... 3 | fieldId     | UINT     |             | Block-ID: Muss immer 0 sein.                        |
| 4 ... 5 | subfieldCnt | UINT     |             | Gesamtanzahl der in der Struktur enthaltenen Blöcke |

### Allgemeine Parameter der Parameterblöcke

Jeder Block enthält folgende allgemeine Parameter:

- **Id**  
Dieser Parameter kennzeichnet den jeweiligen Block und darf nicht verändert werden.
- **Length**  
Dieser Parameter gibt die Länge des Blocks an. Der Wert dient nur Informationszwecken. Blöcke mit Strings und / oder Arrays haben eine variable Länge. Durch versteckte Bytes kann die tatsächliche Länge von Blöcken größer als die Summe der angezeigten Parameter sein.
- **Mode**  
Für diesen Parameter sind die folgenden Werte zulässig:

Tabelle 9-103 Werte von "Mode"

| Wert | Bedeutung  |
|------|--|
| 1    | Permanente Gültigkeit der Projektierungsdaten<br>Nicht relevant beim CP 1242-7   |
| 2    | Temporäre Gültigkeit der Projektierungsdaten einschließlich Löschen vorhandener permanenter Projektierungsdaten<br>Die permanenten Projektierungsdaten werden durch die Parameterblöcke von IF_CONF ersetzt. |

### Block für den Parameterbereich "GPRS-Zugang"

Tabelle 9-104 IF\_CONF\_APN

| Parameter           | Datentyp    | Anfangswert | Beschreibung  |
|---------------------|-------------|-------------|---|
| Id                  | UINT        | 4           | Kennung des Parameterblocks   |
| Length              | UINT        |             | Länge des Parameterblocks in Byte: 174                              |
| Mode                | UINT        |             | Gültigkeit (1: permanent, 2: temporär)                              |
| AccesspointGPRS     | STRING [98] |             | APN: Name des Zugangspunkts des GSM-Netzwerkbetreibers zum Internet |
| AccesspointUser     | STRING [42] |             | APN-Benutzername  |
| AccesspointPassword | STRING [22] |             | APN-Passwort  |

### Block für den Parameterbereich "CP-Identifikation"

Tabelle 9-105 IF\_CONF\_Login

| Parameter | Datentyp | Anfangswert | Beschreibung                           |
|-----------|----------|-------------|--|
| Id        | UINT     | 5           | Kennung des Parameterblocks            |
| Length    | UINT     |             | Länge des Parameterblocks in Byte: 54  |
| Mode      | UINT     |             | Gültigkeit (1: permanent, 2: temporär) |

| Parameter     | Datentyp    | Anfangswert | Beschreibung                                     |
|---------------|-------------|-------------|--|
| ModemName     | STRING [22] |             | Zugangs-ID<br>Der Wert ist nicht parametrierbar. |
| ModemPassword | STRING [22] |             | Telecontrol-Passwort (max. 20 Zeichen)           |

### Block für den Parameterbereich "Telecontrol-Server-Zugang"

Dieser Block ist nur zu verwenden, wenn der Telecontrol-Server mit einem über DNS auflösbaren Namen adressiert wird. Wenn der Telecontrol-Server mit seiner IP-Adresse adressiert wird, dann wird der Block "IF\_CONF\_TCS\_IP\_V4" verwendet.

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Betriebsart".

Bei mehreren Telecontrol-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 9-106 IF\_CONF\_TCS\_Name

| Parameter  | Datentyp     | Anfangswert | Beschreibung   |
|------------|--------------|-------------|--|
| Id         | UINT         | 6           | Kennung des Parameterblocks  |
| Length     | UINT         |             | Länge des Parameterblocks in Byte: 266   |
| Mode       | UINT         |             | Gültigkeit (1: permanent, 2: temporär)   |
| TcsName    | -            | -           | - reserviert -   |
|            | STRING [254] |             | Durch DNS auflösbarer Name des Telecontrol-Servers   |
| RemotePort | UINT         |             | Port des Telecontrol-Servers   |
| Rank       | UINT         |             | Priorität des Servers [1, 2]<br>1 = Haupt-Telecontrol-Server,<br>2 = Ersatz-Telecontrol-Server |

### Block für den Parameterbereich "Telecontrol-Server-Zugang"

Dieser Block ist nur zu verwenden, wenn der Telecontrol-Server mit seiner IP-Adresse adressiert wird. Wenn der Telecontrol-Server mit seinem DNS-Namen adressiert wird, dann wird der Block "IF\_CONF\_TCS\_Name" verwendet.

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Betriebsart".

Bei mehreren Telecontrol-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 9-107 IF\_CONF\_TCS\_IP\_v4

| Parameter     | Datentyp | Anfangswert | Beschreibung   |
|---------------|----------|-------------|--|
| Id            | UINT     | 7           | Kennung des Parameterblocks  |
| Length        | UINT     |             | Länge des Parameterblocks in Byte: 14  |
| Mode          | UINT     |             | Gültigkeit (1: permanent, 2: temporär)   |
| RemoteAddress | IP_V4    |             | IP-Adresse des Telecontrol-Servers   |
| RemotePort    | UINT     |             | Port des Telecontrol-Servers   |
| Rank          | UINT     |             | Priorität des Servers [1, 2]<br>1 = Haupt-Telecontrol-Server,<br>2 = Ersatz-Telecontrol-Server |

**Block für den Parameterbereich "Betriebsart"**

In STEP 7 befinden sich die entsprechenden Daten in den Parameterbereichen "Betriebsart" und "Modem-Einstellungen".

Tabelle 9-108 IF\_CONF\_GPRS\_Mode

| Parameter        | Datentyp | Anfangswert | Beschreibung  |
|------------------|----------|-------------|---|
| Id               | UINT     | 8           | Kennung des Parameterblocks   |
| Length           | UINT     |             | Länge des Parameterblocks in Byte: 10   |
| Mode             | UINT     |             | Gültigkeit (1: permanent, 2: temporär)  |
| GPRSmode         | UINT     |             | Betriebsart des CP: <ul style="list-style-type: none"> <li>• 0 = Telecontrol</li> <li>• 1 = GPRS direkt</li> </ul>  |
| TemporaryStation | BOOL     |             | Bit 0: Temporäre Verbindung<br>Bei aktivierter Option baut der CP eine Verbindung nur zeitweilig zum Senden von Daten auf. Nach erfolgreichem Übertragen der Telegramme baut der CP die Verbindung wieder ab. <ul style="list-style-type: none"> <li>• 1: aktiviert (temporäre Verbindung)</li> <li>• 0: deaktiviert (permanente Verbindung)</li> </ul> |
| SMS_Enabled      | BOOL     |             | Bit 1: SMS ermöglichen<br>Die Aktivierung der Option ermöglicht der S7-Station das Versenden von SMS. <ul style="list-style-type: none"> <li>• 1: aktiviert (SMS ermöglicht)</li> <li>• 0: deaktiviert (keine SMS)</li> </ul>   |

**Block für den Parameter "SMSC"**

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Modem-Einstellungen".

Tabelle 9-109 IF\_CONF\_SMS\_Provider

| Parameter   | Datentyp    | Anfangswert | Beschreibung  |
|-------------|-------------|-------------|---|
| Id          | UINT        | 10          | Kennung des Parameterblocks   |
| Length      | UINT        |             | Länge des Parameterblocks in Byte: 28   |
| Mode        | UINT        |             | Gültigkeit (1: permanent, 2: temporär)  |
| SMSProvider | STRING [20] |             | Teilnehmernummer der SMS-Zentrale (SMSC) des GSM-Netzwerkbetreibers, mit dem der Mobilfunk-Vertrag für diese Station abgeschlossen wurde. |

**Block für den Parameter "PIN"**

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "Modem-Einstellungen".

Tabelle 9-110 IF\_CONF\_PIN

| Parameter | Datentyp   | Anfangswert | Beschreibung   |
|-----------|------------|-------------|--|
| Id        | UINT       | 11          | Kennung des Parameterblocks  |
| Length    | UINT       |             | Länge des Parameterblocks in Byte: 16  |
| Mode      | UINT       |             | Gültigkeit (1: permanent, 2: temporär)   |
| Pin       | STRING [8] |             | PIN der im CP gesteckten SIM-Karte<br>Der Parameter ist nicht relevant, wenn die PIN richtig projektiert wurde. Bei falsch projektierte PIN kann die richtige PIN hiermit eingegeben werden. |

**Block für Überwachungszeiten**

In STEP 7 befinden sich die entsprechenden Daten in den Parameterbereichen "Keep-Alive-Wartezeit" und "Betriebsart" des CP.

Tabelle 9-111 IF\_CONF\_TC\_Timeouts

| Parameter        | Datentyp | Anfangswert | Beschreibung  |
|------------------|----------|-------------|---|
| Id               | UINT     | 12          | Kennung des Parameterblocks   |
| Length           | UINT     |             | Länge des Parameterblocks in Byte: 12   |
| Mode             | UINT     |             | Gültigkeit (1: permanent, 2: temporär)  |
| KeepAliveTimeout | -        | -           | - Reserviert - (nicht parametrierbar)   |
| SendTimeout      | UINT     |             | Verbindungsüberwachungszeit: Überwachungszeit der Verbindung zum Kommunikationspartner (Sekunden)<br>Relevant in den Betriebsarten "Telecontrol" und "GPRS direkt"  |
| RedialTimeout    | UINT     |             | Wahlwiederholungsverzögerung: Basiswert für die Wartezeit bis zum nächsten Verbindungsaufbauversuch nach erfolglosem Verbindungsaufbau. Nach jeweils 3 Versuchen wird der Basiswert bis max. 900 s verdoppelt. Wertebereich: 10...600 s. Bei projektiertem Ersatz-Telecontrol-Server versucht der CP beim 4. Mal, sich mit diesem zu verbinden. Bsp.: Basiswert 20 ergibt folgende Wählintervalle: dreimal 20 s, dreimal 40 s, dreimal 80 s etc. bis max. 900 s.<br>Nicht relevant für SMS-Verbindungen |

**Block für den Parameterbereich "Weckberechtigung"**

Tabelle 9-112 IF\_CONF\_WakeupList

| Parameter            | Datentyp                      | Anfangswert | Beschreibung  |
|----------------------|-------------------------------|-------------|---|
| Id                   | UINT                          | 13          | Kennung des Parameterblocks   |
| Length               | UINT                          |             | Länge des Parameterblocks in Byte: 246  |
| Mode                 | UINT                          |             | Gültigkeit (1: permanent, 2: temporär)  |
| WakeupPhone [1...10] | ARRAY [1...10] of STRING [22] |             | Rufnummer des zum Wecken autorisierten Teilnehmers<br>Der Stern (*) am Ende einer Rufnummer dient als Platzhalter für Durchwahlnummern. |

**Block für den Parameterbereich "Bevorzugte GSM-Netze"**

Tabelle 9-113 IF\_CONF\_PrefProvider

| Parameter        | Datentyp                    | Anfangswert | Beschreibung   |
|------------------|-----------------------------|-------------|--|
| Id               | UINT                        | 14          | Kennung des Parameterblocks  |
| Length           | UINT                        |             | Länge des Parameterblocks in Byte: 46  |
| Mode             | UINT                        |             | Gültigkeit (1: permanent, 2: temporär)   |
| Provider [1...5] | ARRAY [1...5] of STRING [6] |             | Alternative GSM-Netze mit Priorität 1 bis 5, in die sich der CP bevorzugt einwählt. Bis zu 5 Netze sind projektierbar. Nr. 1 mit höchster Priorität, Nr. 5 mit niedrigster Priorität.<br>Eingabe des Public Land Mobile Network (PLMN) des Netzbetreibers, bestehend aus Mobile Country Code (MCC) und Mobile Network Code (MNC).<br>Beispiel (Testnetz der Siemens AG): 26276 |

**Block für den Parameterbereich "DNS-Konfiguration"**

Tabelle 9-114 IF\_CONF\_DNS

| Parameter  | Datentyp | Anfangswert | Beschreibung                                 |
|------------|----------|-------------|--|
| Id         | UINT     | 16          | Kennung des Parameterblocks                  |
| Length     | UINT     |             | Länge des Parameterblocks in Byte: 14        |
| Mode       | UINT     |             | Gültigkeit (1: permanent, 2: temporär)       |
| DNS_IP [1] | IP_V4    |             | IP-Adresse des 1. Domain Name System Servers |
| DNS_IP [2] | IP_V4    |             | IP-Adresse des 2. Domain Name System Servers |



**Block für den Parameterbereich "Uhrzeitsynchronisation"**

Tabelle 9-115 IF\_CONF\_NTP

| Parameter  | Datentyp                  | Anfangswert | Beschreibung                           |
|------------|---------------------------|-------------|--|
| Id         | UINT                      | 17          | Kennung des Parameterblocks            |
| Length     | UINT                      |             | Länge des Parameterblocks in Byte: 24  |
| Mode       | UINT                      |             | Gültigkeit (1: permanent, 2: temporär) |
| NTP_IP [1] | ARRAY [1...4]<br>of IP_V4 |             | IP-Adresse von NTP-Server 1            |
| ...        | ...                       |             | (IP-Adresse von NTP-Server 2...3)      |
| NTP_IP [4] | ARRAY [1...4]<br>of IP_V4 |             | IP-Adresse von NTP-Server 4            |

**Block zur Aktivierung / Deaktivierung von TeleService-Benutzern**

SDT zur Aktivierung oder Deaktivierung von TeleService-Benutzern, die bereits im STEP 7-Projekt des CP projektiert sind. In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen" > "TeleService-Benutzerverwaltung".

Tabelle 9-116 IF\_CONF\_GPRS\_UserList

| Parameter          | Datentyp                       | Anfangswert | Beschreibung                               |
|--------------------|--------------------------------|-------------|--|
| Id                 | UINT                           | 19          | Kennung des Parameterblocks                |
| Length             | UINT                           |             | Länge des Parameterblocks in Byte: 506     |
| Mode               | UINT                           |             | Gültigkeit (1: permanent, 2: temporär)     |
| GPRS_User [1...10] | ARRAY [1...10]<br>of GPRS_User |             | TeleService-Benutzer Nr. 1 bis max. Nr. 10 |

Der Array wird gebildet aus den Parametersätzen für die TeleService-Benutzer ("GPRS\_User" [1...n]).

Tabelle 9-117 GPRS\_User [n] (Parameter für TeleService-Benutzer)

| Parameter            | Datentyp    | Anfangswert | Beschreibung   |
|----------------------|-------------|-------------|--|
| UserName [n]         | STRING [22] |             | TeleService-Benutzername   |
| Password [n]         | STRING [22] |             | - Der String muss leer sein! -   |
| Diag_Allowed [n]     | BOOL        |             | - Reserviert - (nicht parametrierbar)  |
| Teleserv_Allowed [n] | BOOL        |             | Aktivierung des TeleService-Benutzers <ul style="list-style-type: none"> <li>• 0 = Benutzer ist deaktiviert</li> <li>• 1 = Benutzer ist aktiviert</li> </ul> |
| FW_Load_Allowed [n]  | BOOL        |             | - Reserviert - (nicht parametrierbar)  |

**Block für die Parametrierung des TeleService-Zugangs (DNS-Name des Servers)**

Zugangsdaten des TeleService-Servers (Vermittlerstation).

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen".

Bei zwei TeleService-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 9-118 IF\_CONF\_TS\_Name

| Parameter  | Datentyp     | Anfangswert | Beschreibung   |
|------------|--------------|-------------|--|
| Id         | UINT         | 20          | Kennung des Parameterblocks                                      |
| Length     | UINT         |             | Länge des Parameterblocks in Byte: 266                           |
| Mode       | UINT         |             | Gültigkeit (1: permanent, 2: temporär)                           |
| ts_name    | String [254] |             | Durch DNS auflösbarer Name ders TeleService-Servers              |
| RemotePort | UINT         |             | Port der Engineering-Station                                     |
| Rank       | UINT         |             | Priorität des Servers [1] oder [2]<br>1 = Server 1, 2 = Server 2 |

### Block für die Parametrierung des TeleService-Zugangs (IP-Adresse des Servers)

Zugangsdaten des TeleService-Servers (Vermittlerstation).

In STEP 7 befinden sich die entsprechenden Daten im Parameterbereich "TeleService-Einstellungen".

Bei zwei TeleService-Servern verwenden Sie den Block je einmal pro Server.

Tabelle 9-119 IF\_CONF\_TS\_IF\_V4

| Parameter     | Datentyp | Anfangswert | Beschreibung   |
|---------------|----------|-------------|--|
| Id            | UINT     | 21          | Kennung des Parameterblocks                                      |
| Length        | UINT     |             | Länge des Parameterblocks in Byte: 14                            |
| Mode          | UINT     |             | Gültigkeit (1: permanent, 2: temporär)                           |
| RemoteAddress | IP_V4    |             | IP-Adresse des TeleService-Servers                               |
| RemotePort    | UINT     |             | Port des TeleService-Servers                                     |
| Rank          | UINT     |             | Priorität des Servers [1] oder [2]<br>1 = Server 1, 2 = Server 2 |

### Punkt-zu-Punkt

#### PORT\_CFG: Kommunikationsparameter dynamisch projektieren

##### Beschreibung

Die Anweisung "PORT\_CFG" ermöglicht die dynamische Projektierung von Kommunikationsparametern für einen Punkt-zu-Punkt-Kommunikationsport.

Die ursprüngliche statische Projektierung des Ports richten Sie in der Hardware-Konfiguration ein. Diese Projektierung können Sie durch die Ausführung der Anweisung "PORT\_CFG" ändern. Diese Funktion können Sie z. B. auch nutzen, um erstellte Bausteine in Bibliotheken

abzulegen und bei einer Wiederverwendung die Projektierung in der Hardware-Konfiguration zu umgehen.

Mit "PORT\_CFG" können Sie die folgenden Einstellungen der Kommunikationsparameter beeinflussen:

- Parität
- Baudrate
- Anzahl der Bits pro Zeichen
- Anzahl der Bitstopps
- Art und Eigenschaften der Flusskontrolle

Die Änderungen durch die Anweisung "PORT\_CFG" werden im Zielsystem nicht dauerhaft gespeichert.

Serielle Daten können Sie über die elektrischen Verbindungen RS-232 (Halb- und Vollduplex) und RS-485 (Halbduplex) übertragen.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PORT\_CFG":

| Parameter | Deklaration | Datentyp    | Speicherbereich              | Beschreibung   |
|-----------|-------------|-------------|------------------------------|--|
| REQ       | Input       | BOOL        | E, A, M, D, L                | Aktiviert die Projektierungsänderung bei einer steigenden Flanke   |
| PORT      | Input       | PORT (UINT) | D, L oder Konstante          | Kennung des Kommunikationsports (HW-ID)  |
| PROTOCOL  | Input       | UINT        | E, A, M, D, L oder Konstante | Übertragungsprotokoll: <ul style="list-style-type: none"> <li>• 0: Punkt-zu-Punkt-Kommunikationsprotokoll</li> <li>• 1..n: Zukünftige Definition für spezifische Übertragungsprotokolle</li> </ul>   |
| BAUD      | Input       | UINT        | E, A, M, D, L oder Konstante | Baudrate des Ports: <ul style="list-style-type: none"> <li>• 1: 300 Baud</li> <li>• 2: 600 Baud</li> <li>• 3: 1200 Baud</li> <li>• 4: 2400 Baud</li> <li>• 5: 4800 Baud</li> <li>• 6: 9600 Baud (Standardeinstellung)</li> <li>• 7: 19200 Baud</li> <li>• 8: 38400 Baud</li> <li>• 9: 57600 Baud</li> <li>• 10: 76800 Baud</li> <li>• 11: 115200 Baud</li> </ul> |

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| PARITY    | Input       | UINT     | E, A, M, D, L oder Konstante | Parität des Ports: <ul style="list-style-type: none"> <li>• 1: Keine Parität (Standardeinstellung)</li> <li>• 2: Gerade Parität</li> <li>• 3: Ungerade Parität</li> <li>• 4: Markierung Parität</li> <li>• 5: Leerzeichen Parität</li> </ul>                        |
| DATABITS  | Input       | UINT     | E, A, M, D, L oder Konstante | Bits pro Zeichen: <ul style="list-style-type: none"> <li>• 1: 8 Bits pro Zeichen (Standardeinstellung)</li> <li>• 2: 7 Bits pro Zeichen</li> </ul>  |
| STOPBITS  | Input       | UINT     | E, A, M, D, L oder Konstante | Anzahl der Stoppbits: <ul style="list-style-type: none"> <li>• 1: 1 Stoppbit (Standardeinstellung)</li> <li>• 2: 2 Stoppbits</li> </ul>   |
| FLOWCTRL  | Input       | UINT     | E, A, M, D, L oder Konstante | Datenflusskontrolle: <ul style="list-style-type: none"> <li>• 1: Keine (Standard)</li> <li>• 2: XON/XOFF</li> <li>• 3: Hardware-Flusskontrolle (RTS immer eingeschaltet)</li> <li>• 4: Hardware-Flusskontrolle (RTS abschaltbar während der Übertragung)</li> </ul> |
| XONCHAR   | Input       | CHAR     | D                            | Gibt das Zeichen an, das als XON-Zeichen verwendet wird. Als Standard ist das Zeichen DC1 (11H) eingestellt.  |
| XOFFCHAR  | Input       | CHAR     | D                            | Gibt das Zeichen an, das als XOFF-Zeichen verwendet wird. Als Standard ist das Zeichen DC3 (13H) eingestellt.   |
| WAITIME   | Input       | UINT     | E, A, M, D, L oder Konstante | Gibt die Wartezeit für XON oder CTS nach dem Beginn der Übertragung an.<br>Der angegebene Wert muss größer als 0 sein. Als Standard sind 2000 Millisekunden eingestellt.  |
| DONE      | Output      | BOOL     | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>  |
| ERROR     | Output      | BOOL     | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>   |
| STATUS    | Output      | WORD     | E, A, M, D, L                | Status der Anweisung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 80A0  | Das angegebene Protokoll ist ungültig.   |
| 80A1  | Die angegebene Baudrate ist ungültig.  |
| 80A2  | Die angegebene Parität ist ungültig.   |
| 80A3  | Die angegebene Anzahl der Bits pro Zeichen ist ungültig.   |
| 80A4  | Die angegebene Anzahl der Stoppbits ist ungültig.  |
| 80A5  | Die angegebene Art der Flusskontrolle ist ungültig.  |
| 80A6  | Fehlerhafter Wert am Parameter WAITTIME<br>Wenn die Datenflusskontrolle aktiviert ist, muss der Wert am Parameter WAITTIME größer als Null sein. |
| 80A7  | Ungültige Werte an den Parametern XONCHAR und XOFFCHAR.  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 2846)".

## SEND\_CFG: Serielle Übertragungsparameter dynamisch projektieren

### Beschreibung

Die Anweisung "SEND\_CFG" ermöglicht die dynamische Projektierung von seriellen Übertragungsparametern für einen Punkt-zu-Punkt-Kommunikationsport. Alle für die Übertragung anstehenden Nachrichten werden nach Ausführung von "SEND\_CFG" verworfen.

Die ursprüngliche statische Projektierung des Ports richten Sie in der Hardware-Konfiguration ein. Diese Projektierung können Sie durch die Ausführung der Anweisung "SEND\_CFG" ändern. Diese Funktion können Sie z. B. auch nutzen, um erstellte Bausteine in Bibliotheken abzulegen und bei einer Wiederverwendung die Projektierung in der Hardware-Konfiguration zu umgehen. Mit "SEND\_CFG" können Sie die folgenden Einstellungen der Übertragungsparameter beeinflussen:

- Zeitabstand zwischen der Aktivierung von RTS (Request to Send) und dem Start der Übertragung
- Zeitabstand zwischen Übertragungsende und der Deaktivierung von RTS
- Bitzeiten für Pausen festlegen

Die Änderungen durch die Anweisung "SEND\_CFG" werden im Zielsystem nicht dauerhaft gespeichert.

Serielle Daten können Sie über die elektrischen Verbindungen RS-232 (Halb- und Vollduplex) und RS-485 (Halbduplex) übertragen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "SEND\_CFG":

| Parameter | Deklaration | Datentyp    | Speicherbereich              | Beschreibung   |
|-----------|-------------|-------------|------------------------------|--|
| REQ       | Input       | BOOL        | E, A, M, D, L                | Aktiviert die Projektierungsänderung bei einer steigenden Flanke   |
| PORT      | Input       | PORT (UINT) | D, L oder Konstante          | Kennung des Kommunikationsports (HW-ID)  |
| RTSONDLY  | Input       | UINT        | E, A, M, D, L oder Konstante | Zeit, die nach der Aktivierung von RTS bis zum Start der Übertragung vergehen soll.<br>Gültige Werte für diesen Parameter sind:<br><ul style="list-style-type: none"> <li>• 0 (Standard)</li> <li>• 0 bis 65535 ms in Schritten von 1 ms</li> </ul> Dieser Parameter gilt nicht bei RS-485-Baugruppen. |
| RTSOFFDLY | Input       | UINT        | E, A, M, D, L oder Konstante | Zeit, die nach dem Übertragungsende bis zur Deaktivierung von RTS vergehen soll.<br>Gültige Werte für diesen Parameter sind:<br><ul style="list-style-type: none"> <li>• 0 (Standard)</li> <li>• 0 bis 65535 ms in Schritten von 1 ms</li> </ul> Dieser Parameter gilt nicht bei RS-485-Baugruppen.    |
| BREAK     | Input       | UINT        | E, A, M, D, L oder Konstante | Gibt die Bitzeiten für eine Pause an, die zum Nachrichtenbeginn gesendet wird.<br>Als Standard werden 12 Bitzeiten eingestellt.<br>Maximal können 25000 Bitzeiten angegeben werden.  |
| IDLELINE  | Input       | UINT        | E, A, M, D, L oder Konstante | Gibt die Bitzeiten für Idle Line an nach der Pause zum Nachrichtenbeginn.<br>Als Standard werden 12 Bitzeiten eingestellt.<br>Maximal können 25000 Bitzeiten angegeben werden.   |
| DONE      | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten:<br><ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>  |
| ERROR     | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten:<br><ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>   |
| STATUS    | Output      | WORD        | E, A, M, D, L                | Status der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 80B0  | Die Konfiguration einer Übertragungsunterbrechung ist nicht erlaubt                      |
| 80B1  | Die angegebene Pausenzeit überschreitet das erlaubte Maximum von 25000 Bitzeiten         |
| 80B2  | Die für Idle Line angegebene Zeit überschreitet das erlaubte Maximum von 25000 Bitzeiten |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 2846)".

## RCV\_CFG: Serielle Empfangsparameter dynamisch projektieren

### Beschreibung

Die Anweisung "RCV\_CFG" ermöglicht die dynamische Projektierung von seriellen Empfangsparametern für einen Punkt-zu-Punkt-Kommunikationsport. Mit dieser Anweisung können Sie die Bedingungen projektieren, die den Anfang und das Ende einer zu übertragenden Nachricht bestimmen. Der Empfang von Nachrichten, die diesen Bedingungen entsprechen, kann dann durch die Anweisung "RCV\_PTP (Seite 2842)" freigegeben werden.

Die ursprüngliche statische Projektierung des Ports richten Sie in den Eigenschaften der Hardware-Konfiguration ein. Führen Sie die Anweisung "RCV\_CFG" in Ihrem Programm aus, um die Projektierung zu ändern. Diese Funktion können Sie z. B. auch nutzen, um erstellte Bausteine in Bibliotheken abzulegen und bei einer Wiederverwendung die Projektierung in der Hardware-Konfiguration zu umgehen. Die Änderungen durch die Anweisung "RCV\_CFG" werden im Zielsystem nicht dauerhaft gespeichert.

Alle für die Übertragung anstehenden Nachrichten werden nach Ausführung der Anweisung "RCV\_CFG" verworfen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCV\_CFG":

| Parameter  | Deklaration | Datentyp    | Speicherbereich     | Beschreibung   |
|------------|-------------|-------------|---------------------|--|
| REQ        | Input       | BOOL        | E, A, M, D, L       | Aktiviert die Projektierungsänderung bei einer steigenden Flanke.                  |
| PORT       | Input       | PORT (UINT) | D, L oder Konstante | Kennung des Kommunikationsports (HW-ID)  |
| CONDITIONS | Input       | CONDITIONS  | D, L                | Datenstruktur, die Bedingungen für Anfang und Ende der Datenübertragung definiert. |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| DONE      | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul> |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten.</li> </ul>   |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Status der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Datentyp CONDITIONS

Mithilfe der Struktur CONDITIONS legen Sie die Start- und Endbedingungen für die Nachrichtenübertragung fest. Die Struktur CONDITIONS ist in dem Instanz-DB der Anweisung "RCV\_CFG" enthalten. Über die Struktur CONDITIONS definieren Sie die die Start- und Endbedingungen, wann die Übertragung einer Nachricht beendet ist und wann die nächste Nachrichtenübertragung beginnt:

- Die Startbedingung für die Datenübertragung definieren Sie in der Struktur START
- Die Endbedingung für die Datenübertragung definieren Sie in der Struktur END

Dabei können eine oder mehrere Start- und Endbedingungen definiert werden. Bei der Angabe mehrerer Start- oder Endbedingungen werden diese durch ein ODER-Gatter verknüpft.

Die folgende Tabelle zeigt die Struktur CONDITIONS:

| Parameter | Datentyp | Beschreibung   |
|-----------|----------|--|
| START     | STRUCT   | Startbedingungen   |
| STARTCOND | UINT     | Gibt die Startbedingung an (Details, siehe unten).<br>Die Startbedingung kann als ein 16-Bit Hexadezimalwert angegeben werden.<br>Mögliche Werte für die Startbedingung sind: <ul style="list-style-type: none"> <li>• 1: Startzeichen</li> <li>• 2: Beliebiges Zeichen (Standardeinstellung)</li> <li>• 4: Line break</li> <li>• 8: Idle line</li> <li>• 16: Zeichenkette 1</li> <li>• 32: Zeichenkette 2</li> <li>• 64: Zeichenkette 3</li> <li>• 128: Zeichenkette 4</li> </ul> Auch mehrere Startbedingungen können am Parameter STARTCOND festlegt werden. Dafür wird die Summe aus den Werten der einzelnen Bedingungen angegeben. Wenn beispielsweise als Startbedingung "Idle line" ODER "Zeichenkette 1" ODER "Zeichenkette 4" definiert werden soll, muss der Wert "152" angegeben werden. |



| Parameter  | Datentyp | Beschreibung  |
|------------|----------|---|
| IDLETIME   | UINT     | Gibt die maximale Ruhezeit der Leitung an, bevor der Empfang gestartet wird.<br>Gültige Werte für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 40 Bitzeiten (Standardeinstellung)</li> <li>• 0 bis 2500 Bitzeiten</li> </ul>   |
| STARTCHAR  | BYTE     | Gibt das Startzeichen an. Diese Einstellung ist nur dann wirksam, wenn die projektierte Startbedingung "Startzeichen" ist.<br>Gültige Werte für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 02 (STX): Standardeinstellung</li> <li>• B#16#00 bis B#16#FF</li> </ul>   |
| SEQ[1].CTL | BYTE     | Zeichenkette 1: Steuerung für jedes Zeichen<br>Durch die Bitposition der Zeichen kann festgelegt werden, welche Zeichen der Zeichenkette berücksichtigt oder ignoriert werden. Um die Zeichen auszuwerten, müssen die entsprechenden Bits gesetzt werden. <ul style="list-style-type: none"> <li>• Bit 0: 1 Zeichen</li> <li>• Bit 1: 2 Zeichen</li> <li>• Bit 2: 3 Zeichen</li> <li>• Bit 3: 4 Zeichen</li> <li>• Bit 4: 5 Zeichen</li> </ul> Ein Zeichen wird ignoriert, wenn das entsprechende Bit zurückgesetzt ist.  |
| SEQ[1].STR | CHAR[5]  | Zeichenkette 1: Startzeichen (5 Zeichen)  |
| SEQ[2].CTL | BYTE     | Zeichenkette 2: Steuerung für jedes Zeichen ignorieren/vergleichen  |
| SEQ[2].STR | CHAR[5]  | Zeichenkette 2: Startzeichen (5 Zeichen)  |
| SEQ[3].CTL | BYTE     | Zeichenkette 3: Steuerung für jedes Zeichen ignorieren/vergleichen  |
| SEQ[3].STR | CHAR[5]  | Zeichenkette 3: Startzeichen (5 Zeichen)  |
| SEQ[4].CTL | BYTE     | Zeichenkette 4: Steuerung für jedes Zeichen ignorieren/vergleichen  |
| SEQ[4].STR | CHAR[5]  | Zeichenkette 4: Startzeichen (5 Zeichen)  |
| END        | STRUCT   | Endbedingungen  |
| ENDCOND    | UINT     | Gibt die Endbedingung an (Details, siehe unten).<br>Die Endbedingung kann als ein 16-Bit Hexadezimalwert angegeben werden.<br>Mögliche Werte für die Endbedingung sind: <ul style="list-style-type: none"> <li>• 1: Antwort-Zeitüberschreitung</li> <li>• 2: Nachrichten-Zeitüberschreitung</li> <li>• 4: Zeitüberschreitung innerhalb der Zeichenkette</li> <li>• 8: Maximale Länge</li> <li>• 16: N+LEN+M; Die Angabe über die Meldungslänge ist in der Meldung integriert und wird ausgewertet.</li> <li>• 32: Zeichenkette 1</li> </ul> Auch mehrere Endbedingungen können am Parameter ENDCOND festgelegt werden. Dafür wird die Summe aus den Werten der einzelnen Endbedingungen angegeben. Wenn beispielsweise die Endbedingung "Maximale Länge" ODER "Sequenz 1" definiert werden soll, muss der Wert "40" angegeben werden. |

| Parameter  | Datentyp | Beschreibung  |
|------------|----------|---|
| MAXLEN     | UINT     | Gibt die maximale Anzahl der Zeichen in einer Nachricht an.<br>Gültige Werte* für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 1 Zeichen (Standardeinstellung)</li> <li>• 0 bis 1024 Zeichen</li> </ul> Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "Maximale Länge" eingestellt ist.   |
| N          | UINT     | Offset des Längenfeldes in der Nachricht<br>Gültige Werte für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 0 Zeichen (Standardeinstellung)</li> <li>• 0 bis 1024 Zeichen</li> </ul> Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "N+LEN+M" eingestellt ist.  |
| LENGTHSIZE | UINT     | Größe des Längenfeldes in Bytes<br>Gültige Werte* für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 0 Bytes (Standardeinstellung)</li> <li>• 1 Byte</li> <li>• 2 Bytes</li> <li>• 4 Bytes</li> </ul> Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "N+LEN+M" eingestellt ist.  |
| LENGTHM    | UINT     | Gibt die Anzahl von Endzeichen an, die dem Längenfeld folgen aber nicht in der Länge der Nachricht enthalten sind.<br>Gültige Werte für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 0 Zeichen (Standardeinstellung)</li> <li>• 0 bis 255 Zeichen</li> </ul> Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "N+LEN+M" eingestellt ist. |
| RCVTIME    | UINT     | Gibt die Maximaldauer für den Empfang des ersten Zeichens einer Nachricht an.<br>Gültige Werte für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 200 ms (Standardeinstellung)</li> <li>• 0 bis 65535 ms in Schritten von 1 ms</li> </ul> Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "Antwort-Zeitüberschreitung" eingestellt ist.   |
| MSGTIME    | UINT     | Gibt die Maximaldauer für den Empfang einer Nachricht an.<br>Gültige Werte für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 200 ms (Standardeinstellung)</li> <li>• 0 bis 65535 ms in Schritten von 1 ms</li> </ul> Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "Nachrichten-Zeitüberschreitung" eingestellt ist.                   |
| CHARGAP    | UINT     | Gibt den Zeitabstand beim Empfang aufeinander folgender Zeichen an.<br>Gültige Werte für diesen Parameter sind: <ul style="list-style-type: none"> <li>• 12 Bitzeiten (Standardeinstellung)</li> <li>• 0 bis 2500 Bitzeiten</li> </ul> Diese Einstellung ist nur dann wirksam, wenn am Parameter ENDCOND die Endbedingung "Zeitüberschreitung innerhalb der Zeichenkette" eingestellt ist.    |

| Parameter   | Datentyp | Beschreibung   |
|---|----------|--|
| SEQ.CTL   | BYTE     | Zeichenkette: Steuerung für jedes Zeichen<br>Durch die Bitposition der Zeichen kann festgelegt werden, welche Zeichen der Zeichenkette berücksichtigt oder ignoriert werden. Um die Zeichen auszuwerten, müssen die entsprechenden Bits gesetzt werden. <ul style="list-style-type: none"> <li>• Bit 0: 1 Zeichen</li> <li>• Bit 1: 2 Zeichen</li> <li>• Bit 2: 3 Zeichen</li> <li>• Bit 3: 4 Zeichen</li> <li>• Bit 4: 5 Zeichen</li> </ul> Ein Zeichen wird ignoriert, wenn das entsprechende Bit zurückgesetzt ist. |
| SEQ.STR   | CHAR[5]  | Zeichenkette: Startzeichen (5 Zeichen)   |
| * Diese Wertebereiche gelten auch für die entsprechenden Hardware-Einstellungen zum Festlegen des Nachrichtenendes. |          |  |

### Startbedingungen für den Nachrichtenempfang (Parameter STARTCOND)

Der Beginn einer Nachricht wird vom Empfänger erkannt, wenn eine projektierte Startbedingung zutrifft. Die folgenden Bedingungen können als Startbedingungen für den Nachrichtenempfang festgelegt werden:

- **Startzeichen:** Der Beginn einer Nachricht wird beim Auftreten eines bestimmten Zeichens erkannt. Dieses Zeichen wird als erstes Zeichen der Nachricht gespeichert. Alle Zeichen, die vor dem Startzeichen empfangen worden sind, werden verworfen.
- **Beliebiges Zeichen:** Jedes beliebige Zeichen kann den Beginn einer Nachricht definieren. Dieses Zeichen wird als erstes Zeichen der Nachricht gespeichert.
- **Line Break:** Der Beginn einer Nachricht wird erkannt, wenn der Empfangsdatenstrom für mehr als ein Zeichen Länge unterbrochen wurde.
- **Idle Line:** Der Beginn einer Nachricht wird erkannt, wenn die Übertragungsleitung für eine bestimmte Zeit (angegeben in Bitzeiten) im Ruhezustand ist und anschließend wieder Zeichen übertragen werden.
- **Zeichenkette (Sequenz):** Der Beginn einer Nachricht wird erkannt, wenn eine festgelegte Zeichenfolge im Datenstrom auftritt. Sie können bis zu vier Zeichensequenzen mit jeweils bis zu fünf Zeichen festlegen.  
 Beispiel: Eine empfangene hexadezimal-codierte Meldung enthält die folgenden Zeichen: "68 10 aa 68 bb 10 aa 16". Die konfigurierten Startzeichenfolgen finden Sie in der folgenden Tabelle. Startzeichenfolgen werden ausgewertet, nachdem das erste Zeichen 68H erfolgreich empfangen wurde. Nach erfolgreichem Empfang des vierten Zeichens (das zweite 68H) ist die Startbedingung "1" erfüllt. Wenn die Startbedingungen erfüllt sind, beginnt die Auswertung der Endbedingungen.  
 Die Verarbeitung der Startzeichenfolge kann aufgrund verschiedener Fehler bei Parität, Framing oder Zeitabständen zwischen den Zeichen beendet werden. Diese Fehler führen dazu, dass die Meldung nicht empfangen wird, weil die Startbedingung nicht erfüllt wurde.

| Startbedingung | Erstes Zeichen | Erstes Zeichen +1 | Erstes Zeichen +2 | Erstes Zeichen +3 | Erstes Zeichen +4 |
|----------------|----------------|-------------------|-------------------|-------------------|-------------------|
| 1              | 68H            | xx                | xx                | 68H               | xx                |
| 2              | 10H            | aaH               | xx                | xx                | xx                |
| 3              | dcH            | aaH               | xx                | xx                | xx                |
| 4              | e5H            | xx                | xx                | xx                | xx                |

### Endbedingungen für den Nachrichteneingang (Parameter ENDCOND)

Das Ende einer Nachricht wird vom Empfänger erkannt, wenn eine projektierte Endbedingung zutrifft. Die folgenden Bedingungen können als Endbedingungen für den Nachrichteneingang festgelegt werden:

- **Antwort-Zeitüberschreitung:** Der Nachrichteneingang wird beendet, wenn die vorgegebene Maximaldauer für den Empfang eines Zeichens überschritten wird. Die Maximaldauer wird am Parameter RCVTIME definiert. Der Ablauf der definierten Zeit startet, sobald die letzte Übertragung beendet ist und die Anweisung RCV\_PTP den Empfang der Nachricht freigibt. Wenn innerhalb der definierten Zeit (RCVTIME) kein Zeichen empfangen wurde, meldet die Anweisung RCV\_PTP einen Fehler.
- **Nachrichten-Zeitüberschreitung:** Der Nachrichteneingang wird beendet, wenn die vorgegebene Maximaldauer für den Empfang einer Nachricht überschritten wird. Die Maximaldauer wird am Parameter MSGTIME definiert. Der Ablauf der definierten Zeit wird gestartet, sobald das erste Zeichen der Nachricht empfangen wurde.
- **Zeitüberschreitung innerhalb der Zeichenkette:** Der Nachrichteneingang wird beendet, wenn der Zeitabstand beim Empfang von zwei aufeinander folgenden Zeichen größer als der Wert am Parameter CHARGAP ist.
- **Maximale Länge:** Der Nachrichteneingang wird beendet, wenn die am Parameter MAXLEN definierte Länge der Nachricht überschritten wird.
- **Nachrichtenlänge auslesen (N+LEN+M):** Der Nachrichteneingang wird beendet, wenn eine bestimmte Nachrichtenlänge erreicht wird. Diese Länge wird durch die Werte der folgenden Parameter errechnet:
  - N: Position des Zeichens in der Nachricht, ab dem das Längenfeld beginnt.
  - LENGTHSIZE: Größe des Längenfeldes in Bytes
  - LENGTHM: Anzahl der Endzeichen, die dem Längenfeld folgen. Diese Zeichen werden bei der Auswertung der Nachrichtenlänge nicht berücksichtigt.
- **Zeichenkette:** Der Nachrichteneingang wird beendet, wenn eine festgelegte Zeichenfolge empfangen wird. Die Zeichenkette kann maximal fünf Zeichen enthalten. Für jedes Zeichen der Zeichenkette kann über die Bitposition festgelegt werden, ob dieses bei der Auswertung berücksichtigt oder ignoriert wird.

## Parameter STATUS

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 80C0  | Fehler in der Startbedingung  |
| 80C1  | <ul style="list-style-type: none"> <li>Fehler in der Endbedingung</li> <li>Keine Endbedingung definiert</li> </ul>  |
| 80C2  | Empfangsalarm aktiviert   |
| 80C3  | Bei eingestellter Endbedingung "Maximale Länge" wurde am Parameter MAXLEN ein Wert eingegeben, der gleich 0 oder größer als 4132 ist.                                 |
| 80C4  | Bei eingestellter Endbedingung "N+LEN+M" wurde am Parameter N ein Wert eingegeben, der größer als 4131 ist.   |
| 80C5  | Bei eingestellter Endbedingung "N+LEN+M" wurde am Parameter LENGTHSIZE ein Wert eingegeben, der gleich 0 oder ungültig ist.   |
| 80C6  | Bei eingestellter Endbedingung "N+LEN+M" wurde am Parameter LENGTHM ein Wert eingegeben, der größer als 255 ist.  |
| 80C7  | Bei eingestellter Endbedingung "N+LEN+M" wurde eine Nachrichtenlänge errechnet, die größer als 4132 ist.  |
| 80C8  | Bei eingestellter Endbedingung "Antwort-Zeitüberschreitung" wurde am Parameter RCVTIME ein Wert eingegeben, der gleich 0 ist.   |
| 80C9  | Bei eingestellter Endbedingung "Zeitüberschreitung innerhalb der Zeichenkette" wurde am Parameter CHARGAP ein Wert eingegeben, der gleich 0 oder größer als 2500 ist. |
| 80CA  | Bei eingestellter Startbedingung "Idle line" wurde am Parameter IDLETIME ein Wert eingegeben, der gleich 0 oder größer als 2500 ist.                                  |
| 80CB  | Trotz eingestellter Endbedingung "Zeichenkette" sind alle Zeichen der Zeichenkette als irrelevant gekennzeichnet.   |
| 80CC  | Trotz eingestellter Startbedingung "Zeichenkette" sind alle Zeichen der Zeichenkette als irrelevant gekennzeichnet.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 2846)".

## SEND\_PTP: Daten des Sendepuffers übertragen

### Beschreibung

Mit der Anweisung "SEND\_PTP" stoßen Sie die Übertragung von Daten an. Die Anweisung "SEND\_PTP" führt dabei nicht die eigentliche Übertragung der Daten durch. Die Daten des Sendepuffers werden an das Punkt-zu-Punkt-Kommunikationsmodul (CM) übertragen. Das CM führt dann die eigentliche Übertragung durch.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "SEND\_PTP":

| Parameter | Deklaration | Datentyp    | Speicherbereich              | Beschreibung   |
|-----------|-------------|-------------|------------------------------|--|
| REQ       | Input       | BOOL        | E, A, M, D, L                | Freigabe der angeforderten Übertragung bei steigender Flanke dieses Freigabeeingangs. Dadurch wird der Inhalt des Puffers zum Punkt-zu-Punkt-Kommunikationsmodul (CM) übertragen.  |
| PORT      | Input       | PORT (UINT) | D, L oder Konstante          | Kennung des Kommunikationsports (HW-ID)  |
| BUFFER    | Input       | VARIANT     | E, A, M, D, L oder Konstante | Zeiger auf die Anfangsadresse des Sendepuffers. Boolesche Werte oder Array of BOOL werden nicht unterstützt.   |
| LENGTH    | Input       | UINT        | E, A, M, D, L oder Konstante | Länge des Sendepuffers   |
| PTRCL     | Input       | BOOL        | E, A, M, D, L oder Konstante | Dieser Parameter wählt den Puffer für die normale Punkt-zu-Punkt-Kommunikation oder für spezifische Siemens-Protokolle aus, die im angeschlossenen CM implementiert sind.<br>FALSE = vom Anwenderprogramm gesteuerte Punkt-zu-Punkt-Operationen (einzige gültige Option) |
| DONE      | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>   |
| ERROR     | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>  |
| STATUS    | Output      | WORD        | E, A, M, D, L                | Status der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter STATUS**

| Fehlercode* (W#16#...) | Beschreibung  |
|------------------------|---|
| 7000                   | Die Sendeoperation ist nicht aktiv.   |
| 7001                   | Die Sendeoperation verarbeitet den ersten Aufruf.   |
| 7002                   | Die Sendeoperation verarbeitet nachfolgende Aufrufe (Abfragen nach dem ersten Aufruf).                                      |
| 8080                   | Die eingegebene Kennung des Kommunikationsports ist ungültig.   |
| 8088                   | Die Länge des Parameters LENGHT entspricht nicht der Länge der zu sendenden Daten. Siehe auch: Parameter LENGHT und BUFFER. |
| 80D0                   | Eine neue Sendeoperation wurde während einer laufenden Übertragung empfangen.   |
| 80D1                   | Die Übertragung wurde abgebrochen, weil das CTS-Signal innerhalb der angegebenen Wartezeit nicht bestätigt wurde.           |

| Fehlercode*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 80D2  | Die Sendeanforderung wurde abgebrochen, da der Kommunikationspartner (DCE) keine Empfangsbereitschaft (DSR) signalisiert. |
| 80D3  | Die Sendeanforderung wurde abgebrochen, da die maximale Größe der Warteschleife überschritten wurde (mehr als 1024 Byte). |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 2846)".

### Parameter LENGTH und BUFFER

Die Mindestdatengröße, die von der Anweisung "PTP\_SEND" gesendet werden kann, ist ein Byte. Der Parameter BUFFER legt die Größe der zu sendenden Daten fest. Sie können für den Parameter BUFFER weder den Datentyp BOOL noch Array of BOOL verwenden.

| Parameter LENGTH | Parameter BUFFER     | Beschreibung   |
|------------------|----------------------|--|
| LENGTH = 0       | Nicht verwendet      | Die vollständigen Daten werden wie vom Parameter BUFFER definiert gesendet. Ist LENGTH = 0, brauchen Sie die Anzahl der übertragenen Bytes nicht anzugeben.  |
| LENGTH > 0       | Elementarer Datentyp | Der LENGTH-Wert muss die Bytezahl dieses Datentyps enthalten. Andernfalls erfolgt keine Übertragung, und der Fehler 8088 wird ausgegeben.  |
|                  | STRUCT               | Der LENGTH-Wert kann eine Bytezahl kleiner als die vollständige Bytelänge der Struktur enthalten. In diesem Fall werden nur die ersten LENGTH-Bytes übertragen.  |
|                  | ARRAY                | Der LENGTH-Wert kann eine Bytezahl kleiner als die vollständige Bytelänge des Felds enthalten. In diesem Fall werden nur die Feldelemente übertragen, die vollständig in die LENGTH-Bytes passen.<br>Der LENGTH-Wert muss ein Vielfaches der Bytezahl der Datenelemente sein. Andernfalls ist STATUS = 8088, ERROR = 1 und keine Übertragung erfolgt.  |
|                  | STRING               | Die vollständige Speicheranordnung des Zeichenkettenformats wird übertragen, sowie die die Information über maximale Länge der Zeichenkette und aktuelle Länge der Zeichenkette.<br>Der LENGTH-Wert muss Bytes für maximale Länge, tatsächliche Länge und die Zeichen der Zeichenkette enthalten.<br>Beim Datentyp STRING haben alle Längen und Zeichen die Größe eines Bytes.<br>Wird für den Parameter BUFFER eine Zeichenkette verwendet, muss der LENGTH-Wert auch zwei Bytes für die beiden Längenfelder enthalten. |

## RCV\_PTP: Empfang von Nachrichten freigeben

### Beschreibung

Mit der Anweisung "RCV\_PTP" geben Sie den Empfang einer gesendeten Nachricht frei. Dabei muss jede Nachricht einzeln freigegeben werden. Die gesendeten Daten stehen im Empfangsbereich erst dann zur Verfügung, wenn die Nachricht vom entsprechenden Kommunikationspartner quittiert wurde.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCV\_PTP":

| Parameter | Deklaration | Datentyp    | Speicherbereich              | Beschreibung   |
|-----------|-------------|-------------|------------------------------|--|
| EN_R      | Input       | BOOL        | E, A, M, D, L                | Freigabe des Empfangs bei einer steigenden Flanke  |
| PORT      | Input       | PORT (UINT) | D, L oder Konstante          | Kennung des Kommunikationsports (HW-ID)  |
| BUFFER    | Input       | VARIANT     | E, A, M, D, L oder Konstante | Zeigt auf die Anfangsadresse des Empfangspuffers. Verwenden Sie in dem Empfangspuffer keine Variable vom Typ STRING.   |
| NDR       | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul> |
| ERROR     | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>  |
| STATUS    | Output      | WORD        | E, A, M, D, L                | Status der Anweisung   |
| LENGTH    | Output      | UINT        | E, A, M, D, L                | Länge der Nachricht im Empfangspuffer  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

| Fehlercode* (W#16#....) | Beschreibung  |
|-------------------------|---|
| 80E0                    | Der Nachrichtenempfang wurde beendet, weil der Empfangspuffer voll ist.   |
| 80E1                    | Der Nachrichtenempfang wurde aufgrund eines Paritätsfehlers beendet.  |
| 80E2                    | Der Nachrichtenempfang wurde aufgrund eines Framing-Fehlers beendet.  |
| 80E3                    | Der Nachrichtenempfang wurde aufgrund eines Überlauffehlers beendet.  |
| 80E4                    | Der Nachrichtenempfang wurde beendet, weil die errechnete Nachrichtenlänge (N+LEN+M) die Größe des Empfangspuffers überschreitet. |
| 8080                    | Die eingegebene Kennung des Kommunikationsports ist ungültig.   |
| 8088                    | Über den Parameter BUFFER wurde auf einen Datentyp STRING referenziert.   |



| Fehlercode*<br>(W#16#...) | Beschreibung   |
|---------------------------|--|
| 0094                      | Der Nachrichtenempfang wurde beendet, weil die maximale Zeichenlänge empfangen wurde.                    |
| 0095                      | Der Nachrichtenempfang wurde aufgrund einer Zeitüberschreitung beendet.                                  |
| 0096                      | Der Nachrichtenempfang wurde aufgrund einer Zeitüberschreitung innerhalb der Zeichenkette beendet.       |
| 0097                      | Der Nachrichtenempfang wurde aufgrund einer Antwort-Zeitüberschreitung beendet.                          |
| 0098                      | Der Nachrichtenempfang wurde beendet, weil die Längenbedingung "N+LEN+M" erfüllt wurde.                  |
| 0099                      | Der Nachrichtenempfang wurde beendet, weil die als Endbedingung definierte Zeichenkette empfangen wurde. |

\* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 2846)".

## RCV\_RST: Empfangspuffer löschen

### Beschreibung

Mit der Anweisung "RCV\_RST" löschen Sie den Empfangspuffer eines Kommunikationspartners.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "RCV\_RST":

| Parameter | Deklaration | Datentyp    | Speicherbereich     | Beschreibung  |
|-----------|-------------|-------------|---------------------|---|
| REQ       | Input       | BOOL        | E, A, M, D, L       | Freigabe zum Löschen des Empfangspuffers bei einer steigenden Flanke  |
| PORT      | Input       | PORT (UINT) | D, L oder Konstante | Kennung des Kommunikationsports (HW-ID)   |
| DONE      | Output      | BOOL        | E, A, M, D, L       | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>1: Auftrag fehlerfrei ausgeführt</li> </ul>              |
| ERROR     | Output      | BOOL        | E, A, M, D, L       | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>0: Kein Fehler</li> <li>1: Fehler aufgetreten</li> </ul>   |
| STATUS    | Output      | WORD        | E, A, M, D, L       | Status der Anweisung<br>Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 2846)". |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## SGN\_GET: RS-232-Signale abfragen

### Beschreibung

Mit der Anweisung "SGN\_GET" fragen Sie den aktuellen Zustand mehrerer Signale einer RS-232-Kommunikationsbaugruppe ab.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SGN\_GET":

| Parameter | Deklaration | Datentyp    | Speicherbereich              | Beschreibung  |
|-----------|-------------|-------------|------------------------------|---|
| REQ       | Input       | BOOL        | E, A, M, D, L oder Konstante | Freigabe der Abfrage bei einer steigenden Flanke  |
| PORT      | Input       | PORT (UINT) | D, L oder Konstante          | Kennung des Kommunikationsports (HW-ID)   |
| NDR       | Output      | BOOL        | E, A, M, D, L                | Wird für einen Zyklus gesetzt, wenn neue Daten bereit stehen und die Anweisung fehlerfrei ausgeführt wurde.                               |
| DTR       | Output      | BOOL        | E, A, M, D, L                | Datenendgerät bereit, Baugruppe bereit  |
| DSR       | Output      | BOOL        | E, A, M, D, L                | Datensatz bereit, Kommunikationspartner bereit  |
| RTS       | Output      | BOOL        | E, A, M, D, L                | Sendeanforderung, Baugruppe sendebereit   |
| CTS       | Output      | BOOL        | E, A, M, D, L                | Sendebereit, Kommunikationspartner kann Daten empfangen (Reaktion auf RTS = EIN der Baugruppe).   |
| DCD       | Output      | BOOL        | E, A, M, D, L                | Datenträgererkennung, Empfangssignalpegel   |
| RING      | Output      | BOOL        | E, A, M, D, L                | Klingelanzeige, Anzeige eines eingehenden Anrufs  |
| ERROR     | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul> |
| STATUS    | Output      | WORD        | E, A, M, D, L                | Status der Anweisung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

| Fehlercode*<br>(W#16#....)  | Beschreibung   |
|---|--|
| 80F0  | Die Kommunikationsbaugruppe ist eine RS485-Baugruppe, es stehen keine Signale zur Verfügung. |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 2846)".

## SGN\_SET: RS-232-Signale setzen

### Beschreibung

Mit der Anweisung "SGN\_SET" setzen Sie den Zustand der Ausgangssignale einer RS-232-Kommunikationsbaugruppe.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "SGN\_SET":

| Parameter | Deklaration | Datentyp    | Speicherbereich              | Beschreibung  |
|-----------|-------------|-------------|------------------------------|---|
| REQ       | Input       | BOOL        | E, A, M, D, L oder Konstante | Aktiviert den Vorgang bei einer steigenden Flanke<br>Initialwert: FALSE   |
| PORT      | Input       | PORT (UINT) | D, L oder Konstante          | Kennung des Kommunikationsports (HW-ID)<br>Initialwert: 0   |
| SIGNAL    | Input       | BYTE        | E, A, M, D, L oder Konstante | Gib die Signale an, die gesetzt werden: <ul style="list-style-type: none"> <li>• 01H = RTS setzen</li> <li>• 02H = DTR setzen</li> <li>• 04H = DSR setzen</li> </ul> Initialwert: FALSE                           |
| RTS       | Input       | BOOL        | E, A, M, D, L oder Konstante | Sendeanforderung, Baugruppe sendebereit<br>Initialwert: FALSE   |
| DTR       | Input       | BOOL        | E, A, M, D, L oder Konstante | Datenendgerät bereit, Baugruppe bereit<br>Initialwert: FALSE  |
| DSR       | Input       | BOOL        | E, A, M, D, L oder Konstante | Datensatz bereit (gilt nur für Schnittstellen vom Typ DCE)<br>Initialwert: FALSE  |
| DONE      | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul> Initialwert: FALSE |
| ERROR     | Output      | BOOL        | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul> Initialwert: FALSE  |
| STATUS    | Output      | WORD        | E, A, M, D, L                | Status der Anweisung<br>Initialwert: 0  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

| Fehlercode*<br>(W#16#....)  | Beschreibung   |
|---|--|
| 80F0  | Die Kommunikationsbaugruppe ist eine RS485-Baugruppe, es stehen keine Signale zur Verfügung. |
| 80F1  | Es können keine Signale gesetzt werden, da die Hardware-Flusskontrolle aktiviert ist.        |
| 80F2  | Das DSR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DTE ist.                  |
| 80F3  | Das DTR-Signal kann nicht gesetzt werden, da die Baugruppe vom Typ DCE ist.                  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

Weitere Informationen zu den allgemeinen Fehlercodes der Kommunikationsanweisungen finden Sie unter: "Allgemeine Statusinformationen der Kommunikationsbausteine (Seite 2846)".

## Allgemeine Statusinformationen der Kommunikationsbausteine

### Allgemeine Informationen zum Ausführungsstatus der Kommunikationsbausteine

Die folgende Tabelle zeigt, welche allgemeinen Informationen am Parameter STATUS der Kommunikationsbausteine ausgegeben werden können:

| Fehlercode*<br>(W#16#....)  | Beschreibung  |
|---|---|
| 8070  | Alle internen Instanz-Speicher sind im Einsatz  |
| 8080  | Die eingegebene Kennung des Kommunikationsports ist ungültig  |
| 8081  | Zeitüberschreitung, Baugruppenfehler, interner Fehler   |
| 8085  | Fehler bei der Längenangabe am Parameter LENGHT. Die angegebene Länge ist "0" oder größer als der maximal zulässige Wert. |
| 8090  | Länge der Nachricht ungültig, Baugruppe ungültig, Nachricht ungültig  |
| 8091  | Fehlerhafter Typ in der Parametrierungsnachricht  |
| 8092  | Datensatzlänge in der Parametrierungsnachricht ungültig   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## USS

### Überblick über die USS-Anweisungen

#### Einleitung

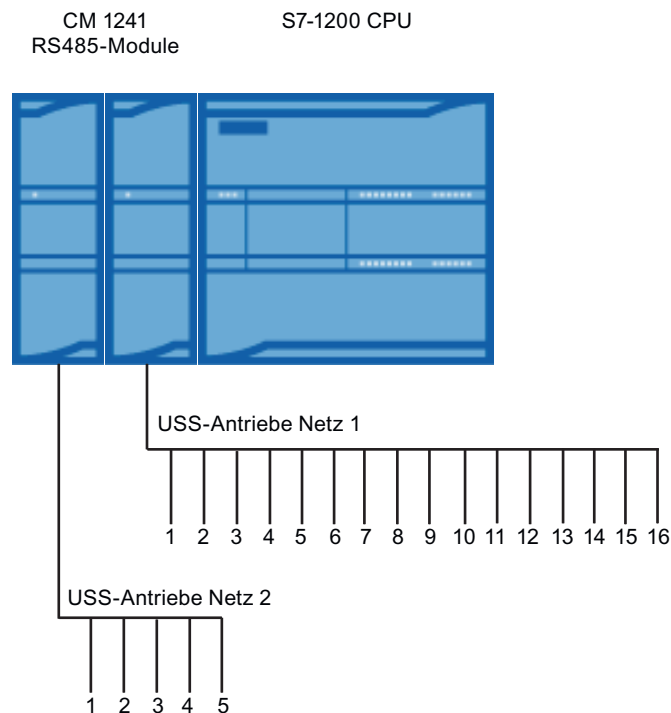
Die USS-Anweisungen steuern den Betrieb von Antrieben, die das Protokoll der universellen seriellen Schnittstelle (USS) unterstützen. Mit den USS-Anweisungen können Sie über eine RS-485-Verbindung mit mehreren Antrieben kommunizieren.

Hierzu benötigen Sie ein CM 1241 RS485-Kommunikationsmodul oder ein CB 1241 RS485-Kommunikationsboard. In einer S7-1200 CPU können bis zu drei CM 1241 RS485-Module und ein CB 1241 RS485-Board eingebaut werden.

Jeder RS485-Port kann bis zu sechzehn Antriebe betreiben.

Das USS-Protokoll nutzt ein Master/Slave-Netzwerk für die Kommunikation über einen seriellen Bus. Der Master verwendet einen Adressparameter, um eine Meldung an einen ausgewählten Slave zu senden. Ein Slave selbst kann niemals senden, ohne dafür zuvor eine Anforderung zu erhalten. Die direkte Meldungsübertragung zwischen den einzelnen Slaves ist nicht möglich. Die USS-Kommunikation funktioniert im Halbduplex-Betrieb.

Das folgende Bild zeigt ein Beispiel eines USS-Netzwerkdigramms:



## Voraussetzungen für den Einsatz des USS-Protokolls

### Allgemeine Voraussetzungen für die Antriebseinrichtung

- Für die Antriebe muss die Verwendung von 4 PKW-Wörtern eingerichtet werden.
- Die Antriebe können für 2, 4, 6 oder 8 PZD-Wörter konfiguriert werden.
- Die Anzahl der PZD-Wörter im Antrieb muss dem Eingang PZD\_LEN der Anweisung "USS\_DRIVE (Seite 2852)" des Antriebs entsprechen.
- Die Baudrate aller Antriebe muss dem der Baudrate am Eingangsparameter BAUD der Anweisung "USS\_PORT (Seite 2851)" entsprechen.
- Der Antrieb muss für die Fernsteuerung eingerichtet werden.

- Für den Frequenzsollwert an der COM-Verbindung des Antriebs muss USS festgelegt werden.
- Für die Antriebsadresse muss 1 bis 16 festgelegt sein. Diese Adresse muss der Adresse am Eingangsparameter DRIVE der Anweisung "USS\_DRIVE (Seite 2852)" entsprechen.
- Für die Richtungssteuerung des Antriebs muss die Verwendung der Polarität des Antriebssollwerts eingerichtet werden.
- Das RS485-Netzwerk muss ordnungsgemäß abgeschlossen sein.

### Definition: PKW- / PZD-Bereich

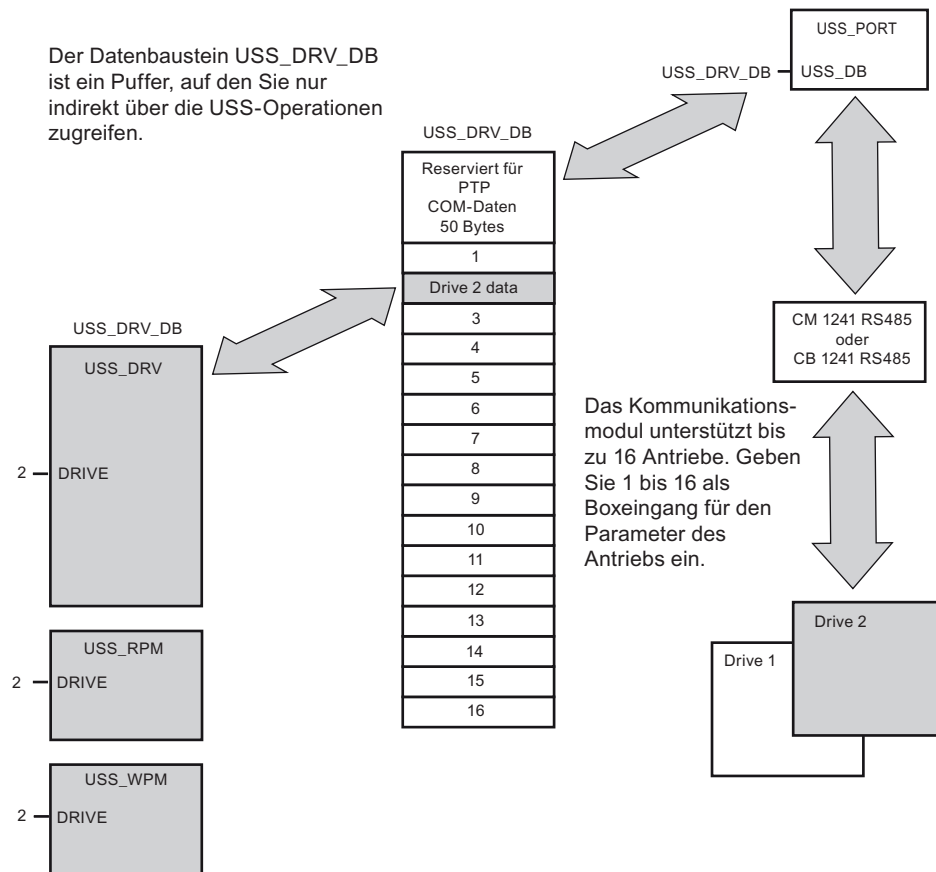
- Der PKW-Bereich bezieht sich dabei auf das Handling der Parameter-Kennung-Wert (PKW)-Schnittstelle.  
Die PKW-Schnittstelle ist keine physikalische Schnittstelle, sondern es wird damit ein Mechanismus beschrieben, der den Parameteraustausch zwischen zwei Kommunikationspartnern regelt. D. h., Lesen und Schreiben von Parameterwerten, Parameterbeschreibungen und zugehörige Texten sowie die Behandlung von Parameteränderung durch Spontanmeldungen. Alle Aufgaben, die über die PKW-Schnittstelle erfolgen, sind im wesentlichen Aufgaben für Bedienen und Beobachten, Service und Diagnose.
- Der PZD-Bereich beinhaltet die für die Automatisierung notwendigen Signale:
  - Steuerwort(e) und Sollwert(e) vom Master zum Slave
  - Zustandswort(e) und Istwert(e) vom Slave zum Master.

Beide Bereiche zusammen ergeben den Nutzdatenblock. Dieser wird als Auftragstelegramm vom Master an den Slave übertragen oder als Antworttelegramm vom Slave an den Master.

### Beschreibung


Jedes Kommunikationsmodul CM 1241 RS485 unterstützt maximal 16 Antriebe. Ein einziger Instanz-Datenbaustein enthält temporäre Speicher- und Pufferfunktionen für alle Antriebe im USS-Netzwerk, die mit einem von Ihnen installierten PtP-Kommunikationsmodul verbunden

sind. Die USS-Anweisungen für diese Antriebe haben gemeinsam Zugriff auf die Informationen in diesem Datenbaustein.



- Alle Antriebe (max. 16), die an einen RS485 Port angeschlossen sind, sind Teil desselben USS-Netzwerks. Alle Antriebe, die an einen anderen RS485 Port angeschlossen sind, sind Teil eines anderen USS-Netzwerks. Weil die S7-1200 bis zu drei Module CM 1241 RS485 unterstützt, können Sie bis zu drei USS-Netzwerke einrichten, jeweils mit maximal 16 Antrieben in jedem Netzwerk, so dass insgesamt 48 USS-Antriebe unterstützt werden.
- Jedes USS-Netzwerk wird über einen eindeutigen Datenbaustein verwaltet (für drei USS-Netzwerke mit drei Modulen CM 1241 RS485 sind drei Datenbausteine erforderlich). Alle Anweisungen, die zu einem USS-Netzwerk gehören, müssen diesen Datenbaustein gemeinsam nutzen. Dies umfasst alle Anweisungen "USS\_DRIVE (Seite 2852)", "USS\_PORT (Seite 2851)", "USS\_RPM (Seite 2855)" und "USS\_WPM (Seite 2856)" für die Steuerung aller Antriebe in einem USS-Netzwerk.

- Die Anweisung "USS\_DRIVE (Seite 2852)" ist ein Funktionsbaustein (FB). Wenn Sie die Anweisung "USS\_DRIVE" im Editor einfügen, werden Sie im Dialog "Aufrufoptionen" aufgefordert, der Anweisung einen DB zuzuweisen.
  - Wenn es sich um die erste Anweisung "USS\_DRIVE" in diesem Programm für dieses USS-Netzwerk handelt, können Sie die DB-Standardzuweisung übernehmen (oder ggf. den Namen ändern), und der neue DB wird erstellt.
  - Wenn es sich jedoch nicht um die erste Anweisung "USS\_DRIVE" für dieses Netzwerk handelt, müssen Sie im Dialog "Aufrufoptionen" in der Klappliste den entsprechenden DB auswählen, der diesem USS-Netzwerk bereits zuvor zugewiesen wurde.
- Bei allen Anweisungen "USS\_PORT (Seite 2851) , USS\_RPM (Seite 2855)" und "USS\_WPM (Seite 2856)" handelt es sich um Funktionen (FCs). Wenn Sie diese FCs im Editor einfügen, wird kein DB zugewiesen. Stattdessen müssen Sie dem Eingang USS\_DB dieser Anweisungen den jeweiligen DB zuweisen (doppelklicken Sie auf das Parameterfeld und dann auf das Symbol, um die verfügbaren DBs anzuzeigen).
- Die Anweisung "USS\_PORT (Seite 2851)" steuert die Kommunikation zwischen der CPU und den Antrieben über das PtP-Kommunikationsmodul. Bei jedem Aufruf dieser Anweisung wird eine Kommunikation mit einem Antrieb bearbeitet. Ihr Programm muss diese Funktion schnell genug aufrufen, so dass die Antriebe keine Zeitüberschreitung melden. Die Anweisung kann aus dem Hauptprogramm oder einem beliebigen Alarm-OB aufgerufen werden.
- Der Funktionsbaustein "USS\_DRIVE (Seite 2852)" gibt Ihrem Programm Zugriff auf einen angegebenen Antrieb im USS-Netzwerk. Seine Ein- und Ausgänge entsprechen den Zuständen und den Bedienfunktionen des Antriebs. Sind 16 Antriebe im Netzwerk vorhanden, so muss "USS\_DRIVE" in Ihrem Programm mindestens 16mal aufgerufen werden, also jeweils einmal für jeden Antrieb. Wie schnell diese Bausteine aufgerufen werden, hängt von der erforderlichen Geschwindigkeit für die Steuerung der Antriebsfunktionen ab.  
Sie können die Anweisung "USS\_DRIVE" nur aus dem Hauptprogramm-OB aufrufen.

 **VORSICHT**

Rufen Sie "USS\_DRIVE", "USS\_RPM", "USS\_WPM" nur aus dem Hauptprogramm-OB auf. Die Anweisung "USS\_PORT" kann aus einem beliebigen OB aufgerufen werden, üblicherweise wird sie aus einem Verzögerungsalarm-OB aufgerufen. Wird die Anweisung "USS\_PORT" während der Ausführung unterbrochen, kann es zu unerwarteten Fehlern kommen.

Mit den Anweisungen "USS\_RPM" und "USS\_WPM" werden die Betriebsparameter des Antriebs gelesen und geschrieben. Diese Parameter steuern die interne Funktionsweise des Antriebs. Eine Definition dieser Parameter finden Sie im Handbuch des Antriebs.

Ihr Programm kann eine beliebige Anzahl dieser Anweisungen enthalten, es kann jedoch immer nur eine Lese- oder Schreibanforderung für einen Antrieb aktiv sein. Sie dürfen die Anweisungen "USS\_RPM" und "USS\_WPM" nur aus einem Hauptprogramm-OB aufrufen.

### Zeit für die Kommunikation mit dem Antrieb berechnen

Die Kommunikation mit dem Antrieb läuft asynchron zum Zyklus der S7-1200 ab. Die S7-1200 durchläuft üblicherweise mehrere Zyklen, bevor die Kommunikation mit einem Antrieb beendet ist.



Das Intervall von "USS\_PORT" ist die Zeit, die für eine Transaktion des Antriebs erforderlich ist. Die folgende Tabelle zeigt die Mindestintervalle für "USS\_PORT" für jede Baudrate. Wenn Sie die Anweisung "USS\_PORT" häufiger aufrufen, als es das "USS\_PORT"-Intervall vorgibt, wird die Anzahl der Transaktionen nicht erhöht. Das Timeout-Intervall des Antriebs ist die Zeitdauer, die für eine Transaktion zur Verfügung steht, wenn zur Fertigstellung der Transaktion aufgrund von Kommunikationsfehlern 3 Versuche nötig sind. Standardmäßig werden beim USS-Protokoll bei jeder Transaktion bis zu 2 Wiederholungen durchgeführt.

| Baudrate | Berechnetes Mindestintervall für Aufruf von USS_PORT (ms) | Intervall-Timeout für Antriebsmeldung pro Antrieb (ms) |
|----------|---|--|
| 1200     | 790   | 2370   |
| 2400     | 405   | 1215   |
| 4800     | 212,5   | 638  |
| 9600     | 116,3   | 349  |
| 19200    | 68,2  | 205  |
| 38400    | 44,1  | 133  |
| 57600    | 36,1  | 109  |
| 115200   | 28,1  | 85   |

## USS\_PORT: Kommunikation über USS-Netzwerk bearbeiten

### Beschreibung

Die Anweisung "USS\_PORT" bearbeitet die Kommunikation über das USS-Netzwerk. Verwenden Sie im Programm jeweils eine Anweisung "USS\_PORT" pro PtP-Kommunikationsport, um die Übertragung zu oder von jeweils einem Antrieb zu steuern.

Alle USS-Anweisungen, die einem USS-Netzwerk und einem PtP-Kommunikationsport zugewiesen sind, müssen den gleichen Instanz-Datenbaustein nutzen.

### Aufruf

Ihr Programm muss die Anweisung "USS\_PORT" oft genug ausführen, damit keine Zeitüberschreitungen im Antrieb auftreten. Rufen Sie daher die Anweisung "USS\_PORT" aus einem Weckalarm-OB auf, um Zeitüberschreitungen beim Antrieb zu verhindern und die letzten USS-Datenaktualisierungen für Aufrufe von "USS\_DRIVE (Seite 2852)" verfügbar zu haben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USS\_PORT":

| Parameter              | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|------------------------|-------------|----------|------------------------------|--|
| PORT                   | Input       | PORT     | D, L oder Konstante          | PtP-Kommunikationsport-Kennung<br>Konstante, auf die in der Standardvariablen-tabelle im Register "Konstanten" verwiesen werden kann.  |
| BAUD                   | Input       | DINT     | E, A, M, D, L oder Konstante | Baudrate für die USS-Kommunikation.  |
| USS_DB                 | InOut       | USS_BASE | D                            | Referenz auf den Instanz-DB der Anweisung "USS_DRIVE (Seite 2852)".  |
| ERROR                  | Output      | BOOL     | E, A, M, D, L                | Tritt ein Fehler auf, wird ERROR auf TRUE gesetzt. Am Ausgang STATUS wird ein entsprechender Fehlercode ausgegeben.  |
| STATUS<br>(Seite 2858) | Output      | WORD     | E, A, M, D, L                | Zustandswert der Anforderung. Er gibt das Ergebnis des Zyklus oder der Initialisierung an. Weitere Informationen stehen für einige Statuscodes in der Variablen "USS_Extended_Error (Seite 2858)" zur Verfügung. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## USS\_DRIVE: Daten mit dem Antrieb austauschen

### Beschreibung

Die Anweisung "USS\_DRIVE" tauscht Daten mit dem Antrieb aus, indem Anfragemeldungen erzeugt und die Antwortmeldungen des Antriebs ausgewertet werden. Für jeden Antrieb muss eine eigene Anweisung verwendet werden, doch müssen alle USS-Anweisungen, die einem USS-Netzwerk und einem PtP-Kommunikationsmodul zugewiesen sind, den gleichen Instanz-Datenbaustein verwenden. Sie müssen den DB-Namen eingeben, wenn Sie die erste Anweisung "USS\_DRIVE" einfügen. Dann nutzen Sie diesen DB weiter, der beim Einfügen der ersten Anweisung angelegt wurde.

Wenn die erste Ausführung von "USS\_DRIVE" erfolgt, wird der von der USS-Adresse (Parameter DRIVE) angegebene Antrieb im Instanz-DB initialisiert. Nach dieser Initialisierung können nachfolgende Anweisungen "USS\_PORT (Seite 2851)" die Kommunikation mit dem Antrieb an dieser Antriebsnummer beginnen.

Wenn Sie die Antriebsnummer ändern, muss das Zielsystem zunächst in STOP und dann wieder in RUN versetzt werden, damit der Instanz-DB initialisiert wird. Die Eingangsparameter werden im USS-Sendepuffer konfiguriert und die Ausgänge werden, sofern vorhanden, aus einem "vorherigen" gültigen Antwortpuffer gelesen. Während der Ausführung der Anweisung "USS\_DRIVE" findet keine Datenübertragung statt. Nach der Ausführung von "USS\_PORT (Seite 2851)" wird mit den Antrieben kommuniziert. "USS\_DRIVE" konfiguriert nur die zu sendenden Meldungen und wertet Daten aus, die in einer vorherigen Anforderung empfangen wurden.

Sie können die Drehrichtung des Antriebs entweder über den Eingang DIR (BOOL) oder über das Vorzeichen (positiv oder negativ) am Eingang SPEED\_SP (REAL) steuern. Die folgende

Tabelle erläutert, wie diese Eingänge zusammen funktionieren, um die Drehrichtung des Antriebs zu bestimmen, vorausgesetzt der Motor dreht vorwärts.

| SPEED_SP | DIR | Drehrichtung des Antriebs |
|----------|-----|---------------------------|
| Wert > 0 | 0   | Rückwärts                 |
| Wert > 0 | 1   | Vorwärts                  |
| Wert < 0 | 0   | Vorwärts                  |
| Wert < 0 | 1   | Rückwärts                 |

## Parameter

Erweitern Sie die Box, um alle Parameter anzuzeigen. Klicken Sie dazu auf den unteren Bereich der Box. Die Parameteranschlüsse, die grau dargestellt sind, sind optional und müssen nicht zugewiesen werden.

Die folgende Tabelle zeigt die Parameter der Anweisung "USS\_DRIVE":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| RUN       | Input       | BOOL     | E, A, M, D, L oder Konstante | Startbit des Antriebs: Hat dieser Parameter den Wert TRUE, so ermöglicht dieser Eingang den Betrieb des Antriebs mit der voreingestellten Drehzahl.  |
| OFF2      | Input       | BOOL     | E, A, M, D, L oder Konstante | Bit "Zum Stillstand auslaufen": Hat dieser Parameter den Wert FALSE, so veranlasst dieses Bit das Auslaufen des Antriebs, ohne zu bremsen.   |
| OFF3      | Input       | BOOL     | E, A, M, D, L oder Konstante | Schnelles Stoppbit - Hat dieser Parameter den Wert FALSE, so verursacht dieses Bit einen schnellen Halt durch Abbremsen des Antriebs.  |
| F_ACK     | Input       | BOOL     | E, A, M, D, L oder Konstante | Fehlerquittierungsbit - Mit diesem Bit wird das Fehlerbit eines Antriebs zurückgesetzt. Es wird nach dem Löschen des Fehlers gesetzt und der Antrieb erkennt damit, dass der vorherige Fehler nicht mehr gemeldet werden muss. |
| DIR       | Input       | BOOL     | E, A, M, D, L oder Konstante | Richtungssteuerung des Antriebs - Dieses Bit wird gesetzt, wenn der Antrieb in Vorwärtsrichtung laufen soll (wenn SPEED_SP positiv ist).   |
| DRIVE     | Input       | USINT    | E, A, M, D, L oder Konstante | Adresse des Antriebs: Dieser Eingang ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.  |
| PZD_LEN   | Input       | USINT    | E, A, M, D, L oder Konstante | Wortlänge - Dies ist die Anzahl der PZD-Datenwörter. Gültige Werte sind 2, 4, 6 oder 8 Wörter. Voreinstellung ist 2.   |
| SPEED_SP  | Input       | REAL     | E, A, M, D, L oder Konstante | Drehzahlsollwert - Dies ist die Drehzahl des Antriebs prozentual zur konfigurierten Frequenz. Ein positiver Wert bedeutet, dass der Antrieb vorwärts läuft (wenn DIR dem Wert TRUE hat).                                       |
| CTRL3     | Input       | WORD     | E, A, M, D, L oder Konstante | Steuerwort 3 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.   |

| Parameter              | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|------------------------|-------------|----------|------------------------------|--|
| CTRL4                  | Input       | WORD     | E, A, M, D, L oder Konstante | Steuerwort 4 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.   |
| CTRL5                  | Input       | WORD     | E, A, M, D, L oder Konstante | Steuerwort 5 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.   |
| CTRL6                  | Input       | WORD     | E, A, M, D, L oder Konstante | Steuerwort 6 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren.   |
| CTRL7                  | Input       | WORD     | E, A, M, D, L oder Konstante | Steuerwort 7 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.   |
| CTRL8                  | Input       | WORD     | E, A, M, D, L oder Konstante | Steuerwort 8 - Wert, der in einen benutzerkonfigurierbaren Parameter des Antriebs geschrieben wird. Der Benutzer muss dies im Antrieb konfigurieren. Optionaler Parameter.   |
| NDR                    | Output      | BOOL     | E, A, M, D, L                | Neue Daten bereit - Hat dieser Parameter den Wert TRUE, so meldet das Bit, dass am Ausgang Daten einer neuen Kommunikationsanforderung bereitstehen.   |
| ERROR                  | Output      | BOOL     | E, A, M, D, L                | Fehler aufgetreten - Hat dieser Parameter den Wert TRUE, so meldet dieses Bit, dass ein Fehler aufgetreten und Ausgang STATUS gültig ist. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung "USS_PORT" gemeldet. |
| STATUS<br>(Seite 2858) | Output      | WORD     | E, A, M, D, L                | Zustandswert der Anforderung. Er gibt das Zyklusergebnis an. Dies ist kein vom Antrieb ausgegebenes Zustandswort.  |
| RUN_EN                 | Output      | BOOL     | E, A, M, D, L                | Betrieb freigegeben - Dieses Bit meldet, ob der Antrieb läuft.   |
| D_DIR                  | Output      | BOOL     | E, A, M, D, L                | Antriebsrichtung - Dieses Bit meldet, ob der Antrieb vorwärts läuft.   |
| INHIBIT                | Output      | BOOL     | E, A, M, D, L                | Antrieb gesperrt - Dieses Bit meldet den Zustand des Sperrbits für den Antrieb.  |
| FAULT                  | Output      | BOOL     | E, A, M, D, L                | Antriebsfehler - Dieses Bit meldet, dass im Antrieb ein Fehler aufgetreten ist. Der Anwender muss die Störung beheben und Bit F_ACK setzen, um dieses Bit zu löschen.  |
| SPEED                  | Output      | REAL     | E, A, M, D, L                | Istwert Antriebsdrehzahl (skalierter Wert von Zustandswort 2 des Antriebs) - Der Wert der Drehzahl des Antriebs prozentual zur konfigurierten Drehzahl.  |
| STATUS1                | Output      | WORD     | E, A, M, D, L                | Zustandswort 1 des Antriebs - Dieser Wert enthält feste Zustandsbits eines Antriebs.   |
| STATUS3                | Output      | WORD     | E, A, M, D, L                | Zustandswort 3 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs.  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| STATUS4   | Output      | WORD     | E, A, M, D, L   | Zustandswort 4 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs. |
| STATUS5   | Output      | WORD     | E, A, M, D, L   | Zustandswort 5 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs. |
| STATUS6   | Output      | WORD     | E, A, M, D, L   | Zustandswort 6 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs. |
| STATUS7   | Output      | WORD     | E, A, M, D, L   | Zustandswort 7 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs. |
| STATUS8   | Output      | WORD     | E, A, M, D, L   | Zustandswort 8 des Antriebs - Dieser Wert enthält ein benutzerkonfigurierbares Zustandswort des Antriebs. |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## USS\_RPM: Parameter aus dem Antrieb auslesen

### Beschreibung

Die Anweisung "USS\_RPM" liest einen Parameter aus dem Antrieb. Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsmodul zugewiesen sind, müssen den gleichen Datenbaustein nutzen. "USS\_RPM" muss aus dem Hauptprogramm-OB aufgerufen werden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USS\_RPM":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| REQ       | Input       | BOOL     | E, A, M, D, L                | Sendeanforderung: Hat dieser Parameter den Wert TRUE, so wird eine neue Leseanforderung benötigt. Dies wird ignoriert, wenn die Anforderung für diesen Parameter bereits ansteht.   |
| DRIVE     | Input       | USINT    | E, A, M, D, L oder Konstante | Adresse des Antriebs: Dieser Eingang ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.   |
| PARAM     | Input       | UINT     | E, A, M, D, L oder Konstante | Parameternummer: Dieser Eingang gibt an, welcher Antriebsparameter geschrieben wird. Der Bereich dieses Parameters ist 0 bis 2047. Weitere Informationen für den Zugriff auf Parameter oberhalb dieses Bereichs finden Sie im Handbuch zu Ihrem Antrieb.  |
| INDEX     | Input       | UINT     | E, A, M, D, L oder Konstante | Parameterindex: Dieser Eingang gibt an, in welchen Antriebsparameterindex geschrieben werden soll. Es handelt sich um einen 16-Bit-Wert, bei dem das niederwertigste Byte der tatsächliche Indexwert ist, mit einem Bereich von (0 bis 255). Das höherwertige Byte kann ebenfalls von dem Antrieb verwendet werden und ist antriebsspezifisch. Weitere Informationen finden Sie im Handbuch Ihres Antriebs. |

| Parameter           | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|---------------------|-------------|----------|-----------------|---|
| USS_DB              | InOut       | USS_BASE | D               | Verweis auf den Instanz-DB, der erstellt und initialisiert wird, wenn eine Anweisung "USS_DRIVE" in Ihr Programm eingefügt wird.  |
| DONE                | Output      | BOOL     | E, A, M, D, L   | Hat dieser Parameter den Wert TRUE, so steht am Ausgang VALUE der zuvor angeforderte Wert des Leseparameters an.<br>Dieses Bit wird gesetzt, wenn die Anweisung "USS_DRIVE" die Leseantwort des Antriebs erkennt.<br>Dieses Bit wird zurückgesetzt, wenn: <ul style="list-style-type: none"> <li>• Sie die Antwortdaten über eine andere Abfrage "USS_RPM" anfordern oder</li> <li>• der zweite der nächsten beiden Aufrufe von "USS_DRIVE (Seite 2852)" ausgeführt wird</li> </ul> |
| ERROR               | Output      | BOOL     | E, A, M, D, L   | Fehler aufgetreten - Hat dieser Parameter den Wert TRUE, so meldet dieses Bit, dass ein Fehler aufgetreten und Ausgang STATUS gültig ist. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung "USS_PORT (Seite 2851)" gemeldet.   |
| STATUS (Seite 2858) | Output      | WORD     | E, A, M, D, L   | Dies ist der Zustandswert der Anforderung. Er gibt das Ergebnis der Leseanforderung an. Weitere Informationen stehen für einige Statuscodes in der Variablen "USS_Extended_Error (Seite 2858)" zur Verfügung.   |
| VALUE               | Output      | VARIANT  | E, A, M, D, L   | Dies ist der Wert des Parameters, der gelesen wurde und er ist nur gültig, wenn das Bit DONE den Wert TRUE hat.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## USS\_WPM: Parameter im Antrieb ändern

### Beschreibung

Die Anweisung "USS\_WPM" ändert einen Parameter im Antrieb. Alle USS-Funktionen, die einem USS-Netzwerk und einem PtP-Kommunikationsmodul zugewiesen sind, müssen den gleichen Datenbaustein nutzen. "USS\_WPM" muss aus dem Hauptprogramm-OB aufgerufen werden.

---

#### Hinweis

##### EEPROM-Schreiboperationen

Übertreiben Sie die Verwendung der EEPROM-Schreiboperation nicht. Halten Sie die Anzahl der EEPROM-Schreiboperationen möglichst gering, um die Lebensdauer des EEPROM zu verlängern.

---

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USS\_WPM":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| REQ       | Input       | BOOL     | E, A, M, D, L                | Sendeanforderung: Hat dieser Parameter den Wert TRUE, so wird eine neue Schreibanforderung benötigt. Dies wird ignoriert, wenn die Anforderung für diesen Parameter bereits ansteht.   |
| DRIVE     | Input       | USINT    | E, A, M, D, L oder Konstante | Adresse des Antriebs: Dieser Eingang ist die Adresse des USS-Antriebs. Der gültige Bereich liegt zwischen Antrieb 1 und Antrieb 16.  |
| PARAM     | Input       | UINT     | E, A, M, D, L oder Konstante | Parameternummer: Dieser Eingang gibt an, welcher Antriebsparameter geschrieben wird. Der Bereich dieses Parameters ist 0 bis 2047. Weitere Informationen für den Zugriff auf Parameter oberhalb dieses Bereichs finden Sie im Handbuch zu Ihrem Antrieb.   |
| INDEX     | Input       | UINT     | E, A, M, D, L oder Konstante | Parameterindex: Dieser Eingang gibt an, in welchen Antriebsparameterindex geschrieben werden soll. Es handelt sich um einen 16-Bit-Wert, bei dem das niederwertigste Byte der tatsächliche Indexwert ist, mit einem Bereich von (0 bis 255). Das höherwertige Byte kann ebenfalls von dem Antrieb verwendet werden und ist antriebsspezifisch. Weitere Informationen finden Sie im Handbuch Ihres Antriebs.  |
| EEPROM    | Input       | BOOL     | E, A, M, D, L oder Konstante | In EEPROM des Antriebs speichern: Hat dieser Parameter den Wert TRUE, so werden in den Antriebsparameter geschriebene Werte im EEPROM des Antriebs gespeichert. Hat dieser Parameter den Wert FALSE, so wird der geschriebene Wert nur temporär gespeichert und geht beim nächsten Einschalten des Antriebs verloren.  |
| VALUE     | Input       | VARIANT  | E, A, M, D, L oder Konstante | Wert des Parameters, in den geschrieben werden soll. Er muss beim Zustandswechsel von REQ gültig sein.   |
| USS_DB    | InOut       | USS_BASE | D                            | Dies ist ein Verweis auf den Instanz-DB, der erstellt und initialisiert wird, wenn eine Anweisung "USS_DRIVE (Seite 2852)" in Ihr Programm eingefügt wird.   |
| DONE      | Output      | BOOL     | E, A, M, D, L                | Hat dieser Parameter den Wert TRUE, wurde der Eingang VALUE in den Antrieb geschrieben.<br>Dieses Bit wird gesetzt, wenn die Anweisung "USS_DRIVE (Seite 2852)" die Schreibanantwort des Antriebs erkennt.<br>Dieses Bit wird zurückgesetzt, wenn:<br>Sie die Bestätigung des Antriebs über eine weitere Abfrage "USS_WPM" anfordern, dass der Schreibvorgang durchgeführt wurde, oder wenn der zweite der nächsten beiden Aufrufe von "USS_DRIVE (Seite 2852)" ausgeführt wird. |

| Parameter              | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|------------------------|-------------|----------|-----------------|--|
| ERROR                  | Output      | BOOL     | E, A, M, D, L   | Fehler aufgetreten: Hat dieser Parameter den Wert TRUE, ist ein Fehler aufgetreten und wird am Ausgang STATUS angezeigt. Alle anderen Ausgänge werden bei einem Fehler auf Null gesetzt. Kommunikationsfehler werden nur an den Ausgängen ERROR und STATUS der Anweisung "USS_PORT (Seite 2851)" gemeldet. |
| STATUS<br>(Seite 2858) | Output      | WORD     | E, A, M, D, L   | Dies ist der Zustandswert der Anforderung. Er gibt das Ergebnis der Schreibanforderung an. Weitere Informationen stehen für einige Statuscodes in der Variablen "USS_Extended_Error (Seite 2858)" zur Verfügung.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS der USS-Anweisungen

#### Parameter STATUS

Die folgende Tabelle enthält die Statuscodes der USS-Operation, die am Ausgang STATUS der USS-Anweisungen ausgegeben werden.

| STATUS*<br>(W#16#....) | Beschreibung  |
|------------------------|---|
| 0000                   | Kein Fehler   |
| 8180                   | Die Länge der Antwort des Antriebs entsprach nicht den vom Antrieb empfangenen Zeichen. Die Nummer des Antriebs, in dem der Fehler aufgetreten ist, wird in der Variablen "USS_Extended_Error" ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle. |
| 8181                   | Der Parameter VALUE ist nicht vom Datentyp WORD, REAL oder DWORD  |
| 8182                   | Der Anwender hat einen Parameterwert vom Typ Wort eingegeben und die Antwort vom Antrieb im Format DWORD oder REAL empfangen  |
| 8183                   | Der Anwender hat einen Parameterwert vom Typ DWORD oder REAL eingegeben und die Antwort vom Antrieb im Format Wort empfangen  |
| 8184                   | Das Antworttelegramm des Antriebs hatte eine falsche Prüfsumme. Die Nummer des Antriebs, in dem der Fehler aufgetreten ist, wird in der Variablen "USS_Extended_Error" ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.                         |
| 8185                   | Unzulässige Antriebsadresse (gültiger Adressbereich für Antriebe: 1-16)   |
| 8186                   | Drehzahlsollwert liegt außerhalb des gültigen Bereichs (gültiger Sollwertbereich für die Drehzahl: -200% bis 200%)  |
| 8187                   | Falsche Antriebsnummer hat auf die gesendete Anforderung geantwortet. Die Nummer des Antriebs, in dem der Fehler aufgetreten ist, wird in der Variablen "USS_Extended_Error" ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.                   |
| 8188                   | Unzulässige PZD-Wortlänge angegeben (gültiger Bereich = 2, 4, 6 oder 8 Wörter)  |
| 8189                   | Unzulässige Baudrate angegeben  |
| 818A                   | Anforderungskanal für Parameter wird von einer anderen Anforderung für diesen Antrieb verwendet   |
| 818B                   | Antrieb hat nicht auf Anforderungen und Wiederholungen reagiert. Die Nummer des Antriebs, in dem der Fehler aufgetreten ist, wird in der Variablen "USS_Extended_Error" ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.                        |



| STATUS*<br>(W#16#...)   | Beschreibung  |
|---|---|
| 818C  | Antrieb hat einen erweiterten Fehler zu einer Parameteranforderung ausgegeben. Die Beschreibung der erweiterten Fehler finden Sie unterhalb dieser Tabelle.   |
| 818D  | Antrieb hat einen Fehler wegen unzulässigen Zugriffs bei einer Parameteranforderung ausgegeben. Weitere Informationen dazu, weshalb der Parameterzugriff begrenzt sein kann, finden Sie im Handbuch zu Ihrem Antrieb.   |
| 818E  | Antrieb wurde nicht initialisiert: Dieser Fehlercode wird an "USS_RPM (Seite 2855)" oder "USS_WPM (Seite 2856)" ausgegeben, wenn die Anweisung "USS_DRIVE (Seite 2852)" nicht mindestens einmal für diesen Antrieb aufgerufen wurde. Dadurch wird verhindert, dass die Initialisierung im ersten Zyklus von "USS_DRIVE (Seite 2852)" eine anstehende Anforderung zum Lesen oder Schreiben von Parametern überschreibt, weil dabei der Antrieb als neuer Eintrag initialisiert wird. Um diesen Fehler zu beheben, rufen Sie die Anweisung "USS_DRIVE (Seite 2852)" für diesen Antrieb auf. |
| 80Ax-80Fx   | Spezifische Fehler, die von der USS-Bibliothek aufgerufenen Anweisungen für die Punkt-zu-Punkt-Kommunikation (PtP) zurückgegeben werden: Diese Fehlercodes werden von der USS-Bibliothek nicht verändert und sind in den Beschreibungen der PtP-Anweisungen definiert.  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### USS\_Extended\_Error - Zusätzliche Fehlercodes für USS-Antriebe

USS-Antriebe unterstützen Lese- und Schreibzugriff auf die internen Parameter eines Antriebs. Diese Funktion ermöglicht die dezentrale Steuerung und Konfiguration des Antriebs. Zugriffe der Antriebsparameter können aufgrund von Fehlern wie Wert außerhalb des Bereichs oder unzulässige Anforderungen in der aktuellen Betriebsart des Antriebs fehlschlagen. Der Antrieb erzeugt einen Fehlercode, der in der Variablen "USS\_Extended\_Error" im Instanz-DB der Anweisung "USS\_DRIVE (Seite 2852)" ausgegeben wird. Dieser Fehlercode gilt nur für die letzte Ausführung der Anweisung "USS\_RPM (Seite 2855)" oder "USS\_WPM (Seite 2856)". Der Fehlercode des Antriebs wird in der Variablen "USS\_Extended\_Error" abgelegt, wenn der Wert von STATUS hexadezimal 818C ist. Der Fehlercode von "USS\_Extended\_Error" richtet sich nach der Variante des Antriebs. Eine Beschreibung der erweiterten Fehlercodes von Lese- und Schreibfunktionen für Parameter finden Sie im Handbuch des Antriebs.

### MODBUS (RTU)

#### MB\_COMM\_LOAD: Port an PtP-Baugruppe für Modbus-RTU konfigurieren

##### Beschreibung

Die Anweisung "MB\_COMM\_LOAD" konfiguriert einen Port für die Kommunikation über das Modbus-RTU-Protokoll. Die folgende Hardware kann hierzu verwendet werden:

- Bis zu drei Punkt-zu-Punkt-Baugruppen (PtP) CM 1241 RS485 oder CM 1241 RS232
- Zusätzlich ein Kommunikationsboard CB 1241 RS485

Nach der Konfiguration des Ports kommunizieren Sie über Modbus, indem Sie die Anweisung "MB\_SLAVE" oder "MB\_MASTER" ausführen.

### Aufruf

"MB\_COMM\_LOAD" muss einmal aufgerufen werden, um den Port für das Modbus-RTU-Protokoll zu konfigurieren. Nach Abschluss der Konfiguration kann der Port von den Anweisungen "MB\_MASTER (Seite 2863)" und "MB\_SLAVE (Seite 2871)" verwendet werden.

"MB\_COMM\_LOAD" muss nur dann erneut aufgerufen werden, wenn einer der Kommunikationsparameter geändert werden muss. Jeder Aufruf von "MB\_COMM\_LOAD" löscht die Kommunikationspuffer. Um Datenverlust bei der Kommunikation zu vermeiden, rufen sie die Anweisung deshalb nicht unnötig auf.

Für die Konfiguration der Ports jedes Kommunikationsmoduls, das für die Modbus-Kommunikation eingesetzt wird, muss eine Instanz von "MB\_COMM\_LOAD" genutzt werden. Sie müssen jedem Port, den Sie nutzen, einen eindeutigen Instanz-Datenbaustein "MB\_COMM\_LOAD" zuweisen. Die S7-1200 CPU ist auf drei Kommunikationsmodule begrenzt.

Wenn Sie die Anweisung "MB\_MASTER (Seite 2863)" oder "MB\_SLAVE (Seite 2871)" einfügen, wird ein Instanz-Datenbaustein zugewiesen. Dieser Instanz-Datenbaustein wird referenziert, indem Sie diesen am Parameter MB\_DB der Anweisung "MB\_COMM\_LOAD" angeben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB\_COMM\_LOAD":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-----------|-------------|----------|------------------------------|---|
| REQ       | Input       | BOOL     | E, A, M, D, L                | Ausführung der Anweisung bei steigender Flanke.   |
| PORT      | Input       | PORT     | E, A, M, D, L oder Konstante | Kennung des Kommunikationsports:<br>Nachdem Sie das Kommunikationsmodul in die Gerätekonfiguration eingefügt haben, erscheint die Portkennung in der Klappliste an der Verbindung der PORT-Box. Auf diese Konstante kann auch in der Variablen-tabelle im Register "Konstanten" verwiesen werden. |
| BAUD      | Input       | UDINT    | E, A, M, D, L oder Konstante | Auswahl der Baudrate:<br>300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200<br>Alle anderen Werte sind ungültig.  |
| PARITY    | Input       | UINT     | E, A, M, D, L oder Konstante | Auswahl der Parität:<br><ul style="list-style-type: none"> <li>• 0 – Keine</li> <li>• 1 – Ungerade</li> <li>• 2 – Gerade</li> </ul>   |
| FLOW_CTRL | Input       | UINT     | E, A, M, D, L oder Konstante | Auswahl der Flusskontrolle:<br><ul style="list-style-type: none"> <li>• 0 – (Standard) Keine Flusskontrolle</li> <li>• 1 – Hardware-Flusskontrolle mit RTS immer EIN (gilt nicht bei RS485-Ports)</li> <li>• 2 - Hardware-Flusskontrolle mit RTS geschaltet</li> </ul>                            |

| Parameter   | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-------------|-------------|----------|------------------------------|---|
| RTS_ON_DLY  | Input       | UINT     | E, A, M, D, L oder Konstante | Auswahl RTS-Einschaltverzögerung: <ul style="list-style-type: none"> <li>• 0 – (Standard) Keine Verzögerung von "RTS aktiv", bis das erste Zeichnung der Meldung gesendet wird.</li> <li>• 1 bis 65535 – Verzögerung in Millisekunden von "RTS aktiv", bis das erste Zeichen der Meldung gesendet wird (gilt nicht bei RS-485-Ports). RTS-Verzögerungen müssen unabhängig von der Auswahl FLOW_CTRL angewendet werden.</li> </ul> |
| RTS_OFF_DLY | Input       | UINT     | E, A, M, D, L oder Konstante | Auswahl RTS-Ausschaltverzögerung: <ul style="list-style-type: none"> <li>• 0 – (Standard) Keine Verzögerung nach der Übertragung des letzten Zeichens bis "RTS inaktiv"</li> <li>• 1 bis 65535 – Verzögerung in Millisekunden von der Übertragung des letzten Zeichens bis "RTS inaktiv" (gilt nicht bei RS-485-Ports). RTS-Verzögerungen müssen unabhängig von der Auswahl FLOW_CTRL angewendet werden.</li> </ul>               |
| RESP_TO     | Input       | UINT     | E, A, M, D, L oder Konstante | Timeout der Antwort:<br>Zeit in Millisekunden, die "MB_MASTER (Seite 2863)" auf eine Antwort vom Slave wartet. Wenn der Slave während dieses Zeitraums nicht antwortet, wiederholt "MB_MASTER (Seite 2863)" die Anforderung oder beendet die Anforderung mit einem Fehler, sofern die angegebene Anzahl von Wiederholungen gesendet wurde.<br>5 ms bis 65535 ms (Standardwert = 1000 ms).   |
| MB_DB       | Input       | MB_BASE  | D                            | Ein Verweis auf den Instanz-Datenbaustein der Anweisungen "MB_MASTER (Seite 2863)" oder "MB_SLAVE (Seite 2871)". Nachdem Sie "MB_SLAVE (Seite 2871)" oder "MB_MASTER (Seite 2863)" in Ihr Programm eingefügt haben, ist die DB-Kennung in der Klappliste an der Verbindung der MB_DB-Box verfügbar.   |
| DONE        | Output      | BOOL     | E, A, M, D, L                | Ausführung der Anweisung ohne Fehler beendet.   |
| ERROR       | Output      | BOOL     | E, A, M, D, L                | Fehler: <ul style="list-style-type: none"> <li>• 0 – Kein Fehler erkannt</li> <li>• 1 – Gibt an, dass ein Fehler erkannt wurde. Am Parameter STATUS wird ein Fehlercode ausgegeben.</li> </ul>  |
| STATUS      | Output      | WORD     | E, A, M, D, L                | Fehlercode der Portprojektierung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter STATUS**

| Fehlercode*<br>(W#16#....) | Beschreibung   |
|----------------------------|--|
| 0000                       | Kein Fehler  |
| 8180                       | Ungültiger Wert für die Port-ID (falsche Adresse für das Kommunikations-Modul).  |
| 8181                       | Ungültiger Wert für die Baudrate.  |
| 8182                       | Ungültiger Wert für die Parität.   |
| 8183                       | Ungültiger Wert für die Flusskontrolle.  |
| 8184                       | Ungültiger Wert für die Zeitüberschreitung der Antwort (der Zeitraum bis eine Zeitüberschreitung gemeldet wird muss mindestens 25ms betragen). |
| 8185                       | Fehlerhafter Zeiger am Parameter MB_DB auf den Instanz-DB der Anweisung "MB_MASTER (Seite 2863)" oder "MB_SLAVE (Seite 2871)".                 |

\* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**MB\_COMM\_LOAD Datenbausteinvariablen**

Die folgende Tabelle zeigt die öffentlichen statischen Variablen im Instanz-DB von MB\_COMM\_LOAD, die Sie in Ihrem Programm verwenden können.

Tabelle 9-120 Statische Variablen im Instanz-DB

| Variable  | Datentyp | Vorbelegung | Beschreibung  |
|-----------|----------|-------------|---|
| ICHAR_GAP | WORD     | 0           | Verzögerung für den Zeichenabstand zwischen den Zeichen. Dieser Parameter wird in Millisekunden angegeben und dient dazu, den erwarteten Zeitraum zwischen den empfangenen Zeichen zu erhöhen. Die entsprechende Anzahl Bitzeiten für diesen Parameter wird zum Modbus-Standardwert von 35 Bitzeiten (3,5 Zeichenzeiten) addiert.   |
| RETRIES   | WORD     | 2           | Die Anzahl der wiederholten Versuche, die der Master durchführt, bevor der Fehlercode 0x80C8 für "Keine Antwort" zurückgegeben wird.  |
| MODE      | USINT    | 0           | Betriebsart<br>Zulässige Betriebsarten sind: <ul style="list-style-type: none"> <li>• 0 = Vollduplex (RS232)</li> <li>• 1 = Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt)</li> <li>• 2 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Master)</li> <li>• 3 = Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)</li> <li>• 4 = Halbduplex (RS485) Zweidraht-Betrieb</li> </ul> |

| Variable    | Datentyp | Vorbelegung | Beschreibung   |
|-------------|----------|-------------|--|
| LINE_PRE    | USINT    | 0           | Vorbelegung der Empfangsleitung<br>Zulässige Vorbelegungen sind: <ul style="list-style-type: none"> <li>• 0 = "Keine" Vorbelegung</li> <li>• 1 = Signal R(A)=5V, Signal R(B)=0 V (Break-Erkennung) :<br/>Bei dieser Vorbelegung ist Break-Erkennung möglich.<br/>Nur auswählbar bei: "Vollduplex (RS422) Vierdraht-Betrieb (Punkt-zu-Punkt-Kopplung)" und "Vollduplex (RS422) Vierdraht-Betrieb (Mehrpunkt Slave)".</li> <li>• 2 = Signal R(A)=0 V, Signal R(B)=5 V :<br/>Diese Vorbelegung entspricht dem Ruhezustand (kein Sendevorgang aktiv). Bei dieser Vorbelegung ist keine Break-Erkennung möglich.</li> </ul> |
| CABLE-BREAK | USINT    | 0           | Kabelbruchererkennung aktivieren: <ul style="list-style-type: none"> <li>• 0 - nicht aktiviert</li> <li>• 1 - aktiviert</li> </ul>   |

## MB\_MASTER: Als Modbus-Master über PtP-Port kommunizieren

### Beschreibung MB\_MASTER

#### Beschreibung

Mit der Anweisung "MB\_MASTER" kann Ihr Programm über den Port eines Punkt-zu-Punkt-Moduls (CM) oder eines Kommunikationsboards (CB) als Modbus-Master kommunizieren. Sie können auf Daten in einem oder mehreren Modbus-Slave-Geräten zugreifen.

Damit die Anweisung "MB\_MASTER" mit einem Port kommunizieren kann, muss "MB\_COMM\_LOAD (Seite 2859)" vorher ausgeführt werden.

Wenn Sie die Anweisung "MB\_MASTER" in Ihr Programm einfügen, wird ein Instanz-DB erstellt. Diesen Instanz-DB geben Sie am Eingangsparameter MB\_DB der Anweisung "MB\_COMM\_LOAD (Seite 2859)" an.

#### Regeln für die Kommunikation des Modbus-Masters

- Ein Port der für Modbus-Masteranforderungen verwendet wird, kann nicht für "MB\_SLAVE" genutzt werden.
- Ein Port kann für einen oder mehrere Aufrufe von "MB\_MASTER" genutzt werden, wenn der gleiche Instanz-DB verwendet wird.
- Die Modbus-Operationen nutzen keine Kommunikations-Alarmereignisse zum Steuern des Kommunikationsprozesses. Ihr Programm muss die Anweisung "MB\_MASTER" auf abgeschlossene Sende- und Empfangsvorgänge abfragen.

- Aufruf der Anweisung:
  - Rufen Sie die Anweisung "MB\_MASTER" nach Möglichkeit in einem zyklischen Programm-OB auf. Die Anweisung kann auch in einem Verzögerungs- oder Weckalarm-OB aufgerufen werden.
  - Rufen Sie nicht mehrere "MB\_MASTER"-Anweisungen in Organisationsbausteinen mit unterschiedlichen Prioritätsklassen auf. Bei einer vorweggenommenen Ausführung einer "MB\_MASTER"-Anweisung aus einer höheren Prioritätsklasse kann es zu einer fehlerhaften Ausrührung kommen.
  - Rufen Sie "MB\_MASTER"-Anweisungen nicht in einem Anlauf-, Diagnose- oder Zeitfehler-OB auf.
- Nach Beginn einer Übertragung muss am Parameter EN (KOP/FUP) durchgehend der Wert "1" anliegen, bis durch die Anweisung an der Ausgangsparameter DONE oder ERROR auf "1" gesetzt wird. Ein erneuter Aufruf durch den Parameter REQ während die Anweisung ausgeführt wird führt zu einem Fehler. Nach der Ausführung der Anweisung bleibt das Bit am Parameter REQ noch für den Zeitraum gesetzt, der im Instanz-DB an dem Parameter BLOCKED\_PROC\_TIMEOUT vorgegeben wurde.
- Wenn "MB\_MASTER" eine Anforderung an einen Slave sendet, müssen Sie die Ausführung von "MB\_MASTER" fortsetzen, bis die Antwort vom Slave eingegangen ist.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB\_MASTER":

| Parameter                 | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|---------------------------|-------------|----------|------------------------------|--|
| REQ<br>(Seite 2866)       | Input       | BOOL     | E, A, M, D, L                | Anforderungseingabe: <ul style="list-style-type: none"> <li>• 0 – Keine Anforderung</li> <li>• 1 – Anforderung, Daten an Modbus-Slave(s) zu senden</li> </ul>  |
| MB_ADDR                   | Input       | UINT     | E, A, M, D, L oder Konstante | Modbus-RTU-Stationsadresse: <ul style="list-style-type: none"> <li>• Standard-Adressbereich: 0 bis 247</li> <li>• Erweiterter Adressbereich: 0 bis 65535</li> </ul> Der Wert "0" ist für den Broadcast einer Meldung an alle Modbus-Slaves reserviert. Für den Broadcast werden nur die Modbus-Funktionscodes 05, 06, 15 und 16 unterstützt. |
| MODE<br>(Seite 2867)      | Input       | USINT    | E, A, M, D, L oder Konstante | Auswahl Modus: Gibt die Art der Anforderung an: Lesen, Schreiben oder Diagnose<br>In der Tabelle der Modbus-Funktionen finden Sie weitere Einzelheiten.  |
| DATA_ADDR<br>(Seite 2867) | Input       | UDINT    | E, A, M, D, L oder Konstante | Anfangsadresse im Slave: Gibt die Anfangsadresse der Daten an, auf die im Modbus-Slave zugegriffen werden soll. Die gültigen Adressen finden Sie in der Tabelle der Modbus-Funktionen.   |
| DATA_LEN                  | Input       | UINT     | E, A, M, D, L oder Konstante | Datenlänge: Gibt die Anzahl der Bits oder Wörter an, auf die diese Anforderung zugreifen soll. Die gültigen Längen finden Sie in der Tabelle der Modbus-Funktionen.  |

| Parameter                | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|--------------------------|-------------|----------|-----------------|---|
| DATA_PTR<br>(Seite 2868) | Input       | VARIANT  | M, D            | Zeiger auf die DB- oder Merker-Adresse der CPU für die zu schreibenden oder zu lesenden Daten. Bei einem DB muss dieser mit der Zugriffsart "Standard - kompatibel mit S7-300/400" erstellt werden. |
| DONE                     | Output      | BOOL     | E, A, M, D, L   | <ul style="list-style-type: none"> <li>0: Transaktion nicht abgeschlossen</li> <li>1: Transaktion ohne Fehler beendet</li> </ul>  |
| BUSY                     | Output      | BOOL     | E, A, M, D, L   | <ul style="list-style-type: none"> <li>0: Keine Transaktion von "MB_MASTER" in Bearbeitung</li> <li>1: Transaktion von "MB_MASTER" in Bearbeitung</li> </ul>  |
| ERROR                    | Output      | BOOL     | E, A, M, D, L   | <ul style="list-style-type: none"> <li>0: Kein Fehler</li> <li>1: Fehler; der Fehlercode wird am Parameter STATUS angezeigt</li> </ul>  |
| STATUS                   | Output      | WORD     | E, A, M, D, L   | Ausführungsbedingung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

## Parameter STATUS

Tabelle 9-121 Kommunikations- und Konfigurationsfehlermeldungen der Anweisung

| Fehlercode*<br>(W#16#....) | Beschreibung   |
|----------------------------|--|
| 0000                       | Kein Fehler  |
| 80C8                       | Slave-Timeout. Überprüfen Sie die Baudrate, Parität und die Anschlüsse des Slaves.   |
| 80D1                       | Der Empfänger hat eine Flusskontrolle zur Unterbrechung einer laufenden Sendung angefordert, aber die Sendung innerhalb der Wartezeit nicht wieder aktiviert.<br>Dieser Fehler wird auch bei der Hardware-Flusskontrolle erzeugt, wenn der Empfänger innerhalb der Wartezeit kein CTS erkennt. |
| 80D2                       | Die Sendeaufforderung wurde abgebrochen, weil vom DCE kein DSR-Signal empfangen wird.  |
| 80E0                       | Die Meldung wurde beendet, weil der Empfangspuffer voll ist.   |
| 80E1                       | Die Meldung wurde wegen eines Paritätsfehlers beendet.   |
| 80E2                       | Die Meldung wurde wegen eines Telegrammfehlers beendet.  |
| 80E3                       | Die Meldung wurde wegen eines Überlauffehlers beendet.   |
| 80E4                       | Die Meldung wurde abgebrochen, weil die angegebene Länge die Gesamtpuffergröße überschreitet.  |
| 8180                       | Ungültiger Wert für die Port-ID.   |
| 8186                       | Ungültige Modbus-Stationsadresse   |
| 8188                       | Ungültiger Wert am Parameter MODE für einen Broadcast-Aufruf.  |
| 8189                       | Ungültiger Wert für die Datenadresse.  |
| 818A                       | Ungültiger Wert für die Datenlänge.  |
| 818B                       | Ungültiger Pointer auf die lokale Datenquelle bzw. das lokale Datenziel: Größe nicht richtig   |
| 818C                       | Ungültiger Zeiger am Parameter DATA_PTR. Verwenden Sie einen Zeiger auf einen Merkerbereich oder einen DB mit der Zugriffsart "Standard - kompatibel mit S7-300/400".  |

| Fehlercode*<br>(W#16#...)   | Beschreibung   |
|---|--|
| 8200  | Port ist durch Bearbeitung einer Sendeanforderung belegt |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

Tabelle 9-122 Fehlermeldungen des Modbus-Protokolls

| Fehlercode*<br>(W#16#...)   | Antwortcode des<br>Slaves | Beschreibung  |
|---|---------------------------|---|
| 8380  | -                         | CRC-Fehler  |
| 8381  | 01                        | Funktionscode nicht unterstützt   |
| 8382  | 03                        | Fehler in der Datenlänge  |
| 8383  | 02                        | Fehler in der Datenadresse oder Adresse außerhalb des gültigen Bereichs von DATA_PTR  |
| 8384  | > 03                      | Fehler im Datenwert   |
| 8385  | 03                        | Wert des Datendiagnosecodes wird nicht unterstützt (Funktionscode 08)   |
| 8386  | -                         | Funktionscode der Antwort stimmt nicht mit dem Funktionscode der Anfrage überein.   |
| 8387  | -                         | Antwort des falschen Slaves   |
| 8388  | -                         | Die Antwort des Slaves auf einen Schreibaufruf ist nicht korrekt. Die vom Slave gesendeten Daten stimmen nicht mit der Anfrage der Masters überein. |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                           |   |

## Parameter REQ

### Beschreibung

- REQ = FALSE: Keine Anforderung
- REQ = TRUE: Anforderung, Daten an Modbus-Slave(s) zu senden

Sie müssen diesen Eingang über einen mittels steigender Flanke gesteuerten Kontakt im ersten Aufruf von "MB\_MASTER" bereitstellen. Der durch eine Flanke ausgelöste Impuls ruft die Übertragungsanforderung einmal auf. Alle Eingänge werden erfasst und bleiben für die Dauer der von diesem Eingang ausgelösten Anforderung und Antwort lang unverändert.

Während eine Instanz der Anweisung "MB\_MASTER" ausgeführt wird, kann keine weitere Instanz der Anweisung aufgerufen werden. Wird während der Ausführung von "MB\_MASTER" die Instanz über den Parameter REQ erneut aufgerufen, wird zudem kein automatischer Folgeaufruf gestartet. Um die Instanz der Anweisung erneut aufzurufen, muss die Ausführung der Anweisung zuerst abgeschlossen werden, um diese erneut am Parameter REQ aufzurufen.



## Parameter DATA\_ADDR und MODE

### Beschreibung

Über den Parameter DATA\_ADDR spezifizieren Sie die Anfangsadresse für den Datenzugriff auf den Modbus-Slave.

Über die Parameter MODE und die Modbus-Adresse legen Sie den Funktionscode fest, der an den Modbus-Slave übertragen wird. Die folgende Tabelle zeigt den Zusammenhang zwischen dem Parameter MODE, dem Funktionscode und dem Modbus-Adressbereich.

| MODE | Modbus-Funktion | Datenlänge                              | Operation und Daten  | Modbus-Adresse                            |
|------|-----------------|---|--|---|
| 0    | 01              | 1 bis 2000<br>1 bis 1992 <sup>(1)</sup> | Ausgangsbits lesen:<br>1 bis (1992 oder 2000) Bits pro Anfrage   | 1 bis 9999                                |
| 0    | 02              | 1 bis 2000<br>1 bis 1992 <sup>(1)</sup> | Eingangsbits lesen:<br>1 bis (1992 oder 2000) Bits pro Anfrage   | 10001 bis 19999                           |
| 0    | 03              | 1 bis 125<br>1 bis 124 <sup>(1)</sup>   | Halteregister lesen:<br>1 bis (124 oder 125) WORD pro Anfrage  | 40001 bis 49999 oder<br>400001 bis 465535 |
| 0    | 04              | 1 bis 125<br>1 bis 124 <sup>(1)</sup>   | Eingangs-WORD lesen:<br>1 bis (124 oder 125) WORD pro Anfrage  | 30001 bis 39999                           |
| 1    | 05              | 1                                       | Ein Ausgangsbit schreiben:<br>Ein Bit pro Anfrage  | 1 bis 9999                                |
| 1    | 06              | 1                                       | Ein Halteregister schreiben:<br>1 WORD pro Anfrage   | 40001 bis 49999 oder<br>400001 bis 465535 |
| 1    | 15              | 2 bis 1968<br>2 bis 1960 <sup>(1)</sup> | Mehrere Ausgangsbits schreiben:<br>2 bis (1960 oder 1968) Bits pro Anfrage   | 1 bis 9999                                |
| 1    | 16              | 2 bis 123<br>2 bis 122 <sup>(1)</sup>   | Mehrere Halteregister schreiben:<br>2 bis (122 oder 123) WORD pro Anfrage  | 40001 bis 49999 oder<br>400001 bis 465535 |
| 2    | 15              | 1 bis 1968<br>2 bis 1960 <sup>(1)</sup> | Ein oder mehrere Ausgangsbits schreiben:<br>1 bis (1960 oder 1968) Bits pro Anfrage  | 1 bis 9999                                |
| 2    | 16              | 1 bis 123<br>2 bis 122 <sup>(1)</sup>   | Ein oder mehrere Halteregister schreiben:<br>1 bis (122 oder 123) WORD pro Anfrage   | 40001 bis 49999 oder<br>400001 bis 465535 |
| 11   | 11              | 0                                       | Auslesen des Kommunikations-Statusworts der Slaves und des Ereigniszählers:<br>Das Statuswort zeigt die Ausführung der Anweisung an (0: wird nicht ausgeführt; 0xFFFF: wird ausgeführt). Der Ereigniszähler wird bei jeder erfolgreichen Übertragung einer Nachricht erhöht.<br>Die Parameter DATA_ADDR und DATA_LEN der Anweisung "MB_MASTER" werden bei dieser Funktion ignoriert. | -   |
| 80   | 08              | 1                                       | Überprüfen des Slave-Status durch Auslesen des Fehlercodes (0x0000):<br>1 WORD pro Anfrage   | -   |

| MODE                                   | Modbus-Funktion | Datenlänge | Operation und Daten   | Modbus-Adresse |
|--|-----------------|------------|---|----------------|
| 81                                     | 08              | 1          | Ereigniszähler des Slaves zurücksetzen über den Diagnose-Code 0x000A:<br>1 WORD pro Anfrage | -              |
| 3 bis 10, 12<br>bis 79, 82<br>bis 2555 |                 |            | Reserviert  | -              |

<sup>(1)</sup> Für den Modus "Erweiterter Adressbereich" wird die maximale Datenlänge um ein Byte bzw. ein WORD reduziert, je nachdem welcher Datentyp für die Funktion verwendet wird.

## Parameter DATA\_PTR

### Beschreibung

Der Parameter DATA\_PTR ist ein Zeiger auf einen Datenbaustein oder einen Merker, von dem Daten geschrieben oder gelesen werden. Wenn Sie einen Datenbaustein verwenden, erstellen Sie einen globalen Datenbaustein mit Zugriffsart "Standard - kompatibel mit S7-300/400".

### Datenbausteinstrukturen für den Parameter DATA\_PTR

- Diese Datentypen gelten für das **Lesen von Wörtern** der Modbus-Adressen 30001 bis 39999, 40001 bis 49999 und 400001 bis 465536 sowie für das **Schreiben von Wörtern** in die Modbus-Adressen 40001 bis 49999 und 400001 bis 465536.
  - Standardfeld der Datentypen WORD, UINT oder INT (siehe unten).
  - Benannte Struktur vom Typ WORD, UINT oder INT, bei der jedes Element einen eindeutigen Namen und einen 16-Bit-Datentyp hat.
  - Benannte komplexe Struktur, bei der jedes Element einen eindeutigen Namen und einen 16- oder 32-Bit-Datentyp hat.
- Zum Lesen und Schreiben von Bits für die Modbus-Adressen 00001 bis 09999 und 10001 bis 19999.
  - Standardfeld aus Booleschen Datentypen.
  - Benannte Boolesche Struktur aus eindeutig benannten Booleschen Variablen.
- Es ist zwar nicht erforderlich, doch empfehlenswert, dass jede Anweisung "MB\_MASTER" einen eigenen getrennten Speicherbereich in einem globalen Datenbaustein hat. Grund hierfür ist, dass die Möglichkeit der Datenzerstörung größer ist, wenn mehrere Anweisungen "MB\_MASTER" in demselben Bereich eines globalen Datenbausteins lesen und schreiben.
- Es ist nicht erforderlich, dass die Speicherbereiche für DATA\_PTR in demselben globalen Datenbaustein liegen. Sie können einen Datenbaustein mit mehreren Bereichen für Modbus-Lesevorgänge, einen Datenbaustein für Modbus-Schreibvorgänge oder einen Datenbaustein für jede Slave-Station anlegen.

## Instanz-DB der Anweisung "MB\_MASTER"

### Statische Variablen des Instanz-DBs

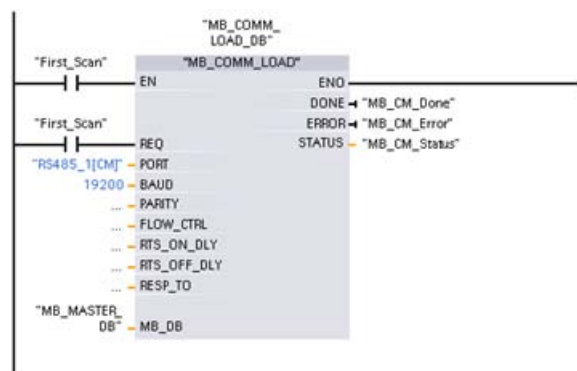
Die folgende Tabelle beschreibt die statischen Variablen des Instanz-DBs der Anweisung, die Sie im Anwenderprogramm verwenden können.

| Variable                 | Datentyp | Beschreibung   |
|--------------------------|----------|--|
| MB_STATE                 | UINT     | Interner Status der Modbus-Anweisung.  |
| BLOCKED_<br>PROC_TIMEOUT | REAL     | Zeitintervall nach Beendigung des Aufrufs der Anweisung und rückerlösen des Bits ACTIVE im Instanz-DB. Der Zeitpuffer wird verwendet, um zu vermeiden, dass das Ausführen der Anweisung beendet wird, bevor ein Auftrag vollständig gesendet wurde. Der voreingestellte Zeitintervall beträgt 500ms.                   |
| EXTENDED_<br>ADDRESSING  | BOOL     | Konfiguration der Adressierung: <ul style="list-style-type: none"> <li>• 0: Standard-Adressbereich (1 Byte)</li> <li>• 1: Erweiterter Adressbereich (2 Byte)</li> </ul> Weitere Informationen zur Variable EXTENDED_ADDRESSING finden Sie in dem folgenden Abschnitt: Instanz-DB der Anweisung "MB_SLAVE" (Seite 2874) |

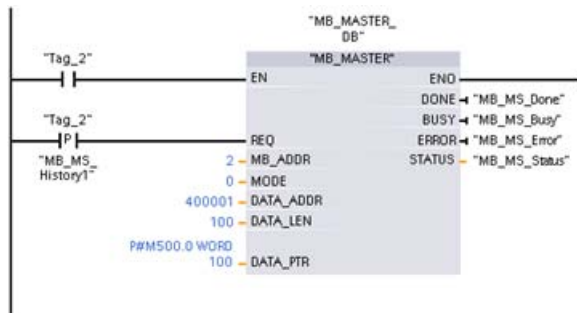
### Beispielprogramm für einen Modbus-Master

#### Netzwerke (KOP)

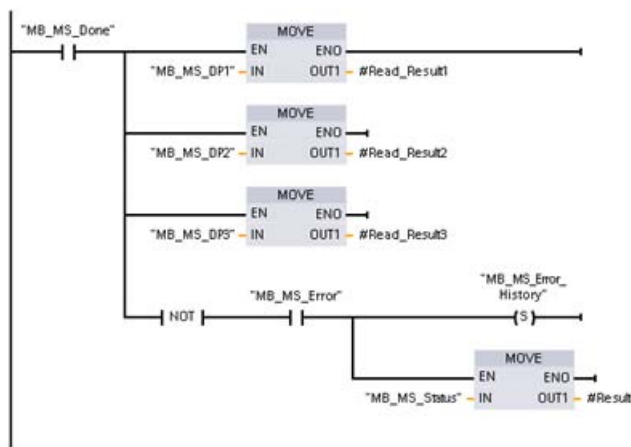
**Netzwerk 1:** Parameter des RS-485-Moduls nur einmal während des ersten Zyklus initialisieren.



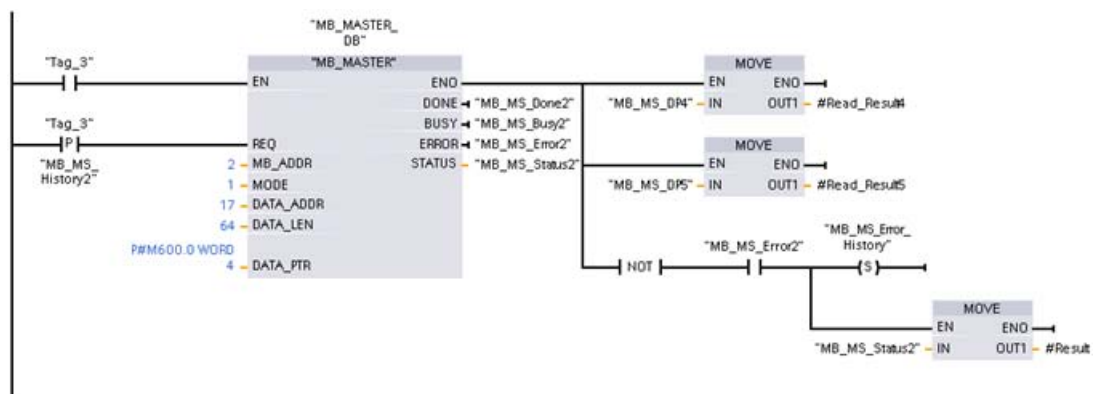
**Netzwerk 2:** 100 Wörter aus dem Haltereister des Slave lesen.



**Netzwerk 3:** Dies ist ein optionales Netzwerk, das die Werte der ersten 3 Wörter anzeigt, sobald die Leseoperation durchgeführt wurde.



**Netzwerk 4:** 64 Bits in das Prozessabbild der Ausgänge schreiben, Beginn an Slave-Adresse A2.0.



**MB\_SLAVE: Als Modbus-Slave über PtP-Port kommunizieren****Beschreibung MB\_SLAVE****Beschreibung**

Mit der Anweisung "MB\_SLAVE" kann Ihr Programm über den Port eines Punkt-zu-Punkt-Moduls (PtP) oder eines Kommunikationsboards (CB) als Modbus-Slave kommunizieren. Ein Modbus-RTU-Master kann eine Anforderung senden, und Ihr Programm antwortet über die Ausführung von "MB\_SLAVE".

Wenn Sie die Anweisung "MB\_SLAVE" in Ihr Programm einfügen, müssen Sie einen eindeutigen Instanz-Datenbaustein zuweisen. Dieser Instanz-Datenbaustein wird verwendet, wenn Sie diesen an dem Parameter MB\_DB der Anweisung "MB\_COMM\_LOAD (Seite 2859)" angeben.

Die Funktionscodes der Modbus-Kommunikation (Funktionscodes 1, 2, 4, 5 und 15) können Bits und Wörter direkt im Prozessabbild der Eingänge und im Prozessabbild der Ausgänge im Zielsystem lesen und schreiben. Die folgende Tabelle zeigt die Zuordnung der Modbus-Adressen zum Prozessabbild in der CPU.

| Modbus-Funktionen von "MB_SLAVE" |                |              |               |     | S7-1200      |                            |                  |
|----------------------------------|----------------|--------------|---------------|-----|--------------|----------------------------|------------------|
| Codes                            | Funktion       | Datenbereich | Adressbereich |     | Datenbereich | CPU-Adresse                |                  |
| 01                               | Bits lesen     | Ausgang      | 1             | bis | 8192         | Prozessabbild der Ausgänge | A0.0 bis A1023.7 |
| 02                               | Bits lesen     | Eingang      | 10001         | bis | 18192        | Prozessabbild der Eingänge | E0.0 bis E1023.7 |
| 04                               | Wörter lesen   | Eingang      | 30001         | bis | 30512        | Prozessabbild der Eingänge | EW0 bis EW1022   |
| 05                               | Bit schreiben  | Ausgang      | 1             | bis | 8192         | Prozessabbild der Ausgänge | A0.0 bis A1023.7 |
| 15                               | Bits schreiben | Ausgang      | 1             | bis | 8192         | Prozessabbild der Ausgänge | A0.0 bis A1023.7 |

Die Funktionscodes der Modbus-Kommunikation (Funktionscodes 3, 6, 16) nutzen ein eigenes Halteregeister. Hierzu kann ein Merker oder ein Datenbaustein mit der Zugriffsart "Standard - kompatibel mit S7-300/400" verwendet werden.

Den Typ des Halteregeisters geben Sie durch den Parameter MB\_HOLD\_REG der Anweisung "MB\_SLAVE" vor. Die folgende Tabelle zeigt die Zuordnung des Modbus-Halteregeisters zur DB-Adresse von MB\_HOLD\_REG im Zielsystem.

| Modbus-Funktionen von "MB_SLAVE" |                |                |                             | S7-1200                     |                              |
|----------------------------------|----------------|----------------|-----------------------------|-----------------------------|------------------------------|
| Codes                            | Funktion       | Datenbereich   | Adressbereich (WORD-Nummer) | Adresse im DB (BYTE-Nummer) | Merker-Adresse (BYTE-Nummer) |
| 03                               | Wörter lesen   | Halteregeister | 40001 bis 49999 oder        | DW0 bis DW19998 oder        | MW0 bis CPU-Limit            |
|                                  |                |                | 400001 bis 465535           | DW0 bis DW131068            |                              |
| 06                               | Wort schreiben | Halteregeister | 40001 bis 49999 oder        | DW0 bis DW19998 oder        |                              |

| Modbus-Funktionen von "MB_SLAVE" |                  |              | S7-1200                                   |  |
|----------------------------------|------------------|--------------|---|--|
|                                  |                  |              | 400001 bis 465535                         | DW0 bis DW131068                         |
| 16                               | Wörter schreiben | Haltregister | 40001 bis 49999 oder<br>400001 bis 465535 | DW0 bis DW19998 oder<br>DW0 bis DW131068 |

Die folgende Tabelle zeigt die unterstützten Modbus-Diagnosefunktionen.

| Modbus-Diagnosefunktionen von "MB_SLAVE" der S7-1200 |              |  |
|--|--------------|--|
| Codes  | Teilfunktion | Beschreibung   |
| 08   | 0000H        | Echotest Abfragedaten ausgeben: Die Anweisung "MB_SLAVE" gibt einem Modbus-Master das Echo eines empfangenen Datenworts zurück.  |
| 08   | 000AH        | Kommunikationsereigniszähler löschen: Die Anweisung "MB_SLAVE" löscht den Kommunikationsereigniszähler, der für Modbus-Funktion 11 verwendet wird.   |
| 11   | -            | Kommunikationsereigniszähler abrufen: Die Anweisung "MB_SLAVE" nutzt einen internen Kommunikationsereigniszähler, um die Anzahl erfolgreicher Modbus-Lese- und Modbus-Schreibanforderungen, die an den Modbus-Slave gesendet werden, zu erfassen. Der Zähler wird bei Funktion 8, Funktion 11 und Broadcast-Anforderungen nicht hochgezählt. Außerdem wird bei Anforderungen, die zu Kommunikationsfehlern führen (z.B. Paritäts- oder CRC-Fehler), nicht hochgezählt. |

Die Anweisung "MB\_SLAVE" unterstützt Broadcast-Schreibanforderungen von Modbus-Mastern, solange die Anforderungen den Zugriff auf gültige Adressen umfassen.

Unabhängig von der Gültigkeit einer Anforderung gibt "MB\_SLAVE" keine Antwort an einen Modbus-Master als Ergebnis einer Broadcast-Anforderung aus.

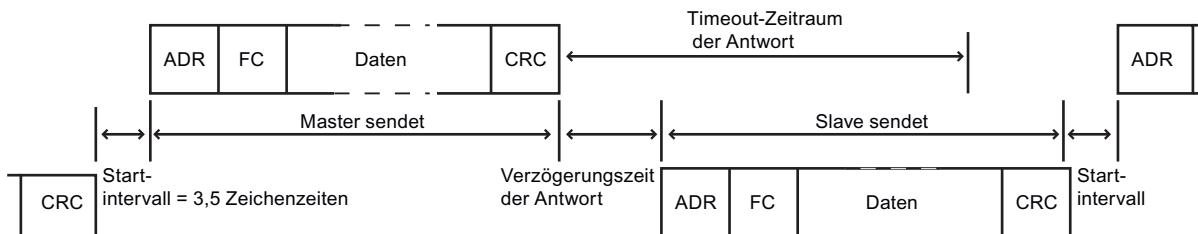
### Regeln für die Modbus-Slave-Kommunikation

- "MB\_COMM\_LOAD" muss ausgeführt werden, um einen Port zu konfigurieren, damit die Anweisung "MB\_SLAVE" mit diesem Port kommunizieren kann.
- Wenn ein Port als Slave auf einen Modbus-Master reagieren soll, kann dieser Port nicht von "MB\_MASTER (Seite 2863)" verwendet werden. Nur eine Instanz von "MB\_SLAVE" kann für einen bestimmten Port verwendet werden.
- Die Modbus-Anweisungen nutzen keine Kommunikations-Alarmereignisse zum Steuern des Kommunikationsprozesses. Ihr Programm muss den Kommunikationsprozess durch Abfrage der Anweisung "MB\_SLAVE" auf abgeschlossene Sende- und Empfangsvorgänge steuern.
- Die Anweisung "MB\_SLAVE" muss regelmäßig mit einer Häufigkeit ausgeführt werden, die eine zeitnahe Antwort auf eingehende Anforderungen eines Modbus-Masters ermöglicht. Es ist daher zu empfehlen, die Anweisung in einem zyklischen Programm-OB aufzurufen. Der Aufruf der Anweisung "MB\_SLAVE" in einem Alarm-OB ist möglich, jedoch nicht zu empfehlen, da es zu hohen Zeitverzögerungen bei der Ausführung kommen kann.

### Häufigkeit der Ausführung von "MB\_SLAVE"

Die Anweisung "MB\_SLAVE" muss regelmäßig ausgeführt werden, um jede Anforderung des Modbus-Masters zu empfangen und entsprechend zu antworten. Die Häufigkeit der

Ausführung von "MB\_SLAVE" richtet sich nach dem vom Modbus-Master vorgegebenen Timeout-Wert für die Antwort. Dies ist in der nachstehenden Abbildung dargestellt.



Der Timeout-Zeitraum für die Antwort ist die Zeitdauer, die ein Modbus-Master auf den Beginn einer Antwort von einem Modbus-Slave wartet. Dieser Zeitraum wird nicht vom Modbus-Protokoll definiert, sondern von einem Parameter des jeweiligen Modbus-Masters. Die Häufigkeit der Ausführung (Zeit zwischen einer Ausführung und der nächsten Ausführung) von "MB\_SLAVE" muss sich nach den jeweiligen Parametern des Modbus-Masters richten. Sie sollten "MB\_SLAVE" mindestens zweimal während des Timeout-Zeitraums für die Antwort des Modbus-Masters ausführen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB\_SLAVE":

| Parameter   | Deklaration | Datentyp | Speicherbereich              | Beschreibung  |
|-------------|-------------|----------|------------------------------|---|
| MB_ADDR     | Input       | UINT     | E, A, M, D, L oder Konstante | Stationsadresse des Modbus-Slaves <ul style="list-style-type: none"> <li>• Standard-Adressbereich: 0 bis 247</li> <li>• Erweiterter Adressbereich: 0 bis 65535</li> </ul> |
| MB_HOLD_REG | Input       | VARIANT  | D                            | Zeiger auf den Modbus-Halteregister-DB. Der DB muss mit der Zugriffsart "Standard - kompatibel mit S7-300/400" erstellt werden.   |
| NDR         | Output      | BOOL     | E, A, M, D, L                | Neue Daten bereit: <ul style="list-style-type: none"> <li>• 0: Keine neuen Daten</li> <li>• 1: Gibt an, dass neue Daten vom Modbus-Master geschrieben wurden</li> </ul>   |
| DR          | Output      | BOOL     | E, A, M, D, L                | Daten lesen: <ul style="list-style-type: none"> <li>• 0: Keine Daten gelesen</li> <li>• 1: Gibt an, dass Daten vom Modbus-Master gelesen wurden</li> </ul>                |
| ERROR       | Output      | BOOL     | E, A, M, D, L                | <ul style="list-style-type: none"> <li>• 0: Kein Fehler erkannt</li> <li>• 1: Fehler; ein entsprechender Fehlercode wird am Parameter STATUS ausgegeben</li> </ul>        |
| STATUS      | Output      | WORD     | E, A, M, D, L                | Fehlercode  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter STATUS**

| STATUS*<br>(W#16#....)  | Beschreibung   |   |
|---|--|---|
| 80C8  | Das angegebene Antwort-Timeout (siehe RCVTIME oder MSGTIME) ist "0".   |   |
| 80D1  | Der Empfänger hat eine Flusskontrolle zur Unterbrechung einer laufenden Sendung angefordert, aber die Sendung innerhalb der Wartezeit nicht wieder aktiviert.<br>Dieser Fehler wird auch bei der Hardware-Flusskontrolle erzeugt, wenn der Empfänger innerhalb der Wartezeit kein CTS erkennt. |   |
| 80D2  | Die Sendeanforderung wurde abgebrochen, weil vom DCE kein DSR-Signal empfangen wird.   |   |
| 80E0  | Die Meldung wurde beendet, weil der Empfangspuffer voll ist  |   |
| 80E1  | Die Meldung wurde wegen eines Paritätsfehlers abgebrochen  |   |
| 80E2  | Die Meldung wurde wegen eines Telegrammfehlers abgebrochen   |   |
| 80E3  | Die Meldung wurde wegen eines Überlauffehlers abgebrochen  |   |
| 80E4  | Die Meldung wurde abgebrochen, weil die angegebene Länge die Gesamtpuffergröße überschreitet   |   |
| 8180  | Ungültiger Wert für die Port-ID.   |   |
| 8186  | Ungültige Modbus-Stationsadresse   |   |
| 8187  | Ungültiger Pointer auf MB_HOLD_REG-DB  |   |
| 818C  | Pointer auf einen typsicheren DB-Typ MB_HOLD_REG (klassischer DB-Typ erforderlich)   |   |
| Antwortcode an Modbus-Master gesendet (B#16#...)  |  |   |
| 8380  | Keine Antwort  | CRC-Fehler  |
| 8381  | 01   | Funktionscode nicht unterstützt oder nicht unterstützt innerhalb eines Broadcasts       |
| 8382  | 03   | Fehler in der Datenlänge  |
| 8383  | 02   | Fehler in der Datenadresse oder Adresse außerhalb des gültigen Bereichs von MB_HOLD_REG |
| 8384  | 03   | Fehler im Datenwert   |
| 8385  | 03   | Wert des Datendiagnosecodes wird nicht unterstützt (Funktionscode 08)                   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |   |

**Instanz-DB der Anweisung "MB\_SLAVE"**

**Statische Variablen des Instanz-DBs**

Die folgende Tabelle beschreibt die statischen Variablen des Instanz-DBs der Anweisung, die Sie im Anwenderprogramm verwenden können. Ihr Programm kann in die Variablen HR\_Start\_Offset und Extended\_Addressinge Werte schreiben und die Modbus-Slave-Operationen steuern.

Die anderen Variablen können gelesen werden, um den Modbus-Zustand zu überwachen.



| Variable            | Datentyp | Beschreibung  |
|---------------------|----------|---|
| HR_Start_Offset     | WORD     | Startadresse des Modbus-Halteregisters (Standard="0")   |
| Extended_Addressing | BOOL     | Konfiguration der Adressierung: <ul style="list-style-type: none"> <li>• 0: Standard-Adressbereich (1 Byte)</li> <li>• 1: Erweiterter Adressbereich (2 Byte)</li> </ul> |
| Request_Count       | WORD     | Anzahl aller Anfragen, die durch den Slave empfangen wurden   |
| Slave_Message_Count | WORD     | Anzahl der Anfragen, die an diesen spezifischen Slave gesendet wurden   |
| Bad_CRC_Count       | WORD     | Anzahl der empfangenen Anfragen mit CRC-Fehler  |
| Broadcast_Count     | WORD     | Anzahl der empfangenen Broadcast-Anfragen   |
| Exception_Count     | WORD     | Anzahl der Modbus-spezifischer Fehler, die das Rücksenden einer Ausnahme erfordern  |
| Success_Count       | WORD     | Die Anzahl der für diesen spezifischen Slave empfangenen Anforderungen ohne Protokollfehler   |

## HR\_Start\_Offset

Die Adressen des Modbus-Halteregisters beginnen bei 40001 oder 40001. Diese Adressen entsprechen der Anfangsadresse des Halteregisters im Zielsystemspeicher. Über die Variable HR\_Start\_Offset können Sie den Versatz zu einer anderen Anfangsadresse vorgeben.

Beispiel: Ein Halteregister beginnt an MW100 und hat eine Länge von 100 WORD. Bei einem Versatz von 20 am Parameter HR\_Start\_Offset beginnt das Halteregister an der Adresse 40021 statt 40001. Jede Adresse unter 40021 und über 400119 führt zu einem Adressierungsfehler.

|         | HR_Start_Offset = 0 |                     | HR_Start_Offset = 20 |                     |
|---------|---------------------|---------------------|----------------------|---------------------|
|         | Modbus-Wortadresse  | S7-1200 Byteadresse | Modbus-Wortadresse   | S7-1200 Byteadresse |
| Minimum | 40001               | MW100               | 40021                | MW100               |
| Maximum | 40099               | MW198               | 40119                | MW198               |

## Extended\_Addressing

Für die Modbus-Slave-Adressierung kann ein einzelnes Byte (Standard-Adressbereich) oder ein doppeltes Byte (Erweiterter Adressbereich) konfiguriert werden. Die erweiterte Adressierung dient zur Adressierung von mehr als 247 Geräten in einem einzigen Netzwerk. Wenn Sie sich für die erweiterte Adressierung entscheiden, können Sie maximal 64.000

Adressen ansprechen. Im Folgenden wird ein Telegramm der Modbus-Funktion 1 als Beispiel gezeigt.

Tabelle 9-123 Slave-Adresse mit einem Byte (Byte 0)

| Funktion 1      | Byte 0        | Byte 1 | Byte 2         | Byte 3      | Byte 4           | Byte 5 |  |
|-----------------|---------------|--------|----------------|-------------|------------------|--------|--|
| Anforderung     | Slave-Adresse | F-Code | Anfangsadresse |             | Länge der Spulen |        |  |
| Gültige Antwort | Slave-Adresse | F-Code | Länge          | Spulendaten |                  |        |  |
| Fehlerantwort   | Slave-Adresse | 0x81   | E-Code         |             |                  |        |  |

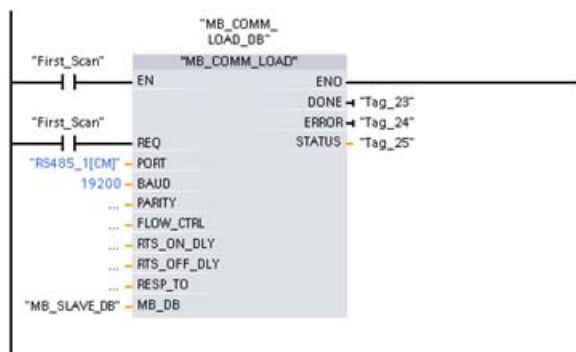
Tabelle 9-124 Slave-Adresse mit zwei Bytes (Byte 0 und Byte 1)

| Funktion 1      | Byte 0        | Byte 1 | Byte 2 | Byte 3         | Byte 4      | Byte 5           | Byte 6 |
|-----------------|---------------|--------|--------|----------------|-------------|------------------|--------|
| Anforderung     | Slave-Adresse |        | F-Code | Anfangsadresse |             | Länge der Spulen |        |
| Gültige Antwort | Slave-Adresse |        | F-Code | Länge          | Spulendaten |                  |        |
| Fehlerantwort   | Slave-Adresse |        | 0x81   | F-Code         |             |                  |        |

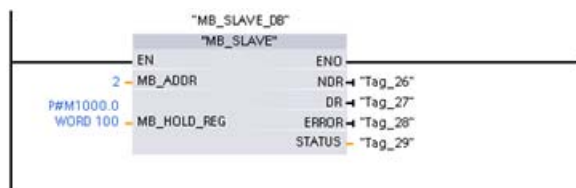
### Beispielprogramm für einen Modbus-Slave

#### Netzwerke (KOP)

**Netzwerk 1:** Parameter des RS-485-Moduls nur einmal während des ersten Zyklus initialisieren.



**Netzwerk 2:** In jedem Zyklus auf Anforderungen des Modbus-Master prüfen. Für das Modbus-Haltregister sind 100 Wörter mit Beginn an MW1000 konfiguriert.



**MODBUS (TCP)****MODBUS (TCP)****MB\_CLIENT: Als Modbus-TCP-Client über PROFINET kommunizieren****Beschreibung MB\_CLIENT****Beschreibung**

Die Anweisung "MB\_CLIENT" kommuniziert als Modbus-TCP-Client über die PROFINET-Verbindung der S7-1200 CPU. Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul. Über die Anweisung "MB\_CLIENT" bauen Sie eine Verbindung zwischen dem Client und dem Server auf, senden Anfragen und empfangen Antworten und steuern den Verbindungsabbau des Modbus-TCP-Servers.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "MB\_CLIENT":

| Parameter               | Deklaration | Datentyp | Beschreibung  |
|-------------------------|-------------|----------|---|
| REQ (Seite 2880)        | Input       | BOOL     | Kommunikationsanfrage mit dem Modbus TCP-Server bei steigender Flanke.<br><br>Mit der Kommunikationsanfrage wird der Instanz-DB für andere Clients gesperrt. Änderungen an den Eingangsparametern wirken sich erst aus, wenn eine Antwort des Servers erfolgt ist oder eine Fehlermeldung ausgegeben wurde. Wird während einer laufenden Abfrage der Parameter REQ erneut gesetzt, wird im Anschluss keine weitere Übertragung durchgeführt.  |
| DISCONNECT (Seite 2880) | Input       | BOOL     | Über den Parameter steuern Sie den Verbindungsaufbau- und abbau zu dem Modbus-Server: <ul style="list-style-type: none"> <li>• 0: Kommunikationsverbindung zu der angegebenen IP-Adresse und Port-Nummer herstellen.</li> <li>• 1: Kommunikationsverbindung trennen. Während des Verbindungsabbaus wird keine andere Funktion ausgeführt. Nach dem erfolgreichen Abbau der Verbindung wird am Parameter STATUS der Wert 7003 ausgegeben.</li> </ul> Ist der Parameter REQ beim Verbindungsaufbau gesetzt, wird die Abfrage sofort gesendet. |
| CONNECT_ID              | Input       | UINT     | Eindeutige ID zur Identifikation der Verbindung. Jeder Instanz der Anweisungen "MB_CLIENT" und "MB_SERVER (Seite 2885)" muss eine eindeutige Verbindungs-ID zugewiesen werden.  |
| IP_OCTET_1              | Input       | USINT    | 1. Oktett der IP-Adresse* des Modbus TCP -Servers.  |
| IP_OCTET_2              | Input       | USINT    | 2. Oktett der IP-Adresse* des Modbus TCP -Servers.  |
| IP_OCTET_3              | Input       | USINT    | 3. Oktett der IP-Adresse* des Modbus TCP -Servers.  |
| IP_OCTET_4              | Input       | USINT    | 4. Oktett der IP-Adresse* des Modbus TCP -Servers.  |

| Parameter                    | Deklaration | Datentyp | Beschreibung  |
|------------------------------|-------------|----------|---|
| IP_PORT                      | Input       | UINT     | IP-Port-Nummer des Servers, zu dem der Client die Verbindung herstellt und über das TCP/IP-Protokoll kommuniziert (Standard-Wert: 502).   |
| MB_MODE<br>(Seite 2880)      | Input       | USINT    | Auswahl des Modus der Anfrage (Lesen, Schreiben oder Diagnose).   |
| MB_DATA_ADDR<br>(Seite 2880) | Input       | UDINT    | Anfangsadresse der Daten, auf welche die Anweisung "MB_CLIENT" zugreift.  |
| DATA_LEN                     | Input       | UINT     | Datenlänge: Anzahl der Bits oder Wörter für den Datenzugriff (siehe "Parameter MB_MODE und MB_DATA_ADDR" - Datenlänge).   |
| MB_DATA_PTR<br>(Seite 2881)  | InOut       | VARIANT  | Zeiger auf das Modbus-Datenregister: Das Register ist ein Puffer für die vom Modbus-Server empfangenen oder zum Modbus-Server zu sendenden Daten. Der Zeiger muss auf einen globalen Datenbaustein mit Standardzugriff verweisen.<br>Die Anzahl der Bits, die adressiert werden, muss durch 8 teilbar sein. |
| DONE                         | Out         | BOOL     | Das Bit am Ausgangsparameter DONE wird auf "1" gesetzt, sobald der letzte Auftrag ohne Fehler ausgeführt wurde.   |
| BUSY                         | Out         | BOOL     | <ul style="list-style-type: none"> <li>0: Kein Auftrag von "MB_CLIENT" in Bearbeitung</li> <li>1: Auftrag von "MB_CLIENT" in Bearbeitung</li> </ul>   |
| ERROR                        | Out         | BOOL     | <ul style="list-style-type: none"> <li>0: Kein Fehler</li> <li>1: Fehler aufgetreten. Die Fehlerursache wird über den Parameter STATUS angezeigt.</li> </ul>  |
| STATUS<br>(Seite 2882)       | Out         | WORD     | Fehlercode der Anweisung.   |

\* 8 Bit langer Anteil der 32 Bit IPv4 IP-Adresse des Modbus TCP Servers.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Hinweis**

**Konsistente Input-Daten während eines "MB\_CLIENT"-Aufrufs**

Sobald ein Modbus-Client eine Modbus-Anweisung aufruft, wird der Status der Input-Parameter intern gespeichert und dann bei Folgeaufruf verglichen. Der Vergleich wird verwendet um zu bestimmen, ob dieser bestimmte Aufruf die laufende Abfrage initialisiert hat. Mehrere Aufrufe von "MB\_CLIENT" können durchgeführt werden, indem Sie einen gemeinsamen Instanz-DB verwenden. Die Werte der Input-Parameter dürfen hierbei nicht verändert werden, solange eine "MB\_CLIENT"-Instanz ausgeführt wird. Werden die Input-Parameter während der Ausführung verändert, kann durch "MB\_CLIENT" nicht überprüft werden, ob die Instanz gerade ausgeführt wird.

**Mehrfachverbindungen von Clients**

Ein Modbus TCP-Client kann mehrere TCP-Verbindungen unterstützen, wobei die maximale Anzahl der Verbindungen von der verwendeten CPU abhängt. Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus-TCP Clients und Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten. Modbus-TCP Verbindungen können hierbei auch gemeinsam von Client- und/oder Server-Verbindungen genutzt werden.

Bei einzelnen Client-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB\_CLIENT"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Für jede "MB\_CLIENT"-Verbindung muss eine eindeutige IP-Adresse des Servers spezifiziert sein.
- Jede "MB\_CLIENT"-Verbindung benötigt eine eindeutige Verbindungs-ID. Die einzelne Verbindungs-ID muss jeweils für jeden einzelnen Instanz-DB der Anweisung verwendet werden. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.
- Eindeutige Nummern des IP-Ports werden je nach Server-Konfiguration benötigt oder nicht.

### Statische Variablen der Anweisung

Die folgende Tabelle beschreibt die editierbaren statischen Variablen des Instanz-Datenbausteins der Anweisung "MB\_CLIENT".

| Variable             | Datentyp | Startwert | Beschreibung  |
|----------------------|----------|-----------|---|
| Blocked_Proc_Timeout | REAL     | 3.0       | Wartezeit in Sekunden, bevor bei einer blockierten Modbus-Instanz die statische Variable ACTIVE rückgesetzt wird. Dies kann zum Beispiel auftreten, wenn bei einer ausgegebenen Client-Anfrage die Ausführung der Client-Funktion abgebrochen wird, bevor die Anfrage vollständig ausgeführt wurde. Die maximale Wartezeit beträgt 55 Sekunden. |
| MB_Transaction_ID    | WORD     | 1         | Transaktions-ID des Modbus-TCP-Protokolls. Der Startwert von "1" sollte nur geändert werden, wenn der Modbus-TCP Server einen anderen Wert benötigt.  |
| MB_Unit_ID           | WORD     | 65535     | Unit-ID des Modbus-Protokolls. Die Variable entspricht der Slave-Adresse des Modbus-RTU-Protokolls. Ändern Sie diesen Wert nur, wenn der Modbus-TCP-Server als Gateway verwendet werden kann und durch das Anwendungsprogramm innerhalb des Modbus-Servers kontrolliert wird.   |
| RCV_TIMEOUT          | REAL     | 2.0       | Zeitintervall in Sekunden, in der die Anweisung "MB_CLIENT" auf eine Antwort des Servers wartet.  |
| Connected            | BOOL     | 0         | Gibt an, ob die Verbindung zum zugewiesenen Client verbunden oder nicht verbunden ist: 1 = verbunden, 0 = nicht verbunden.  |

### Siehe auch

MB\_CLIENT Beispiel 1: Mehrere Anfragen über eine TCP-Verbindung senden (Seite 2890)

MB\_CLIENT Beispiel 2: Mehrere Anfragen über mehrere TCP-Verbindungen senden (Seite 2891)

MB\_CLIENT Beispiel 3: Mehrere Anfragen koordinieren (Seite 2892)

## Parameter REQ und DISCONNECT

### Beschreibung

Wenn keine Instanz der Anweisung "MB\_CLIENT" ausgeführt wird und wenn am Parameter DISCONNECT der Wert "0" anliegt, wird bei steigender Flanke am Parameter REQ ein neuer Auftrag ausgeführt. Besteht noch keine Verbindung, wird diese bei der Ausführung hergestellt.

Wird die gleiche Instanz der Anweisung "MB\_CLIENT" erneut ausgeführt (DISCONNECT=0 und REQ=1), bevor der aktive Auftrag ausgeführt wurde, wird dieser nicht im Anschluss an den aktiven Auftrag ausgeführt. Erst nach Beendigung des aktiven Auftrags kann ein neuer Auftrag angestoßen werden (REQ=1).

Den Status der Ausführung können Sie über den Ausgangsparameter DONE überwachen. Diesen können Sie nutzen, um bei einer sequenziellen Ausführung der Anweisung "MB\_CLIENT" den Ausführungsstatus zu überwachen.

### Siehe auch

Beschreibung MB\_CLIENT (Seite 2877)

## Parameter MB\_MODE und MB\_DATA\_ADDR

### Beschreibung

Die Anweisung "MB\_CLIENT" verwendet anstelle eines Funktionscodes den Parameter MB\_MODE. Über den Parameter MB\_DATA\_ADDR legen Sie die Modbus-Startadresse fest der Daten fest, auf die Sie zugreifen möchten. Die Kombination der Parameter MB\_MODE und MB\_DATA\_ADDR legt den Funktionscode fest, der in der aktuellen Modbus-Nachricht verwendet wird.

Die folgende Tabelle zeigt den Zusammenhang zwischen dem Parameter MB\_MODE, der Modbus-Funktion und dem Adressraum.

| MB_MODE | Modbus Funktion | Datenlänge | Funktion und Datentyp                             | MB_DATA_ADDR    |
|---------|-----------------|------------|---|-----------------|
| 0       | 01              | 1 bis 2000 | Output-Bits lesen:<br>1 bis 2000 Bits pro Aufruf  | 1 bis 9999      |
| 0       | 02              | 1 bis 2000 | Input-Bits lesen:<br>1 bis 2000 Bits pro Aufruf   | 10001 bis 19999 |
| 0       | 03              | 1 bis 125  | Halteregister lesen:<br>1 bis 125 WORD pro Aufruf | 40001 bis 49999 |
| 0       | 04              | 1 bis 125  | Input-Wörter lesen:<br>1 bis 125 WORD pro Aufruf  | 30001 bis 39999 |
| 1       | 05              | 1          | Ein Output-Bit schreiben:<br>Ein Bit pro Aufruf   | 1 bis 9999      |
| 1       | 06              | 1          | Ein Halteregister schreiben:<br>1 WORD pro Aufruf | 40001 bis 49999 |

| MB_MODE                               | Modbus Funktion | Datenlänge | Funktion und Datentyp   | MB_DATA_ADDR    |
|---------------------------------------|-----------------|------------|---|-----------------|
| 1                                     | 15              | 2 bis 1968 | Mehrere Output-Bits schreiben:<br>2 bis 1968 Bits pro Aufruf  | 1 bis 9999      |
| 1                                     | 16              | 2 bis 123  | Mehrere Haltereister schreiben:<br>2 bis 123 WORD pro Aufruf  | 40001 bis 49999 |
| 2                                     | 15              | 1 bis 1968 | Ein oder mehrere Output-Bits schreiben:<br>1 bis 1968 Bits pro Aufruf   | 1 bis 9999      |
| 2                                     | 16              | 1 bis 123  | Ein oder mehrere Haltereister schreiben:<br>1 bis 123 WORD pro Aufruf   | 40001 bis 49999 |
| 11                                    | 11              | 0          | Status-Wort und Ereigniszähler der Server-Kommunikation lesen: <ul style="list-style-type: none"> <li>Das Status-Wort bildet den Bearbeitungszustand ab (0 - nicht in Bearbeitung, 0xFFFF - in Bearbeitung).</li> <li>Der Ergebniszähler wird nach jedem erfolgreichen Senden einer Nachricht inkrementiert.</li> </ul> Die Parameter MB_DATA_ADDR und MB_DATA_LEN der Anweisung "MB_CLIENT" werden bei der Ausführung dieser Funktion nicht ausgewertet. | -               |
| 80                                    | 08              | 1          | Überprüfung des Server-Status durch den Fehlercode 0x0000 (Rückschleifentest - der Server sendet die Abfrage zurück):<br>1 WORD pro Aufruf  | -               |
| 81                                    | 08              | 1          | Ereigniszähler des Servers über den Fehlercode 0x000A rücksetzen:<br>1 WORD pro Aufruf  |                 |
| 3 bis 10,<br>12 bis 79,<br>82 bis 255 |                 |            | reserviert  |                 |

## Siehe auch

Beschreibung MB\_CLIENT (Seite 2877)

## Parameter MB\_DATA\_PTR

### Beschreibung

Der Parameter MB\_DATA\_PTR ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Datenpuffer können Sie einen globalen Datenbaustein oder einen Speicherbereich (M) verwenden.

Verwenden Sie für einen Puffer im Speicherbereich (M) einen Zeiger im Format ANY nach dem Muster "P#Bit-Adresse" "Datentyp" "Länge" (Beispiel: P#M1000.0 WORD 500).

Der Parameter MB\_DATA\_PTR verwendet einen Kommunikationspuffer:

- Bei den Kommunikationsfunktionen der Anweisung "MB\_CLIENT":
  - Lesen und schreiben von 1-Bit Daten der Modbus-Server Adressen 00001 bis 09999 und 10001 bis 19999.
  - Lesen von 16-Bit WORD-Daten der Modbus-Server Adressen 30001 bis 39999 und 40001 bis 49999.
  - Schreiben von 16-Bit WORD-Daten der Modbus-Server Adressen 40001 bis 49999.
- Bei der Datenübertragung (Länge: Bit oder WORD) von oder zu dem globalen DB oder dem Speicherbereich (M), den Sie über den Parameter MB\_DATA\_PTR zugewiesen haben.

Wenn Sie für den Zeiger auf den Puffer am Parameter MB\_DATA\_PTR einen Datenbaustein verwenden, müssen Sie den DB-Elementen Datentypen zuweisen.

- Verwenden Sie den 1-Bit Datentyp BOOL für eine Modbus Bit-Adresse
- Verwenden Sie einen 16-Bit Datentyp wie WORD, UINT, INT oder REAL für eine Modbus WORD-Adresse.
- Verwenden Sie einen 32-Bit Datentyp (Doppelwort) wie DWORD, DINT oder REAL für zwei Modbus WORD-Adressen.
- Über MB\_DATA\_PTR können Sie auch auf komplexe DB-Elemente zugreifen wie:
  - Standard-Arrays
  - Strukturen mit eindeutigen Element-Namen
  - Komplexe Strukturen mit eindeutiger Benennung der Elemente Datentypplängen von 16 oder 32 Bit.
- Die Datenbereiche für den Parameter MB\_DATA\_PTR können auch unterschiedlichen globalen Datenbausteinen stehen (oder in unterschiedlichen Speicherbereichen). Sie können zum Beispiel einen Datenbaustein für die Lesevorgänge und einen anderen für die Schreibvorgänge verwenden oder für jede "MB\_CLIENT"-Station einen eigenen Datenbaustein.

**Siehe auch**

Beschreibung MB\_CLIENT (Seite 2877)

**Parameter STATUS**

**Parameter STATUS (allgemeine Statusinformationen)**

| STATUS* (W#16#) | Beschreibung                     |
|-----------------|----------------------------------|
| 0000            | Anweisung fehlerfrei ausgeführt. |
| 0001            | Verbindung hergestellt.          |
| 0003            | Verbindungsabbau durchgeführt.   |



| STATUS* (W#16#) | Beschreibung  |
|-----------------|---|
| 7000            | Kein Aufruf aktiv (REQ=0).  |
| 7001            | Erstaufruf mit REQ=1: Bearbeitung angestoßen; BUSY hat den Wert 1.  |
| 7002            | Zwischenaufruf (REQ irrelevant). Bearbeitung bereits aktiv; BUSY hat den Wert 1.  |
| 7003            | Verbindung wird abgebaut.   |
| 7004            | Verbindung hergestellt und überwacht. Keine Auftragsbearbeitung aktiv.  |
| 7005            | Daten wurden gesendet.  |
| 7006            | Daten wurden empfangen.   |
| 80BB            | Ungültiger Wert am Parameter ACTIVE_EST (Kennung für die Art des Verbindungsaufbaus, siehe T_CON_PARAM): <ul style="list-style-type: none"> <li>• Serverseitig ist nur der passive Verbindungsaufbau zulässig (ACTIVE_EST = FALSE).</li> <li>• Clientseitig ist nur der aktive Verbindungsaufbau zulässig (ACTIVE_EST = TRUE).</li> </ul> |
| 8380            | Empfangenes Modbus-Frame hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.  |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Parameter STATUS (Protokoll-Fehler)

| STATUS* (W#16#) | Code der Antwort an den Modbus-Client (B#16#) | Beschreibung   |
|-----------------|---|--|
| 8381            | 01  | Funktions-Code wird nicht unterstützt.   |
| 8382            | 03  | Fehler bei Datenlänge.   |
| 8383            | 02  | Fehler bei Daten-Adresse oder Zugriff außerhalb des Adressbereichs von MB_DATA_PTR (Seite 2881). |
| 8384            | 03  | Fehler bei Datenwert.  |
| 8385            | 03  | Fehler-Codes der Diagnose nicht unterstützt (Funktions-Code 08).                                 |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Parameter STATUS (Parameter-Fehler)

Zusätzlich zu den Fehlern, die in der folgenden Tabelle aufgelistet sind, können bei der Anweisung "MB\_CLIENT" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND" und "TRCV") auftreten.

| STATUS* (W#16#) | Beschreibung   |
|-----------------|--|
| 80C8            | Keine Antwort des Servers im definierten Zeitraum. Überprüfen Sie die Verbindung zu dem Modbus-Server. Dieser Fehler wird erst nach Abschluss der konfigurierten erneuten Versuche gemeldet.<br>Empfängt die Anweisung "MB_CLIENT" keine Antwort mit der ursprünglich übermittelten Transaktions-ID (Variable MB_TRANSACTION_ID) innerhalb des definierten Zeitraums, wird dieser Fehlercode ausgegeben. |
| 8188            | Ungültiger Wert am Parameter MB_MODE.  |
| 8189            | Ungültige Adressierung der Daten am Parameter MB_DATA_ADDR.  |
| 818A            | Ungültige Datenlänge am Parameter MB_DATA_LEN.   |

| STATUS*<br>(W#16#)  | Beschreibung   |
|---|--|
| 818B  | Ungültiger Zeiger am Parameter MB_DATA_PTR. Überprüfen Sie zudem die Werte der Parameter MB_DATA_ADDR (Seite 2880) und MB_DATA_LEN.  |
| 818C  | <ul style="list-style-type: none"> <li>• Der Zeiger am Parameter MB_DATA_PTR (Seite 2881) verweist auf einen optimierten Datenbaustein. Verwenden Sie entweder einen Datenbaustein mit Standardzugriff oder einen Speicherbereich.</li> <li>• Zeitüberschreitung bei dem Parameter BLOCKED_PROC_TIMEOUT (siehe statische Variablen der Anweisung). Das Limit von 55 Sekunden wurde überschritten.</li> </ul> |
| 818D  | Die Transaktions-ID (Variable MB_TRANSACTION_ID) entspricht nicht der ursprünglich gesendeten (siehe statische Variablen der Anweisung).   |
| 8200  | <ul style="list-style-type: none"> <li>• Eine weitere Modbus-Anfrage wird zur Zeit über den Port verarbeitet.</li> <li>• Eine weitere Instanz von MB_CLIENT mit den gleichen Verbindungsparametern bearbeitet eine bereits bestehende Modbus-Anfrage.</li> </ul>   |
| 8380  | Empfänger Modbus-Datenübertragungsblock ist nicht wohlgeformt oder zu wenige Bytes wurden empfangen.   |
| 8386  | Empfänger Funktions-Code entspricht nicht dem ursprünglich gesendetem.   |
| 8387  | <ul style="list-style-type: none"> <li>• Die zugewiesene Verbindungs-ID ist eine andere als bei vorherigen Abfragen. Es kann nur eine Verbindungs-ID für jeden Instanz-DB der Anweisung "MB_CLIENT" verwendet werden.</li> <li>• Der Fehlercode wird auch ausgegeben, wenn die ID des vom Server empfangenen Modbus TCP Protokolls ungleich "0" ist.</li> </ul>  |
| 8388  | Durch den Modbus-Server wurde eine andere Datenlänge gesendet, als abgefragt wurde. Dieser Fehler tritt nur bei Verwendung der Modbus-Funktionen 15 oder 16 auf.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**Hinweis**

**Fehlercodes der intern verwendeten Kommunikationsanweisungen.**

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB\_CLIENT" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND" und "TRCV") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB\_CLIENT" möglich. Im Abschnitt Static werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

**Siehe auch**

Beschreibung MB\_CLIENT (Seite 2877)

Parameter MB\_HOLD\_REG (Seite 2888)

## MB\_SERVER: Als Modbus-TCP-Server über PROFINET kommunizieren

### Beschreibung MB\_SERVER

#### Beschreibung

Die Anweisung "MB\_SERVER" kommuniziert als Modbus-TCP-Server über die PROFINET-Verbindung der S7-1200 CPU. Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul. Die Anweisung "MB\_SERVER" verarbeitet Verbindungsanfragen eines Modbus-TCP-Clients, empfängt Anfragen von Modbus-Funktionen und sendet Antwort-Meldungen.

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB\_SERVER":

| Parameter                   | Deklaration | Datentyp | Beschreibung  |
|-----------------------------|-------------|----------|---|
| DISCONNECT                  | Input       | BOOL     | Die Anweisung "MB_SERVER" geht eine passive Verbindung mit einer Partner-Baugruppe ein, d.h. der Server reagiert auf eine TCP-Verbindungsanfrage von jeder anfragenden IP-Adresse. Über diesen Parameter können Sie steuern, wann eine Verbindungsanfrage akzeptiert wird: <ul style="list-style-type: none"> <li>• 0: Wenn keine Kommunikationsverbindung besteht, wird eine passive Verbindung aufgebaut.</li> <li>• 1: Initialisierung des Verbindungsabbaus. Ist der Eingang gesetzt, werden keine anderen Vorgänge ausgeführt. Nach erfolgreichem Verbindungsabbau wird am Parameter STATUS der Wert 7003 ausgegeben.</li> </ul> |
| CONNECT_ID                  | Input       | UINT     | Über den Parameter wird eine Verbindung innerhalb der CPU eindeutig identifiziert. Jede einzelne Instanz der Anweisungen "MB_CLIENT (Seite 2877)" und "MB_SERVER" muss eine eindeutige ID am Parameter CONNECT_ID verwenden.  |
| IP_PORT                     | Input       | UINT     | Startwert=502. Die Nummer des IP-Ports legt fest, welcher IP-Port für Verbindungsanfragen des Modbus-Clients überwacht wird.<br>Die folgenden TCP-Port-Nummern dürfen nicht für die passive Verbindung der "MB_SERVER"-Anweisung verwendet werden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964.  |
| MB_HOLD_REG<br>(Seite 2888) | InOut       | VARIANT  | Zeiger auf das Modbus-Halteregister der "MB_SERVER"-Anweisung: Verwenden Sie als Halteregister einen globalen Datenbaustein mit Standardzugriff. Das Halteregister beinhaltet die Werte, auf die ein Modbus-Client über die Modbus-Funktionen 3 (lesen), 6 (schreiben) und 16 (lesen) zugreifen darf.   |
| NDR                         | Output      | BOOL     | "New Data Ready": <ul style="list-style-type: none"> <li>• 0: Keine neuen Daten</li> <li>• 1: Neue Daten durch den Modbus-Client geschrieben</li> </ul>   |
| DR                          | Output      | BOOL     | "Data Read": <ul style="list-style-type: none"> <li>• 0: Keine Daten gelesen</li> <li>• 1: Daten durch den Modbus-Client gelesen</li> </ul>   |

| Parameter              | Deklaration | Datentyp | Beschreibung  |
|------------------------|-------------|----------|---|
| ERROR                  | Output      | BOOL     | Tritt während eines Aufrufs der Anweisung "MB_SERVER" ein Fehler auf, wird der Ausgang am Parameter ERROR auf TRUE gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS angezeigt. |
| STATUS<br>(Seite 2889) | Output      | WORD     | Fehlercode der Anweisung.   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Abbilden der Modbus-Adressen auf dem Prozessabbild

Die Anweisung "MB\_SERVER" ermöglicht eingehenden Modbus-Funktionen (1, 2, 4, 5 und 15) direkt lesend und schreibend auf das Prozessabbild der Eingänge und Ausgänge der S7-1200 CPU zuzugreifen (Verwendung der Datentypen BOOL und WORD).

Für den Datentransfer der Funktionscodes 3, 6 und 16 muss das Halteregeister (Parameter MB\_HOLD\_REG) länger als ein Byte definiert werden. Die folgende Tabelle zeigt die Abbildung der Modbus-Adressen auf das Prozessabbild der CPU.

| Modbus Funktion |                 |              |            |     |       | S7-1200               |                  |
|-----------------|-----------------|--------------|------------|-----|-------|-----------------------|------------------|
| Code            | Funktion        | Datenbereich | Adressraum |     |       | Datenbereich          | CPU Adresse      |
| 01              | Lesen: Bits     | Output       | 1          | bis | 8192  | Prozessabbild Ausgang | A0.0 bis A1023.7 |
| 02              | Lesen: Bits     | Input        | 10001      | bis | 18192 | Prozessabbild Eingang | E0.0 bis E1023.7 |
| 04              | Lesen: WORD     | Input        | 30001      | bis | 30512 | Prozessabbild Eingang | EW0 bis EW1022   |
| 05              | Schreiben: Bits | Output       | 1          | bis | 8192  | Prozessabbild Ausgang | A0.0 bis A1023.7 |
| 15              | Schreiben: Bits | Output       | 1          | bis | 8192  | Prozessabbild Ausgang | A0.0 bis A1023.7 |

Eingehende Modbus-Meldungen mit den Funktionscodes 3, 6 und 16 schreiben oder lesen vom Modbus-Halteregister (das Halteregeister spezifizieren Sie am Parameter MB\_HOLD\_REG).

### Mehrfache Server-Verbindungen

Sie können mehrfache Server-Verbindungen erstellen. Dies erlaubt es einer einzelnen CPU gleichzeitig Verbindungen zu mehreren Modbus-TCP-Clients herzustellen.

Ein Modbus-TCP-Server kann mehrere TCP-Verbindungen unterstützen, wobei die maximale Anzahl der Verbindungen von der verwendeten CPU abhängt.

Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus-TCP Clients und Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten.

Modbus-TCP Verbindungen können hierbei auch gemeinsam von Client- und/oder Server-Verbindungen genutzt werden.

Bei Server-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB\_SERVER"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Jede "MB\_SERVER"-Verbindung muss mit einer eindeutigen Nummer des IP-Ports erstellt werden. Für jeden Port wird nur eine Verbindung unterstützt.

- Jede "MB\_SERVER"-Verbindung muss eine eindeutige Verbindungs-ID verwenden. Die einzelne Verbindungs-ID muss jeweils für jeden einzelnen Instanz-DB der Anweisung verwendet werden. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.
- Für jede Verbindung muss die Anweisung "MB\_SERVER" einzeln aufgerufen werden.

## Modbus-Diagnosefunktionen

Die folgende Tabelle enthält die Beschreibung der Modbus-Diagnosefunktionen.

| Code | Sub-Funktion | Beschreibung  |
|------|--------------|---|
| 08   | 0x0000       | Echo-Test: Die Anweisung "MB_SERVER" empfängt ein Datenwort und sendet dieses unverändert an den Modbus-Master zurück.  |
| 08   | 0x000A       | Ereigniszähler zurücksetzen: Die Anweisung "MB_SERVER" setzt den Ereigniszähler für die Kommunikation zurück, welcher für die Modbus-Funktion 11 verwendet wird.  |
| 11   | -            | Ereigniszähler der Kommunikation holen: Die Anweisung "MB_SERVER" verwendet einen internen Ereigniszähler bei der Kommunikation zur Erfassung der Anzahl der erfolgreich ausgeführten Lese- und Schreibanfragen, die an den Modbus-Server gesendet werden.<br>Der Ereigniszähler wird bei den Funktionen 8, 11 oder Broadcast-Anfragen nicht inkrementiert. Gleiches gilt für Anfragen, die einen Kommunikationsfehler zur Folge haben (zum Beispiel bei Paritätsfehlern oder CRC-Fehlern). Die Broadcast-Funktion ist für Modbus-TCP nicht verfügbar, da nur eine Client/Server-Verbindung gleichzeitig bestehen kann. |

## Statische Variablen der Anweisung

Die folgende Tabelle beschreibt die statischen Variablen des Instanz-Datenbausteins der Anweisung "MB\_SERVER", die im Programm verwendet werden können. Auf die Variable HR\_Start\_Offset können Sie schreibend zugreifen. Auf die übrigen Variablen können Sie lesend zugreifen, um den Modbus-Status zu überwachen.

| Variable             | Datentyp | Startwert | Beschreibung  |
|----------------------|----------|-----------|---|
| HR_Start_Offset      | WORD     | 0         | Zuweisen der Anfangsadresse des Modbus-Halteregisters.  |
| Request_Count        | WORD     | 0         | Anzahl aller Anfragen, die durch den Server empfangen wurden.   |
| Server_Message_Count | WORD     | 0         | Anzahl aller empfangenen Meldungen für den jeweiligen Server.   |
| Xmt_Rcv_Count        | WORD     | 0         | Zähler zur Erfassung der Anzahl der Übertragungen, bei denen ein Fehler aufgetreten ist. Der Zähler wird auch erhöht, wenn eine ungültige Modbus-Meldung empfangen wurde. |
| Exception_Count      | WORD     | 0         | Zähler zur Erfassung der Anzahl von Fehlern, die spezifisch für Modbus sind und ein Ausnahmefehler hervorrufen.   |
| Success_Count        | WORD     | 0         | Zähler zur Erfassung der Anzahl der Anfragen, die keine Fehler in dem übertragenen Protokoll enthalten.   |
| Connected            | BOOL     | 0         | Gibt an, ob die Verbindung zum zugewiesenen Client verbunden oder nicht verbunden ist: 1 = verbunden, 0 = nicht verbunden.  |

**Beispiel: Adressierung über statische Variable HR\_Start\_Offset**

Die Adressen des Modbus-Haltregisters beginnen bei 40001. Diese Adressen entsprechen dem Adressraum des CPU-Speicherbereichs für das Haltregister. Sie können die Variable HR\_Start\_Offset auch so definieren, dass das Modbus-Haltregister eine andere Anfangsadresse als 40001 erhält.

Beispiel: Das Haltregister beginnt am MW100 und hat eine Länge von 100 WORD. Ein Offset-Wert am Parameter HR\_Start\_Offset bewirkt, dass sich die Anfangsadresse des Haltregisters von 40001 auf 40021 verschiebt. Jede Adressierung des Haltregisters, die unterhalb der Adresse 40021 und oberhalb der Adresse 40119 liegt, ruft einen Fehler hervor.

| HR_Start_Offset | Adresse               | Minimum | Maximum |
|-----------------|-----------------------|---------|---------|
| 0               | Modbus-Adresse (WORD) | 40001   | 40099   |
|                 | S7-1200-Adresse       | MW100   | MW298   |
| 20              | Modbus-Adresse (WORD) | 40021   | 40119   |
|                 | S7-1200-Adresse       | MW100   | MW298   |

**Siehe auch**

MB\_SERVER Beispiel: Mehrere TCP-Verbindungen (Seite 2893)

**Parameter MB\_HOLD\_REG**

**Beschreibung**

Der Parameter MB\_HOLD\_REG ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Datenpuffer können Sie einen globalen Datenbaustein oder einen Speicherbereich (M) verwenden.

Verwenden Sie als Zeiger auf einen Puffer im Speicherbereich (M) das Format ANY nach dem Muster "P#Bit-Adresse" "Datentyp" "Länge" (Beispiel: P#M1000.0 WORD 500).

Die folgende Tabelle zeigt einige Beispiele zur Abbildung von Modbus-Adressen auf das Haltregister für die Modbus-Funktionen 3 (WORD lesen), 6 (WORD schreiben) und 16 (mehrere WORD schreiben). Die Obergrenze für die Anzahl von Adressen im Datenbaustein ist durch den maximalen Arbeitsspeicher der CPU vorgegeben. Wenn Sie einen Speicherbereich verwenden, ist die Obergrenze für die Adressen durch die Größe des Speicherbereichs der CPU vorgegeben.

| Modbus-Adressen | MB_HOLD_REG Parameter - Beispiele |                     |                        |
|-----------------|-----------------------------------|---------------------|------------------------|
|                 | P#DB10.DBx0.0 WORD 5              | "Recipe".ingredient |                        |
| 40001           | MW100                             | DB10.DBW0           | "Recipe".ingredient[1] |
| 40002           | MW102                             | DB10.DBW2           | "Recipe".ingredient[2] |
| 40003           | MW104                             | DB10.DBW4           | "Recipe".ingredient[3] |
| 40004           | MW106                             | DB10.DBW6           | "Recipe".ingredient[4] |
| 40005           | MW108                             | DB10.DBW8           | "Recipe".ingredient[5] |

**Siehe auch**

Beschreibung MB\_SERVER (Seite 2885)

**Parameter STATUS****Parameter STATUS (allgemeine Statusinformationen)**

| STATUS* (W#16#)   | Beschreibung  |
|---|---|
| 0000  | Anweisung fehlerfrei ausgeführt.  |
| 0001  | Verbindung hergestellt.   |
| 0003  | Verbindungsabbau durchgeführt.  |
| 7000  | Kein Aufruf aktiv (REQ=0).  |
| 7001  | Erstaufruf mit REQ=1: Bearbeitung angestoßen; BUSY hat den Wert 1.  |
| 7002  | Zwischenaufruf (REQ irrelevant). Bearbeitung bereits aktiv; BUSY hat den Wert 1.  |
| 7003  | Verbindung wird abgebaut.   |
| 7004  | Verbindung hergestellt und überwacht. Keine Auftragsbearbeitung aktiv.  |
| 7005  | Daten wurden gesendet.  |
| 7006  | Daten wurden empfangen.   |
| 80BB  | Ungültiger Wert am Parameter ACTIVE_EST (Kennung für die Art des Verbindungsaufbaus, siehe T_CON_PARAM): <ul style="list-style-type: none"> <li>• Serverseitig ist nur der passive Verbindungsaufbau zulässig (ACTIVE_EST = FALSE).</li> <li>• Clientseitig ist nur der aktive Verbindungsaufbau zulässig (ACTIVE_EST = TRUE).</li> </ul> |
| 8380  | Empfangenes Modbus-Frame hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

**Parameter STATUS (Parameter-Fehler)**

| STATUS* (W#16#) | Code der Antwort an den Modbus-Server (B#16#) | Beschreibung   |
|-----------------|---|--|
| 8187            | Keine Antwort                                 | Ungültiger Zeiger am Parameter MB_HOLD_REG. Datenbereich ist zu klein.   |
| 818C            | Keine Antwort                                 | <ul style="list-style-type: none"> <li>• Der Zeiger am Parameter MB_HOLD_REG verweist auf einen optimierten Datenbaustein. Verwenden Sie entweder einen Datenbaustein mit Standardzugriff oder einen Speicherbereich.</li> <li>• Fehler durch Zeitüberschreitung der Ausführung (bei mehr als 55 Sekunden).</li> </ul> |
| 8381            | 01  | Funktions-Code wird nicht unterstützt.   |
| 8382            | 03  | Fehler bei Datenlänge  |
| 8383            | 02  | Fehler bei Datenadresse oder Zugriff außerhalb des Adressbereichs des Haltereisters (Parameter MB_HOLD_REG (Seite 2888)).  |
| 8384            | 03  | Fehler bei Datenwert   |

| STATUS* (W#16#)   | Code der Antwort an den Modbus-Server (B#16#) | Beschreibung   |
|---|---|--|
| 8385  | 03  | Wert des Diagnosecodes wird nicht unterstützt (nur bei Funktions-Code 08). |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |  |

---

**Hinweis**

**Fehlercodes der intern verwendeten Kommunikationsanweisungen.**

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB\_SERVER" noch Fehler bei den durch die Anweisung verwendeten Kommunikationsanweisungen ("TCON", "TDISCON", "TSEND" und "TRCV") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB\_SERVER" möglich. Im Abschnitt Static werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

---

**Siehe auch**

Beschreibung MB\_SERVER (Seite 2885)

**Beispiele**

**MB\_CLIENT Beispiel 1: Mehrere Anfragen über eine TCP-Verbindung senden**

**Beschreibung**

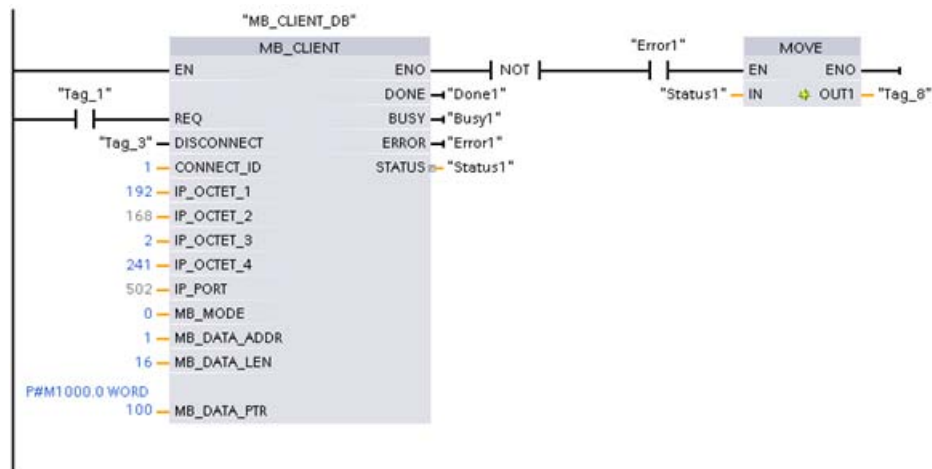
Über eine TCP-Verbindung können mehrere Anfragen des Modbus-Client gesendet werden. Hierzu verwenden Sie den gleichen Instanz-DB, die gleiche Verbindungs-ID und die gleiche Port-Nummer.

Zur gleichen Zeit kann immer nur ein Client aktiv sein. Nachdem die Ausführung eines Clients beendet wurde, wird der nächste Client ausgeführt. Die Reihenfolge der Ausführung müssen sie im Programm definieren.

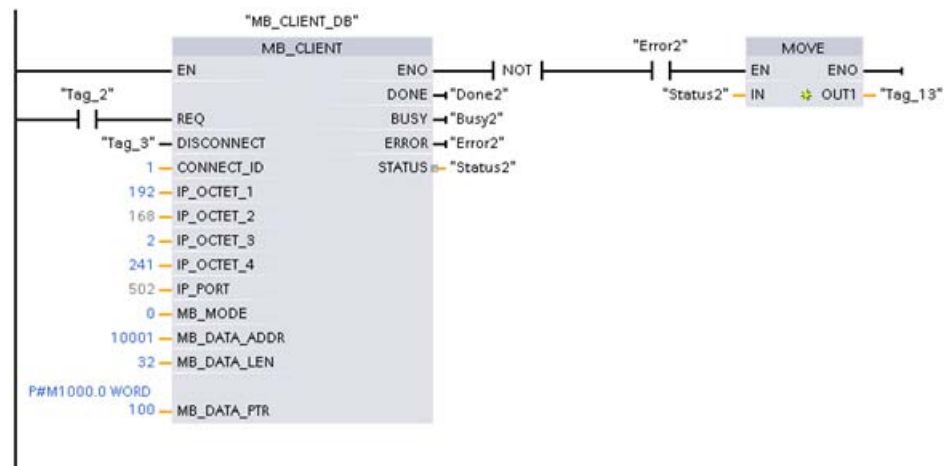
In dem folgenden Beispielprogramm wird als Ergänzung der Wert des Ausgangsparameters STATUS kopiert.



### Netzwerk 1: Modbus Funktion 1 - 16 Output-Bits lesen



### Netzwerk 2: Modbus Funktion 2 - 32 Input-Bits lesen



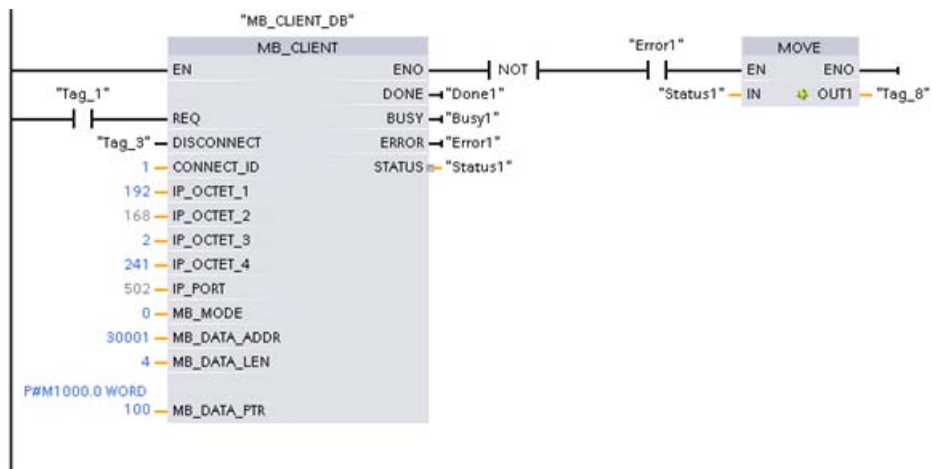
### MB\_CLIENT Beispiel 2: Mehrere Anfragen über mehrere TCP-Verbindungen senden

#### Beschreibung

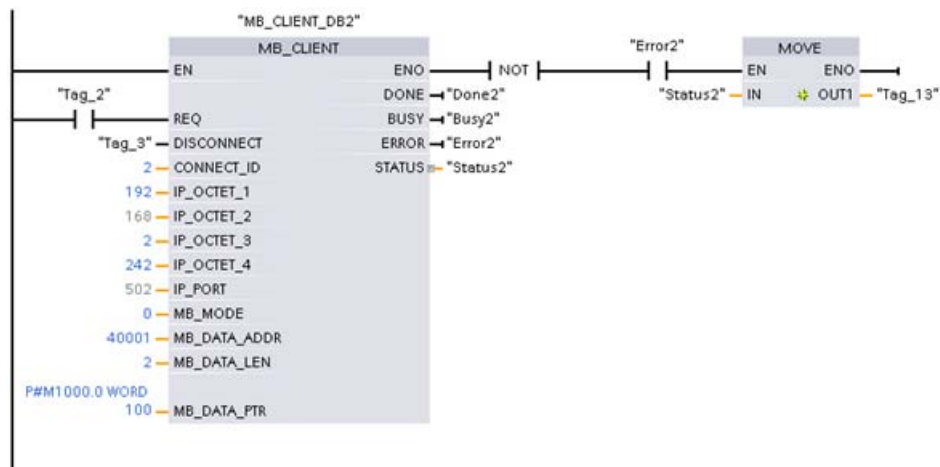
Anfragen des Modbus-Client können über verschiedene TCP-Verbindungen gesendet werden. Hierzu verwenden Sie jeweils einen unterschiedlichen Instanz-DB und eine andere Verbindungs-ID.

Verwenden Sie jeweils eine unterschiedliche Port-Nummer, wenn die Verbindungen zu dem gleichen Modbus-Server erfolgen. Erfolgen die Verbindungen zu jeweils unterschiedlichen Modbus-Servern, können Sie die Port-Nummer frei vergeben.

### Netzwerk 1: Modbus-Funktion 4 - Input (WORD) lesen



### Netzwerk 2: Modbus-Funktion 3 - Halteregeister (WORD) lesen

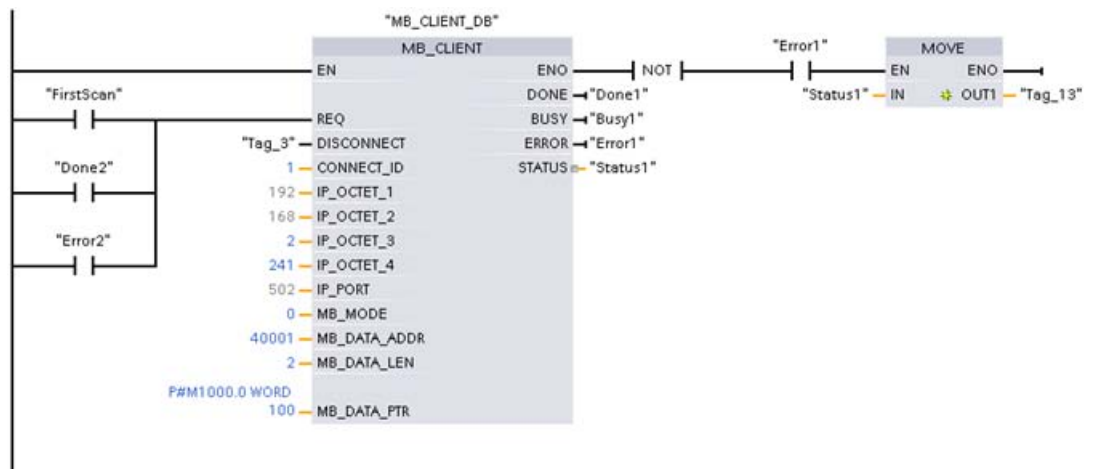


### MB\_CLIENT Beispiel 3: Mehrere Anfragen koordinieren

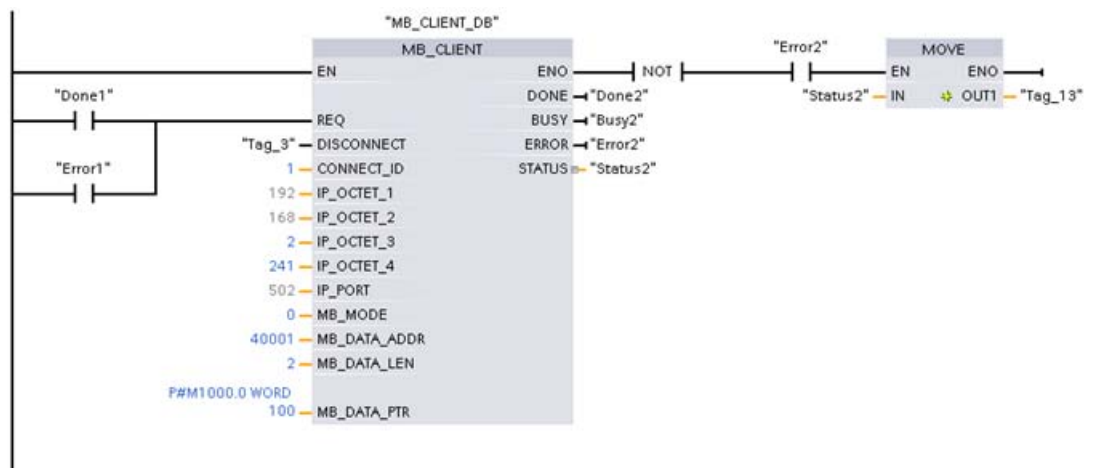
#### Beschreibung

Sie müssen sicherstellen, dass die einzelnen Modbus- Anfragen ausgeführt werden. Die Koordination der Anfragen steuern Sie über das Programm. Das folgende Beispiel zeigt, wie die Output-Parameter der ersten und zweiten Client-Anfrage verwendet werden können, um die Ausführung der Anweisungen zu koordinieren.

### Netzwerk 1: Modbus-Funktion 3 - Halteregeister lesen (WORD)



### Netzwerk 2: Modbus-Funktion 3 - Halteregeister lesen (WORD)



### MB\_SERVER Beispiel: Mehrere TCP-Verbindungen

#### Beschreibung

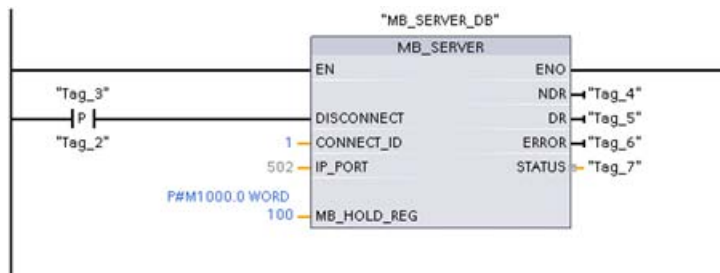
Sie können mehrere Modbus-TCP Server-Verbindungen verwenden. Hierzu muss die Anweisung "MB\_SERVER" für jede Verbindung unabhängig aufgerufen werden.

Jede Verbindung benötigt:

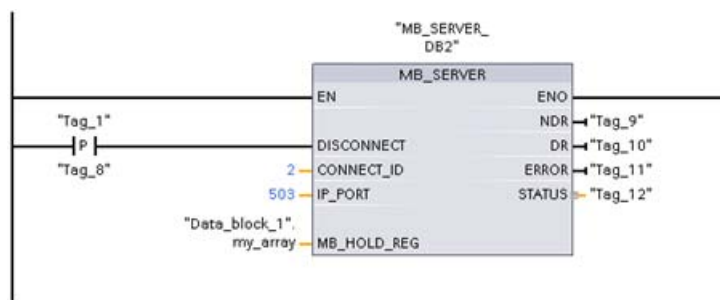
- einen unabhängigen Instanz-Datenbaustein der Anweisung
- eine eindeutige Verbindungs-ID
- einen eigenen IP-Port (bei der S7-1200 ist pro IP-Port nur eine Verbindung zulässig)

Zur Optimierung der Leistung sollte "MB\_SERVER" für jede Verbindung einmal pro Programmzyklus ausgeführt werden.

**Netzwerk 1: Verbindung #1 mit zugehörigem IP-Port, Verbindungs-ID und Instanz-DB**



**Netzwerk 2: Verbindung #1 mit zugehörigem IP-Port, Verbindungs-ID und Instanz-DB**



**MODBUS (TCP)**

**MB\_CLIENT: Als Modbus-TCP-Client über PROFINET kommunizieren**

**Beschreibung MB\_CLIENT**

**Beschreibung**

Die Anweisung "MB\_CLIENT" kommuniziert als Modbus-TCP-Client über die PROFINET-Verbindung. Über die Anweisung "MB\_CLIENT" bauen Sie eine Verbindung zwischen dem Client und dem Server auf, senden Modbus-Anfragen und empfangen Antworten und steuern den Verbindungsabbau des Modbus-TCP-Clients.

Die Anweisung "MB\_CLIENT" V3.0 kann sowohl für die S7-1500 als auch für die S7-1200 ab Version 4.0 verwendet werden. Die Verbindung kann über die lokale Schnittstelle der CPU oder CM/CP erfolgen.

Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul.

## Mehrfache Client-Verbindungen

Ein Modbus TCP-Client kann mehrere TCP-Verbindungen unterstützen, wobei die maximale Anzahl der Verbindungen von der verwendeten CPU abhängt. Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus-TCP Clients und Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten. Modbus-TCP Verbindungen können hierbei auch gemeinsam von "MB\_CLIENT"- und/oder "MB\_SERVER"-Instanzen genutzt werden.

Bei einzelnen Client-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB\_CLIENT"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Für jede "MB\_CLIENT"-Verbindung muss eine eindeutige IP-Adresse des Servers spezifiziert sein.
- Jede "MB\_CLIENT"-Verbindung benötigt eine eindeutige Verbindungs-ID.  
Die einzelne Verbindungs-ID muss jeweils für jeden einzelnen Instanz-DB der Anweisung verwendet werden. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.
- Eindeutige Nummern des IP-Ports werden je nach Server-Konfiguration benötigt oder nicht.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "MB\_CLIENT":

| Parameter                 | Deklaration | Datentyp | Beschreibung   |
|---------------------------|-------------|----------|--|
| REQ (Seite 2897)          | Input       | BOOL     | Kommunikationsanfrage mit dem Modbus TCP-Server bei steigender Flanke.<br><br>Mit der Kommunikationsanfrage wird der Instanz-DB für andere Clients gesperrt. Änderungen an den Eingangsparametern wirken sich erst aus, wenn eine Antwort des Servers erfolgt ist oder eine Fehlermeldung ausgegeben wurde. Wird während einer laufenden Modbus-Anfrage der Parameter REQ erneut gesetzt, wird im Anschluss keine weitere Übertragung durchgeführt.  |
| DISCONNECT (Seite 2897)   | Input       | BOOL     | Über den Parameter steuern Sie den Verbindungsaufbau- und abbau zu dem Modbus-Server: <ul style="list-style-type: none"> <li>• 0: Kommunikationsverbindung zu dem am Parameter CONNECT konfigurierten Verbindungspartner herstellen (siehe Parameter CONNECT).</li> <li>• 1: Kommunikationsverbindung trennen. Während des Verbindungsabbaus wird keine andere Funktion ausgeführt. Nach dem erfolgreichen Abbau der Verbindung wird am Parameter STATUS der Wert 0003 ausgegeben.</li> </ul> Ist der Parameter REQ beim Verbindungsaufbau gesetzt, wird die Modbus-Anfrage sofort gesendet. |
| MB_MODE (Seite 2897)      | Input       | USINT    | Auswahl des Modus der Modbus-Anfrage (Lesen, Schreiben oder Diagnose).   |
| MB_DATA_ADDR (Seite 2897) | Input       | UDINT    | Anfangsadresse der Daten, auf welche die Anweisung "MB_CLIENT" zugreift.   |
| MB_DATA_LEN               | Input       | UINT     | Datenlänge: Anzahl der Bits oder Wörter für den Datenzugriff (siehe "Parameter MB_MODE und MB_DATA_ADDR" - Datenlänge).  |

| Parameter                   | Deklaration | Datentyp | Beschreibung   |
|-----------------------------|-------------|----------|--|
| MB_DATA_PTR<br>(Seite 2900) | InOut       | VARIANT  | Zeiger auf das Modbus-Datenregister: Das Register ist ein Puffer für die vom Modbus-Server empfangenen oder zum Modbus-Server zu sendenden Daten. Der Zeiger muss auf einen globalen Datenbaustein mit optimiertem Zugriff verweisen.<br><br>Die Anzahl der Bits, die adressiert werden, muss durch 8 teilbar sein.  |
| CONNECT<br>(Seite 2901)     | InOut       | VARIANT  | Zeiger auf die Struktur der Verbindungsbeschreibung<br>Die folgenden Strukturen (SDTs) können verwendet werden: <ul style="list-style-type: none"> <li>• TCON_IP_v4: Enthält alle Adressierungsparameter, die für den Aufbau einer programmierten Verbindung benötigt werden. Bei Verwendung von TCON_IP_v4 wird die Verbindung beim Aufruf der Anweisung "MB_SERVER" hergestellt.</li> <li>• TCON_Configured: Enthält die Adressierungsparameter einer konfigurierten Verbindung. Bei Verwendung von TCON_Configured wird eine vorhandene Verbindung genutzt, die nach dem Laden der Hardware-Konfiguration durch die CPU hergestellt wurde.</li> </ul> |
| DONE                        | Out         | BOOL     | Das Bit am Ausgangsparameter DONE wird auf "1" gesetzt, sobald der letzte Auftrag ohne Fehler ausgeführt wurde.  |
| BUSY                        | Out         | BOOL     | <ul style="list-style-type: none"> <li>• 0: Keine Modbus-Anfrage in Bearbeitung</li> <li>• 1: Modbus-Anfrage wird bearbeitet</li> </ul> Der Ausgangsparameter BUSY wird beim Verbindungsaufbau und -abbau nicht gesetzt.   |
| ERROR                       | Out         | BOOL     | <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten. Die Fehlerursache wird über den Parameter STATUS angezeigt.</li> </ul>   |
| STATUS<br>(Seite 2903)      | Out         | WORD     | Detaillierte Statusinformation der Anweisung.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Hinweis**

**Konsistente Input-Daten während eines "MB\_CLIENT"-Aufrufs**

Sobald ein Modbus-Client Anweisung aufgerufen wird, wird der Status der Input-Parameter intern gespeichert und dann bei Folgeaufruf verglichen. Der Vergleich wird verwendet, um zu bestimmen, ob dieser bestimmte Aufruf die laufende Abfrage initialisiert hat. Mehrere Aufrufe von "MB\_CLIENT" können durchgeführt werden, indem Sie einen gemeinsamen Instanz-DB verwenden. Die Werte der Input-Parameter dürfen hierbei nicht verändert werden, solange eine "MB\_CLIENT"-Instanz ausgeführt wird. Werden die Input-Parameter während der Ausführung verändert, kann durch "MB\_CLIENT" nicht überprüft werden, ob die Instanz gerade ausgeführt wird.

## Statische Variablen der Anweisung

Die folgende Tabelle beschreibt die editierbaren statischen Variablen des Instanz-Datenbausteins der Anweisung "MB\_CLIENT".

| Variable             | Datentyp | Startwert | Beschreibung  |
|----------------------|----------|-----------|---|
| Blocked_Proc_Timeout | REAL     | 3.0       | Wartezeit in Sekunden, bevor bei einer blockierten Modbus-Instanz die statische Variable ACTIVE rückgesetzt wird. Dies kann zum Beispiel auftreten, wenn bei einer ausgegebenen Client-Anfrage die Ausführung der Client-Funktion abgebrochen wird, bevor die Anfrage vollständig ausgeführt wurde. Die maximale Wartezeit beträgt 55 Sekunden. |
| MB_Transaction_ID    | WORD     | 1         | Transaktions-ID des Modbus-TCP-Protokolls. Der Startwert von "1" sollte nur geändert werden, wenn der Modbus-TCP Server einen anderen Wert benötigt.  |
| MB_Unit_ID           | WORD     | 65535     | Unit-ID des Modbus-Protokolls. Die Variable entspricht der Slave-Adresse des Modbus-RTU-Protokolls. Ändern Sie diesen Wert nur, wenn der Modbus-TCP-Server als Gateway verwendet werden kann und durch das Anwendungsprogramm innerhalb des Modbus-Servers kontrolliert wird.   |
| RCV_TIMEOUT          | REAL     | 2.0       | Zeitintervall in Sekunden, in dem die Anweisung "MB_CLIENT" auf eine Antwort des Servers wartet.  |
| Connected            | BOOL     | 0         | Gibt an, ob die Verbindung zum zugewiesenen Server verbunden oder nicht verbunden ist: 1 = verbunden, 0 = nicht verbunden.  |

## Parameter REQ und DISCONNECT

### Beschreibung

Wenn keine Instanz der Anweisung "MB\_CLIENT" ausgeführt wird und wenn am Parameter DISCONNECT der Wert "0" anliegt, wird bei steigender Flanke am Parameter REQ ein neuer Auftrag ausgeführt. Besteht noch keine Verbindung, wird diese bei der Ausführung hergestellt.

Wird die gleiche Instanz der Anweisung "MB\_CLIENT" erneut ausgeführt (DISCONNECT=0 und REQ=1), bevor der aktive Auftrag ausgeführt wurde, wird dieser nicht im Anschluss an den aktiven Auftrag ausgeführt. Erst nach Beendigung des aktiven Auftrags kann ein neuer Auftrag angestoßen werden (REQ=1).

Der Status der Ausführung wird über die Ausgangsparameter ausgegeben. Diese können Sie nutzen, um bei einer sequenziellen Ausführung der Anweisung "MB\_CLIENT" den Ausführungsstatus zu überwachen.

## Parameter MB\_MODE und MB\_DATA\_ADDR

### Beschreibung

Die Anweisung "MB\_CLIENT" verwendet anstelle eines Funktionscodes den Parameter MB\_MODE. Über den Parameter MB\_DATA\_ADDR legen Sie die Modbus-Startadresse fest der Daten fest, auf die Sie zugreifen möchten.

9.7 Referenzen

Die Kombination der Parameter MB\_MODE, MB\_DATA\_ADDR und MB\_DATA\_LEN legt den Funktionscode fest, der in der aktuellen Modbus-Nachricht verwendet wird. Zum Beispiel:

- Funktionscode 5
  - MB\_MODE=1
  - MB\_DATA\_ADDR=1
  - MB\_DATA\_LEN=1
- Funktionscode 15
  - MB\_MODE=1
  - MB\_DATA\_ADDR=1
  - MB\_DATA\_LEN=2

Die folgende Tabelle zeigt den Zusammenhang zwischen den Eingangsparametern der Anweisung "MB\_CLIENT" und der Modbus-Funktion.

| MB_MODE | MB_DATA_ADDR  | MB_DATA_LEN   | Modbus Funktion | Funktion und Datentyp   |
|---------|---|---|-----------------|---|
| 0       | Anfangsadresse:<br>• 1 bis 9999                             | Datenlänge (Bits) pro Aufruf:<br>• 1 bis 2000               | 01              | Eingangs-Bits lesen:<br>• 1 bis 2000                          |
| 0       | Anfangsadresse:<br>• 10001 bis 19999                        | Datenlänge (Bits) pro Aufruf:<br>• 1 bis 2000               | 02              | Eingangs-Bits lesen:<br>• 1 bis 2000                          |
| 0       | Anfangsadresse:<br>• 40001 bis 49999<br>• 400001 bis 465535 | Datenlänge (WORD) pro Aufruf:<br>• 1 bis 125<br>• 1 bis 125 | 03              | Halteregister lesen:<br>• 0 bis 9998<br>• 0 bis 65534         |
| 0       | Anfangsadresse:<br>• 30001 bis 39999                        | Datenlänge (WORD) pro Aufruf:<br>• 1 bis 125                | 04              | Eingangswörter lesen:<br>• 0 bis 9998                         |
| 1       | Anfangsadresse:<br>• 1 bis 9999                             | Datenlänge (Bit) pro Aufruf:<br>• 1                         | 05              | Ein Ausgangsbit schreiben:<br>• 0 bis 9998                    |
| 1       | Anfangsadresse:<br>• 40001 bis 49999<br>• 400001 bis 465535 | Datenlänge (WORD) pro Aufruf:<br>• 1<br>• 1                 | 06              | Ein Halteregister schreiben:<br>• 0 bis 9998<br>• 0 bis 65524 |



| MB_MODE                               | MB_DATA_ADDR  | MB_DATA_LEN   | Modbus Funktion | Funktion und Datentyp  |
|---------------------------------------|---|---|-----------------|--|
| 1                                     | Anfangsadresse:<br>• 1 bis 9999   | Datenlänge (Bits) pro Aufruf:<br>• 2 bis 1968               | 15              | Mehrere Ausgangsbits schreiben:<br>• 0 bis 9998  |
| 1                                     | Anfangsadresse:<br>• 40001 bis 49999<br>• 400001 bis 465534   | Datenlänge (WORD) pro Aufruf:<br>• 2 bis 123<br>• 2 bis 123 | 16              | Mehrere Haltereister schreiben:<br>• 0 bis 9998<br>• 0 bis 65534   |
| 2                                     | Anfangsadresse:<br>• 1 bis 9999   | Datenlänge (Bits) pro Aufruf:<br>• 1 bis 1968               | 15              | Ein oder mehrere Ausgangsbits schreiben:<br>• 0 bis 9998   |
| 2                                     | Anfangsadresse:<br>• 40001 bis 49999<br>• 400001 bis 465535   | Datenlänge (WORD) pro Aufruf:<br>• 1 bis 123<br>• 1 bis 123 | 16              | Ein oder mehrere Haltereister schreiben:<br>• 0 bis 9998<br>• 0 bis 65534  |
| 11                                    | Die Parameter MB_DATA_ADDR und MB_DATA_LEN werden bei der Ausführung dieser Funktion nicht ausgewertet. |   | 11              | Status-Wort und Ereigniszähler des Servers lesen:<br>• Das Status-Wort bildet den Bearbeitungszustand ab (0 - nicht in Bearbeitung, 0xFFFF - in Bearbeitung).<br>• Der Ereigniszähler wird inkrementiert, wenn die Modbus-Anfrage erfolgreich ausgeführt wurde. Wenn bei der Ausführung einer Modbus-Funktion ein Fehler aufgetreten ist, wird zwar eine Nachricht vom Server gesendet, der Ereigniszähler jedoch nicht inkrementiert. |
| 80                                    | -   | Datenlänge (WORD) pro Aufruf:<br>• 1                        | 08              | Überprüfung des Server-Status durch den Diagnosecode 0x0000 (Rückschleifentest - der Server sendet die Abfrage zurück):<br>• 1 WORD pro Aufruf   |
| 81                                    | -   | Datenlänge (WORD) pro Aufruf:<br>• 1                        | 08              | Ereigniszähler des Servers über den Diagnosecode 0x000A rücksetzen:<br>• 1 WORD pro Aufruf   |
| 3 bis 10,<br>12 bis 79,<br>82 bis 255 |   |   |                 | reserviert   |

## Parameter MB\_DATA\_PTR

### Beschreibung

Der Parameter MB\_DATA\_PTR ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Datenpuffer können Sie einen globalen Datenbaustein oder einen Speicherbereich (M) verwenden.

Verwenden Sie für einen Puffer im Speicherbereich (M) einen Zeiger im Format ANY nach dem Muster "P#Bit-Adresse" "Datentyp" "Länge" (Beispiel: P#M1000.0 WORD 500).

Der Parameter MB\_DATA\_PTR verwendet einen Kommunikationspuffer:

Der Kommunikationspuffer wird verwendet:

- Bei den Kommunikationsfunktionen der Anweisung "MB\_CLIENT":
  - Lesen und schreiben von 1-Bit Daten der Modbus-Server Adressen 00001 bis 09999 und 10001 bis 19999.
  - Lesen von 16-Bit WORD-Daten der Modbus-Server Adressen 30001 bis 39999 und 40001 bis 49999.
  - Schreiben von 16-Bit WORD-Daten der Modbus-Server Adressen 40001 bis 49999.
- Bei der Datenübertragung (Länge: Bit oder WORD) von oder zu dem globalen DB oder dem Speicherbereich (M), den Sie über den Parameter MB\_DATA\_PTR zugewiesen haben.

Wenn Sie für den Zeiger auf den Puffer am Parameter MB\_DATA\_PTR einen Datenbaustein verwenden, müssen Sie den DB-Elementen Datentypen zuweisen.

- Verwenden Sie den 1-Bit Datentyp BOOL für eine Modbus Bit-Adresse
- Verwenden Sie einen 16-Bit Datentyp wie WORD, UINT, INT oder REAL für eine Modbus WORD-Adresse.
- Verwenden Sie einen 32-Bit Datentyp (Doppelwort) wie DWORD, DINT oder REAL für zwei Modbus WORD-Adressen.
- Über MB\_DATA\_PTR können Sie auch auf komplexe DB-Elemente zugreifen wie:
  - Standard-Arrays
  - Strukturen mit eindeutigen Element-Namen
  - Komplexe Strukturen mit eindeutiger Benennung der Elemente Datentypenlängen von 16 oder 32 Bit.
- Die Datenbereiche für den Parameter MB\_DATA\_PTR können auch unterschiedlichen globalen Datenbausteinen stehen (oder in unterschiedlichen Speicherbereichen). Sie können zum Beispiel einen Datenbaustein für die Lesevorgänge und einen anderen für die Schreibvorgänge verwenden oder für jede "MB\_CLIENT"-Station einen eigenen Datenbaustein.

## Parameter CONNECT

### Verbindungsbeschreibungen am Parameter CONNECT

Für die Anweisung "MB\_CLIENT" können zwei verschiedene Verbindungsbeschreibungen verwendet werden:

- Programmierte Verbindungen mit der Struktur TCON\_IP\_v4  
Die Verbindungsparameter werden in der Struktur TCON\_IP\_v4 hinterlegt und die Verbindung wird erst mit Aufruf der Anweisung "MB\_CLIENT" eingerichtet.
- Konfigurierte Verbindungen mit der Struktur TCON\_Configured  
Die konfigurierte Verbindung wurde bereits durch die CPU aufgebaut. Über die Struktur TCON\_Configured geben Sie an, welche bestehende Verbindung für die Anweisung verwendet werden soll.

Jede Instanz der Anweisung "MB\_CLIENT" benötigt eine eindeutige Verbindung. Erstellen Sie entsprechend für jede Instanz der Anweisung eine eigene Struktur TCON\_IP\_v4 oder TCON\_Configured zur Verbindungsbeschreibung.

### Verbindungsbeschreibung für programmierte Verbindungen

Verwenden Sie für programmierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON\_IP\_v4:

- Achten Sie darauf, in der Struktur TCON\_IP\_v4 nur Verbindungen vom Typ TCP anzugeben.
- Die Verbindung darf die folgenden TCP-Port-Nummern nicht verwenden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964.

| Byte    | Parameter         | Datentyp                   | Startwert | Beschreibung   |
|---------|-------------------|----------------------------|-----------|--|
| 0 ... 1 | InterfaceID       | HW_ANY                     | -         | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).  |
| 2 ... 3 | ID                | CONN_OUC                   | -         | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Über den Parameter wird eine Verbindung innerhalb der CPU eindeutig identifiziert. Jede einzelne Instanz der Anweisung "MB_CLIENT" muss eine eindeutige ID verwenden.            |
| 4       | ConnectionType    | BYTE                       | 11        | Verbindungstyp<br>Wählen Sie 11 (dezimal) für TCP. Andere Verbindungstypen sind nicht zulässig. Wird ein anderer Verbindungstyp (z. B. UDP) verwendet, wird am Parameter STATUS der Anweisung eine entsprechende Fehlermeldung ausgegeben. |
| 5       | ActiveEstablished | BOOL                       | TRUE      | Kennung für die Art des Verbindungsaufbaus<br>Wählen Sie TRUE für den aktiven Verbindungsaufbau.   |
| 6 ... 9 | RemoteAddress     | ARRAY<br>[1..4] of<br>BYTE | -         | IP-Adresse des Verbindungspartners (Modbus Server), z. B. für 192.168.0.1: <ul style="list-style-type: none"> <li>• addr[1] = 192</li> <li>• addr[2] = 168</li> <li>• addr[3] = 0</li> <li>• addr[4] = 1</li> </ul>                        |

| Byte         | Parameter  | Datentyp | Startwert | Beschreibung   |
|--------------|------------|----------|-----------|--|
| 10 ...<br>11 | RemotePort | UINT     | 502       | Port-Nummer des entfernten Verbindungspartners (Wertebereich: 1 bis 49151).<br>Verwenden Sie die IP-Port-Nummer des Servers, zu dem der Client die Verbindung herstellt und über das TCP/IP-Protokoll kommuniziert (Standard-Wert: 502). |
| 12 ...<br>13 | LocalPort  | UINT     | 0         | Port-Nummer des lokalen Verbindungspartners: <ul style="list-style-type: none"> <li>• Port-Nummern: 1 bis 49151</li> <li>• Beliebiger Port: "0"</li> </ul>   |

**Hinweis**

**Migration der Anweisung "MB\_CLIENT" Version 2.1**

Die Parameter CONNECT\_ID, IP\_PORT und IP\_OCTET\_x sind in Version 3.0 der Anweisung "MB\_CLIENT" in der Struktur TCON\_IP\_v4 abgebildet:

- Der Parameter CONNECT\_ID der Anweisung "MB\_CLIENT" V2.1 entspricht dem Parameter ID von TCON\_IP\_v4.
- Der Parameter IP\_PORT der Anweisung "MB\_CLIENT" V2.1 entspricht dem Parameter RemotePort von TCON\_IP\_v4.
- Die vier Parameter IP\_OCTET\_x der Anweisung "MB\_CLIENT" V2.1 entsprechen dem Array des Parameters RemoteAddress von TCON\_IP\_v4.

**Verbindungsbeschreibung für konfigurierte Verbindungen**

Verwenden Sie für programmierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON\_Configured.

- Achten Sie darauf, in der Struktur TCON\_Configured nur Verbindungen vom Typ TCP anzugeben.
- Die Verbindung darf die folgenden TCP-Port-Nummern nicht verwenden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964.

| Byte    | Parameter      | Datentyp | Startwert | Beschreibung   |
|---------|----------------|----------|-----------|--|
| 0 ... 1 | Interfaceld    | HW_ANY   | -         | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).  |
| 2 ... 3 | ID             | CONN_OUC | -         | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095).<br>Geben Sie die Verbindungs-ID der bestehenden Verbindung an. |
| 4       | ConnectionType | BYTE     | 0         | Verbindungstyp<br>Wählen Sie 254 (dezimal) für eine konfigurierte Verbindung.  |

## Parameter STATUS

### Parameter STATUS (allgemeine Statusinformationen)

| STATUS*<br>(W#16#)  | Beschreibung  |
|---|---|
| 0000  | Anweisung fehlerfrei ausgeführt.  |
| 0001  | Verbindung hergestellt.   |
| 0003  | Verbindungsabbau durchgeführt.  |
| 7000  | Kein Aufruf aktiv und keine Verbindung aufgebaut (REQ=0).   |
| 7001  | Verbindungsaufbau angestoßen.   |
| 7002  | Zwischenaufruf. Verbindung wird aufgebaut.  |
| 7003  | Verbindung wird abgebaut.   |
| 7004  | Verbindung hergestellt und überwacht. Keine Auftragsbearbeitung aktiv.  |
| 7005  | Daten werden gesendet.  |
| 7006  | Daten werden empfangen.   |
| 80BB  | Ungültiger Wert am Parameter ActiveEstablished (Kennung für die Art des Verbindungsaufbaus, siehe Parameter CONNECT (Seite 2901)): <ul style="list-style-type: none"> <li>• Serverseitig ist nur der passive Verbindungsaufbau zulässig (ActiveEstablished = FALSE).</li> <li>• Clientseitig ist nur der aktive Verbindungsaufbau zulässig (ActiveEstablished = TRUE).</li> </ul> |
| 8380  | Empfangenes Modbus-Frame hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.  |
| * Die Statuscodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Parameter STATUS (Protokoll-Fehler)

| STATUS*<br>(W#16#) | Fehlerode in der Fehlermeldung von MB_SERVER (B#16#) | Beschreibung   |
|--------------------|--|--|
| 8381               | 01   | Funktionscode wird nicht unterstützt.  |
| 8382               | 03   | Fehler bei Datenlänge: <ul style="list-style-type: none"> <li>• Unzulässige Längenangabe im empfangenen Modbus-Frame.</li> <li>• Die Länge des Modbus-Frames im Frame-Header stimmt mit der Anzahl empfangenen Bytes nicht überein.</li> <li>• Die Anzahl der Bytes stimmt mit den tatsächlich übertragenen Bytes nicht überein (nur Funktionen 1-4).</li> <li>• Die empfangene Startadresse stimmt mit der ursprünglich gesendeten nicht überein (Funktionen 5, 6, 15, 16).</li> <li>• Anzahl der Wörter stimmt mit den tatsächlich übertragenen Wörtern nicht überein (Funktionen 15 und 16).</li> </ul> |
| 8383               | 02   | Fehler beim Lesen oder Schreiben der Daten oder Zugriff außerhalb des Adressbereichs von MB_DATA_PTR (Seite 2900).<br>Der Fehler kann sowohl lokal als auch bei der Anweisung "MB_SERVER" auftreten.   |

| STATUS* (W#16#)   | Fehlerode in der Fehlermeldung von MB_SERVER (B#16#) | Beschreibung  |
|---|--|---|
| 8384  | 03   | Fehler bei Datenwert: <ul style="list-style-type: none"> <li>• Fehler bei Datenwert für die Funktion 5 (Fehler auf der Serverseite).</li> <li>• Es wurde ein anderer Datenwert empfangen, als ursprünglich vom Client gesendet wurde (Funktionen 5 und 6) (lokaler Fehler).</li> <li>• Ungültiger Exception Code empfangen.</li> </ul>                                |
| 8385  | 03   | Diagnosecode nicht unterstützt (Funktions-Code 08). Der Fehler kann sowohl lokal als auch auf der Serverseite auftreten.  |
| 8386  | -  | Empfangener Funktionscode entspricht nicht dem ursprünglich gesendeten.   |
| 8387  | -  | <ul style="list-style-type: none"> <li>• Die zugewiesene Verbindungs-ID ist eine andere als bei vorherigen Abfragen. Es kann nur eine Verbindungs-ID für jeden Instanz-DB der Anweisung "MB_CLIENT" verwendet werden.</li> <li>• Der Fehlercode wird auch ausgegeben, wenn die Protokoll ID des vom Server empfangenen Modbus TCP Frames ungleich "0" ist.</li> </ul> |
| 8388  | 15 oder 16   | Durch den Modbus-Server wurde eine andere Datenlänge gesendet, als abgefragt wurde. Dieser Fehler tritt nur bei Verwendung der Modbus-Funktionen 15 oder 16 auf.  |
| * Die Statuscodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |   |

### Parameter STATUS (Parameter-Fehler)

| STATUS* (W#16#)   | Beschreibung   |
|---|--|
| 80B6  | Ungültiger Verbindungstyp, nur TCP-Verbindungen werden unterstützt.  |
| 80C8  | Keine Antwort des Servers im definierten Zeitraum. Überprüfen Sie die Verbindung zu dem Modbus-Server. Dieser Fehler wird erst nach Abschluss der konfigurierten erneuten Versuche gemeldet.<br>Empfängt die Anweisung "MB_CLIENT" keine Antwort mit der ursprünglich übermittelten Transaktions-ID (siehe statische Variable MB_TRANSACTION_ID) innerhalb des definierten Zeitraums, wird dieser Fehlercode ausgegeben. |
| 8188  | Ungültiger Wert am Parameter MB_MODE.  |
| 8189  | Ungültige Adressierung der Daten am Parameter MB_DATA_ADDR.  |
| 818A  | Ungültige Datenlänge am Parameter MB_DATA_LEN.   |
| 818B  | Ungültiger Zeiger am Parameter MB_DATA_PTR. Überprüfen Sie zudem die Werte der Parameter MB_DATA_ADDR (Seite 2900) und MB_DATA_LEN.  |
| 818C  | Zeitüberschreitung bei dem Parameter BLOCKED_PROC_TIMEOUT oder RCV_TIMEOUT (siehe statische Variablen der Anweisung). Das Limit von 55 Sekunden wurde überschritten.   |
| 818D  | Die Transaktions-ID (Variable MB_TRANSACTION_ID) entspricht nicht der ursprünglich gesendeten (siehe statische Variablen der Anweisung).   |
| 8200  | <ul style="list-style-type: none"> <li>• Eine andere Modbus-Anfrage wird zur Zeit über den Port verarbeitet.</li> <li>• Eine weitere Instanz von MB_CLIENT mit den gleichen Verbindungsparametern bearbeitet eine bereits bestehende Modbus-Anfrage.</li> </ul>  |
| * Die Statuscodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |  |

**Hinweis****Fehlercodes der intern verwendeten Kommunikations-Anweisungen**

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB\_CLIENT" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND", "TRCV", "T\_DIAG" und "TRESET") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB\_CLIENT" möglich. Im Abschnitt "Static" werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

---

**Siehe auch**

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

**MB\_SERVER: Als Modbus-TCP-Server über PROFINET kommunizieren****Beschreibung MB\_SERVER****Beschreibung**

Die Anweisung "MB\_SERVER" kommuniziert als Modbus-TCP-Server über eine PROFINET-Verbindung. Die Anweisung "MB\_SERVER" verarbeitet Verbindungsanfragen eines Modbus-TCP-Clients, empfängt und bearbeitet Modbus-Anfragen und sendet Antwort-Meldungen.

Die Anweisung "MB\_SERVER" V3.0 kann sowohl für die S7-1500 als auch für die S7-1200 ab Version 4.0 verwendet werden. Die Verbindung kann über die lokale Schnittstelle der CPU oder CM/CP erfolgen.

Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul.

**Mehrfache Server-Verbindungen**

Sie können mehrfache Server-Verbindungen erstellen. Dies erlaubt es einer einzelnen CPU, gleichzeitig Verbindungen von mehreren Modbus-TCP-Clients anzunehmen.

Ein Modbus-TCP-Server kann mehrere TCP-Verbindungen unterstützen, wobei die maximale Anzahl der Verbindungen von der verwendeten CPU abhängt.

Die gesamte Anzahl der Verbindungen einer CPU, inklusive der Modbus-TCP Clients und Server, darf die maximale Anzahl der unterstützten Verbindungen nicht überschreiten.

Modbus-TCP Verbindungen können hierbei auch gemeinsam von "MB\_CLIENT"- und/oder "MB\_SERVER"-Instanzen genutzt werden.

Bei Server-Verbindungen müssen Sie folgende Regeln beachten:

- Jede "MB\_SERVER"-Verbindung muss einen eindeutigen Instanz-DB verwenden.
- Jede "MB\_SERVER"-Verbindung muss eine eindeutige Verbindungs-ID verwenden. Die einzelne Verbindungs-ID muss jeweils für jeden einzelnen Instanz-DB der Anweisung verwendet werden. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.
- Für jede Verbindung muss die Anweisung "MB\_SERVER" einzeln aufgerufen werden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "MB\_SERVER":

| Parameter                   | Deklaration | Datentyp | Beschreibung  |
|-----------------------------|-------------|----------|---|
| DISCONNECT                  | Input       | BOOL     | <p>Die Anweisung "MB_SERVER" geht eine passive Verbindung mit einer Partner-Baugruppe ein. Der Server reagiert auf eine Verbindungsanfrage von der IP-Adresse, die in der SDT "TCON_IP_v4" am CONNECT-Parameter eingegeben ist.</p> <p>Über diesen Parameter können Sie steuern, wann eine Verbindungsanfrage akzeptiert wird:</p> <ul style="list-style-type: none"> <li>• 0: Wenn keine Kommunikationsverbindung besteht, wird eine passive Verbindung aufgebaut.</li> <li>• 1: Initialisierung des Verbindungsabbaus. Ist der Eingang gesetzt, werden keine anderen Vorgänge ausgeführt. Nach erfolgreichem Verbindungsabbau wird am Parameter STATUS der Wert 0003 ausgegeben.</li> </ul>   |
| MB_HOLD_REG<br>(Seite 2909) | InOut       | VARIANT  | <p>Zeiger auf das Modbus-Halterregister der "MB_SERVER"-Anweisung. Das Halterregister beinhaltet die Werte, auf die ein Modbus-Client über die Modbus-Funktionen 3 (lesen), 6 (schreiben) und 16 (lesen) zugreifen darf.</p> <p>Verwenden Sie als Halterregister einen globalen Datenbaustein mit optimiertem Zugriff oder den Speicherbereich der Merker.</p>  |
| CONNECT<br>(Seite 2909)     | InOut       | VARIANT  | <p>Zeiger auf die Struktur der Verbindungsbeschreibung. Die folgenden Strukturen (SDTs) können verwendet werden:</p> <ul style="list-style-type: none"> <li>• TCON_IP_v4: Enthält alle Adressierungsparameter, die für den Aufbau einer programmierten Verbindung benötigt werden. Defaultmäßig ist sie 0.0.0.0 (beliebige IP-Adresse), Sie können auch eine bestimmte IP-Adresse angeben, wodurch der Server nur auf die Anfragen von dieser Adresse reagiert. Bei Verwendung von TCON_IP_v4 wird die Verbindung beim Aufruf der Anweisung "MB_SERVER" hergestellt.</li> <li>• TCON_Configured: Enthält die Adressierungsparameter einer konfigurierten Verbindung. Bei Verwendung von TCON_Configured wird die Verbindung nach dem Laden der Hardware-Konfiguration durch die CPU hergestellt.</li> </ul> |
| NDR                         | Output      | BOOL     | <p>"New Data Ready":</p> <ul style="list-style-type: none"> <li>• 0: Keine neuen Daten</li> <li>• 1: Neue Daten durch den Modbus-Client geschrieben</li> </ul>  |
| DR                          | Output      | BOOL     | <p>"Data Read":</p> <ul style="list-style-type: none"> <li>• 0: Keine Daten gelesen</li> <li>• 1: Daten durch den Modbus-Client gelesen</li> </ul>  |



| Parameter              | Deklaration | Datentyp | Beschreibung   |
|------------------------|-------------|----------|--|
| ERROR                  | Output      | BOOL     | Tritt während eines Aufrufs der Anweisung "MB_SERVER" ein Fehler auf, wird der Ausgang am Parameter ERROR auf "1" gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS angezeigt. |
| STATUS<br>(Seite 2911) | Output      | WORD     | Detaillierte Statusinformation der Anweisung.  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Statische Variablen der Anweisung

Die folgende Tabelle beschreibt die statischen Variablen des Instanz-Datenbausteins der Anweisung "MB\_SERVER", die im Programm verwendet werden können. Auf die Variable HR\_Start\_Offset können Sie schreibend zugreifen. Auf die übrigen Variablen können Sie lesend zugreifen, um den Modbus-Status zu überwachen.

| Variable             | Datentyp | Startwert | Beschreibung  |
|----------------------|----------|-----------|---|
| HR_Start_Offset      | WORD     | 0         | Zuweisen der Anfangsadresse des Modbus-Haltregisters.   |
| Request_Count        | WORD     | 0         | Anzahl aller Anfragen, die durch den Server empfangen wurden.   |
| Server_Message_Count | WORD     | 0         | Anzahl aller empfangenen Meldungen für den jeweiligen Server.   |
| Xmt_Rcv_Count        | WORD     | 0         | Zähler zur Erfassung der Anzahl der Übertragungen, bei denen ein Fehler aufgetreten ist. Der Zähler wird nur dann erhöht, wenn eine ungültige Modbus-Anfrage empfangen wurde. |
| Exception_Count      | WORD     | 0         | Zähler zur Erfassung der Anzahl von Fehlern, die spezifisch für Modbus sind und eine Fehlermeldung an "MB_CLIENT" hervorrufen.  |
| Success_Count        | WORD     | 0         | Ereigniszähler zur Erfassung der Anzahl der Anfragen, die vom Server erfolgreich ausgeführt werden konnten.   |
| Connected            | BOOL     | 0         | Gibt an, ob die Verbindung zum zugewiesenen Client verbunden oder nicht verbunden ist: 1 = verbunden, 0 = nicht verbunden.  |

### Abbilden der Modbus-Adressen auf dem Prozessabbild

Die Anweisung "MB\_SERVER" ermöglicht eingehenden Modbus-Funktionen (1, 2, 4, 5 und 15) direkt lesend und schreibend auf das Prozessabbild der Eingänge und Ausgänge der CPU zuzugreifen (Verwendung der Datentypen BOOL und WORD).

Für den Datentransfer der Funktionscodes 3, 6 und 16 muss das Haltregister (Parameter MB\_HOLD\_REG) länger als ein Byte definiert werden. Die folgende Tabelle zeigt die Abbildung der Modbus-Adressen auf das Prozessabbild der CPU.

| Modbus Funktion |             |              |            |     |       | S7-1500, S7-1200 V4.0 |                  |
|-----------------|-------------|--------------|------------|-----|-------|-----------------------|------------------|
| Funktion scode  | Funktion    | Datenbereich | Adressraum |     |       | Datenbereich          | CPU Adresse      |
| 01              | Lesen: Bits | Output       | 1          | bis | 9999  | Prozessabbild Ausgang | A0.0 bis A1248.6 |
| 02              | Lesen: Bits | Input        | 10001      | bis | 19999 | Prozessabbild Eingang | E0.0 bis E1248.6 |
| 04              | Lesen: WORD | Input        | 30001      | bis | 39999 | Prozessabbild Eingang | EW0 bis EW19996  |

| Modbus Funktion |                 |        |   |     |      | S7-1500, S7-1200 V4.0 |                  |
|-----------------|-----------------|--------|---|-----|------|-----------------------|------------------|
| 05              | Schreiben: Bits | Output | 1 | bis | 9999 | Prozessabbild Ausgang | A0.0 bis A1248.6 |
| 15              | Schreiben: Bits | Output | 1 | bis | 9999 | Prozessabbild Ausgang | A0.0 bis A1248.6 |

Eingehende Modbus-Anfragen mit den Funktionscodes 3, 6, 16 und 23 schreiben oder lesen vom Modbus-Halteregister (das Halteregister spezifizieren Sie am Parameter MB\_HOLD\_REG).

**Beispiel: Adressierung über statische Variable HR\_Start\_Offset**

Die Adressen des Modbus-Halteregisters beginnen bei 40001. Diese Adressen entsprechen dem Adressraum des CPU-Speicherbereichs für das Halteregister. Sie können die Variable HR\_Start\_Offset auch so definieren, dass das Modbus-Halteregister eine andere Anfangsadresse als 40001 erhält.

Beispiel: Das Halteregister beginnt am MW100 und hat eine Länge von 100 WORD. Ein Offset-Wert am Parameter HR\_Start\_Offset bewirkt, dass sich die Anfangsadresse des Halteregisters von 40001 auf 40021 verschiebt. Jede Adressierung des Halteregisters, die unterhalb der Adresse 40021 und oberhalb der Adresse 40120 liegt, ruft einen Fehler hervor.

| HR_Start_Offset | Adresse               | Minimum | Maximum |
|-----------------|-----------------------|---------|---------|
| 0               | Modbus-Adresse (WORD) | 40001   | 40100   |
|                 | CPU-Adresse           | MW100   | MW298   |
| 20              | Modbus-Adresse (WORD) | 40021   | 40120   |
|                 | CPU-Adresse           | MW100   | MW298   |

**Modbus-Diagnosefunktionen**

Die folgende Tabelle enthält die Beschreibung der Modbus-Diagnosefunktionen.

| Funktionscode | Diagnosecode | Beschreibung   |
|---------------|--------------|--|
| 08            | 0x0000       | Echo-Test: Die Anweisung "MB_SERVER" empfängt ein Datenwort und sendet dieses unverändert an den Modbus-Client zurück.   |
| 08            | 0x000A       | Ereigniszähler zurücksetzen: Die Anweisung "MB_SERVER" setzt die folgenden Ereigniszähler zurück: "Success_Count", "Xmt_Rcv_Count", "Exception_Count", "Server_Message_Co" und "Request_Count".  |
| 11            | -            | Ereigniszähler der Kommunikation holen: Die Anweisung "MB_SERVER" verwendet einen internen Ereigniszähler bei der Kommunikation zur Erfassung der Anzahl der erfolgreich ausgeführten Lese- und Schreibanfragen, die an den Modbus-Server gesendet werden.<br>Der Ereigniszähler wird bei den Funktionen 8 oder 11 nicht inkrementiert. Gleiches gilt für Anfragen, die einen Kommunikationsfehler zur Folge haben, zum Beispiel falls ein Protokollfehler aufgetreten ist (z. B. der Funktionscode in der empfangenen Modbus-Anfrage nicht unterstützt wird). |

## Parameter MB\_HOLD\_REG

### Beschreibung

Der Parameter MB\_HOLD\_REG ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Speicherbereich können Sie einen globalen Datenbaustein oder einen Merker (M) verwenden.

- Die Obergrenze für die Anzahl von Adressen im Datenbaustein (D) ist durch den maximalen Arbeitsspeicher der CPU vorgegeben.
- Die Obergrenze für die Anzahl von Merkern (M) ist durch die Größe des Speicherbereichs der CPU vorgegeben.

Die folgende Tabelle zeigt einige Beispiele zur Abbildung von Modbus-Adressen auf das Haltereister für die Modbus-Funktionen 3 (WORD lesen), 6 (WORD schreiben), 16 (mehrere WORD schreiben) und 23 (mehrere Wörter schreiben und lesen).

| Modbus-Adressen | MB_HOLD_REG Parameter - Beispiele |           |                        |
|-----------------|-----------------------------------|-----------|------------------------|
| 40001           | MW100                             | DB10.DBW0 | "Recipe".ingredient[1] |
| 40002           | MW102                             | DB10.DBW2 | "Recipe".ingredient[2] |
| 40003           | MW104                             | DB10.DBW4 | "Recipe".ingredient[3] |
| 40004           | MW106                             | DB10.DBW6 | "Recipe".ingredient[4] |
| 40005           | MW108                             | DB10.DBW8 | "Recipe".ingredient[5] |

## Parameter CONNECT

### Verbindungsbeschreibungen am Parameter CONNECT

Für die Anweisung "MB\_SERVER" können zwei verschiedene Verbindungsbeschreibungen verwendet werden:

- Programmierte Verbindungen mit der Struktur TCON\_IP\_v4  
Die Verbindungsparameter werden in der Struktur TCON\_IP\_v4 hinterlegt und die Verbindung wird erst mit Aufruf der Anweisung "MB\_SERVER" eingerichtet.
- Konfigurierte Verbindungen mit der Struktur TCON\_Configured  
Die konfigurierte Verbindung wurde bereits durch die CPU aufgebaut. Über die Struktur TCON\_Configured geben Sie an, welche bestehende Verbindung für die Anweisung verwendet werden soll.

Jede Instanz der Anweisung "MB\_SERVER" benötigt eine eindeutige Verbindung. Erstellen Sie entsprechend für jede Instanz der Anweisung eine eigene Struktur TCON\_IP\_v4 oder TCON\_Configured zur Verbindungsbeschreibung.

### Verbindungsbeschreibung für programmierte Verbindungen

Verwenden Sie für programmierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON\_IP\_v4.

- Achten Sie darauf, in der Struktur TCON\_IP\_v4 nur Verbindungen vom Typ TCP anzugeben.
- Die Verbindung darf die folgenden TCP-Port-Nummern nicht verwenden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964.

| Byte      | Parameter         | Datentyp             | Startwert | Beschreibung  |
|-----------|-------------------|----------------------|-----------|---|
| 0 ... 1   | InterfacelD       | HW_ANY               | -         | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).   |
| 2 ... 3   | ID                | CONN_OUC             | -         | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Über den Parameter wird eine Verbindung innerhalb der CPU eindeutig identifiziert. Jede einzelne Instanz der Anweisung "MB_SERVER" muss eine eindeutige ID verwenden.   |
| 4         | ConnectionType    | BYTE                 | 11        | Verbindungstyp<br>Wählen Sie 11 (dezimal) für TCP. Andere Verbindungstypen sind nicht zulässig. Wird ein anderer Verbindungstyp (z. B. UDP) verwendet, wird am Parameter STATUS der Anweisung eine entsprechende Fehlermeldung ausgegeben.  |
| 5         | ActiveEstablished | BOOL                 | FALSE     | Kennung für die Art des Verbindungsaufbaus<br>Wählen Sie FALSE für den passiven Verbindungsaufbau.  |
| 6 ... 9   | RemoteAddress     | ARRAY [1..4] of BYTE | 0.0.0.0   | IP-Adresse des Verbindungspartners, z. B. für 192.168.0.1:<br><ul style="list-style-type: none"> <li>• addr[1] = 192</li> <li>• addr[2] = 168</li> <li>• addr[3] = 0</li> <li>• addr[4] = 1</li> </ul> Soll die Anweisung "MB_SERVER" Verbindungsanfragen von jedem Verbindungspartner akzeptieren, verwenden Sie als IP-Adresse "0.0.0.0".                             |
| 10 ... 11 | RemotePort        | UINT                 | 0         | Port-Nummer des entfernten Verbindungspartners (Wertebereich: 1 bis 49151).<br>Soll die Anweisung "MB_SERVER" Verbindungsanfragen von jedem entfernten Partner akzeptieren, verwenden Sie als Port-Nummer "0".  |
| 12 ... 13 | LocalPort         | UINT                 | 502       | Port-Nummer des lokalen Verbindungspartners (Wertebereich: 1 bis 49151).<br>Die Nummer des IP-Ports legt fest, welcher IP-Port für Verbindungsanfragen des Modbus-Clients überwacht wird.<br>Die folgenden TCP-Port-Nummern dürfen nicht für die passive Verbindung der "MB_SERVER"-Anweisung verwendet werden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964. |

**Hinweis****Migration der Anweisung "MB\_SERVER" Version 2.1**

Die Parameter CONNECT\_ID und IP\_PORT sind in Version 3.0 der Anweisung "MB\_SERVER" in der Struktur TCON\_IP\_v4 abgebildet:

- Der Parameter CONNECT\_ID der Anweisung "MB\_SERVER" V2.1 entspricht dem Parameter ID von TCON\_IP\_v4.
- Der Parameter IP\_PORT der Anweisung "MB\_SERVER" V2.1 entspricht dem Parameter LocalPort von TCON\_IP\_v4.

**Verbindungsbeschreibung für konfigurierte Verbindungen**

Verwenden Sie für konfigurierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON\_Configured.

- Achten Sie darauf, in der Struktur TCON\_Configured nur Verbindungen vom Typ TCP anzugeben.
- Die Verbindung darf die folgenden TCP-Port-Nummern nicht verwenden: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 und 34964.

| Byte    | Parameter      | Datentyp | Startwert | Beschreibung  |
|---------|----------------|----------|-----------|---|
| 0 ... 1 | InterfaceID    | HW_ANY   | -         | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).   |
| 2 ... 3 | ID             | CONN_OUC | -         | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Geben Sie die Verbindungs-ID der bestehenden Verbindung an. |
| 4       | ConnectionType | BYTE     | -         | Verbindungstyp<br>Wählen Sie 254 (dezimal) für eine konfigurierte Verbindung.   |

**Parameter STATUS****Parameter STATUS (allgemeine Statusinformationen)**

| STATUS*<br>(W#16#) | Beschreibung                               |
|--------------------|--|
| 0000               | Anweisung fehlerfrei ausgeführt.           |
| 0001               | Verbindung hergestellt.                    |
| 0003               | Verbindungsabbau durchgeführt.             |
| 7000               | Kein Aufruf aktiv (REQ=0).                 |
| 7001               | Erstaufruf. Verbindungsaufbau angestoßen.  |
| 7002               | Zwischenaufruf. Verbindung wird aufgebaut. |
| 7003               | Verbindung wird abgebaut.                  |
| 7005               | Daten werden gesendet.                     |

| STATUS* (W#16#)   | Beschreibung  |
|---|---|
| 7006  | Daten werden empfangen.   |
| 80BB  | Ungültiger Wert am Parameter ActiveEstablished (Kennung für die Art des Verbindungsaufbaus, siehe Parameter CONNECT (Seite 2909)): <ul style="list-style-type: none"> <li>• Serverseitig ist nur der passive Verbindungsaufbau zulässig (active_established = FALSE).</li> <li>• Clientseitig ist nur der aktive Verbindungsaufbau zulässig (active_established = TRUE).</li> </ul> |
| 8380  | Empfangenes Modbus-Frame hat nicht das richtige Format oder es wurden zu wenige Bytes empfangen.  |
| * Die Statuscodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

### Parameter STATUS (Parameter-Fehler)

| STATUS* (W#16#)   | Fehlercode in der Fehlermeldung von "MB_SERVER" (B#16#) | Beschreibung  |
|---|---|---|
| 8187  | Keine Antwort   | Ungültiger Zeiger am Parameter MB_HOLD_REG. Datenbereich ist zu klein.  |
| 8381  | 01  | Funktionscode wird nicht unterstützt.   |
| 8382  | 03  | Fehler bei Datenlänge: <ul style="list-style-type: none"> <li>• Unzulässige Längenangabe im empfangenen Modbus-Telegramm</li> <li>• Die im Header des Modbus-Frames eingetragene Frame-Länge stimmt mit der Anzahl von tatsächlich empfangenen Bytes nicht überein.</li> <li>• Die im Header des Modbus-Frames eingetragene Anzahl der Bytes stimmt mit der Anzahl von tatsächlich empfangenen Bytes nicht überein (Funktionen 15 und 16).</li> </ul> |
| 8383  | 02  | Fehler bei Datenadresse oder Zugriff außerhalb des Adressbereichs des Haltereisters (Parameter MB_HOLD_REG (Seite 2909)).   |
| 8384  | 03  | Ungültiger Datenwert (Funktion 5).  |
| 8385  | 03  | Diagnosecode wird nicht unterstützt (nur bei Funktion 08).  |
| * Die Statuscodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |   |

#### Hinweis

#### Fehlercodes der intern verwendeten Kommunikations-Anweisungen

Zusätzlich zu den Fehlern, die in den Tabellen aufgelistet sind, können bei der Anweisung "MB\_SERVER" noch Fehler bei den durch die Anweisung verwendeten Kommunikations-Anweisungen ("TCON", "TDISCON", "TSEND", "TRCV", "T\_DIAG" und "T\_RESET") auftreten.

Die Zuordnung der Fehlercodes ist über den Instanz-Datenbaustein der Anweisung "MB\_SERVER" möglich. Im Abschnitt "Static" werden die Fehlercodes bei der jeweiligen Anweisung unter STATUS angezeigt.

Die Bedeutung der Fehlercodes entnehmen Sie der Dokumentation der entsprechenden Kommunikations-Anweisung.

**Siehe auch**

GET\_ERR\_ID: Fehler-ID lokal abfragen (Seite 1832)

**9.7.5.2 S7-Kommunikation****Datenkonsistenz****Definition**

Die Größe des Datenbereichs, der nicht gleichzeitig durch konkurrierende Prozesse verändert werden kann, wird als konsistenter Datenbereich bezeichnet. Ein Datenbereich, der größer als der konsistente Datenbereich ist, kann somit in seiner Gesamtheit verfälscht werden.

Das heißt, ein in sich zusammengehöriger Datenbereich, der größer als der konsistente Datenbereich ist, kann zu einem Zeitpunkt teilweise aus neuen und aus alten konsistenten Datenblöcken bestehen.

**Beispiel**

Eine Inkonsistenz kann entstehen, wenn ein Kommunikations-Baustein z. B. durch einen Prozessalarm-OB mit höherer Priorität unterbrochen wird. Verändert das Anwenderprogramm in diesem OB jetzt die Daten, die teilweise bereits vom Kommunikations-Baustein verarbeitet wurden, stammen die übertragenen Daten:

- zum einen Teil aus der Zeit vor der Prozessalarm-Bearbeitung
- und zum anderen Teil aus der Zeit nach der Prozessalarm-Bearbeitung.

Das bedeutet, dass diese Daten inkonsistent (nicht zusammengehörig) sind.

**Auswirkung**

Wenn große Datenmengen konsistent übertragen werden sollen, dann darf die Übertragung nicht unterbrochen werden. Dadurch kann z. B. die Alarmreaktionszeit der CPU verlängert werden.

D. h.: Je größer die Menge der garantiert konsistent zu übertragenen Daten, desto länger die Alarmreaktionszeit eines Systems.

## Datenkonsistenz bei SIMATIC

- Existiert im Anwenderprogramm eine Kommunikationsfunktion, welche auf gemeinsame Daten zugreift, kann der Zugriff auf diesen Datenbereich z. B. über den Parameter DONE selbst koordiniert werden. Die Datenkonsistenz der Kommunikationsbereiche, die lokal mit einem Kommunikationsbaustein übertragen wird, kann deshalb im Anwenderprogramm sichergestellt werden.
- Bei S7-Kommunikationsanweisungen "PUT (Seite 2920)" / "GET (Seite 2918)" muss bereits bei der Programmierung bzw. Projektierung die Größe der konsistenten Datenbereiche berücksichtigt werden, da im Anwenderprogramm des Zielgerätes (Server) kein Kommunikationsbaustein vorhanden ist, der die Kommunikationsdaten in das Anwenderprogramm einsynchronisiert:
- Bei der S7-300 und C7-300 (Ausnahme: CPU 318-2 DP) werden die Kommunikationsdaten in Blöcken zu 32 Bytes im Zykluskontrollpunkt des Betriebssystems, konsistent in den Anwenderspeicher kopiert. Für alle größeren Datenbereiche wird keine Datenkonsistenz garantiert. Ist eine definierte Datenkonsistenz gefordert, so dürfen die Kommunikationsdaten im Anwenderprogramm nicht größer als 32 Bytes sein (je nach Ausgabestand maximal 8 Byte).
- Bei der S7-400 und S7-1500 werden im Gegensatz dazu die Kommunikationsdaten in Blöcken zu 462 Bytes nicht im Zykluskontrollpunkt, sondern in festen Zeitscheiben während des Programmzyklusses bearbeitet. Systemseitig wird die Konsistenz einer Variable garantiert. Auf diese Kommunikationsbereiche kann dann, z. B. von einem OP oder von einer OS, mit den "PUT (Seite 2920)" / "GET (Seite 2918)"-Anweisungen bzw. Lesen/Schreiben von Variablen konsistent zugegriffen werden.

---

### Hinweis

Weitere Hinweise zu Datenkonsistenz finden Sie bei der Beschreibung der einzelnen Anweisungen.

---

## Gemeinsame Parameter der Anweisungen zur S7-Kommunikation

### Klassifikation

Die Parameter der Anweisungen zur S7-Kommunikation lassen sich von ihrer Funktion her in die folgenden fünf Klassen einteilen:

1. Steuerparameter dienen der Aktivierung einer Anweisung.
2. Adressierungsparameter dienen der Adressierung des remoten Kommunikationspartners.
3. Sendeparameter zeigen auf diejenigen Datenbereiche, die zum remoten Partner gesendet werden sollen.
4. Empfangsparameter zeigen auf diejenigen Datenbereiche, in welche die vom remoten Partner empfangenen Daten eingetragen werden.
5. Zustandsparameter dienen zur Überwachung, ob die Anweisung ihre Aufgabe fehlerfrei beendet hat, bzw. zur Analyse der aufgetretenen Fehler.



## Steuerparameter

Die Aktivierung des Datenaustauschs erfolgt nur dann, wenn beim Aufruf der Anweisung die zugehörigen Steuerparameter einen definierten Wert haben (z. B. gesetzt sind) bzw. wenn sich der Wert gegenüber dem letzten Aufruf in definierter Weise geändert hat (z. B. positive Flanke).

## Adressierungsparameter

| Parameter | Beschreibung   |
|-----------|--|
| ID        | Referenz auf die lokale Verbindungsbeschreibung (wird durch die Verbindungsprojektierung vorgegeben). Hinweis: Bei den Anweisungen der S7-Kommunikation ist die ID W#16#EEEE nicht erlaubt.  |
| R_ID      | Mit dem Parameter R_ID legen Sie die Zusammengehörigkeit einer Sende- und einer Empfangs-Anweisung fest: Der Parameter R_ID muss bei der Anweisung auf der Sendeseite und bei der Anweisung auf der Empfangsseite übereinstimmen.<br>Damit wird die Kommunikation mehrerer Anweisungs-paare über dieselbe logische Verbindung ermöglicht. <ul style="list-style-type: none"> <li>• R_ID muss in der Form DW#16#wxyzWXYZ angegeben werden.</li> <li>• Die über R_ID festgelegten Anweisungs-paare einer logischen Verbindung müssen für diese Verbindung eindeutig sein.</li> </ul> |

### Hinweis

#### Adressierungsparameter ID und R\_ID

Sie können die Adressierungsparameter ID und R\_ID zur Laufzeit umparametrieren. Die neuen Parameter werden mit jedem neuen Auftrag nach Abschluss des vorangegangenen Auftrags wirksam.

Mit folgenden Möglichkeiten können Sie Instanz-DBs und somit Arbeitsspeicher sparen:

1. Durch variable IDs können Sie mehrere Verbindungen über einen Dateninstanzbaustein nutzen.
2. Durch variable R\_IDs können Sie mit einer Instanz mehrere Zusammengehörigkeiten von Sende- und Empfangs-Anweisungs-paaren für einen Auftrag festlegen.
3. Sie können die Fälle 1 und 2 miteinander kombinieren.

Beachten Sie, dass die neuen Parameter erst nach Abschluss des vorangegangenen Auftrags gültig werden. Wenn Sie den Sendevorgang aktivieren, muss bei der Anweisung auf der Sendeseite und bei der Anweisung auf der Empfangsseite der Parameter R\_ID übereinstimmen.

## Zustandsparameter

Mit den Zustandsparametern überwachen Sie, ob die Anweisung ihre Aufgabe ordnungsgemäß beendet hat oder ob sie noch tätig ist. Darüber hinaus zeigen diese aufgetretene Fehler an.

---

### Hinweis

Die Zustandsparameter sind nur einen Zyklus lang gültig, nämlich vom ersten Befehl, der dem Aufruf folgt, bis zum nächsten Aufruf. Daraus folgt, dass Sie diese Parameter nach jedem Durchlauf der Anweisung auswerten müssen.

---

## Sende- und Empfangsparameter

Bei den Anweisungen zur zweiseitigen Kommunikation

- muss die Anzahl der verwendeten SD\_i- und RD\_i-Parameter auf Sende- und Empfangsseite übereinstimmen
- müssen die Datentypen bei zusammengehörigen SD\_i- und RD\_i-Parametern auf Sende- und Empfangsseite übereinstimmen
- darf die Anzahl der über den Parameter SD\_i zu sendenden Daten nicht größer sein als der Bereich, der durch den zugehörigen Parameter RD\_i aufgespannt wird (gilt nicht für "BSEND (Seite 2926)" / "BRCV (Seite 2929)"). Die Parameter RD\_i müssen (mit Ausnahme von "BSEND"/"BRCV") die gleich Datengröße besitzen.

Wenn Sie gegen die genannten Regeln verstoßen, so wird Ihnen dies mit ERROR = 1 und STATUS = 4 angezeigt.

---

### Hinweis

#### Versorgung der Sende- und Empfangsparameter

Sende- und Empfangsparameter mit dem Datentyp VARIANT müssen beim Aufruf aller Kommunikations-Anweisung immer versorgt werden. Es ist nicht möglich, z. B. die Sendepuffer der Kommunikations-Anweisung im Anlauf zu versorgen und im zyklischen Betrieb nur den Sendeauftrag anzustoßen.

---

## Nutzdatengröße

Bei den Anweisungen "USEND (Seite 2923)", "URCV (Seite 2925)", "GET (Seite 2918)" und "PUT (Seite 2920)" darf die Anzahl der zu übertragenden Daten eine bestimmte Nutzdatengröße nicht überschreiten. Die maximale Nutzdatengröße ist abhängig von:

- verwendeter Anweisung und
- Kommunikationspartner.

Die garantierte Mindestgröße der Nutzdaten für eine Anweisung mit 1-4 Variablen können Sie folgender Tabelle entnehmen:

| Anweisung    | Partner: S7-300  | Partner: S7-400 | Partner: S7-1200 | Partner: S7-1500 |
|--------------|------------------|-----------------|------------------|------------------|
| PUT / GET    | 160 Byte         | 400 Byte        | 160 Byte         | 880 Byte         |
| USEND / URCV | 160 Byte         | 440 Byte        | -                | 920 Byte         |
| BSEND / BRCV | 32768/65534 Byte | 65534 Byte      | -                | 65534 Byte       |

Beachten Sie die Einschränkungen zur Nutzdatengröße in den Technischen Daten der jeweiligen CPU.

### Exakte Nutzdatengröße

Wenn die oben angegebene Nutzdatengröße nicht ausreicht, können Sie die exakte maximale Nutzdatenlänge in Byte wie folgt ermitteln:

Zunächst lesen Sie aus der folgenden Tabelle die bei der Kommunikation gültige Datenblockgröße ab:

| Eigene CPU | Remote CPU | Datenblockgröße in Byte |
|------------|------------|-------------------------|
| S7-1200    | beliebig   | 240                     |
| S7-1500    | S7-300     | 240                     |
|            | S7-400     | 480                     |
|            | S7-1200    | 240                     |
|            | S7-1500    | 960                     |

Mit diesem Wert gehen Sie in die folgende Tabelle und lesen daraus die maximal mögliche Nutzdatenlänge in Byte als Summe der benutzten Parameter ab. Sie gilt für gerade Längen der Bereiche SD\_i, RD\_i, ADDR\_i.

Für jeden Bereich ungerader Länge reduziert sich die maximal mögliche Nutzdatenlänge um ein Byte.

| Datenblockgröße                            | Anweisung         | Anzahl benutzter Parameter SD_i, RD_i, ADDR_i |     |     |     |
|--|-------------------|---|-----|-----|-----|
|  |                   | 1   | 2   | 3   | 4   |
| 240 (S7-300)                               | PUT/GET/<br>USEND | 160   | -   | -   | -   |
| 240 (S7-300 via integrierte Schnittstelle) | PUT               | 212   | -   | -   | -   |
|  | GET               | 222   | -   | -   | -   |
|  | USEND             | 212   | -   | -   | -   |
| 240 (S7-400)                               | PUT               | 212   | 196 | 180 | 164 |
|  | GET               | 222   | 218 | 214 | 210 |
|  | USEND             | 212   | -   | -   | -   |
| 480 (S7-400)                               | PUT               | 452   | 436 | 420 | 404 |
|  | GET               | 462   | 458 | 454 | 450 |
|  | USEND             | 452   | 448 | 444 | 440 |
| 240 (S7-1200)                              | PUT               | 212   | 196 | 180 | 164 |
|  | GET               | 222   | 218 | 214 | 210 |
| 960 (S7-1500)                              | PUT               | 932   | 916 | 900 | 884 |

| Datenblockgröße | Anweisung | Anzahl benutzter Parameter SD_i, RD_i, ADDR_i |     |     |     |
|-----------------|-----------|---|-----|-----|-----|
|                 |           | 1   | 2   | 3   | 4   |
|                 | GET       | 942   | 938 | 934 | 930 |
|                 | USEND     | 932   | 928 | 924 | 920 |

## GET: Daten aus einer remoten CPU lesen

### Beschreibung

Mit der Anweisung "GET" können Sie Daten aus einer remoten CPU auslesen. Dies ist nur möglich, wenn für die Partner-CPU in den Eigenschaften der CPU unter "Schutz" die Funktion "Zugriff über PUT/GET-Kommunikation durch entfernten Partner erlauben" aktiviert wurde. Mit der Anweisung "GET" können sie nicht auf Bausteine zugreifen, die mit der Zugriffsart "optimiert" erstellt wurden.

Bei einer positiven Flanke am Steuereingang REQ wird die Anweisung gestartet. Dabei werden die relevanten Zeiger auf die auszulesenden Bereiche (ADDR\_i) an die Partner-CPU gesendet. Die remote CPU kann sich im Betriebszustand RUN oder STOP befinden. Der remote Partner sendet die Dateninhalte zurück.

- Bei einem Nullwert an einem der Parameter ADDR\_i, wird dieser ignoriert. Die empfangenen Daten werden beim nächsten Anweisung-Aufruf in die projizierten Empfangsbereiche (RD\_i) kopiert.
- Sie müssen darauf achten, dass die über die Parameter ADDR\_i und RD\_i definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinanderpassen. Der zu lesende Bereich (Parameter ADDR\_i) darf nicht kleiner sein als der Bereich zur Datenablage (Parameter RD\_i). Beide Bereiche müssen in Anzahl, Länge und Datentyp übereinstimmen. Überschreitet die Antwort die maximale Nutzdatenlänge, wird dies am Parameter STATUS mit dem Fehlercode "2" angezeigt.

Änderungen in den adressierten Datenbereichen auf der Partner-CPU werden durch die Anweisung "GET" nicht registriert.

Der Abschluss dieses Vorgangs wird am Zustandsparameter NDR mit "1" angezeigt. Eine erneute Aktivierung eines Lesevorgangs ist erst nach dem Abschluss des letzten möglich.

Wenn beim Lesen der Daten Zugriffsprobleme auftraten, oder die Datentypprüfung einen Fehler ergab, werden Fehler und Warnungen über ERROR und STATUS ausgegeben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "GET":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------|------------------------------|--|
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke. |
| ID        | Input       | CONN_PRG (WORD) | E, A, M, D, L oder Konstante | Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.         |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| NDR       | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter NDR: <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder läuft noch.</li> <li>• 1: Auftrag wurde erfolgreich abgeschlossen.</li> </ul>   |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter ERROR und STATUS, Fehleranzeige: <ul style="list-style-type: none"> <li>• ERROR=0<br/>STATUS hat den Wert:                             <ul style="list-style-type: none"> <li>- 0000H: weder Warnung noch Fehler</li> <li>- &lt;&gt; 0000H: Warnung, STATUS liefert detaillierte Auskunft.</li> </ul> </li> <li>• ERROR=1<br/>Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul> |
| STATUS    | Output      | WORD     | E, A, M, D, L   |  |
| ADDR_1    | InOut       | REMOTE   | E, A, M, D      | Zeiger auf diejenigen Bereiche in der Partner-CPU, die gelesen werden sollen. Wenn der REMOTE-Zeiger auf einen DB zugreift, ist der DB immer zu spezifizieren. Beispiel: P#DB10.DBX5.0 Byte 10. Bei der Übertragung von Datenstrukturen (z.B. Struct, Array) muss an dem Parameter ADDR der Datentyp CHAR, BYTE, WORD oder DWORD verwendet werden.   |
| ADDR_2    | InOut       | REMOTE   |                 |  |
| ADDR_3    | InOut       | REMOTE   |                 |  |
| ADDR_4    | InOut       | REMOTE   |                 |  |
| RD_1      | InOut       | VARIANT  | E, A, M, D, L   | Zeiger auf diejenigen Bereiche in der eigenen CPU, in der die gelesenen Daten abgelegt werden.   |
| RD_2      | InOut       | VARIANT  |                 |  |
| RD_3      | InOut       | VARIANT  |                 |  |
| RD_4      | InOut       | VARIANT  |                 |  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "GET" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

| ERROR | STATUS (dezimal) | Erläuterung  |
|-------|------------------|--|
| 0     | 11               | Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.   |
| 0     | 25               | Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.  |
| 1     | 1                | Kommunikationsprobleme, z. B. <ul style="list-style-type: none"> <li>• Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>• Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP)</li> <li>• Verbindung zum Partner noch nicht aufgebaut</li> </ul> |

| ERROR | STATUS (dezimal) | Erläuterung  |
|-------|------------------|--|
| 1     | 2                | <ul style="list-style-type: none"> <li>Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.</li> <li>Antwort der remoten Station überschreitet die maximale Nutzdatenlänge.</li> <li>Bei S7-1500: Zugriffsschutz bei remoter Station aktiviert.</li> </ul> |
| 1     | 4                | Fehler in den Zeigern zur Datenablage RD_i: <ul style="list-style-type: none"> <li>Datentypen der Parameter RD_i und ADDR_i passen nicht zueinander.</li> <li>Die Länge des Bereichs RD_i ist kleiner als die Länge der zu lesenden Daten der Parameter ADDR_i.</li> </ul>   |
| 1     | 8                | Zugriffsfehler bei der Partner-CPU.  |
| 1     | 10               | Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).  |
| 1     | 20               | <ul style="list-style-type: none"> <li>Maximale Anzahl an parallelen Aufträgen überschritten.</li> <li>Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufwurf).</li> </ul>  |

**Hinweis**

**Datenkonsistenz**

Die Daten werden konsistent empfangen, wenn Sie den aktuell benutzten Teil des Empfangsbereichs RD\_i vollständig auslesen, bevor Sie einen erneuten Auftrag aktivieren.

**Siehe auch**

Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 2914)

**PUT: Daten in eine remote CPU schreiben**

**Beschreibung**

Mit der Anweisung "PUT" können Sie Daten in eine remote CPU schreiben. Dies ist nur möglich, wenn für die Partner-CPU in den Eigenschaften der CPU unter "Schutz" die Funktion "Zugriff über PUT/GET-Kommunikation durch entfernten Partner erlauben" aktiviert wurde. Mit der Anweisung "PUT" können sie nicht auf Bausteine zugreifen, die mit der Zugriffsart "optimiert" erstellt wurden. Die remote CPU kann sich im Betriebszustand RUN oder STOP befinden.

Bei einer positiven Flanke am Steuereingang REQ wird die Anweisung gestartet. Dabei werden die Zeiger auf die zu schreibenden Bereiche (ADDR\_i) und die Daten (SD\_i) an die Partner-CPU gesendet. Sie müssen darauf achten, dass die über die Parameter ADDR\_i und SD\_i definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinander passen.

- Die zu sendenden Daten werden bei einer positiven Flanke am Parameter REQ aus den projektierten Sendebereichen (SD\_i) kopiert.
- Sie müssen darauf achten, dass die über die Parameter ADDR\_i und SD\_i definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinander passen. Der zu schreibende Bereich (Parameter ADDR\_i) darf nicht größer sein als der Sendebereich (Parameter SD\_i). Beide Bereiche müssen in Anzahl, Länge und Datentyp übereinstimmen.

Der remote Partner legt die gesendeten Daten unter den mitgeführten Adressen ab und sendet eine Ausführungsquittung zurück. Falls keine Fehler auftraten, wird dies beim nächsten Anweisungs-Aufruf am Zustandsparameter DONE mit "1" angezeigt. Eine erneute Aktivierung eines Schreibvorgangs ist erst nach dem Abschluss des letzten möglich.

Wenn beim Schreiben der Daten Zugriffsprobleme auftraten, oder die Prüfung der Ausführungsquittung einen Fehler ergab, werden Fehler und Warnungen über ERROR und STATUS ausgegeben.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "PUT":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------------|------------------------------|---|
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.  |
| ID        | Input       | CONN_PRG (WORD) | E, A, M, D, L oder Konstante | Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.  |
| DONE      | Output      | BOOL            | E, A, M, D, L                | Zustandsparameter DONE: <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt</li> <li>• 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>  |
| ERROR     | Output      | BOOL            | E, A, M, D, L                | Zustandsparameter ERROR und STATUS, Fehleranzeige:  |
| STATUS    | Output      | WORD            | E, A, M, D, L                | <ul style="list-style-type: none"> <li>• ERROR=0<br/>STATUS hat den Wert: <ul style="list-style-type: none"> <li>– 0000H: weder Warnung noch Fehler</li> <li>– &lt;&gt; 0000H: Warnung, STATUS liefert detaillierte Auskunft.</li> </ul> </li> <li>• ERROR=1<br/>Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul> |
| ADDR_1    | InOut       | REMOTE          | E, A, M, D                   | Zeiger auf diejenigen Bereiche in der Partner-CPU, in die geschrieben werden soll.<br>Wenn der REMOTE-Zeiger auf einen DB zugreift, ist der DB immer zu spezifizieren.<br>Beispiel: P#DB10.DBX5.0 Byte 10.<br>Bei der Übertragung von Datenstrukturen (z.B. Struct, Array) muss an den Parametern ADDR_i der Datentyp CHAR verwendet werden.                        |
| ADDR_2    | InOut       | REMOTE          |                              |   |
| ADDR_3    | InOut       | REMOTE          |                              |   |
| ADDR_4    | InOut       | REMOTE          |                              |   |
| SD_1      | InOut       | VARIANT         | E, A, M, D, L                | Zeiger auf diejenigen Bereiche in der eigenen CPU, die die zu versendenden Daten enthalten.<br><br>Zulässig sind nur die Datentypen BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL.   |
| SD_2      | InOut       | VARIANT         |                              |   |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| SD_3      | InOut       | VARIANT  |                 | Bei der Übertragung von Datenstrukturen (z.B. Struct, Array) muss an den Parametern SD_i der Datentyp CHAR verwendet werden. |
| SD_4      | InOut       | VARIANT  |                 |  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für die Anweisung "PUT" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

| ERROR | STATUS (dezimal) | Erläuterung  |
|-------|------------------|--|
| 0     | 11               | Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.   |
| 0     | 25               | Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.  |
| 1     | 1                | Kommunikationsprobleme, z. B. <ul style="list-style-type: none"> <li>• Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>• Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP)</li> <li>• Verbindung zum Partner noch nicht aufgebaut</li> </ul>   |
| 1     | 2                | <ul style="list-style-type: none"> <li>• Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.</li> <li>• Bei S7-1500: Zugriffsschutz bei remoter Station aktiviert.</li> </ul>   |
| 1     | 4                | Fehler in den Zeigern zur Datenablage: <ul style="list-style-type: none"> <li>• Datentypen der Parameter SD_i und ADDR_i passen nicht zueinander.</li> <li>• Die Länge des Bereichs SD_i ist größer als die Länge der zu schreibenden Daten der Parameter ADDR_i.</li> <li>• Auf SD_i kann nicht zugegriffen werden.</li> <li>• Maximale Nutzdatengröße überschritten.</li> <li>• Anzahl der Parameter SD_i und ADDR_i stimmen nicht überein.</li> </ul> |
| 1     | 8                | Zugriffsfehler bei der Partner-CPU (z. B. DB nicht geladen oder schreibgeschützt).   |
| 1     | 10               | Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).  |
| 1     | 20               | <ul style="list-style-type: none"> <li>• Maximale Anzahl an parallelen Aufträgen überschritten.</li> <li>• Der Auftrag wird gerade in einer Prioritätsklasse mit niedriger Priorität aufgerufen (Erstaufruf).</li> </ul>   |

### Datenkonsistenz

Mit dem Aktivieren eines Sendevorgangs (steigende Flanke an REQ) sind die zu sendenden Daten der Sendebereiche SD\_i aus dem Anwenderprogramm kopiert. Sie können diese



Bereiche nach dem Bausteinaufruf neu beschreiben, ohne die aktuellen Sendedaten zu verfälschen.

---

#### Hinweis

Der gesamte Sendevorgang ist erst dann abgeschlossen, wenn der Zustandsparameter DONE den Wert "1" annimmt.

---

## Weitere

### USEND: Daten unkoordiniert senden

#### Beschreibung

Die Anweisung "USEND" sendet Daten an eine remote Partner-Anweisung vom Typ "URCV (Seite 2925)". Der Sendevorgang verläuft ohne Koordination mit der Partner-Anweisung, d. h. die Datenübertragung erfolgt ohne Quittierung durch die Partner-Anweisung.

Mit dem Aktivieren eines Sendevorgangs (steigende Flanke an REQ) werden die zu sendenden Daten der Sendebereiche SD\_i aus dem Anwenderprogramm kopiert. Sie können diese Sendebereiche nach dem Aufruf der Anweisung neu beschreiben, ohne die aktuellen Sendedaten zu verfälschen.

Der erfolgreiche Abschluss des Sendevorgangs wird am Zustandsparameter DONE mit "1" angezeigt.

#### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "USEND":

| Parameter | Deklaration | Datentyp  | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------|------------------------------|---|
| REQ       | Input       | BOOL      | E, A, M, D, L oder Konstante | Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke.  |
| ID        | Input       | CONN_PRG  | M, D oder Konstante          | Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.  |
| R_ID      | Input       | CONN_R_ID | E, A, M, D, L oder Konstante | Adressierungsparameter R_ID zur Festlegung der Anweisungs-Paare "USEND" und "URCV".<br>Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 2914)             |
| DONE      | Output      | BOOL      | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul> |

| Parameter        | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|------------------|-------------|----------|-----------------|---|
| ERROR            | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter<br><ul style="list-style-type: none"> <li>• 0: Weder Warnung noch Fehler.</li> <li>• 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul>  |
| STATUS           | Output      | WORD     | E, A, M, D, L   | Zustandsparameter<br>Siehe Tabelle "Parameter ERROR und STATUS".  |
| SD_i (1 ≤ i ≤ 4) | InOut       | VARIANT  | M, D            | Zeiger auf i-ten Sendebereich.<br>Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL.<br>Die maximale Nutzdatengröße für die Parameter SD_i ist abhängig von der Partner-CPU (Anweisung "URCV") sowie der Anzahl der verwendeten Parameter.<br>Weitere Informationen finden Sie unter: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 2914) |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

| ERROR | STATUS (dezimal) | Erläuterung  |
|-------|------------------|--|
| 0     | 11               | Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.   |
| 0     | 25               | Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.  |
| 1     | 1                | Kommunikationsproblem aufgetreten. Mögliche Gründe: <ul style="list-style-type: none"> <li>• Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>• Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP)</li> <li>• Verbindung zum Partner noch nicht aufgebaut</li> </ul> |
| 1     | 4                | <ul style="list-style-type: none"> <li>• Fehler in den Sendebereichszeigern SD_i bezüglich der Datenlänge oder des Datentyps.</li> <li>• Maximale Nutzdatenlänge wurde überschritten.</li> </ul>   |
| 1     | 10               | Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).  |
| 1     | 18               | Wert am Parameter R_ID existiert bereits in der am Parameter ID angegebenen Verbindung (Wert für R_ID muss für die Verbindung eindeutig sein).   |
| 1     | 20               | <ul style="list-style-type: none"> <li>• Maximale Anzahl an parallelen Aufträgen überschritten.</li> <li>• Der Auftrag wird gerade in einer Prioritätsklasse mit niedrigerer Priorität aufgerufen (Erstaufruf).</li> </ul>   |

## URCV: Daten unkoordiniert empfangen

### Beschreibung

Die Anweisung "URCV" empfängt asynchron Daten von einer remoten Partner-Anweisung vom Typ "USEND (Seite 2923)" und kopiert sie in die projizierten Empfangsbereiche.

Die Anweisung ist empfangsbereit, wenn am Eingang EN\_R eine logische "1" anliegt. Mit EN\_R=0 kann ein laufender Auftrag abgebrochen werden.

Die Empfangsdatenbereiche werden durch die Parameter RD\_1, ... RD\_4 referenziert. Sie müssen beachten, dass die über die Parameter RD\_i / RD\_1 und SD\_i / SD\_1 (bei der zugehörigen Partner-Anweisung "USEND (Seite 2923)") definierten Bereiche übereinstimmen in Anzahl und Länge.

Der erfolgreiche Abschluss des Kopiervorgangs wird am Zustandsparameter NDR mit einer logischen "1" angezeigt. Nachdem der Zustandsparameter NDR den Wert "1" angenommen hat, befinden sich neue Empfangsdaten in Ihren Empfangsbereichen (RD\_i). Ein erneuter Bausteinaufruf kann diese Daten mit neuen Empfangsdaten überschreiben. Wenn Sie dies verhindern wollen, müssen Sie "URCV" mit dem Wert "0" an EN\_R so lange aufrufen (z. B. bei zyklischer Bausteinbearbeitung), bis Sie die Bearbeitung der Empfangsdaten beendet haben.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "URCV":

| Parameter | Deklaration | Datentyp      | Speicherbereich              | Beschreibung   |
|-----------|-------------|---------------|------------------------------|--|
| EN_R      | Input       | BOOL          | E, A, M, D, L oder Konstante | Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.   |
| ID        | Input       | CONN_PR<br>G  | M, D oder Konstante          | Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.   |
| R_ID      | Input       | CONN_R_I<br>D | E, A, M, D, L oder Konstante | Adressierungsparameter zur Festlegung der Anweisungs-Paare "USEND" und "URCV".<br>Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 2914)                                     |
| NDR       | Output      | BOOL          | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder läuft noch.</li> <li>• 1: Auftrag wurde erfolgreich abgeschlossen.</li> </ul>                      |
| ERROR     | Output      | BOOL          | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Weder Warnung noch Fehler</li> <li>• 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul> |

| Parameter        | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|------------------|-------------|----------|-----------------|---|
| STATUS           | Output      | WORD     | E, A, M, D, L   | Zustandsparameter<br>Siehe Tabelle "Parameter ERROR und STATUS".  |
| RD_i (1 ≤ i ≤ 4) | InOut       | VARIANT  | E, A, M, D      | Zeiger auf den i-ten Empfangsbereich:<br>Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL.<br><br>Weitere Informationen finden Sie unter:<br>Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 2914) |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

| ERROR | STATUS (dezimal) | Erläuterung  |
|-------|------------------|--|
| 0     | 9                | Warnung: Ältere Empfangsdaten wurden von neueren Empfangsdaten überschrieben.  |
| 0     | 11               | Warnung: Die Empfangsdaten werden bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet (Fehler kann beim Kopieren der Daten in den Empfangsbereich auftreten).   |
| 0     | 25               | Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.  |
| 1     | 1                | Kommunikationsproblem aufgetreten. Mögliche Gründe: <ul style="list-style-type: none"> <li>• Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>• Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP)</li> <li>• Verbindung zum Partner noch nicht aufgebaut</li> </ul> |
| 1     | 4                | Fehler in den Empfangsbereichszeigern RD_i bezüglich der Datenlänge oder des Datentyps.  |
| 1     | 10               | Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).  |
| 1     | 18               | Wert am Parameter R_ID existiert bereits in der am Parameter ID angegebenen Verbindung (Wert für R_ID muss für die Verbindung eindeutig sein).   |
| 1     | 19               | Die zugehörige Anweisung "USEND (Seite 2923)" sendet schneller Daten, als diese von "URCV" in die Empfangsbereiche kopiert werden können.  |
| 1     | 20               | <ul style="list-style-type: none"> <li>• Maximale Anzahl an parallelen Aufträgen überschritten.</li> <li>• Der Auftrag wird gerade in einer Prioritätsklasse mit niedrigerer Priorität aufgerufen (Erstaufruf).</li> </ul>   |

### BSEND: Daten in Segmenten senden

#### Beschreibung

Die Anweisung "BSEND" sendet Daten an eine remote Partner-Anweisung vom Typ "BRCV (Seite 2929)". Bei diesem Datentransfer kann eine größere Datenmenge zwischen den Kommunikationspartnern transportiert werden, als dies mit allen anderen Kommunikations-Anweisungen für projektierte S7-Verbindungen möglich ist. Die maximale Datenmenge beträgt jeweils 65534 Byte bei der integrierten Schnittstelle als auch bei SIMATIC-Net-CP.

## Arbeitsweise

Das Anweisungspaar "BSEND" und "BRCV" legen Sie über den Parameter R\_ID fest. Der Parameter R\_ID muss bei den zusammengehörenden Anweisungen identisch sein.

Die Aktivierung des Sendevorgangs erfolgt nach Aufruf der Anweisung und positiver Flanke am Steuereingang REQ. Die Bearbeitung von "BSEND" erfolgt nach dem Aufruf nicht im Hintergrund, d.h. die Daten können nur innerhalb des Anwenderprogramms gelesen werden.

Der zu sendende Datenbereich wird segmentiert. Jedes Segment wird einzeln an den Partner gesendet. Das letzte Segment wird vom Partner nach der Übernahme dieses Segments durch "BRCV (Seite 2929)" quittiert. Bei einer Segmentierung der Daten muss "BSEND" mehrfach aufgerufen werden, bis alle Segmente übertragen worden sind.

Der Datenbereich der zu sendenden Daten wird durch SD\_1 vorgegeben. Um Datenkonsistenz zu gewährleisten, dürfen Sie den aktuell benutzten Teil des Sendebereichs SD\_1 erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter DONE den Wert "1" annimmt.

Die Länge der Sendedaten legen Sie auftragsbezogen durch LEN fest. Bei LEN = "0" werden alle Daten gesendet, die über den Parameter SD\_1 adressiert sind.

Bei einer positiven Flanke am Steuereingang R wird ein laufender Sendevorgang abgebrochen.

Aufgrund der asynchronen Datenübertragung kann ein erneutes Senden von Daten erst gestartet werden, wenn die vorhergehenden Daten durch Aufruf der Partner-Anweisung abgeholt wurden. Wenn die Daten abgeholt wurden, wird in der Partner-Anweisung "BRCV" der Zustandsparameter "NDR" gesetzt.

---

### Hinweis

#### Migration von S7-400 Anwenderprogrammen

Eine S7-400 CPU interpretiert den Parameter SD\_1 als Zeiger, nicht als Datenbereich.

Bei der S7-1500 darf LEN den Bereich von SD1 nicht überschreiten. Bei der S7-400 war dies zulässig. Empfehlung: Verwenden Sie als Größe des Datenbereichs am Parameter SD\_1 die maximale Größe für den Parameter LEN (65534 Byte bei der integrierten Schnittstelle).

---

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "BSEND":

| Parameter | Deklaration | Datentyp     | Speicherbereich              | Beschreibung  |
|-----------|-------------|--------------|------------------------------|---|
| REQ       | Input       | BOOL         | E, A, M, D, L oder Konstante | Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke                         |
| R         | Input       | BOOL         | E, A, M, D, L oder Konstante | Steuerparameter reset, aktiviert Abbruch eines noch laufenden Datenaustauschs bei steigender Flanke |
| ID        | Input       | CONN_PR<br>G | M, D oder Konstante          | Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.                                |

| Parameter | Deklaration | Datentyp  | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------|------------------------------|---|
| R_ID      | Input       | CONN_R_ID | E, A, M, D, L oder Konstante | Adressierungsparameter zur Festlegung der Anweisungs-Paare "BSEND" und "BRCV" (Seite 2929).<br>Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 2914)                         |
| SD_1      | InOut       | VARIANT   | E, A, M, D                   | Zeiger auf Sendebereich   |
| LEN       | InOut       | WORD      | E, A, M, D, L                | Länge des zu sendenden Datenblocks in Byte.<br>Bei LEN = "0" werden alle Daten von SD_1 gesendet.   |
| DONE      | Output      | BOOL      | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>• 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>                 |
| ERROR     | Output      | BOOL      | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Weder Warnung noch Fehler.</li> <li>• 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul> |
| STATUS    | Output      | WORD      | E, A, M, D, L                | Zustandsparameter<br>Siehe Tabelle "Parameter ERROR und STATUS".  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "BSEND" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

| ERROR | STATUS (dezimal) | Erläuterung  |
|-------|------------------|--|
| 0     | 11               | Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.   |
| 0     | 25               | Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.  |
| 1     | 1                | Kommunikationsproblem aufgetreten. Mögliche Gründe: <ul style="list-style-type: none"> <li>• Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>• Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP)</li> <li>• Verbindung zum Partner noch nicht aufgebaut</li> </ul>   |
| 1     | 2                | Negative Quittung von Partner-Anweisung. Die Anweisung ist nicht ausführbar.   |
| 1     | 3                | R_ID ist auf der durch ID vorgegebenen Kommunikationsverbindung nicht bekannt, oder der Empfangsbaustein wurde noch nie aufgerufen.  |
| 1     | 4                | <ul style="list-style-type: none"> <li>• Fehler im Sendebereichszeiger SD_1 bezüglich der Datenlänge oder des Datentyps.</li> <li>• Wert für LEN größer als der Bereich SD_1.</li> <li>• Es wird ein Speicherbereich mit optimiertem Zugriff verwendet und der Wert für LEN ist nicht "0".</li> <li>• Es wird ein Speicherbereich mit Standardzugriff verwendet und der Wert für LEN ist "0".</li> </ul> |
| 1     | 5                | Resetanforderung wurde ausgeführt.   |

| ERROR | STATUS (dezimal) | Erläuterung  |
|-------|------------------|--|
| 1     | 6                | Partner-Anweisung befindet sich im Zustand DISABLED (EN_R hat den Wert "0"). Überprüfen Sie zusätzlich die Eingangsparameter von "BRCV (Seite 2929)" auf Konsistenz zu "BSEND".  |
| 1     | 7                | Partner-Anweisung "BRCV (Seite 2929)" wurde nach der letzten Datenübertragung nicht mehr aufgerufen.   |
| 1     | 8                | Zugriff auf remotes Objekt im Anwenderspeicher wurde abgelehnt: Der Zielbereich beim zugehörigen "BRCV (Seite 2929)" ist zu klein.<br>An den Ausgangsparametern von "BRCV (Seite 2929)" wird ERROR = 1, STATUS = 4 oder ERROR = 1, STATUS = 10 gemeldet. |
| 1     | 10               | Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).  |
| 1     | 18               | R_ID existiert bereits in der Verbindung.  |
| 1     | 20               | <ul style="list-style-type: none"> <li>• Maximale Anzahl an parallelen Aufträgen überschritten.</li> <li>• Der Auftrag wird gerade in einer Prioritätsklasse mit niedrigerer Priorität aufgerufen (Erstaufwurf).</li> </ul>                              |

## BRCV: Daten in Segmenten empfangen

### Beschreibung

Die Anweisung "BRCV" empfängt Daten von einer remoten Partner-Anweisung vom Typ "BSEND (Seite 2926)". Der Parameter R\_ID muss bei den zusammengehörenden Anweisungen identisch sein.

Die Anweisung ist nach Aufruf mit dem Wert "1" am Steuereingang EN\_R empfangsbereit (STATUS = 25). Mit EN\_R=0 kann ein laufender Auftrag abgebrochen werden.

Der maximale Empfangsbereich wird durch RD\_1 vorgegeben. Die Daten werden konsistent empfangen, wenn Sie den aktuell benutzten Teil des Empfangsbereichs RD\_1 vollständig auswerten, bevor Sie den Baustein wieder mit dem Wert 1 am Steuereingang EN\_R aufrufen.

Nach jedem empfangenen Datensegment wird eine Quittung an die Partner-Anweisung geschickt. Bei mehreren Segmenten ist es nötig, "BRCV" mehrfach aufzurufen bis alle Segemente empfangen worden sind. Der asynchrone Datenempfang wird durch STATUS = 17 angezeigt. Die Anzahl der derzeit empfangenen Daten wird am Parameter LEN angezeigt. Der Parameter RD\_1 muss während der Operation konstant bleiben.

Der fehlerfreie Empfang sämtlicher Datensegmente wird am Zustandsparameter NDR mit dem Wert "1" angezeigt. Die empfangenen Daten bleiben unverändert bis zum nächsten Aufruf mit EN\_R=1.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "BRCV":

| Parameter | Deklaration | Datentyp | Speicherbereich              | Beschreibung   |
|-----------|-------------|----------|------------------------------|--|
| EN_R      | Input       | BOOL     | E, A, M, D, L oder Konstante | Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist. |
| ID        | Input       | CONN_PRG | M, D oder Konstante          | Adressierungsparameter zur Angabe der Verbindung zu der Partner-CPU.                                 |

| Parameter | Deklaration | Datentyp  | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------|------------------------------|--|
| R_ID      | Input       | CONN_R_ID | E, A, M, D, L oder Konstante | Adressierungsparameter zur Festlegung der Anweisungs-Paare "BSEND (Seite 2926)" und "BRCV".<br>Siehe auch: Gemeinsame Parameter der Anweisungen zur S7-Kommunikation (Seite 2914)                        |
| RD_1      | InOut       | VARIANT   | E, A, M, D                   | Zeiger auf Empfangsbereich. Die Längenangabe gibt die maximale Länge des zu empfangenden Blocks vor.   |
| LEN       | InOut       | WORD      | E, A, M, D, L                | Länge der bisher empfangenen Daten in Byte.  |
| NDR       | Output      | BOOL      | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder läuft noch.</li> <li>• 1: Auftrag wurde erfolgreich abgeschlossen.</li> </ul>                      |
| ERROR     | Output      | BOOL      | E, A, M, D, L                | Zustandsparameter <ul style="list-style-type: none"> <li>• 0: Weder Warnung noch Fehler</li> <li>• 1: Es liegt ein Fehler vor. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul> |
| STATUS    | Output      | WORD      | E, A, M, D, L                | Zustandsparameter<br>Siehe Tabelle "Parameter ERROR und STATUS".   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

Die folgende Tabelle enthält alle für "BRCV" spezifischen Fehlerinformationen, die über die Parameter ERROR und STATUS ausgegeben werden können.

| ERROR | STATUS (dezimal) | Erläuterung  |
|-------|------------------|--|
| 0     | 11               | Warnung: Die Empfangsdaten werden bereits in einer Prioritätsklasse mit niedrigerer Priorität bearbeitet.  |
| 0     | 17               | Warnung: Anweisung empfängt asynchron Daten. Der Parameter LEN zeigt die Anzahl der bisher empfangenen Daten in Byte an.   |
| 0     | 25               | Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.  |
| 1     | 1                | Kommunikationsproblem aufgetreten. Mögliche Gründe: <ul style="list-style-type: none"> <li>• Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>• Verbindung unterbrochen (z. B. Kabel, CPU ausgeschaltet, CP in STOP)</li> <li>• Verbindung zum Partner noch nicht aufgebaut</li> </ul> |
| 1     | 2                | Funktion nicht ausführbar (Protokollfehler)  |
| 1     | 4                | Fehler im Empfangsbereichszeiger RD_1 bezüglich der Datenlänge oder des Datentyps. Der gesendete Datenblock ist länger als der Empfangsbereich.  |
| 1     | 5                | Resetanforderung eingetroffen, unvollständige Übertragung.   |
| 1     | 8                | Zugriffsfehler beim zugehörigen "BSEND (Seite 2926)": Nach dem Versenden des letzten gültigen Datensegments wird ERROR = 1 und STATUS = 4 oder ERROR = 1 und STATUS = 10 gemeldet.   |



| ERROR | STATUS<br>(dezimal) | Erläuterung   |
|-------|---------------------|---|
| 1     | 10                  | Zugriff auf lokalen Anwenderspeicher nicht möglich (z. B. Zugriff auf gelöschten DB).   |
| 1     | 18                  | Wert am Parameter R_ID existiert bereits in der am Parameter ID angegebenen Verbindung (Wert für R_ID muss für die Verbindung eindeutig sein).  |
| 1     | 20                  | <ul style="list-style-type: none"> <li>• Maximale Anzahl an parallelen Aufträgen überschritten.</li> <li>• Der Auftrag wird gerade in einer Prioritätsklasse mit niedrigerer Priorität aufgerufen (Erstaufwurf).</li> </ul> |

### 9.7.5.3 Open User Communication

#### TSEND\_C: Daten über Ethernet senden

#### TSEND\_C: Daten über Ethernet senden

#### Beschreibung

Die folgende Beschreibung der Anweisung "TSEND\_C" ist gültig für die CPU S7-1200 bis Version 3.0.

Die Anweisung "TSEND\_C" richtet eine TCP- oder ISO-on-TCP-Kommunikationsverbindung ein und baut diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht. Zum Einrichten der Kommunikationsverbindung wird die am Parameter CONNECT angegebene Verbindungsbeschreibung verwendet.

Die Anweisung wird asynchron ausgeführt und hat die folgenden Funktionen:

- Eine Kommunikationsverbindung einrichten und aufbauen:  
Die Kommunikationsverbindung wird mit CONT=1 eingerichtet und aufgebaut. Bei einem erfolgreichen Aufbau der Verbindung wird der Parameter DONE für einen Zyklus auf "1" gesetzt. Wenn die CPU in den Betriebszustand STOP geht, wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung muss "TSEND\_C" erneut ausgeführt werden. Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU.
- Daten über die bestehende Kommunikationsverbindung senden:  
Den Sendebereich geben Sie am Parameter DATA an. In diesem sind Adresse und die Länge der zu sendenden Daten enthalten. Verwenden Sie am Parameter DATA keinen Datenbereich mit dem Datentyp BOOL oder Array of BOOL. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.

- Der Sendeauftrag wird ausgeführt, wenn am Parameter REQ eine steigende Flanke erfasst wird. Am Parameter LEN legen Sie die maximale Anzahl der Bytes fest, die mit einem Sendeauftrag gesendet werden. Beim Absenden von Daten (steigende Flanke am Parameter REQ) muss der Parameter CONT den Wert "1" haben, um eine Verbindung aufzubauen oder zu halten. Die zu sendenden Daten dürfen nicht editiert werden, bis der Sendeauftrag vollständig ausgeführt ist. Bei einer erfolgreichen Ausführung des Sendeauftrags wird der Parameter DONE auf "1" gesetzt. Der Signalzustand "1" am Parameter DONE ist dabei keine Bestätigung, dass die gesendeten Daten vom Kommunikationspartner bereits gelesen wurden.
- Die Kommunikationsverbindung abbauen:  
Die Kommunikationsverbindung wird abgebaut, wenn der Parameter CONT auf den Wert "0" gesetzt wird, auch wenn eine laufende Datenübertragung noch nicht abgeschlossen ist. Dies gilt jedoch nicht, wenn Sie für "TSEND\_C" eine bereits projektierte Verbindung verwenden.

Beim Setzen des Parameters COM\_RST auf "1" kann der laufende Verbindungsaufbau oder eine laufende Datenübertragung jederzeit zurücksetzen werden. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut. Wenn zum Zeitpunkt der Neuausführung Daten übertragen werden, kann es zum Datenverlust kommen.

Um "TSEND\_C" nach der Ausführung (DONE = 1) wieder freizugeben, rufen Sie die Anweisung einmal mit REQ = 0 auf.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND\_C":

| Parameter | Deklaration | Datentyp   | Speicherbereich              | Beschreibung  |
|-----------|-------------|------------|------------------------------|---|
| REQ       | Input       | BOOL       | E, A, M, D, L                | Startet den Sendeauftrag bei einer steigenden Flanke.   |
| CONT      | Input       | BOOL       | E, A, M, D, L                | Steuert die Kommunikationsverbindung: <ul style="list-style-type: none"> <li>• 0: Kommunikationsverbindung trennen</li> <li>• 1: Kommunikationsverbindung aufbauen und halten</li> </ul> Beim Absenden von Daten (steigende Flanke am Parameter REQ) muss der Parameter CONT den Wert TRUE haben, um eine Verbindung aufzubauen oder zu halten. |
| LEN       | Input       | UINT       | E, A, M, D, L oder Konstante | Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.   |
| CONNECT   | InOut       | TCON_Param | D                            | Zeiger auf die Verbindungsbeschreibung<br>Siehe auch: Auto-Hotspot  |
| DATA      | InOut       | VARIANT    | E, A, M, D, L                | Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält.  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| COM_RST   | InOut       | BOOL     | E, A, M, D, L   | Bewirkt einen Neustart der Anweisung: <ul style="list-style-type: none"> <li>• 0: Irrelevant</li> <li>• 1: Vollständiger Neustart der Anweisung, wodurch die vorhandene Verbindung abgebaut und eine neue Verbindung aufgebaut wird.</li> </ul> |
| DONE      | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>  |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>• 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.</li> </ul>         |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>   |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Status der Anweisung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TSEND\_C" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| BUSY | DONE | ERROR | Beschreibung  |
|------|------|-------|---|
| 1    | -    | -     | Der Auftrag wird bearbeitet.  |
| 0    | 1    | 0     | Der Auftrag wurde erfolgreich durchgeführt.   |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben. |
| 0    | 0    | 0     | Kein neuer Auftrag wurde zugewiesen.  |

### Parameter ERROR und STATUS

| ERROR | STATUS*<br>(W#16#...) | Beschreibung                             |
|-------|-----------------------|--|
| 0     | 0000                  | Auftrag fehlerfrei ausgeführt.           |
| 0     | 0001                  | Der Verbindungsaufbau ist abgeschlossen. |

| ERROR | STATUS*<br>(W#16#...) | Beschreibung   |
|-------|-----------------------|--|
| 0     | 0003                  | Der Verbindungsabbau ist abgeschlossen.  |
| 0     | 7000                  | Keine Auftragsbearbeitung aktiv.   |
| 0     | 7001                  | <ul style="list-style-type: none"> <li>• Auftragsbearbeitung starten</li> <li>• Verbindung aufbauen</li> <li>• Auf Verbindungspartner warten</li> </ul>  |
| 0     | 7002                  | Daten wurden gesendet.   |
| 0     | 7003                  | Verbindung wird abgebaut.  |
| 0     | 7004                  | Verbindung aufgebaut und überwacht, keine Auftragsverarbeitung aktiv.  |
| 0     | 7005                  | Daten werden gerade gesendet.  |
| 1     | 80A0                  | Sammelfehler für Fehlercodes 80A1 und 80A2.  |
| 1     | 80A1                  | <ul style="list-style-type: none"> <li>• Verbindung oder Port wird vom Anwender bereits verwendet.</li> <li>• Kommunikationsfehler: <ul style="list-style-type: none"> <li>– Die angegebene Verbindung wurde noch nicht aufgebaut.</li> <li>– Die angegebene Verbindung wird gerade beendet. Eine Übertragung über diese Verbindung ist nicht möglich.</li> <li>– Die Schnittstelle wird neu initialisiert.</li> </ul> </li> </ul> |
| 1     | 80A2                  | Lokaler oder remoter Port wird vom System verwendet.   |
| 1     | 80A3                  | Es wird versucht, eine nicht vorhandene Verbindung zu beenden.   |
| 1     | 80A4                  | IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, d.h. sie entspricht der IP-Adresse des lokalen Partners.   |
| 1     | 80A7                  | Kommunikationsfehler: Sie haben die Anweisung mit COM_RST = 1 aufgerufen, bevor der Sendeauftrag beendet war.  |
| 1     | 80B2                  | Der Parameter CONNECT zeigt auf einen Datenbaustein, der mit dem Attribut "Nur im Ladespeicher ablegen" generiert wurde.   |
| 1     | 80B3                  | Inkonsistente Parametrierung: Sammelfehler für Fehlercodes 80A0 bis 80A2, 80A4, 80B4 bis 80B9.   |
| 1     | 80B4                  | Bei Verwendung der Protokollvariante ISO on TCP (connection_type = B#16#12) für den passiven Aufbau einer Verbindung (active_est = FALSE) wurde eine oder beide der folgenden Bedingungen verletzt: "local_tsap_id_len >= B#16#02" und/oder "local_tsap_id[1] = B#16#E0".  |
| 1     | 80B5                  | Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.   |
| 1     | 80B6                  | Parametrierfehler im Parameter connection_type des Datenbausteins für Verbindungsbeschreibung.   |
| 1     | 80B7                  | Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.  |
| 1     | 8085                  | Parameter LEN ist größer als der größte zulässige Wert.  |
| 1     | 8086                  | Der Parameter ID innerhalb des Parameters CONNECT liegt außerhalb des zulässigen Bereichs.   |
| 1     | 8087                  | Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich.   |
| 1     | 8088                  | Der Wert am Parameter LEN passt nicht zu den am Parameter DATA angegebene Empfangsbereich.   |
| 1     | 8089                  | Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.   |
| 1     | 8091                  | Maximale Schachtelungstiefe überschritten.   |
| 1     | 809A                  | Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.  |
| 1     | 809B                  | Die ID des lokalen Geräts in der Verbindungsbeschreibung entspricht nicht der CPU.   |

| ERROR | STATUS*<br>(W#16#...) | Beschreibung  |
|-------|-----------------------|---|
| 1     | 80C3                  | <ul style="list-style-type: none"> <li>• Alle Verbindungsressourcen sind belegt.</li> <li>• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.</li> </ul>  |
| 1     | 80C4                  | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>• Die Verbindung kann derzeit nicht aufgebaut werden.</li> <li>• Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut.</li> <li>• Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON" entfernt.</li> <li>• Die benutzte Verbindung wird gerade durch einen Aufruf mit COM_RST= 1 beendet</li> </ul> |
| 1     | 80C6                  | Remoter Netzwerkfehler. Remoter Partner kann nicht erreicht werden.   |
| 1     | 8722                  | Parameter CONNECT: Der Quellbereich ist ungültig. Der Bereich ist im DB nicht vorhanden.  |
| 1     | 873A                  | Parameter CONNECT: Zugriff auf Verbindungsbeschreibung nicht möglich (z.B. DB nicht verfügbar).   |
| 1     | 877F                  | Parameter CONNECT: Interner Fehler.   |
| 1     | 8822                  | Parameter DATA: Quellbereich ungültig, Bereich im DB nicht vorhanden.   |
| 1     | 8824                  | Parameter DATA: Bereichsfehler im VARIANT-Pointer.  |
| 1     | 8832                  | Parameter DATA: DB-Nummer ist zu groß.  |
| 1     | 883A                  | Parameter CONNECT: Zugriff auf die angegebene Verbindungsdaten nicht möglich (z.B. weil der DB nicht vorhanden ist).  |
| 1     | 887F                  | Parameter DATA: Interner Fehler, z. B. unzulässige VARIANT-Referenz.  |
| 1     | 893A                  | Parameter DATA: Zugriff auf den Sendebereich nicht möglich (z.B. weil der DB nicht vorhanden ist).  |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### Hinweis

#### Fehlermeldungen der Anweisungen "TCON", "TSEND", "T\_DIAG" und "TDISCON"

Die Anweisung "TSEND\_C" verwendet intern die Anweisungen "TCON (Seite 2968)", "TSEND (Seite 2981)", "T\_DIAG (Seite 3002)", "T\_RESET (Seite 3000)" und "TDISCON (Seite 2976)". Die Fehlermeldungen dieser Anweisungen können ebenfalls am Parameter STATUS ausgegeben werden. Die Bedeutung der Fehlercodes ist bei den entsprechenden Anweisungen beschrieben. Bei gleichen Fehlercodes intern verwendeter Anweisungen mit unterschiedlicher Bedeutung kann über den Instanzdatenbaustein von "TSEND\_C" ermittelt werden, durch welche Anweisung der Fehler ausgegeben wurde.

## TSEND\_C: Daten über Ethernet senden

### Beschreibung

Die folgende Beschreibung der Anweisung "TSEND\_C" ist gültig für die CPU S7-1500 und S7-1200 V4.0.

Die Anweisung "TSEND\_C" richtet eine Kommunikationsverbindung ein und baut diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht.

Die Anweisung wird asynchron ausgeführt und hat die folgenden Funktionen:

- Eine Kommunikationsverbindung einrichten und aufbauen
- Daten über die bestehende Kommunikationsverbindung senden
- Die Kommunikationsverbindung abbauen oder zurücksetzen

Die Anweisung "TSEND\_C" verwendet dabei intern die Kommunikations-Anweisungen "TCON", "TSEND", "T\_DIAG", "T\_RESET" und "TDISCON".

### Kommunikationsverbindung einrichten und aufbauen

Die Kommunikationsverbindung wird mit CONT=1 eingerichtet und aufgebaut. Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU. Zum Einrichten der Kommunikationsverbindung wird die am Parameter CONNECT angegebene Verbindungsbeschreibung verwendet. Die folgenden Verbindungstypen können verwendet werden:

- Programmierte Verbindungen (Aufbau der Verbindung über "TCON"):
  - TCP / UDP: Verbindungsbeschreibung über den Systemdatentyp TCON\_IP\_v4
  - ISO-on-TCP: Verbindungsbeschreibung über den Systemdatentyp TCON\_IP\_RFC
  - ISO: Verbindungsbeschreibung über den Systemdatentyp TCON\_ISONative (nur bei CP1543-1)
- Konfigurierte Verbindungen
  - Angabe einer bestehenden Verbindung in dem Systemdatentyp TCON\_Configured.

Wenn die CPU in den Betriebszustand STOP geht, wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung muss "TSEND\_C" erneut ausgeführt werden.

### Daten über die bestehende Kommunikationsverbindung senden

Der Sendeauftrag wird ausgeführt, wenn am Parameter REQ eine steigende Flanke erfasst wird. Wie oben beschrieben wird zuvor die Kommunikationsverbindung aufgebaut.

Den Sendebereich geben Sie am Parameter DATA an. In diesem sind Adresse und die Länge der zu sendenden Daten enthalten. Verwenden Sie am Parameter DATA keinen Datenbereich mit dem Datentyp BOOL oder Array of BOOL. Am Parameter LEN legen Sie die maximale Anzahl der Bytes fest, die mit einem Sendeauftrag gesendet werden. Wenn Sie einen Sendebereich mit optimiertem Zugriff am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.

Die zu sendenden Daten dürfen nicht editiert werden, bis der Sendeauftrag vollständig ausgeführt ist.

### Die Kommunikationsverbindung abbauen und zurücksetzen

Die Kommunikationsverbindung wird abgebaut, wenn der Parameter CONT auf den Wert "0" gesetzt wird, auch wenn eine laufende Datenübertragung noch nicht abgeschlossen ist. Dies gilt jedoch nicht, wenn Sie für "TSEND\_C" eine konfigurierte Verbindung verwenden.

Durch Setzen des Parameters COM\_RST auf "1" kann die Verbindung jederzeit zurückgesetzt werden. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut. Wenn zu diesem Zeitpunkt Daten übertragen werden, kann es zum Datenverlust kommen.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND\_C":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung  |
|-----------|-------------|----------|------------------------------------|---|
| REQ       | Input       | BOOL     | E, A, M, D, L, T, Z oder Konstante | Startet den Sendeauftrag bei einer steigenden Flanke.   |
| CONT      | Input       | BOOL     | E, A, M, D, L                      | Steuert die Kommunikationsverbindung: <ul style="list-style-type: none"> <li>• 0: Kommunikationsverbindung trennen.</li> <li>• 1: Kommunikationsverbindung aufbauen und halten.</li> </ul>  |
| LEN       | Input       | UDINT    | E, A, M, D, L oder Konstante       | Optionaler Parameter (ausgeblendet)<br>Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden. Wenn Sie einen Sendebereich mit optimiertem Zugriff am Parameter DATA verwenden, muss am Parameter LEN der Wert "0" verwendet werden.  |
| CONNECT   | InOut       | VARIANT  | D                                  | Zeiger auf die Struktur zur Verbindungsbeschreibung: <ul style="list-style-type: none"> <li>• Programmierter Verbindung: <ul style="list-style-type: none"> <li>– Bei TCP verwenden Sie den Systemdatentyp TCON_IP_v4<br/>Zur Beschreibung siehe: Auto-Hotspot</li> <li>– Bei ISO-on-TCP verwenden Sie den Systemdatentyp TCON_IP_RFC<br/>Zur Beschreibung siehe: Auto-Hotspot</li> <li>– Bei ISO verwenden Sie den Systemdatentyp TCON_ISOnative (nur bei CP1543-1)<br/>Zur Beschreibung siehe Anweisung "TCON (Seite 2971)".</li> </ul> </li> <li>• Konfigurierte Verbindung: <ul style="list-style-type: none"> <li>– Für bestehende Verbindungen verwenden Sie den Systemdatentyp TCON_Configured<br/>Zur Beschreibung siehe unten "Systemdatentyp für konfigurierte Verbindungen"</li> </ul> </li> </ul> |
| DATA      | InOut       | VARIANT  | E, A, M, D, L                      | Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält.  |
| ADDR      | InOut       | VARIANT  | D                                  | Optionaler Parameter (ausgeblendet)<br>Zeiger auf die Adresse des Empfängers.   |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| COM_RST   | InOut       | BOOL     | E, A, M, D, L   | <p>Optionaler Parameter (ausgeblendet)</p> <p>Bewirkt ein Zurücksetzen der Verbindung:</p> <ul style="list-style-type: none"> <li>• 0: Irrelevant</li> <li>• 1: Die vorhandene Verbindung wird zurückgesetzt.</li> </ul> <p>Der Parameter COM_RST wird nach Auswertung durch die Anweisung "TSEND_C" rückgesetzt und sollte daher nicht statisch verschaltet werden.</p>  |
| DONE      | Output      | BOOL     | E, A, M, D, L   | <p>Statusparameter mit folgenden Werten:</p> <ul style="list-style-type: none"> <li>• 0: Sendeauftrag noch nicht gestartet oder noch in Bearbeitung.</li> <li>• 1: Sendeauftrag fehlerfrei ausgeführt. Dieser Zustand wird nur für einen Zyklus angezeigt.</li> </ul> <p>Der Ausgangsparameter DONE wird gesetzt, wenn ein Zwischenschritt bei der Bearbeitung erfolgreich abgeschlossen wurde (Verbindungsaufbau, Senden, Verbindungsabbau) und wenn die Ausführung von "TSEND_C" erfolgreich abgeschlossen ist.</p> |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | <p>Statusparameter mit folgenden Werten:</p> <ul style="list-style-type: none"> <li>• 0: Sendeauftrag noch nicht gestartet oder bereits beendet.</li> <li>• 1: Sendeauftrag noch nicht beendet. Ein neuer Sendeauftrag kann nicht gestartet werden.</li> </ul>  |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | <p>Statusparameter mit folgenden Werten:</p> <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten beim Verbindungsaufbau, beim Senden der Daten oder beim Verbindungsabbau.</li> </ul> <p>Der Ausgangsparameter ERROR kann auf Grund eines Fehlers bei der Anweisung "TSEND_C" oder den intern verwendeten Kommunikations-Anweisungen gesetzt werden.</p>   |
| STATUS    | Output      | WORD     | E, A, M, D, L   | <p>Status der Anweisung (siehe Beschreibung "Parameter ERROR und STATUS").</p>  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".



## Parameter REQ, CONT und COM\_RST

Der Parameter CONT steuert den Verbindungsaufbau der Anweisung "TSEND\_C" unabhängig von dem Parameter REQ. Das Verhalten des Parameters CONT hängt zum Teil davon ab, ob eine programmierte oder konfigurierte Verbindung verwendet wird:

- Bei CONT = "0": Es werden keine Daten gesendet (unabhängig davon, ob eine programmierte oder konfigurierte Verbindung verwendet wird).
- Beim Wechsel von CONT = "0" auf "1":
  - Bei einer programmierten Verbindung wird diese mit "TCON" aufgebaut.
  - Bei einer konfigurierten Verbindung wird diese mit "T\_DIAG" überprüft.
- Bei CONT = "1":
  - Solange keine Daten gesendet werden (REQ="0"), wird die Verbindung mit "T\_DIAG" überprüft.
  - Wird durch die intern verwendeten Kommunikations-Anweisungen gemeldet, dass kein Verbindungsendpunkt vorhanden ist, wird die Verbindung über "TCON" automatisch wieder aufgebaut.
- Beim Wechsel von CONT = "1" auf "0":
  - Bei einer programmierten Verbindung wird diese mit "TDISCON" abgebaut.
  - Bei einer konfigurierten Verbindung wird diese mit "T\_RESET" zurückgesetzt.

Der Parameter COM\_RST setzt die Verbindung beim Wechsel von "0" auf "1" zurück:

- Ist eine Verbindung aufgebaut, wird diese über "T\_RESET" zurückgesetzt (unabhängig davon, ob eine programmierte oder konfigurierte Verbindung verwendet wird).
- Ist keine Verbindung aufgebaut, hat das Setzen des Parameters keine Auswirkungen.

Die Parameter REQ und COM\_RST sind nur aktiv, wenn CONT auf "1" gesetzt wurde. Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern REQ, CONT und COM\_RST:

| REQ        | CONT  | COM_RST    | Status der Anweisung   | Beschreibung  |
|------------|-------|------------|------------------------|---|
| irrelevant | 0     | irrelevant | Noch nicht ausgeführt  | Kein Auftrag aktiv (STATUS = 7000).   |
| irrelevant | 0     | irrelevant | Initialisierung        | Verbindung wird abgebaut. Die Anweisung wird zurückgesetzt.                     |
| irrelevant | 0 > 1 | irrelevant | Verbindungsaufbau      | Verbindung wird aufgebaut. Es werden noch keine Daten übertragen.               |
| 0          | 1     | 0          | Verbindung hergestellt | Die Verbindung ist hergestellt und wird mit der Anweisung "T_DIAG" überwacht.   |
| irrelevant | 1     | 0 > 1      | Verbindung hergestellt | Die Verbindung wird durch "T_RESET" kurzfristig unterbrochen und zurückgesetzt. |
| 0 > 1      | 1     | 0          | Verbindung hergestellt | Anweisung beginnt zu senden.  |
| irrelevant | 1     | 0 > 1      | Daten werden gesendet  | Datenübertragung wird unterbrochen. Die Verbindung wird zurückgesetzt.          |

### Systemdatentyp für konfigurierte Verbindungen

Verwenden Sie für konfigurierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON\_Configured:

| Byte    | Parameter      | Datentyp | Startwert | Beschreibung  |
|---------|----------------|----------|-----------|---|
| 0 ... 1 | InterfacelD    | HW_ANY   | -         | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).   |
| 2 ... 3 | ID             | CONN_OUC | -         | Referenz auf die Verbindung (Wertebereich: 1 bis 4095). Geben Sie die Verbindungs-ID der bestehenden Verbindung an. |
| 4       | ConnectionType | BYTE     | -         | Verbindungstyp<br>Wählen Sie 254 (dezimal) für eine konfigurierte Verbindung.                                       |

### Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Sendeauftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TSEND\_C" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| DONE | BUSY | ERROR | Beschreibung  |
|------|------|-------|---|
| 0    | 0    | 0     | Die Anweisung wurde noch nicht ausgeführt (keine steigende Flanke am Parameter REQ).  |
| 0    | 1    | 0     | Die Anweisung wird ausgeführt und ruft die intern verwendeten Kommunikations-Anweisungen auf.   |
| 1    | 0    | 0     | Der Sendeauftrag wurde erfolgreich durchgeführt. Am Parameter STATUS wird "0000" ausgegeben. DONE = "1" wird nur für einen Zyklus angezeigt.  |
| 0    | 0    | 1     | Die Ausführung der Anweisung oder ein Zwischenschritt bei der Bearbeitung wurde mit einem Fehler beendet. Kommt es durch eine intern verwendete Kommunikations-Anweisung zu einem Folgefehler, wird der Fehler angezeigt, welcher bei der Bearbeitung zuerst aufgetreten ist. Dieser Zustand wird nur für einen Zyklus angezeigt. |

### Parameter ERROR und STATUS

| ERROR | STATUS*<br>(W#16#...) | Beschreibung  |
|-------|-----------------------|---|
| 0     | 0000                  | Sendeauftrag wurde fehlerfrei ausgeführt.                                       |
| 0     | 0001                  | Der Verbindungsaufbau ist abgeschlossen.  |
| 0     | 0003                  | Der Verbindungsabbau ist abgeschlossen.   |
| 0     | 7000                  | Keine Sendeauftragsbearbeitung aktiv, keine Kommunikationsverbindung aufgebaut. |

| ERROR | STATUS*<br>(W#16#...) | Beschreibung   |
|-------|-----------------------|--|
| 0     | 7001                  | <ul style="list-style-type: none"> <li>• Sendeauftragsbearbeitung starten.</li> <li>• Verbindung aufbauen.</li> <li>• Auf Verbindungspartner warten.</li> </ul>  |
| 0     | 7002                  | Daten werden gesendet.   |
| 0     | 7003                  | Kommunikationsverbindung wird abgebaut.  |
| 0     | 7004                  | Kommunikationsverbindung aufgebaut und überwacht, keine Sendeauftragsverarbeitung aktiv.   |
| 0     | 7005                  | Kommunikationsverbindung wird rückgesetzt.   |
| 1     | 80A1                  | <ul style="list-style-type: none"> <li>• Verbindung oder Port wird vom Anwender bereits verwendet.</li> <li>• Kommunikationsfehler: <ul style="list-style-type: none"> <li>– Die angegebene Verbindung wurde noch nicht aufgebaut.</li> <li>– Die angegebene Verbindung wird gerade beendet.<br/>Eine Übertragung über diese Verbindung ist nicht möglich.</li> <li>– Die Schnittstelle wird neu initialisiert.</li> </ul> </li> </ul> |
| 1     | 80A3                  | Es wird versucht, eine nicht vorhandene Verbindung zu beenden.   |
| 1     | 80A4                  | IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, oder sie passt zur IP-Adresse des lokalen Partners.  |
| 1     | 80A7                  | Kommunikationsfehler: Sie haben die Anweisung mit COM_RST = 1 aufgerufen, bevor der Sendeauftrag beendet war.  |
| 1     | 80B4                  | Bei Verwendung der Protokollvariante ISO on TCP (connection_type = B#16#12) für den passiven Aufbau einer Verbindung (active_est = FALSE) wurde eine oder beide der folgenden Bedingungen verletzt: <ul style="list-style-type: none"> <li>• local_tsap_id_len &gt;= B#16#02</li> <li>• local_tsap_id[1] = B#16#E0</li> </ul>  |
| 1     | 80B5                  | Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.   |
| 1     | 80B6                  | Parametrierfehler im Parameter connection_type des Datenbausteins für Verbindungsbeschreibung.   |
| 1     | 80B7                  | Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.  |
| 1     | 8085                  | Parameter LEN ist größer als der größte zulässige Wert.  |
| 1     | 8086                  | Der Parameter ID innerhalb des Parameters CONNECT liegt außerhalb des zulässigen Bereichs.   |
| 1     | 8087                  | Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich.   |
| 1     | 8088                  | Der Wert am Parameter LEN passt nicht zu den am Parameter DATA angegebenen Empfangsbereich.  |
| 1     | 8089                  | Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.   |
| 1     | 8091                  | Maximale Schachtelungstiefe überschritten.   |
| 1     | 809A                  | Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.  |
| 1     | 809B                  | InterfacelD ist ungültig. Sie ist entweder Null, oder sie zeigt nicht auf eine lokale CPU-Schnittstelle oder einen CP.   |
| 1     | 80C3                  | <ul style="list-style-type: none"> <li>• Alle Verbindungsressourcen sind belegt.</li> <li>• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.</li> </ul>   |

| ERROR | STATUS*<br>(W#16#...) | Beschreibung  |
|-------|-----------------------|---|
| 1     | 80C4                  | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>• Die Verbindung kann derzeit nicht aufgebaut werden.</li> <li>• Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut.</li> <li>• Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON (Seite 2976)" entfernt.</li> <li>• Die benutzte Verbindung wird gerade durch einen Aufruf mit COM_RST = 1 beendet</li> </ul> |
| 1     | 80C6                  | Remoter Netzwerkfehler. Remoter Partner kann nicht erreicht werden.   |
| 1     | 8722                  | Parameter CONNECT: Der Quellbereich ist ungültig. Der Bereich ist im DB nicht vorhanden.  |
| 1     | 873A                  | Parameter CONNECT: Zugriff auf Verbindungsbeschreibung nicht möglich (z. B. weil der DB nicht verfügbar ist).   |
| 1     | 877F                  | Parameter CONNECT: Interner Fehler.   |
| 1     | 8822                  | Parameter DATA: Quellbereich ungültig, Bereich im DB nicht vorhanden.   |
| 1     | 8824                  | Parameter DATA: Bereichsfehler im VARIANT-Zeiger.   |
| 1     | 8832                  | Parameter DATA: DB-Nummer ist zu groß.  |
| 1     | 883A                  | Parameter CONNECT: Zugriff auf die angegebenen Verbindungsdaten nicht möglich (z. B. weil der DB nicht vorhanden ist).  |
| 1     | 887F                  | Parameter DATA: Interner Fehler, z. B. unzulässige VARIANT-Referenz.  |
| 1     | 893A                  | Parameter DATA: Zugriff auf den Sendebereich nicht möglich (z.B. weil der DB nicht vorhanden ist).  |

\* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**Hinweis**

**Fehlermeldungen der Anweisungen "TCON", "TSEND", "T\_DIAG", "T\_RESET" und "TDISCON"**

Die Anweisung "TSEND\_C" verwendet intern die Anweisungen "TCON (Seite 2971)", "TSEND (Seite 2981)", "T\_DIAG (Seite 3002)", "T\_RESET (Seite 3000)" und "TDISCON (Seite 2976)". Die Fehlermeldungen dieser Anweisungen können ebenfalls am Parameter STATUS ausgegeben werden. Die Bedeutung der Fehlercodes ist bei den entsprechenden Anweisungen beschrieben. Bei gleichen Fehlercodes intern verwendeter Anweisungen mit unterschiedlicher Bedeutung kann über den Instanzdatenbaustein von "TSEND\_C" ermittelt werden, durch welche Anweisung der Fehler ausgegeben wurde.

**TRCV\_C: Daten über Ethernet empfangen**

**TRCV\_C: Daten über Ethernet empfangen**

**Beschreibung**

Die Anweisung "TRCV\_C" wird asynchron ausgeführt und hat die folgenden Funktionen:

1. **Eine Kommunikationsverbindung einrichten und aufbauen:**

"TRCV\_C" richtet eine TCP- oder ISO-on-TCP-Kommunikationsverbindung ein und baut diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht.

Zum Einrichten der Kommunikationsverbindung wird die am Parameter CONNECT angegebene Verbindungsbeschreibung verwendet. Um die Verbindung aufzubauen, muss der Parameter CONT auf den Wert "1" gesetzt werden. Bei einem erfolgreichen Aufbau der Verbindung wird der Parameter DONE auf "1" gesetzt.

Wenn die CPU in den Betriebszustand STOP geht, wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung muss "TRCV\_C" erneut ausgeführt werden.

Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU.
  2. **Daten über die bestehende Kommunikationsverbindung empfangen:**

Wenn der Parameter EN\_R auf den Wert "1" gesetzt wird, wird der Datenempfang aktiviert. Beim Empfangen von Daten (steigende Flanke am Parameter EN\_R) muss der Parameter CONT den Wert TRUE haben, um eine Verbindung aufzubauen oder zu halten.

Die empfangenen Daten werden in einen Empfangsbereich eingetragen. Die Länge des Empfangsbereichs legen Sie abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls  $LEN \neq 0$ ) oder die Längenangabe des Parameters DATA (falls  $LEN = 0$ ) fest. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.

Nach einem erfolgreichen Datenempfang ist der Signalzustand am Parameter DONE "1". Wenn bei der Datenübertragung Fehler auftreten, wird der Parameter DONE auf "0" zurückgesetzt.
  3. **Die Kommunikationsverbindung abbauen:**

Die Kommunikationsverbindung wird abgebaut, wenn der Parameter CONT auf den Wert "0" gesetzt wird.
- Beim Setzen des Parameters COM\_RST wird "TRCV\_C" neu ausgeführt. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut. Wenn zum Zeitpunkt der Neuausführung Daten empfangen werden, kann es zum Datenverlust kommen.

## Empfangsmodi von TRCV\_C

Die folgende Tabelle zeigt, wie die empfangenen Daten in den Empfangsbereich eingetragen werden.

| Protokollvariante                                      | Verfügbarkeit der Daten im Empfangsbereich   | Parameter connection_type der Verbindungsbeschreibung | Parameter LEN   |
|--|--|---|---|
| TCP<br>(Ad-hoc-Modus)                                  | Die Daten sind sofort verfügbar.   | Hexadezimalwert: B#16#11<br>Integerwert: 17           | 0   |
| TCP<br>(Datenempfang mit angegebener Länge)            | Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde. | Hexadezimalwert: B#16#11<br>Integerwert: 17           | 1 bis 8192  |
| ISO on TCP<br>(Nachrichtenorientiert Datenübertragung) | Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde. | Hexadezimalwert: B#16#12<br>Integerwert: 18           | <ul style="list-style-type: none"> <li>• 1 bis 1452, wenn ein CP verwendet wird.</li> <li>• 1 bis 8192, wenn kein CP verwendet wird.</li> </ul> |

### TCP (Ad-hoc-Modus)

Den Ad-hoc-Modus gibt es nur bei der Protokollvariante TCP. Den Ad-hoc-Modus verwenden Sie, um mit der Anweisung "TRCV" Daten mit dynamischen Längen zu empfangen.

Den Ad-hoc-Modus stellen Sie ein, indem Sie dem Parameter LEN den Wert "0" zuweisen. Bei Verwendung des Ad-hoc-Modus können für Datenbausteine mit Standardzugriff alle Datentypen verwendet werden. Für Datenbausteine mit optimiertem Zugriff können als Datentypen nur ARRAY of BYTE oder Datentypen mit einer Länge von 8 Bit verwendet werden (z. B. CHAR, USINT, SINT, etc.). Die tatsächlich empfangene Datenlänge wird am Parameter RCVD\_LEN ausgegeben.

### TCP (Datenempfang mit angegebener Länge)

Die Länge für den Datenempfang geben Sie durch den Wert des Parameters LEN vor. Der Datenempfang wird erst abgeschlossen, wenn die am Parameter LEN angegebene Länge der Daten vollständig empfangen wurde. Erst danach sind die Daten im Empfangsbereich (Parameter DATA) verfügbar. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD\_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

### ISO on TCP (Nachrichtenorientiert Datenübertragung)

Bei der Protokollvariante ISO on TCP werden über eine Verbindung abgeschlossene Nachrichtenblöcke gesendet, die vom Empfänger auch als solche erkannt werden. Bei Verwendung von ISO on TCP meldet "TRCV\_C" den Datenempfang, sobald der Nachrichtenblock vollständig empfangen wurde. Der Empfangsbereich wird von den Parametern LEN und DATA definiert. Ist der Empfangspuffer (Parameter DATA) zu klein für die gesendeten Daten, meldet "TRCV\_C" einen Fehler. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD\_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV\_C":

| Parameter | Deklaration | Datentyp   | Speicherbereich              | Beschreibung  |
|-----------|-------------|------------|------------------------------|---|
| EN_R      | Input       | BOOL       | E, A, M, D, L                | Empfangsfreigabe  |
| CONT      | Input       | BOOL       | E, A, M, D, L                | Steuert die Kommunikationsverbindung: <ul style="list-style-type: none"> <li>• 0: Kommunikationsverbindung nach Senden der Daten automatisch trennen</li> <li>• 1: Kommunikationsverbindung aufbauen und halten</li> </ul> Beim Empfangen von Daten (steigende Flanke am Parameter EN_R) muss der Parameter CONT den Wert TRUE haben, um eine Verbindung aufzubauen oder zu halten. |
| LEN       | Input       | UINT       | E, A, M, D, L oder Konstante | Maximale Länge der zu empfangenen Daten. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.  |
| CONNECT   | InOut       | TCON_Param | D                            | Zeiger auf die Verbindungsbeschreibung<br>Siehe auch: Auto-Hotspot  |
| DATA      | InOut       | VARIANT    | E, A, M, D, L                | Zeiger auf den Empfangsbereich  |
| COM_RST   | InOut       | BOOL       | E, A, M, D, L                | Bewirkt einen Neustart der Anweisung: <ul style="list-style-type: none"> <li>• 0: Irrelevant</li> <li>• 1: Vollständiger Neustart der Anweisung, wodurch die vorhandene Verbindung abgebaut wird</li> </ul>   |
| DONE      | Output      | BOOL       | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>  |
| BUSY      | Output      | BOOL       | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>• 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden</li> </ul>  |
| ERROR     | Output      | BOOL       | E, A, M, D, L                | Statusparameter ERROR: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>  |
| STATUS    | Output      | WORD       | E, A, M, D, L                | Status der Anweisung  |
| RCVD_LEN  | Output      | UDINT      | E, A, M, D, L                | Tatsächlich empfangene Datenmenge in Byte   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TRCV\_C" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| BUSY | DONE | ERROR | Beschreibung   |
|------|------|-------|--|
| 1    | -    | -     | Der Auftrag wird bearbeitet.   |
| 0    | 1    | 0     | Der Auftrag wurde erfolgreich durchgeführt.  |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben. |
| 0    | 0    | 0     | Kein neuer Auftrag wurde zugewiesen.   |

### Parameter ERROR und STATUS

| ERROR | STATUS*<br>(W#16#...) | Beschreibung  |
|-------|-----------------------|---|
| 0     | 0000                  | Auftrag fehlerfrei ausgeführt.  |
| 0     | 0001                  | Der Verbindungsaufbau ist abgeschlossen.  |
| 0     | 0003                  | Der Verbindungsabbau ist abgeschlossen.   |
| 0     | 7000                  | Keine Auftragsbearbeitung aktiv.  |
| 0     | 7001                  | <ul style="list-style-type: none"> <li>• Auftragsbearbeitung starten</li> <li>• Verbindung aufbauen</li> <li>• Auf Verbindungspartner warten</li> </ul>   |
| 0     | 7002                  | Daten werden empfangen.   |
| 0     | 7003                  | Verbindung wird beendet.  |
| 0     | 7004                  | <ul style="list-style-type: none"> <li>• Verbindung aufgebaut und überwacht</li> <li>• Keine Auftragsbearbeitung aktiv</li> </ul>   |
| 0     | 7006                  | Daten werden gerade empfangen.  |
| 1     | 8085                  | <ul style="list-style-type: none"> <li>• Parameter LEN ist größer als der größte zulässige Wert.</li> <li>• Der Wert am Parameter LEN oder DATA wurde nach dem ersten Aufruf geändert.</li> </ul> |
| 1     | 8086                  | Der Parameter ID liegt außerhalb des zulässigen Wertebereichs.  |
| 1     | 8087                  | Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich   |
| 1     | 8088                  | Der Wert am Parameter LEN passt nicht zu dem am Parameter DATA angegebenen Empfangsbereich.   |
| 1     | 8089                  | Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.  |
| 1     | 8091                  | Maximale Schachtelungstiefe überschritten.  |
| 1     | 809A                  | Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.   |
| 1     | 809B                  | Die ID des lokalen Geräts (local_device_id) in der Verbindungsbeschreibung entspricht nicht der CPU.  |



| ERROR | STATUS*<br>(W#16#...) | Beschreibung   |
|-------|-----------------------|--|
| 1     | 80A0                  | Sammelfehler für Fehlercodes W#16#80A1 und W#16#80A2.  |
| 1     | 80A1                  | <ul style="list-style-type: none"> <li>• Verbindung oder Port wird vom Anwender bereits verwendet.</li> <li>• Kommunikationsfehler: <ul style="list-style-type: none"> <li>– Die angegebene Verbindung wurde noch nicht aufgebaut.</li> <li>– Die angegebene Verbindung wird gerade beendet. Übertragung über diese Verbindung ist nicht möglich.</li> <li>– Die Schnittstelle wird neu initialisiert.</li> </ul> </li> </ul>              |
| 1     | 80A2                  | Lokaler oder remoter Port wird vom System verwendet.   |
| 1     | 80A3                  | <ul style="list-style-type: none"> <li>• Es wird versucht, eine vorhandene Verbindung erneut aufzubauen.</li> <li>• Es wird versucht, eine nicht vorhandene Verbindung zu beenden.</li> </ul>  |
| 1     | 80A4                  | IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, d.h. sie entspricht der IP-Adresse des lokalen Partners.   |
| 1     | 80A7                  | Kommunikationsfehler: Sie haben die Anweisung mit COM_RST = 1 aufgerufen, bevor der Sendeauftrag beendet war.  |
| 1     | 80B2                  | Der Parameter CONNECT zeigt auf einen Datenbaustein, der mit dem Attribut "Nur im Ladespeicher ablegen" generiert wurde.   |
| 1     | 80B3                  | Inkonsistente Parametrierung: Sammelfehler für Fehlercodes W#16#80A0 bis W#16#80A2, W#16#80A4, W#16#80B4 bis W#16#80B9.  |
| 1     | 80B4                  | Bei Verwendung der Protokollvariante ISO on TCP (connection_type = B#16#12) für den passiven Aufbau einer Verbindung (active_est = FALSE) wurde eine oder beide der folgenden Bedingungen verletzt: "local_tsap_id_len >= B#16#02" und/oder "local_tsap_id[1] = B#16#E0".  |
| 1     | 80B5                  | Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.   |
| 1     | 80B6                  | Parametrierfehler im Parameter connection_type des Datenbausteins für Verbindungsbeschreibung.   |
| 1     | 80B7                  | Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.  |
| 1     | 80C3                  | <ul style="list-style-type: none"> <li>• Alle Verbindungsressourcen sind belegt.</li> <li>• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.</li> </ul>   |
| 1     | 80C4                  | <p>Temporärer Kommunikationsfehler:</p> <ul style="list-style-type: none"> <li>• Die Verbindung kann derzeit nicht aufgebaut werden.</li> <li>• Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut.</li> <li>• Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON" entfernt.</li> <li>• Die benutzte Verbindung wird gerade durch einen Aufruf mit COM_RST= 1 beendet</li> </ul> |
| 1     | 8722                  | Fehler im Parameter CONNECT: Ungültiger Quellbereich (der Bereich ist im Datenbaustein nicht deklariert).  |
| 1     | 873A                  | Fehler im Parameter CONNECT: Zugriff auf die Verbindungsbeschreibung nicht möglich (kein Zugriff auf den Datenbaustein).   |
| 1     | 877F                  | Fehler im Parameter CONNECT: Interner Fehler   |
| 1     | 8922                  | Parameter DATA: Zielbereich ungültig, Bereich im DB nicht vorhanden.   |
| 1     | 8924                  | Parameter DATA: Bereichsfehler im VARIANT-Pointer.   |
| 1     | 8932                  | Parameter DATA: DB-Nummer ist zu groß.   |
| 1     | 893A                  | Parameter CONNECT: Zugriff auf die angegebenen Verbindungsdaten nicht möglich (z.B. weil der DB nicht vorhanden ist).  |
| 1     | 897F                  | Parameter DATA: Interner Fehler, z. B. unzulässige VARIANT-Referenz.   |

| ERROR   | STATUS*<br>(W#16#...) | Beschreibung  |
|---|-----------------------|---|
| 1   | 8A3A                  | Parameter DATA: Kein Zugriff auf den Datenbereich, da z.B. der Datenbaustein nicht existiert. |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                       |   |

**Hinweis**

**Fehlermeldungen der Anweisungen "TCON", "TRCV" und "TDISCON"**

Die Anweisung "TRV\_C" verwendet intern die Anweisungen "TCON (Seite 2968)", "TRCV (Seite 2984)" und "TDISCON (Seite 2976)". Die Fehlermeldungen dieser Anweisungen sind in den jeweiligen Beschreibungen enthalten.

**TRCV\_C: Daten über Ethernet empfangen**

**Beschreibung**

Die folgende Beschreibung der Anweisung "TRCV\_C" ist gültig für die CPU S7-1500 und S7-1200 V4.0.

Die Anweisung "TRCV\_C" wird asynchron ausgeführt und führt der Reihe nach die folgenden Funktionen aus:

- Eine Kommunikationsverbindung einrichten und aufbauen
- Daten über die bestehende Kommunikationsverbindung empfangen
- Die Kommunikationsverbindung abbauen oder zurücksetzen

Die Anweisung "TRCV\_C" verwendet dabei intern die Kommunikations-Anweisungen "TCON", "TRCV", "T\_DIAG", "T\_RESET" und "TDISCON".

**Eine Kommunikationsverbindung einrichten und aufbauen**

Die Kommunikationsverbindung wird mit CONT=1 eingerichtet und aufgebaut. Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU. Zum Einrichten der Kommunikationsverbindung wird die am Parameter CONNECT angegebene Verbindungsbeschreibung verwendet. Die folgenden Verbindungstypen können verwendet werden:

- Programmierte Verbindungen (Aufbau der Verbindung über "TCON"):
  - TCP / UDP: Verbindungsbeschreibung über den Systemdatentyp TCON\_IP\_v4
  - ISO-on-TCP: Verbindungsbeschreibung über den Systemdatentyp TCON\_IP\_RFC
  - ISO: Verbindungsbeschreibung über den Systemdatentyp TCON\_ISONative (nur bei CP1543-1)
- Konfigurierte Verbindungen
  - Angabe einer bestehenden Verbindung in dem Systemdatentyp TCON\_Configured.

Wenn die CPU in den Betriebszustand STOP geht, wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung muss "TRCV\_C" erneut ausgeführt werden.

## Daten über die bestehende Kommunikationsverbindung empfangen

Wenn der Parameter EN\_R auf den Wert "1" gesetzt wird, wird der Datenempfang aktiviert. Die empfangenen Daten werden in einen Empfangsbereich eingetragen. Die Länge des Empfangsbereichs legen Sie abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls  $LEN \neq 0$ ) oder die Längenangabe des Parameters DATA (falls  $LEN = 0$ ) fest. Wenn Sie rein symbolische Werte am Parameter DATA verwenden, muss für den Parameter LEN als Wert "0" verwendet werden.

Empfangsmodi von TRCV\_C:

- **TCP (Ad-hoc-Modus)**

Den Ad-hoc-Modus gibt es nur bei der Protokollvariante TCP. Den Ad-hoc-Modus verwenden Sie, um mit der Anweisung "TRCV\_C" Daten mit dynamischen Längen zu empfangen.

Den Ad-hoc-Modus stellen Sie ein, indem Sie dem Parameter ADHOC den Wert "1" zuweisen. Bei Verwendung des Ad-hoc-Modus können für Datenbausteine mit Standardzugriff alle Datentypen verwendet werden. Für Datenbausteine mit optimiertem Zugriff können als Datentypen nur ARRAY of BYTE oder Datentypen mit einer Länge von 8 Bit verwendet werden (z. B. CHAR, USINT, SINT, etc.). Die tatsächlich empfangene Datenlänge wird am Parameter RCVD\_LEN ausgegeben.

- **TCP (Datenempfang mit angegebener Länge)**

Für einen Datenempfang mit angegebener Länge weisen Sie dem Parameter ADHOC den Wert "0" zu. Ist der Ad-hoc-Modus deaktiviert, wird der Datenempfang erst abgeschlossen, wenn die am Parameter LEN angegebene Länge der Daten vollständig empfangen wurde. Erst danach sind die Daten im Empfangsbereich (Parameter DATA) verfügbar. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD\_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

- **ISO on TCP (Nachrichtenorientiert Datenübertragung)**

Bei der Protokollvariante ISO on TCP werden über eine Verbindung abgeschlossene Nachrichtenblöcke gesendet, die vom Empfänger auch als solche erkannt werden. Der Empfangsbereich wird von den Parametern LEN und DATA definiert. Ist der Empfangspuffer (Parameter DATA) zu klein für die gesendeten Daten, meldet "TRCV\_C" einen Fehler. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD\_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

Die folgende Tabelle zeigt, wie die empfangenen Daten in den Empfangsbereich eingetragen werden.

| Protokollvariante                                       | Verfügbarkeit der Daten im Empfangsbereich   | Parameter connection_type der Verbindungsbeschreibung | Parameter LEN                       |
|---|--|---|-------------------------------------|
| TCP<br>(Ad-hoc-Modus)                                   | Die Daten sind sofort verfügbar.   | Hexadezimalwert: B#16#11<br>Integerwert: 17           | 1 bis maximale Länge (CPU-abhängig) |
| TCP<br>(Datenempfang mit angegebener Länge)             | Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde. | Hexadezimalwert: B#16#11<br>Integerwert: 17           | 1 bis 8192                          |
| ISO on TCP<br>(Nachrichtenorientierte Datenübertragung) | Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde. | Hexadezimalwert: B#16#12<br>Integerwert: 18           | 1 bis 8192                          |

### Die Kommunikationsverbindung abbauen

Die Kommunikationsverbindung wird abgebaut, wenn der Parameter CONT auf den Wert "0" gesetzt wird, auch wenn eine laufende Datenübertragung noch nicht abgeschlossen ist. Dies gilt jedoch nicht, wenn Sie eine konfigurierte Verbindung verwenden.

Durch Setzen des Parameters COM\_RST auf "1" kann die Verbindung jederzeit zurückgesetzt werden. Die bestehende Kommunikationsverbindung wird dadurch abgebaut und eine neue Verbindung aufgebaut. Wenn zu diesem Zeitpunkt Daten übertragen werden, kann es zum Datenverlust kommen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV\_C":

| Parameter | Deklaration | Datentyp | Speicherbereich                    | Beschreibung   |
|-----------|-------------|----------|------------------------------------|--|
| EN_R      | Input       | BOOL     | E, A, M, D, L, T, Z oder Konstante | Empfangsfreigabe   |
| CONT      | Input       | BOOL     | E, A, M, D, L                      | Steuert die Kommunikationsverbindung: <ul style="list-style-type: none"> <li>• 0: Kommunikationsverbindung nach Empfangen der Daten automatisch trennen.</li> <li>• 1: Kommunikationsverbindung aufbauen und nach Empfangen der Daten halten.</li> </ul> |
| LEN       | Input       | UDINT    | E, A, M, D, L oder Konstante       | Maximale Länge der zu empfangenen Daten. Wenn Sie einen Empfangsbereich mit optimiertem Zugriff am Parameter DATA verwenden, muss am Parameter LEN der Wert "0" verwendet werden.  |
| ADHOC     | Input       | BOOL     | E, A, M, D, L oder Konstante       | Optionaler Parameter (ausgeblendet)<br>Ad-hoc Modus für die Protokollvariante TCP verwenden.   |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| CONNECT   | InOut       | VARIANT  | D               | <p>Zeiger auf die Verbindungsbeschreibung</p> <ul style="list-style-type: none"> <li>• Programmierte Verbindung: <ul style="list-style-type: none"> <li>– Bei TCP oder UDP verwenden Sie die Struktur TCON_IP_v4<br/>Zur Beschreibung siehe: Auto-Hotspot</li> <li>– Bei ISO-on-TCP verwenden Sie die Struktur TCON_IP_RFC<br/>Zur Beschreibung siehe: Auto-Hotspot</li> <li>– Bei ISO verwenden Sie die Struktur TCON_ISOnative (nur bei CP1543-1)<br/>Zur Beschreibung siehe Anweisung "TCON (Seite 2971)": "Aufbau der Verbindungsbeschreibung nach TCON_ISOnative"</li> </ul> </li> <li>• Konfigurierte Verbindung: <ul style="list-style-type: none"> <li>– Für bestehende Verbindungen verwenden Sie den Systemdatentyp TCON_Configured. Zur Beschreibung siehe unten "Systemdatentyp für konfigurierte Verbindungen".</li> </ul> </li> </ul> |
| DATA      | InOut       | VARIANT  | E, A, M, D, L   | Zeiger auf den Empfangsbereich.   |
| ADDR      | InOut       | VARIANT  | D               | <p>Optionaler Parameter (ausgeblendet)</p> <p>Zeiger auf die Adresse des Senders bei dem Verbindungstyp UDP.</p>  |
| COM_RST   | InOut       | BOOL     | E, A, M, D, L   | <p>Optionaler Parameter (ausgeblendet)</p> <p>Bewirkt ein Zurücksetzen der Verbindung:</p> <ul style="list-style-type: none"> <li>• 0: Irrelevant</li> <li>• 1: Die vorhandene Verbindung wird zurückgesetzt.</li> </ul> <p>Der Parameter COM_RST wird nach Auswertung durch die Anweisung "TRCV_C" rückgesetzt und sollte daher nicht statisch verschaltet werden.</p>   |
| DONE      | Output      | BOOL     | E, A, M, D, L   | <p>Statusparameter mit folgenden Werten:</p> <ul style="list-style-type: none"> <li>• 0: Empfang noch nicht gestartet oder noch in Bearbeitung.</li> <li>• 1: Empfang fehlerfrei ausgeführt. Dieser Zustand wird nur für einen Zyklus angezeigt.</li> </ul> <p>Der Ausgangsparameter DONE wird gesetzt, wenn ein Zwischenschritt bei der Bearbeitung erfolgreich abgeschlossen wurde (Verbindungsaufbau, Empfang, Verbindungsabbau) und wenn die Ausführung von "TRCV_C" erfolgreich abgeschlossen ist.</p>   |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | <p>Statusparameter mit folgenden Werten:</p> <ul style="list-style-type: none"> <li>• 0: Empfang noch nicht gestartet oder bereits beendet.</li> <li>• 1: Empfang noch nicht beendet. Ein neuer Sendeauftrag kann nicht gestartet werden.</li> </ul>  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten beim Verbindungsaufbau, beim Empfang der Daten oder beim Verbindungsabbau.</li> </ul> Der Ausgangsparameter ERROR kann auf Grund eines Fehlers bei der Anweisung "TRCV_C" oder den intern verwendeten Kommunikations-Anweisungen gesetzt werden. |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Status der Anweisung  |
| RCVD_LEN  | Output      | UDINT    | E, A, M, D, L   | Tatsächlich empfangene Datenmenge in Byte   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter EN\_R, CONT und COM\_RST

Der Parameter CONT steuert den Verbindungsaufbau der Anweisung "TRCV\_C" unabhängig von dem Parameter EN\_R. Das Verhalten des Parameters CONT hängt zum Teil davon ab, ob eine programmierte oder konfigurierte Verbindung verwendet wird:

- Bei CONT = "0": Es werden keine Daten empfangen (unabhängig davon, ob eine programmierte oder konfigurierte Verbindung verwendet wird).
- Beim Wechsel von CONT = "0" auf "1":
  - Bei einer programmierten Verbindung wird diese mit "TCON" aufgebaut.
  - Bei einer konfigurierten Verbindung wird diese mit "T\_DIAG" überprüft.
- Bei CONT = "1":
  - Solange keine Daten empfangen werden (EN\_R = "0"), wird die Verbindung mit "T\_DIAG" überprüft.
  - Wird durch die intern verwendeten Kommunikations-Anweisungen gemeldet, dass kein Verbindungsendpunkt vorhanden ist, wird die Verbindung über "TCON" automatisch wieder aufgebaut.
- Beim Wechsel von CONT = "1" auf "0":
  - Bei einer programmierten Verbindung wird diese mit "TDISCON" abgebaut.
  - Bei einer konfigurierten Verbindung wird diese mit "T\_RESET" zurückgesetzt.

Der Parameter COM\_RST setzt die Verbindung beim Wechsel von "0" auf "1" zurück:

- Ist eine Verbindung aufgebaut, wird diese über "T\_RESET" zurückgesetzt (unabhängig davon, ob eine programmierte oder konfigurierte Verbindung verwendet wird).
- Ist keine Verbindung aufgebaut, hat das Setzen des Parameters keine Auswirkungen.

Die Parameter EN\_R und COM\_RST sind nur aktiv, wenn CONT auf "1" gesetzt wurde. Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern EN\_R, CONT und COM\_RST:

| EN_R       | CONT  | COM_RST    | Status der Anweisung   | Beschreibung  |
|------------|-------|------------|------------------------|---|
| irrelevant | 0     | irrelevant | Noch nicht ausgeführt  | Kein Auftrag aktiv (STATUS = 7000).   |
| irrelevant | 0     | irrelevant | Initialisierung        | Verbindung wird abgebaut. Die Anweisung wird zurückgesetzt.                     |
| irrelevant | 0 > 1 | irrelevant | Verbindungsaufbau      | Verbindung wird aufgebaut. Es werden noch keine Daten übertragen.               |
| 0          | 1     | 0          | Verbindung hergestellt | Die Verbindung ist hergestellt und wird mit der Anweisung "T_DIAG" überwacht.   |
| irrelevant | 1     | 0 > 1      | Verbindung hergestellt | Die Verbindung wird durch "T_RESET" kurzfristig unterbrochen und zurückgesetzt. |
| 0 > 1      | 1     | 0          | Verbindung hergestellt | Anweisung beginnt zu empfangen.   |
| irrelevant | 1     | 0 > 1      | Daten werden empfangen | Datenübertragung wird unterbrochen. Die Verbindung wird zurückgesetzt.          |

### Systemdatentyp für konfigurierte Verbindungen

Verwenden Sie für konfigurierte Verbindungen am Parameter CONNECT die folgende Struktur zur Verbindungsbeschreibung nach TCON\_Configured:

| Byte    | Parameter      | Datentyp | Startwert | Beschreibung   |
|---------|----------------|----------|-----------|--|
| 0 ... 1 | InterfaceID    | HW_ANY   | -         | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).  |
| 2 ... 3 | ID             | CONN_OUC | -         | Referenz auf die Verbindung (Wertebereich: 1 bis 4095).<br>Geben Sie die Verbindungs-ID der bestehenden Verbindung an. |
| 4       | ConnectionType | BYTE     | -         | Verbindungstyp<br>Wählen Sie 254 (dezimal) für eine konfigurierte Verbindung.  |

### Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TRCV\_C" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| DONE | BUSY | ERROR | Beschreibung  |
|------|------|-------|---|
| 0    | 0    | 0     | Die Anweisung wurde noch nicht ausgeführt (keine steigende Flanke am Parameter EN_R).         |
| 0    | 1    | 0     | Die Anweisung wird ausgeführt und ruft die intern verwendeten Kommunikations-Anweisungen auf. |

| DONE | BUSY | ERROR | Beschreibung  |
|------|------|-------|---|
| 1    | 0    | 0     | Der Empfang wurde erfolgreich durchgeführt. Am Parameter STATUS wird "0000" ausgegeben. DONE = "1" wird nur für einen Zyklus angezeigt.   |
| 0    | 0    | 1     | Die Ausführung der Anweisung oder ein Zwischenschritt bei der Bearbeitung wurde mit einem Fehler beendet. Kommt es durch eine intern verwendete Kommunikations-Anweisung zu einem Folgefehler, wird der Fehler angezeigt, welcher bei der Bearbeitung zuerst aufgetreten ist. Dieser Zustand wird nur für einen Zyklus angezeigt. |

### Parameter ERROR und STATUS

| ERROR | STATUS (W#16#...) | Beschreibung  |
|-------|-------------------|---|
| 0     | 0000              | Empfangsauftrag fehlerfrei ausgeführt.  |
| 0     | 0001              | Der Verbindungsaufbau ist abgeschlossen.  |
| 0     | 0003              | Der Verbindungsabbau ist abgeschlossen.   |
| 0     | 7000              | Keine Auftragsbearbeitung aktiv.  |
| 0     | 7001              | <ul style="list-style-type: none"> <li>• Empfangsauftragsbearbeitung starten</li> <li>• Kommunikationsverbindung aufbauen</li> <li>• Auf Verbindungspartner warten</li> </ul>   |
| 0     | 7002              | Daten werden empfangen.   |
| 0     | 7003              | Kommunikationsverbindung wird beendet.  |
| 0     | 7004              | <ul style="list-style-type: none"> <li>• Kommunikationsverbindung aufgebaut und überwacht.</li> <li>• Keine Empfangsauftragsbearbeitung aktiv.</li> </ul>   |
| 0     | 7005              | Kommunikationsverbindung wird rückgesetzt.  |
| 0     | 7006              | Daten werden gerade empfangen.  |
| 1     | 8085              | <ul style="list-style-type: none"> <li>• Parameter LEN ist größer als der größte zulässige Wert.</li> <li>• Der Wert am Parameter LEN oder DATA wurde nach dem ersten Aufruf geändert.</li> </ul>   |
| 1     | 8086              | Der Parameter ID liegt außerhalb des zulässigen Wertebereichs.  |
| 1     | 8087              | Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich   |
| 1     | 8088              | Der Wert am Parameter LEN passt nicht zu dem am Parameter DATA angegebenen Empfangsbereich.   |
| 1     | 8089              | Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.  |
| 1     | 8091              | Maximale Schachtelungstiefe überschritten.  |
| 1     | 809A              | Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.   |
| 1     | 809B              | InterfaceID ist ungültig. Sie ist entweder Null, oder sie zeigt nicht auf eine lokale CPU-Schnittstelle oder einen CP.  |
| 1     | 80A1              | <ul style="list-style-type: none"> <li>• Verbindung oder Port wird vom Anwender bereits verwendet.</li> <li>• Kommunikationsfehler:                             <ul style="list-style-type: none"> <li>– Die angegebene Verbindung wurde noch nicht aufgebaut.</li> <li>– Die angegebene Verbindung wird gerade beendet. Übertragung über diese Verbindung ist nicht möglich.</li> <li>– Die Schnittstelle wird neu initialisiert.</li> </ul> </li> </ul> |
| 1     | 80A3              | <ul style="list-style-type: none"> <li>• Es wird versucht, eine vorhandene Verbindung erneut aufzubauen.</li> <li>• Es wird versucht, eine nicht vorhandene Verbindung zu beenden.</li> </ul>   |



| ERROR | STATUS<br>(W#16#...) | Beschreibung   |
|-------|----------------------|--|
| 1     | 80A4                 | IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, oder sie passt zur IP-Adresse des lokalen Partners.  |
| 1     | 80A7                 | Kommunikationsfehler: Sie haben die Anweisung mit COM_RST = 1 aufgerufen, bevor der Sendeauftrag beendet war.  |
| 1     | 80B4                 | Bei Verwendung der Protokollvariante ISO on TCP (connection_type = B#16#12) für den passiven Aufbau einer Verbindung (active_est = FALSE) wurde eine oder beide der folgenden Bedingungen verletzt: "local_tsap_id_len >= B#16#02" und/oder "local_tsap_id[1] = B#16#E0".  |
| 1     | 80B5                 | Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.   |
| 1     | 80B6                 | Parametrierfehler im Parameter connection_type des Datenbausteins für Verbindungsbeschreibung.   |
| 1     | 80B7                 | Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.  |
| 1     | 80C3                 | <ul style="list-style-type: none"> <li>• Alle Verbindungsressourcen sind belegt.</li> <li>• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.</li> </ul>   |
| 1     | 80C4                 | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>• Die Verbindung kann derzeit nicht aufgebaut werden.</li> <li>• Die Schnittstelle empfängt gerade neue Parameter oder die Verbindung wird gerade aufgebaut.</li> <li>• Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON" entfernt.</li> <li>• Die benutzte Verbindung wird gerade durch einen Aufruf mit COM_RST = 1 beendet</li> </ul> |
| 1     | 8722                 | Fehler im Parameter CONNECT: Ungültiger Quellbereich (der Bereich ist im Datenbaustein nicht deklariert).  |
| 1     | 873A                 | Fehler im Parameter CONNECT: Zugriff auf die Verbindungsbeschreibung nicht möglich (kein Zugriff auf den Datenbaustein).   |
| 1     | 877F                 | Fehler im Parameter CONNECT: Interner Fehler   |
| 1     | 8922                 | Parameter DATA: Zielbereich ungültig, Bereich im DB nicht vorhanden.   |
| 1     | 8924                 | Parameter DATA: Bereichsfehler im VARIANT-Zeiger.  |
| 1     | 8932                 | Parameter DATA: DB-Nummer ist zu groß.   |
| 1     | 893A                 | Parameter CONNECT: Zugriff auf die angegebenen Verbindungsdaten nicht möglich (z.B. weil der DB nicht vorhanden ist).  |
| 1     | 897F                 | Parameter DATA: Interner Fehler, z. B. unzulässige VARIANT-Referenz.   |
| 1     | 8A3A                 | Parameter DATA: Kein Zugriff auf den Datenbereich, da z.B. der Datenbaustein nicht existiert.  |

### Hinweis

#### Fehlermeldungen der Anweisungen "TCON", "TRCV" und "TDISCON"

Die Anweisung "TRV\_C" verwendet intern die Anweisungen "TCON (Seite 2971)", "TRCV (Seite 2984)" und "TDISCON (Seite 2976)". Die Fehlermeldungen dieser Anweisungen sind in den jeweiligen Beschreibungen enthalten.

### Siehe auch

TSEND\_C: Daten über Ethernet senden (Seite 2935)

## TMAIL\_C: E-Mail übertragen

### Beschreibung TMAIL\_C

#### Beschreibung

Über die Anweisung "TMAIL\_C" versenden Sie eine E-Mail über die Ethernet-Schnittstelle der S7-1500 oder S7-1200 V4.0, eines Kommunikationsmoduls (CM) oder eines Kommunikationsprozessors (CP).

Voraussetzung zur Verwendung der Anweisung ist, dass die Hardware bereits konfiguriert ist und die Infrastruktur des Netzwerks eine Kommunikationsverbindung zu dem Mailserver zulässt.

Die Inhalte der E-Mail sowie die Verbindungsdaten definieren Sie über die folgenden Parameter:

- Die Empfängeradressen definieren Sie über die Parameter TO\_S und CC.
- Den Inhalt der E-Mail definieren über die Parameter SUBJECT und TEXT.
- Einen Anhang können Sie über VARIANT-Zeiger an den Parametern ATTACHMENT und ATTACHMENT\_NAME definieren.
- Die Definition der Verbindungsdaten sowie die Adressierung und Authentisierung für den Mailserver erfolgt über einen der Systemdatentypen Tmail\_v4, Tmail\_v6 oder Tmail\_FQDN am Parameter MAIL\_ADDR\_PARAM.
  - Bei Verwendung der Schnittstelle der S7-1500 CPU verwenden Sie ausschließlich den Systemdatentyp Tmail\_v4. Der Versand der Mail ist in diesem Fall nur über SMTP möglich.
  - Bei Verwendung der Schnittstelle eines CMs/CPs können Sie alle Systemdatentypen verwenden. Der Versand der Mail ist auch über SMTPS möglich.
- Sie starten das Versenden einer E-Mail mit einem Flankenwechsel von "0" auf "1" am Parameter REQ.
- Über die Ausgangsparameter "BUSY", "DONE", "ERROR" und "STATUS" wird der Zustand des Auftrags angezeigt.

Das direkte Versenden einer SMS über die Anweisung "TMAIL\_C" ist nicht möglich. Ob eine Weiterleitung der E-Mail als SMS durch den Mailserver erfolgen kann, ist abhängig von dem Telekommunikationsanbieter.

#### Arbeitsweise der Anweisung

Die Anweisung "TMAIL\_C" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Beim Aufruf der Anweisung "TMAIL\_C" ist die Angabe einer Instanz zwingend erforderlich.

In den folgenden Fällen wird die Verbindung zum Mailserver abgebrochen:

- Beim Wechsel der CPU in den Betriebszustand STOP, während "TMAIL\_C" aktiv ist.
- Wenn am Industrial Ethernet-Bus Kommunikationsprobleme auftreten.

In diesen Fällen wird das Versenden der E-Mail unterbrochen und sie erreicht ihre Empfänger nicht. Nach erfolgreicher Ausführung der Anweisung und dem Versenden der Mail wird die Verbindung ebenfalls abgebaut.

#### **ACHTUNG**

##### **Ändern von Anwenderprogrammen**

Sie dürfen die Teile Ihres Anwenderprogramms, die sich unmittelbar auf die Aufrufe von "TMAIL\_C" auswirken, nur ändern:

- Wenn die CPU im Betriebszustand "STOP" ist.
- Wenn keine Mail gesendet wird (REQ = 0 und BUSY = 0).

Dies bezieht sich insbesondere auf das Löschen und Austauschen von Programmbausteinen, die Aufrufe von "TMAIL\_C" oder Aufrufe der Instanz von "TMAIL\_C" enthalten.

Bei Nichtbeachtung können Verbindungsressourcen belegt bleiben. Das Automatisierungssystem kann in einen nicht definierten Zustand bei den TCP/IP-Kommunikationsfunktionen über Industrial Ethernet geraten.

Nach dem Übertragen der Änderungen müssen Sie einen Neustart (Warmstart) oder Kaltstart bei der CPU durchführen.

## **Datenkonsistenz**

Die Parameter TO\_S, CC, SUBJECT, TEXT, ATTACHMENT und MAIL\_ADDR\_PARAM werden während des Betriebs der Anweisung "TMAIL\_C" von diesem übernommen und dürfen somit erst geändert werden, wenn der Auftrag beendet wurde (BUSY = 0).

## **SMTP-Authentisierung**

Unter Authentisierung versteht man ein Verfahren zur Sicherstellung einer Identität z. B. durch eine Passwortabfrage.

Die Anweisung "TMAIL\_C" unterstützt bei Verwendung der Schnittstelle der S7-1500 CPU das SMTP-Authentisierungsverfahren AUTH-LOGIN, das von den meisten Mailservern gefordert wird. Informationen über die Authentisierungsverfahren Ihres Mail-Servers entnehmen Sie bitte dem Handbuch des Mail-Servers oder der Website Ihres Internet Service Providers.

- Um das Authentisierungsverfahren AUTH-LOGIN zu nutzen, benötigt die Anweisung "TMAIL\_C" den Benutzernamen, mit dem er sich beim Mailserver anmelden kann. Dieser Benutzername entspricht dem Nutzernamen mit dem Sie ein Mailkonto auf Ihrem Mailserver eingerichtet haben. Er wird über den Parameter UserName der Struktur am Parameter MAIL\_ADDR\_PARAM übergeben.  
Wird in der Struktur am Parameter MAIL\_ADDR\_PARAM kein Benutzername angegeben, so wird das Authentisierungsverfahren AUTH-LOGIN auch nicht verwendet. Die E-Mail wird dann ohne Authentisierung versendet.
- Des Weiteren benötigt die Anweisung "TMAIL\_C" zur Anmeldung das dazugehörige Passwort. Dieses Passwort entspricht dem Passwort, das Sie beim Einrichten Ihres Mailkontos vergeben haben. Es wird über den Parameter PassWord der Struktur am Parameter MAIL\_ADDR\_PARAM übergeben.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "TMAIL\_C":

| Parameter                       | Deklaration | Datentyp | Speicherbereich                    | Beschreibung   |
|---------------------------------|-------------|----------|------------------------------------|--|
| REQ                             | Input       | BOOL     | E, A, M, D, L, T, Z oder Konstante | Steuerparameter REQUEST: Aktiviert das Versenden einer E-Mail bei steigender Flanke.   |
| TO_S<br>(Seite 2959)            | Input       | STRING   | D                                  | Empfängeradressen<br>STRING mit einer maximalen Länge von 180 Zeichen (Byte).<br>Zur Schreibweise der Mailadresse siehe Beispiel in der Beschreibung des Parameters.   |
| CC<br>(Seite 2959)              | Input       | STRING   | D                                  | CC-Empfängeradressen (optional)<br>STRING mit einer maximalen Länge von 180 Zeichen (Byte).<br>Gleiche Schreibweise der Mailadresse wie bei dem Parameter TO_S. Wird hier ein Leerstring zugewiesen, wird die E-Mail nicht an einen CC-Empfänger gesendet.   |
| SUBJECT                         | Input       | STRING   | D                                  | Betreff der E-Mail<br>STRING mit einer maximalen Länge von 180 Zeichen (Byte).   |
| TEXT                            | Input       | STRING   | D                                  | Text der E-Mail (optional)<br>STRING mit einer maximalen Länge von 180 Zeichen (Byte). Wird an diesem Parameter ein Leerstring zugewiesen, wird die E-Mail ohne Text versendet.  |
| ATTACHMENT                      | Input       | VARIANT  | D                                  | Anhang der E-Mail (optional)<br>Referenz auf ein Byte-/Wort-/Doppelwort-Feld (ArrayOfByte, ArrayOfWord oder ArrayOfDWord) mit einer maximalen Länge von 64 kBytes. Wird kein Wert zugewiesen, wird die E-Mail ohne Anhang versendet.   |
| ATTACHMENT_NAME                 | Input       | VARIANT  | D                                  | Name des Anhangs der E-Mail (optional)<br>Referenz auf eine Zeichenkette mit einer maximalen Länge von 50 Zeichen (Byte) zur Definition des Dateinamens des Anhangs. Wird an diesem Parameter ein Leerstring zugewiesen, wird der Anhang der E-Mail mit dem Dateinamen "attachment.bin" versendet. |
| MAIL_ADDR_PARAM<br>(Seite 2960) | Input       | VARIANT  | D                                  | Parameter der Verbindung und Adresse des E-Mail-Servers<br>Verwenden Sie zur Definition der Verbindungsparameter die Struktur Tmail_v4, Tmail_v6 oder Tmail_FQDN (siehe Beschreibung des Parameters).  |

| Parameter              | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|------------------------|-------------|----------|-----------------|--|
| DONE<br>(Seite 2963)   | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter <ul style="list-style-type: none"> <li>• DONE = 0: Auftrag ist noch nicht gestartet oder wird noch ausgeführt.</li> <li>• DONE = 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>  |
| BUSY<br>(Seite 2963)   | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter <ul style="list-style-type: none"> <li>• BUSY = 0: Die Bearbeitung von "TMAIL_C" wurde beendet.</li> <li>• BUSY = 1: Das Senden der E-Mail ist noch nicht abgeschlossen.</li> </ul>  |
| ERROR<br>(Seite 2963)  | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter <ul style="list-style-type: none"> <li>• ERROR = 0: Kein Fehler aufgetreten.</li> <li>• ERROR = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefern detaillierte Auskunft über die Art des Fehlers.</li> </ul> |
| STATUS<br>(Seite 2964) | Output      | WORD     | E, A, M, D, L   | Zustandsparameter<br>Rückgabewert bzw. Fehlerinformation der Anweisung "TMAIL_C" (siehe Beschreibung des Parameters).  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Hinweis

#### Optionale Parameter

Die optionalen Parameter CC, TEXT und ATTACHMENT werden nur mit der E-Mail versendet, wenn die entsprechenden Parameter einen String der Länge > 0 enthalten.

## Parameter TO\_S und CC

### Beschreibung

Die Parameter TO\_S und CC sind Strings mit z. B. folgendem Inhalt:

- <wenna@mydomain.com>, <ruby@mydomain.com>
- <admin@mydomain.com>, <judy@mydomain.com>

Folgende Regeln sind bei der Eingabe der Parameter zu beachten:

- Vor jeder Adresse muss ein Leerzeichen und eine spitze Klammer auf "<" eingegeben werden.
- Nach jeder Adresse muss eine spitze Klammer zu ">" eingegeben werden.
- Zwischen den Adressen in TO und CC muss ein Komma eingegeben werden.

Aus Gründen der Laufzeit und des Speicherplatzes führt die Anweisung "TMAIL\_C" keine Syntaxprüfung der Parameter TO\_S und CC durch.

## Parameter MAIL\_ADDR\_PARAM

### Beschreibung

Am Parameter MAIL\_ADDR\_PARAM definieren Sie in der Struktur Tmail\_v4, Tmail\_v6 oder Tmail\_FQDN, über welche Verbindung die E-Mail gesendet werden soll und hinterlegen die Adresse des E-Mail-Servers sowie die Login-Daten.

Je nachdem, in welchem Format Sie den E-Mail-Server adressieren möchten verwenden Sie am Parameter MAIL\_ADDR\_PARAM die Struktur:

- Tmail\_v4: Adressierung über die IP-Adresse nach IPv4.
- Tmail\_v6: Adressierung über die IP-Adresse nach IPv6.
- Tmail\_FQDN: Adressierung über voll qualifizierten Domain-Namen (FQDN).

Welche Struktur Sie verwenden können, ist abhängig von der am Parameter Interfaceld adressierten Schnittstelle:

- Zur Verwendung der Anweisung "TMAIL\_C" mit der internen Schnittstelle muss am Parameter MAIL\_ADDR\_PARAM die Struktur Tmail\_v4 verwendet werden.
- Bei der Verwendung eines Kommunikationsprozessors (CP) oder Kommunikationsmoduls (CM) können Sie alle drei Adressierungsmöglichkeiten (IPv4, IPv6 und FQDN) verwendet werden.

Tabelle 9-125 Tmail\_v4: Adressieren des Mailservers über die IP-Adresse nach IPv4

| Parameter         | Datentyp | Beschreibung   |
|-------------------|----------|--|
| Tmail_v4          | Struct   |  |
| Interfaceld       | LADDR    | Hardware-Kennung der Schnittstelle   |
| ID                | CONN_OUC | Verbindungs-ID   |
| ConnectionType    | BYTE     | Verbindungstyp. Wählen Sie für IPv4 als Verbindungstyp 16#20.  |
| ActiveEstablished | BOOL     | Zustandsbit. Wird auf "1" gesetzt, wenn die Verbindung hergestellt ist.  |
| CertIndex         | BYTE     | <ul style="list-style-type: none"> <li>• =0: Verwendung von SMTP (<b>S</b>imple <b>M</b>ail <b>T</b>ransfer <b>P</b>rotocol). Beim Versenden der Mail über die Schnittstelle einer S7-1500 CPU muss SMTP verwendet werden.</li> <li>• ≠0: Verwendung von SMTPS zur Absicherung der Verbindung vor dem Verbindungsaufbau (bei CPs/CMs). Über den Parameter CertIndex geben Sie das zu verwendende Zertifikat an (siehe "Projektnavigation" &gt; "Globale Security-Einstellungen" &gt; "Zertifikatsmanager").</li> </ul> |

| Parameter               | Datentyp    | Beschreibung   |
|-------------------------|-------------|--|
| WatchDogTime            | TIME        | Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs.<br><br>Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH_DOG_TIME berücksichtigen.<br><br>Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut. |
| MailServerAddress       | IP_v4       | IP Adresse des Mailservers. Nach IPv4 im Format XXX.XXX.XXX.XXX (dezimal).<br>Beispiel: 192.142.131.237.   |
| UserName                | STRING[254] | Loginname des Mailservers  |
| PassWord                | STRING[254] | Passwort des Mailservers   |
| From                    | EMAIL_ADDR  | Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymailserver.com".  |
| LocalPartPlusAtSign     | STRING[64]  | Lokaler Teil der Absenderadresse inklusive @-Zeichen.<br>Beispiel: "myname@".  |
| FullQualifiedDomainName | STRING[254] | Fully Qualified Domain Name ( kurz: FQDN) des Mailservers. Beispiel: "mymailserver.com".   |

Tabelle 9-126 Tmail\_v6: Adressieren des Mailservers über die IP-Adresse nach IPv6

| Parameter         | Datentyp | Beschreibung   |
|-------------------|----------|--|
| Tmail_v6          | Struct   |  |
| Interfaceld       | LADDR    | Hardware-Kennung der Schnittstelle   |
| ID                | CONN_OUC | Verbindungs-ID   |
| ConnectionType    | BYTE     | Verbindungstyp. Wählen Sie für IPv6 als Verbindungstyp 16#21.  |
| ActiveEstablished | BOOL     | Zustandsbit. Wird auf "1" gesetzt, wenn die Verbindung hergestellt ist.  |
| CertIndex         | BYTE     | <ul style="list-style-type: none"> <li>• =0: Verwendung von SMTP (Simple Mail Transfer Protocol). Beim Versenden der Mail über die Schnittstelle einer S7-1500 CPU muss SMTP verwendet werden.</li> <li>• ≠0: Verwendung von SMTPS zur Absicherung der Verbindung vor dem Verbindungsaufbau (bei CPs/CMs). Über den Parameter CertIndex geben Sie das zu verwendende Zertifikat an (siehe "Projektnavigation" &gt; "Globale Security-Einstellungen" &gt; "Zertifikatsmanager").</li> </ul> |

| Parameter               | Datentyp    | Beschreibung   |
|-------------------------|-------------|--|
| WatchDogTime            | TIME        | Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs.<br><br>Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH_DOG_TIME berücksichtigen.<br><br>Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut. |
| MailServerAddress       | IP_v6       | IP Adresse des Mailservers nach IPv6 im Format XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX (hexadezimal).<br><br>Die Adresse ist in 8 Blöcke von jeweils 2 Byte unterteilt (insgesamt 16 Byte).<br><br>Beispiel: 2001:db8:1f11:08d3:290:27ff:0370:2093   |
| UserName                | STRING[254] | Loginname des Mailservers  |
| PassWord                | STRING[254] | Passwort des Mailservers   |
| From                    | EMAIL_ADDR  | Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymailserver.com".  |
| LocalPartPlusAtSign     | STRING[64]  | Lokaler Teil der Absenderadresse inklusive @-Zeichen.<br>Beispiel: "myname@".  |
| FullQualifiedDomainName | STRING[254] | Fully Qualified Domain Name (kurz: FQDN) des Mailservers.<br>Beispiel: "mymailserver.com".   |

Tabelle 9-127 Tmail\_FQDN: Adressieren des Mailservers über den FQDN

| Parameter         | Datentyp | Beschreibung   |
|-------------------|----------|--|
| Tmail_v6          | Struct   |  |
| Tmail_FQDN        | LADDR    | Hardware-Kennung der Schnittstelle   |
| ID                | CONN_OUC | Verbindungs-ID   |
| ConnectionType    | BYTE     | Verbindungstyp. Wählen Sie für FQDN als Verbindungstyp 16#22.  |
| ActiveEstablished | BOOL     | Zustandsbit. Wird auf "1" gesetzt, wenn die Verbindung hergestellt ist.  |
| CertIndex         | BYTE     | <ul style="list-style-type: none"> <li>• =0: Verwendung von SMTP (Simple Mail Transfer Protocol). Beim Versenden der Mail über die Schnittstelle einer S7-1500 CPU muss SMTP verwendet werden.</li> <li>• ≠0: Verwendung von SMTPS zur Absicherung der Verbindung vor dem Verbindungsaufbau (bei CPs/CMs). Über den Parameter CertIndex geben Sie das zu verwendende Zertifikat an (siehe "Projektnavigation" &gt; "Globale Security-Einstellungen" &gt; "Zertifikatsmanager").</li> </ul> |



| Parameter               | Datentyp    | Beschreibung   |
|-------------------------|-------------|--|
| WatchDogTime            | TIME        | Zeitüberwachung der Ausführung. Definieren Sie über diesen Parameter die maximale Ausführungsdauer des Sendevorgangs.<br><br>Hinweis: Bei einer langsamen Verbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH_DOG_TIME berücksichtigen.<br><br>Nach Ablauf der vorgegebenen Zeit wird die Verbindung abgebaut. |
| MailServerAddress       | STRING[254] | FQDN (Fully Qualified Domain Name) des Mailservers. Die Adressierung des Mailservers erfolgt über den vollständigen Namen der Domain.<br><br>Beispiel: "www.mymailserver.com".   |
| UserName                | STRING[254] | Loginname des Mailservers  |
| PassWord                | STRING[254] | Passwort des Mailservers   |
| From                    | Struct      | Absenderadresse der E-Mail, die über die folgenden beiden STRING-Parameter definiert wird. Zum Beispiel: "myname@mymailserver.com".  |
| LocalPartPlusAtSign     | STRING[64]  | Lokaler Teil der Absenderadresse inklusive @-Zeichen.<br>Beispiel: "myname@".  |
| FullQualifiedDomainName | STRING[254] | Fully Qualified Domain Name ( kurz: FQDN) des Mailservers.<br>Beispiel: "mymailserver.com".  |

## Parameter DONE, BUSY und ERROR

### Beschreibung

Die Ausgangsparameter DONE, BUSY und ERROR werden jeweils nur einen Zyklus lang angezeigt, wenn der Zustand des Ausgangsparameters BUSY von "1" auf "0" wechselt.

In der folgenden Tabelle ist der Zusammenhang zwischen DONE, BUSY und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich die Anweisung "TMAIL\_C" aktuell befindet bzw. wann das Senden der E-Mail beendet ist.

| DONE | BUSY | ERROR | Beschreibung   |
|------|------|-------|--|
| 0    | 1    | 0     | Der Auftrag ist in Bearbeitung.  |
| 1    | 0    | 0     | Der Auftrag wurde erfolgreich abgeschlossen.   |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache entnehmen Sie dem Parameter STATUS (Seite 2964). |
| 0    | 0    | 0     | Der Anweisung "TMAIL_C" wurde kein (neuer) Auftrag erteilt.  |

## Parameter STATUS

### Beschreibung

Die folgende Tabelle zeigt die Rückgabewerte am Parameter STATUS von "TMAIL\_C":

| Rückgabewert<br>STATUS*<br>(W#16#...): | Erläuterung  | Hinweise   |
|--|--|--|
| 0000                                   | Die Bearbeitung von "TMAIL_C" wurde fehlerfrei abgeschlossen.  | Ein fehlerfreies Beenden von "TMAIL_C" bedeutet nicht, dass die versendete E-Mail ankommt.<br>Eine fehlerhafte Eingabe der Empfängeradressen erzeugt keinen Statusfehler der Anweisung "TMAIL_C". Das Versenden der E-Mail an weitere Empfänger, auch wenn diese richtig eingegeben wurden, ist in diesem Fall nicht sichergestellt. |
| 7001                                   | "TMAIL_C" ist aktiv (BUSY = 1).  | Erstaufruf: Auftrag angestoßen.  |
| 7002                                   | "TMAIL_C" ist aktiv (BUSY = 1).  | Zwischenaufruf: Auftrag ist bereits aktiv.   |
| 8xxx                                   | Die Bearbeitung von "TMAIL_C" wurde mit einem Fehlercode der intern aufgerufenen Kommunikations-Anweisungen abgeschlossen. | Ausführliche Informationen finden Sie in den Beschreibungen des Parameters STATUS der Kommunikations-Anweisungen "TCON (Seite 2971)", "TDISCON (Seite 2976)", "TSEND (Seite 2981)" und "TRCV (Seite 2984)".  |
| 8010                                   | Fehler beim Verbindungsaufbau  | Weitere Informationen zur Auswertung finden Sie im Parameter SFB_STATUS des Instanzdatenbausteins. Die Bedeutung des an SFB_STATUS angezeigten Fehlercodes ist in der Beschreibung des Parameters STATUS der Anweisung "TCON (Seite 2971)" enthalten.  |
| 8011                                   | Fehler beim Senden der Daten   | Weitere Informationen zur Auswertung finden Sie im Parameter SFB_STATUS des Instanzdatenbausteins. Die Bedeutung des an SFB_STATUS angezeigten Fehlercodes ist in der Beschreibung des Parameters STATUS der Anweisung "TSEND (Seite 2981)" enthalten.   |
| 8012                                   | Fehler beim Empfangen der Daten  | Weitere Informationen zur Auswertung finden Sie im Parameter SFB_STATUS des Instanzdatenbausteins. Die Bedeutung des an SFB_STATUS angezeigten Fehlercodes ist in der Beschreibung des Parameters STATUS der Anweisung "TRCV (Seite 2984)" enthalten.  |

| Rückgabewert<br>STATUS*<br>(W#16#...): | Erläuterung   | Hinweise   |
|--|---|--|
| 8013                                   | Fehler beim Verbindungsaufbau   | Weitere Informationen zur Auswertung finden Sie im Parameter SFB_STATUS des Instanzdatenbausteins. Die Bedeutung des an SFB_STATUS angezeigten Fehlercodes ist in der Beschreibung des Parameters STATUS der Anweisungen "TCON (Seite 2971)" und "TDISCON (Seite 2976)" enthalten.   |
| 8014                                   | Aufbau einer Verbindung nicht möglich.  | Möglicherweise haben Sie eine fehlerhafte Mailserver-IP-Adresse (MailServerAddress (Seite 2960)) oder eine zu kleine Zeitspanne (WatchDogTime (Seite 2960)) für den Verbindungsaufbau eingegeben. Es besteht auch die Möglichkeit, dass die CPU keine Verbindung zum Netzwerk hat oder die CPU-Konfiguration fehlerhaft ist. |
| 8015                                   | Falscher Datentyp für MAIL_ADDR_PARAM   | Erlaubte Datentypen sind die Systemdatentypen (Strukturen) Tmail_v4, Tmail_v6 oder TMail_FQDN.   |
| 8016                                   | Falscher Datentyp für den Parameter ATTACHMENT  | Erlaubte Datentypen sind ArrayOfByte, ArrayOfWord und ArrayOfDWord.  |
| 8017                                   | Falsche Datenlänge für den Parameter ATTACHMENT   | Erlaubte Datenlänge beträgt <= 65534 Byte.   |
| 8401                                   | Keine Kanäle verfügbar. Mögliche Ursache: Es besteht bereits eine E-Mail-Verbindung über den CP. Eine zweite Verbindung kann nicht parallel eingerichtet werden.                        | Spezifischer Fehler für CP 1543  |
| 8403                                   | Es konnte keine TCP/IP-Verbindung zum Mailserver aufgebaut werden.  | Spezifischer Fehler für CP 1543  |
| 8405                                   | Server hat Login-Anfrage verweigert.  | Spezifischer Fehler für CP 1543  |
| 8406                                   | Ein interner SSL-Fehler oder ein Problem mit der Struktur des Zertifikats wurde durch den SMTP-Client festgestellt.   | Spezifischer Fehler für CP 1543  |
| 8407                                   | Anfrage zur Verwendung von SSL wurde verweigert.  | Spezifischer Fehler für CP 1543  |
| 8408                                   | Der Client konnte kein Socket zur Erstellung einer TCP/IP-Verbindung zum Mailserver ermitteln.  | Spezifischer Fehler für CP 1543  |
| 8409                                   | Über die Verbindung kann nicht geschrieben werden. Mögliche Ursache: Durch den Kommunikationspartner wurde ein Reset der Verbindung durchgeführt oder die Verbindung wurde abgebrochen. | Spezifischer Fehler für CP 1543  |

| Rückgabewert<br>STATUS*<br>(W#16#...): | Erläuterung   | Hinweise                        |
|--|---|---------------------------------|
| 8410                                   | Über die Verbindung kann nicht gelesen werden. Mögliche Ursache: Durch den Kommunikationspartner wurde ein Reset der Verbindung durchgeführt oder die Verbindung wurde abgebrochen. | Spezifischer Fehler für CP 1543 |
| 8411                                   | Senden der E-Mail fehlgeschlagen. Ursache: Speicherplatz war nicht ausreichend, um den Sendevorgang durchzuführen.  | Spezifischer Fehler für CP 1543 |
| 8412                                   | Konfigurierter DNS-Server konnte den angegebenen Domain-Namen nicht auflösen.   | Spezifischer Fehler für CP 1543 |
| 8413                                   | Aufgrund eines internen Fehlers im DNS-Subsystem konnte der Domain-Name nicht aufgelöst werden.   | Spezifischer Fehler für CP 1543 |
| 8414                                   | Als Domain-Name wurde eine leere Zeichenkette angegeben.  | Spezifischer Fehler für CP 1543 |
| 8415                                   | Ein interner Fehler ist im Curl-Modul aufgetreten. Ausführung wurde gestoppt.   | Spezifischer Fehler für CP 1543 |
| 8416                                   | Ein interner Fehler ist im SMTP-Modul aufgetreten. Ausführung wurde gestoppt.   | Spezifischer Fehler für CP 1543 |
| 8417                                   | Anfrage an SMTP auf bereits verwendetem Kanal oder ungültige Kanal-ID. Ausführung wurde gestoppt.   | Spezifischer Fehler für CP 1543 |
| 8418                                   | Senden der E-Mail wurde abgebrochen. Mögliche Ursachen: Überschreitung der Ausführungszeit (Parameter WatchDogTime) oder Start/Stop-Übergang des CPs.                               | Spezifischer Fehler für CP 1543 |
| 8419                                   | Der Kanal wurde unterbrochen und kann nicht verwendet werden, bevor die Verbindung geschlossen wird.  | Spezifischer Fehler für CP 1543 |
| 8420                                   | Zertifikatskette von dem Server konnte mit dem Root-Zertifikat des CPs nicht verifiziert werden.  | Spezifischer Fehler für CP 1543 |
| 8421                                   | Interner Fehler aufgetreten. Ausführung wurde gestoppt.   | Spezifischer Fehler für CP 1543 |

| Rückgabewert<br>STATUS*<br>(W#16#...):   | Erläuterung   | Hinweise  |
|--|---|---|
| 82xx, 84xx,<br>bzw. 85xx   | Die Fehlermeldung stammt vom Mailserver und entspricht bis auf die "8" der Fehlernummer des SMTP-Protokolls.<br><br>In den nachfolgenden Zeilen sind einige Fehlercodes aufgeführt, die auftreten können. | Weitere Informationen zum SMTP-Fehlercode und weitere Fehlercodes finden Sie im SMTP-Protokoll im Internet bzw. in der Fehlerdokumentation des Mailservers. Darüber hinaus können Sie die letzte, vom Mailserver gesendete Fehlermeldung in Textform in ihrem Instanz-DB im Parameter BUFFER1 einsehen. In dem Instanz-DB finden Sie unter DATEN die zuletzt von der Anweisung "TMAIL_C" versendeten Daten. |
| 8450   | Aktion nicht ausgeführt: Mailbox nicht verfügbar/nicht erreichbar   | Versuchen Sie es später noch einmal.  |
| 8451   | Aktion abgebrochen: Lokaler Fehler in der Verarbeitung  | Versuchen Sie es später noch einmal.  |
| 8500   | Syntax-Fehler: Fehler nicht erkannt. Das schließt auch den Fehler einer zu langen Befehlskette ein. Ursache kann auch sein, dass der E-Mail-Server das Authentisierungsverfahren LOGIN nicht unterstützt. | Überprüfen Sie die Parameter von "TMAIL_C". Versuchen Sie, eine E-Mail ohne Authentisierung zu versenden. Ersetzen Sie dazu den Inhalt des Parameters UserName durch einen leeren String. Wird kein Benutzername angegeben, wird das Authentisierungsverfahren LOGIN auch nicht verwendet.  |
| 8501   | Syntax-Fehler: Falsche Eingabe an einem Parameter   | Mögliche Ursache: Fehlerhafte Adresse an den Parametern TO_S oder CC (siehe auch: Parameter TO_S und CC (Seite 2959)).  |
| 8502   | Befehl unbekannt bzw. nicht implementiert   | Überprüfen Sie Ihre Eingaben, insbesondere den Parameter FROM. Möglicherweise ist dieser unvollständig und Sie haben "@" oder "." vergessen (siehe auch: Parameter TO_S und CC (Seite 2959)).   |
| 8535   | SMTP-Authentisierung unvollständig  | Eventuell haben Sie einen fehlerhaften Benutzernamen oder ein fehlerhaftes Passwort eingegeben.   |
| 8550   | Mailserver kann nicht erreicht werden. Sie haben keine Zugriffsrechte.  | Möglicherweise haben Sie einen fehlerhaften Benutzernamen bzw. Passwort eingegeben oder der Mailserver unterstützt Ihr Login nicht. Eine weitere Fehlerursache könnte eine fehlerhafte Eingabe des Domainnamens nach dem "@" an den Parametern TO_S bzw. CC sein (siehe auch: Parameter TO_S und CC (Seite 2959)).  |
| 8552   | Aktion abgebrochen: Überschreitung der zugewiesenen Speichergröße   | Versuchen Sie es später noch einmal.  |
| 8554   | Übertragung fehlgeschlagen  | Versuchen Sie es später noch einmal.  |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |   |

**Weitere**

**TCON: Kommunikationsverbindung aufbauen**

**TCON: Kommunikationsverbindung aufbauen**

**Beschreibung**

Mit der Anweisung "TCON" richten Sie eine Kommunikationsverbindung ein und bauen diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht. "TCON" wird asynchron ausgeführt.

Zum Einrichten der Kommunikationsverbindung werden die am Parameter CONNECT und ID angegebenen Verbindungsdaten verwendet. Um die Verbindung aufzubauen, muss am Parameter REQ eine steigende Flanke erfasst werden. Bei einem erfolgreichen Aufbau der Verbindung wird der Parameter DONE auf "1" gesetzt.

**Anzahl möglicher Verbindungen**

Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU.

**Verbindung bei TCP und ISO on TCP**

Beide Kommunikationspartner rufen die Anweisung "TCON" zum Einrichten und Aufbauen der Kommunikationsverbindung auf. In der Parametrierung hinterlegen Sie, welcher der aktive und welcher der passive Kommunikationsendpunkt ist.

Bei Verbindungsabbruch durch z. B. Leitungsunterbrechung oder durch den remoten Kommunikationspartner versucht der aktive Partner, die eingerichtete Verbindung wieder aufzubauen. Sie müssen "TCON" nicht erneut aufrufen. Dies gilt jedoch nur, wenn "TCON" einmal erfolgreich durchlaufen wurde (DONE = 1).

Mit der Ausführung der Anweisung "TDISCON (Seite 2976)" oder im Betriebszustand STOP der CPU wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung müssen Sie "TCON" erneut ausführen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "TCON":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------|------------------------------|--|
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | Startet den Auftrag zum Aufbauen der in der ID angegebenen Verbindung bei einer steigenden Flanke. |
| ID        | Input       | CONN_OUC (WORD) | E, A, M, D, L oder Konstante | Referenz auf die zugewiesene Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF                     |

| Parameter | Deklaration | Datentyp   | Speicherbereich | Beschreibung   |
|-----------|-------------|------------|-----------------|--|
| CONNECT   | InOut       | TCON_Param | D               | Zeiger auf die Verbindungsbeschreibung<br>Siehe auch: Auto-Hotspot   |
| DONE      | Output      | BOOL       | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>   |
| BUSY      | Output      | BOOL       | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>• 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden</li> </ul> |
| ERROR     | Output      | BOOL       | E, A, M, D, L   | Statusparameter ERROR: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>   |
| STATUS    | Output      | WORD       | E, A, M, D, L   | Status der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob einen Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TCON" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| BUSY | DONE | ERROR | Beschreibung   |
|------|------|-------|--|
| 1    | 0    | 0     | Der Auftrag wird bearbeitet.   |
| 0    | 1    | 0     | Der Auftrag wurde erfolgreich durchgeführt.  |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben. |
| 0    | 0    | 0     | Kein neuer Auftrag wurde zugewiesen.   |

### Parameter ERROR und STATUS

| ERROR | STATUS* (W#16#...) | Erläuterung                                       |
|-------|--------------------|---|
| 0     | 0000               | Verbindung wurde erfolgreich aufgebaut.           |
| 0     | 7000               | Keine Auftragsbearbeitung aktiv                   |
| 0     | 7001               | Auftragsbearbeitung starten, Verbindung aufbauen. |
| 0     | 7002               | Verbindung wird aufgebaut (REQ irrelevant).       |

| ERROR   | STATUS*<br>(W#16#...) | Erläuterung  |
|---|-----------------------|--|
| 1   | 8085                  | ID wird von einer projektierten Verbindung verwendet   |
| 1   | 8086                  | Der Parameter ID liegt außerhalb des zulässigen Bereichs.  |
| 1   | 8087                  | Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich  |
| 1   | 8089                  | Der Parameter CONNECT zeigt nicht auf einen Datenbaustein.   |
| 1   | 809A                  | Der Parameter CONNECT zeigt auf ein Feld, das nicht der Länge der Verbindungsbeschreibung entspricht.  |
| 1   | 809B                  | Das Element Interfaceld innerhalb der TCON_xxx Struktur referenziert keine Hardware-Kennung einer CPU- oder CM/CP Schnittstelle oder hat den Wert "0".   |
| 1   | 80A0                  | Sammelfehler für Fehlercodes W#16#80A1 und W#16#80A2   |
| 1   | 80A2                  | Lokaler oder remoter Port wird vom System verwendet.   |
| 1   | 80A3                  | Es wird versucht, eine vorhandene Verbindung erneut aufzubauen.  |
| 1   | 80A4                  | IP-Adresse des remoten Endpunkts der Verbindung ist ungültig, d.h. sie entspricht der IP-Adresse des lokalen Partners.   |
| 1   | 80A5                  | Verbindungs-ID wird bereits verwendet.   |
| 1   | 80A7                  | Kommunikationsfehler: Sie haben "TDISCON (Seite 2976)" ausgeführt, bevor "TCON" beendet war.   |
| 1   | 80B2                  | Der Parameter CONNECT zeigt auf einen Datenbaustein, der mit dem Attribut "Nur im Ladespeicher ablegen" generiert wurde.   |
| 1   | 80B4                  | Sie haben beim passiven Aufbauen einer Verbindung mit der Protokollvariante ISO on TCP (connection_type = B#16#12) eine oder mehrere der folgenden Bedingungen verletzt: <ul style="list-style-type: none"> <li>• local_tsap_id_len &gt;= B#16#02</li> <li>• local_tsap_id[1] = B#16#E0</li> <li>• Bei local_tsap_id_len &gt;= B#16#03 ist local_tsap_id[1] ein ASCII-Zeichen.</li> <li>• local_tsap_id[1] ist ein ASCII-Zeichen und local_tsap_id_len &gt;= B#16#03.</li> </ul> |
| 1   | 80B5                  | Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig.   |
| 1   | 80B6                  | Parametrierungsfehler im Parameter connection_type des SDTs TCON_Param.  |
| 1   | 80B7                  | Fehler in einem der folgenden Parameter des Datenbausteins für Verbindungsbeschreibung: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.  |
| 1   | 80B8                  | Verbindungsbeschreibung des Strukturelements ID und der Bausteinparameter ID sind unterschiedlich.   |
| 1   | 80C3                  | Alle Verbindungsressourcen sind belegt.  |
| 1   | 80C4                  | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>• Die Verbindung kann derzeit nicht aufgebaut werden.</li> <li>• Die Schnittstelle empfängt gerade neue Parameter.</li> <li>• Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON (Seite 2976)" entfernt.</li> </ul>  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                       |  |



## TCON: Kommunikationsverbindung aufbauen

### Beschreibung

Mit der Anweisung "TCON" richten Sie eine Kommunikationsverbindung ein und bauen diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der CPU gehalten und überwacht. "TCON" wird asynchron ausgeführt.

Zum Einrichten der Kommunikationsverbindung werden die am Parameter CONNECT und ID angegebenen Verbindungsdaten verwendet.

- Nutzen Sie am Parameter CONNECT nach Möglichkeit nur vordefinierte Strukturen, wie sie durch die Verbindungsparametrierung im Inspektorfenster des Programmiereditors erstellt werden.
- Wird der Parameter CONNECT nicht mit einer der in der Parametertabelle aufgelisteten Strukturen verschaltet oder enthält die Struktur einen Fehler, wird am Parameter STATUS der Fehlercode 8089 ausgegeben.

Um die Verbindung aufzubauen, muss am Parameter REQ eine steigende Flanke erfasst werden. Bei einem erfolgreichen Aufbau der Verbindung wird der Parameter DONE auf "1" gesetzt.

Die Anweisung "TCON" V3.0 kann auch mit der CPU S7-1200 ab der Version 4.0 verwendet werden.

### Anzahl möglicher Verbindungen

Die Anzahl der möglichen Kommunikationsverbindungen entnehmen Sie den technischen Daten Ihrer CPU.

### Verbindung bei TCP und ISO on TCP

Beide Kommunikationspartner rufen die Anweisung "TCON" zum Einrichten und Aufbauen der Kommunikationsverbindung auf. In der Parametrierung hinterlegen Sie, welcher der aktive und welcher der passive Kommunikationsendpunkt ist.

Bei Verbindungsabbruch durch z. B. Leitungsunterbrechung oder durch den remoten Kommunikationspartner versucht der aktive Partner, die eingerichtete Verbindung wieder aufzubauen. Sie müssen "TCON" nicht erneut aufrufen. Dies gilt jedoch nur, wenn "TCON" einmal erfolgreich durchlaufen wurde (DONE = 1).

Mit der Ausführung der Anweisung "TDISCON (Seite 2976)" oder im Betriebszustand STOP der CPU wird eine bestehende Verbindung abgebrochen und die eingerichtete Verbindung entfernt. Zum erneuten Einrichten und Aufbauen der Verbindung müssen Sie "TCON" erneut ausführen.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "TCON":

| Parameter | Deklaration | Datentyp        | Speicherbereich               | Beschreibung  |
|-----------|-------------|-----------------|-------------------------------|---|
| REQ       | Input       | BOOL            | E, A, M, D, L, oder Konstante | Startet den Auftrag zum Aufbauen der Verbindung bei einer steigenden Flanke.  |
| ID        | Input       | CONN_OUC (WORD) | E, A, M, D, L, oder Konstante | Referenz auf die zugewiesene Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF  |
| CONNECT   | InOut       | VARIANT         | D                             | Zeiger auf die Verbindungsbeschreibung <ul style="list-style-type: none"> <li>• Bei TCP oder UDP verwenden Sie die Struktur TCON_IP_v4<br/>Zur Beschreibung siehe: Auto-Hotspot</li> <li>• Bei ISO-on-TCP verwenden Sie die Struktur TCON_IP_RFC<br/>Zur Beschreibung siehe: Auto-Hotspot</li> <li>• Bei ISO verwenden Sie die Struktur TCON_ISOnative (nur bei CP1543-1)<br/>Zur Beschreibung siehe: "Aufbau der Verbindungsbeschreibung nach TCON_ISOnative"</li> </ul> |
| DONE      | Output      | BOOL            | E, A, M, D, L                 | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>  |
| BUSY      | Output      | BOOL            | E, A, M, D, L                 | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>• 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden</li> </ul>  |
| ERROR     | Output      | BOOL            | E, A, M, D, L                 | Statusparameter ERROR: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>  |
| STATUS    | Output      | WORD            | E, A, M, D, L                 | Status der Anweisung  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter BUSY, DONE und ERROR**

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob einen Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TCON" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die Anweisung "TCON" erzeugt in Version 3.0 eine Fehlermeldung, wenn ein aktiver Verbindungsaufbau zu einem remoten Partner scheitert. Um einen erneuten Verbindungsaufbau durchzuführen, erzeugen Sie am Parameter REQ eine steigende Flanke.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| BUSY | DONE | ERROR | Beschreibung   |
|------|------|-------|--|
| 1    | 0    | 0     | Der Auftrag wird bearbeitet.   |
| 0    | 1    | 0     | Der Auftrag wurde erfolgreich durchgeführt.  |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben. |
| 0    | 0    | 0     | Kein neuer Auftrag wurde zugewiesen.   |

### Aufbau der Verbindungsbeschreibung nach TCON\_ISOnative

Um die Kommunikationsverbindungen bei ISO zu parametrieren, wird ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON\_ISOnative verwendet. Die feste Datenstruktur des TCON\_ISOnative enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden.

| Byte          | Parameter         | Datentyp                | Startwert | Beschreibung   |
|---------------|-------------------|-------------------------|-----------|--|
| 0 ...<br>1    | InterfaceId       | HW_ANY                  | 64        | Hardware-Kennung der lokalen Schnittstelle (Wertebereich: 0 bis 65535).  |
| 2 ...<br>3    | ID                | CONN_OUC                | 1         | Referenz auf diese Verbindung (eindeutige ID im Wertebereich: 1 bis 4095).   |
| 4             | ConnectionType    | BYTE                    | 16#16     | Verbindungstyp: ISO  |
| 5             | ActiveEstablished | BOOL                    | TRUE      | Kennung für die Art des Verbindungsaufbaus: <ul style="list-style-type: none"> <li>FALSE: passiver Verbindungsaufbau</li> <li>TRUE: aktiver Verbindungsaufbau</li> </ul>   |
| 8 ...<br>13   | RemoteMacAddress  | ARRAY [1..6] of<br>BYTE | -         | MAC-Adresse des Partner-Endpunkts, z. B. für 00-74-41-FD-AE-84: <ul style="list-style-type: none"> <li>MacAddr[1] = 00</li> <li>MacAddr[2] = 74</li> <li>MacAddr[3] = 41</li> <li>MacAddr[4] = FD</li> <li>MacAddr[5] = AE</li> <li>MacAddr[6] = 84</li> </ul> |
| 14 ..<br>. 19 | LocalMacAddress   | ARRAY [1..6] of<br>BYTE | -         | MAC-Adresse des lokalen Endpunkts, z. B. für 00-74-41-FD-AE-84: <ul style="list-style-type: none"> <li>MacAddr[1] = 00</li> <li>MacAddr[2] = 74</li> <li>MacAddr[3] = 41</li> <li>MacAddr[4] = FD</li> <li>MacAddr[5] = AE</li> <li>MacAddr[6] = 84</li> </ul> |
| 20 ..<br>. 53 | RemoteTSelector   | Struct                  | -         | TSelector des entfernten Verbindungspartners: <ul style="list-style-type: none"> <li>Byte 20 bis 21 = TSelLength</li> <li>Byte 22 bis 53: = TSel[1-32]</li> </ul>  |
|               | TSelLength        | UINT                    | -         | Wertebereich 0 bis 32  |

| Byte               | Parameter              | Datentyp              | Startwert | Beschreibung  |
|--------------------|------------------------|-----------------------|-----------|---|
|                    | TSel                   | ARRAY [1..32] of BYTE | -         | Wertebereich jeweils 0 bis 255 in Bytes   |
| 54 ..<br>. 87      | LocalTSelector         | Struct                | -         | TSelector des entfernten Verbindungspartners: <ul style="list-style-type: none"> <li>• Byte 20 bis 21 = TSelLength</li> <li>• Byte 22 bis 53: = TSel[1-32]</li> </ul> |
|                    | TSelLength             | UINT                  |           | Wertebereich 0 bis 32   |
|                    | TSel                   | ARRAY [1..32] of BYTE | -         | Wertebereich jeweils 0 bis 255 in Bytes   |
| 88 ..<br>...<br>89 | CrRetransmissionTime   | UINT                  | -         | Zeitdauer bis zur Wiederholung des Verbindungsaufbaus in Sekunden.  |
| 90 ..<br>. 91      | DataRetransmissionTime | UINT                  | 100 ms    | Zeitdauer bis zur Wiederholung des Datentransfers in Millisekunden.   |
| 92 ..<br>. 93      | MaxRetransmissionCount | UINT                  | -         | Maximale Anzahl der Wiederholungen.   |
| 94 ..<br>. 95      | InactivityTime         | UINT                  | -         | in Sekunden   |
| 96 ..<br>. 97      | WindowTime             | UINT                  |           | in Sekunden   |

**Parameter ERROR und STATUS**

| ERROR | STATUS* (W#16#...) | Erläuterung  |
|-------|--------------------|--|
| 0     | 0000               | Verbindung wurde erfolgreich aufgebaut.  |
| 0     | 7000               | Keine Auftragsbearbeitung aktiv  |
| 0     | 7001               | Auftragsbearbeitung starten, Verbindung aufbauen.  |
| 0     | 7002               | Verbindung wird aufgebaut (REQ irrelevant).  |
| 1     | 8085               | ID wird von einer projektierten Verbindung verwendet.  |
| 1     | 8086               | Der Parameter ID liegt außerhalb des zulässigen Bereichs.  |
| 1     | 8087               | Maximale Anzahl von Verbindungen erreicht, keine weitere Verbindung möglich  |
| 1     | 8089               | Der Parameter CONNECT zeigt nicht auf eine Verbindungsbeschreibung oder die Verbindungsbeschreibung wurde manuell erstellt.  |
| 1     | 809A               | Die Struktur am Parameter CONNECT wird nicht unterstützt oder die Länge ist ungültig.  |
| 1     | 809B               | Das Element Interfaceld innerhalb der TCON_xxx Struktur referenziert keine Hardware-Kennung einer CPU- oder CM/CP Schnittstelle oder hat den Wert "0".                                 |
| 1     | 80A2               | Lokaler oder remoter Port wird vom System verwendet. Lokal sind folgende Ports reserviert: 20, 21, 80, 102, 135, 161, 162, 443, 34962, 34963, 34964 sowie der Bereich 49152 bis 65535. |
| 1     | 80A3               | ID wird von einer durch das Anwenderprogramm erstellten Verbindung genutzt, die auch die gleiche Verbindungsbeschreibung am Parameter CONNECT verwendet.                               |
| 1     | 80A4               | IP-Adresse des remoten Endpunkts der Verbindung ist ungültig oder sie entspricht der IP-Adresse des lokalen Partners.  |
| 1     | 80A7               | Kommunikationsfehler: Sie haben "TDISCON (Seite 2976)" ausgeführt, bevor "TCON" beendet war.   |

| ERROR | STATUS*<br>(W#16#...) | Erläuterung   |
|-------|-----------------------|---|
| 1     | 80B4                  | Nur bei TCON_IP_RFC: Der lokale T-Selektor wurde nicht angegeben oder das erste Byte enthält nicht den Wert 0x0E (nur bei einer Länge von T-Selector = 2) oder der lokale T-Selektor beginnt mit "SIMATIC-".  |
| 1     | 80B5                  | Bei Verbindungstyp 13 = UDP ist nur ein passiver Verbindungsaufbau zulässig (Parameter ActiveEstablished der Struktur TCON_IP_v4 / TCON_PARAM hat den Wert TRUE).   |
| 1     | 80B6                  | Parametrierungsfehler im Parameter ConnectionType des Datenbausteins für Verbindungsbeschreibung. <ul style="list-style-type: none"> <li>Nur gültig bei TCON_IP_v4: 0x11, 0x0B und 0x13.</li> <li>Nur gültig bei TCON_IP_RFC: 0x0C und 0x12</li> </ul>  |
| 1     | 80B7                  | Bei TCON_IP_v4: <ul style="list-style-type: none"> <li>TCP (aktiver Verbindungsaufbau): Entfernter Port ist "0".</li> <li>TCP (passiver Verbindungsaufbau): Lokaler Port ist "0".</li> <li>UDP: Lokaler Port ist "0".</li> </ul> Bei TCON_IP_RFC: <ul style="list-style-type: none"> <li>Lokaler (LocalTSelector) oder entfernter (RemoteTSelector) T-Selektor wurde mit einer Länge von mehr als 32 Byte angegeben.</li> <li>Bei TselLength des T-Selektors (lokal oder entfernt) wurde eine Länge größer als 32 eingegeben.</li> <li>Fehler bei der Länge der IP-Adresse des jeweiligen Verbindungspartners.</li> </ul> |
| 1     | 80B8                  | Parameter ID in der lokalen Verbindungsbeschreibung (Struktur am Parameter CONNECT) und Parameter ID der Anweisung sind unterschiedlich.  |
| 1     | 80C3                  | Alle Verbindungsressourcen sind belegt.   |
| 1     | 80C4                  | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die Verbindung kann derzeit nicht aufgebaut werden.</li> <li>Die Schnittstelle empfängt gerade neue Parameter.</li> <li>Die projektierte Verbindung wird gerade von einer Anweisung "TDISCON (Seite 2976)" entfernt.</li> </ul>   |
| 1     | 80C5                  | Der Verbindungspartner verweigert den Verbindungsaufbau, hat die Verbindung abgebaut oder aktiv beendet.  |
| 1     | 80C6                  | Der Verbindungspartner kann nicht erreicht werden (Netzwerkfehler).   |
| 1     | 80C7                  | Zeitüberschreitung der Ausführung.  |
| 1     | 80C8                  | Wert am Parameter ID wird bereits von einer Verbindung genutzt, die über das Anwenderprogramm erstellt wurde. Die Verbindung verwendet die gleiche ID, aber unterschiedliche Verbindungseinstellungen am Parameter CONNECT.   |
| 1     | 80C9                  | Validierung des Verbindungspartners fehlgeschlagen. Der Verbindungspartner, der eine Verbindung aufbauen will, entspricht nicht dem definierten Partner der Struktur am Parameter CONNECT.  |
| 1     | 80CE                  | Die IP-Adresse des lokalen Interfaces ist 0.0.0.0.  |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## TDISCON: Kommunikationsverbindung abbauen

### Beschreibung

Die Anweisung "TDISCON" baut eine Kommunikationsverbindung der CPU zu einem Verbindungspartner ab.

### Arbeitsweise

"TDISCON" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Auftrag zum Verbindungsabbau, indem Sie "TDISCON" mit REQ = 1 aufrufen.

Nach dem erfolgreichen Durchlauf von "TDISCON" ist die bei "TCON" angegebene ID nicht mehr gültig und kann damit weder zum Senden noch zum Empfangen verwendet werden.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET\_VAL der asynchron arbeitenden Anweisungen (siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620)).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich "TDISCON" aktuell befindet bzw. wann der Verbindungsaufbau beendet ist.

| BUSY | DONE | ERROR | Beschreibung   |
|------|------|-------|--|
| 1    | 0    | 0     | Der Auftrag ist in Bearbeitung.  |
| 0    | 1    | 0     | Der Auftrag wurde erfolgreich abgeschlossen.   |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen. |
| 0    | 0    | 0     | Der Anweisung wurde kein (neuer) Auftrag erteilt.  |

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TDISCON":

| Parameter | Deklaration | Datentyp        | Speicherbereich              |                              | Beschreibung   |
|-----------|-------------|-----------------|------------------------------|------------------------------|--|
|           |             |                 | S7-1200                      | S7-1500                      |  |
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Steuerparameter REQUEST, startet den Auftrag zum Abbau der durch ID gegebenen Verbindung. Der Auftragsstart erfolgt bei steigender Flanke. |
| ID        | Input       | CONN_OUT (WORD) | D, L oder Konstante          | E, A, M, D, L oder Konstante | Referenz auf die abzubauenen Verbindung (Verbindungs-ID)<br>Wertebereich: W#16#0001 bis W#16#0FFF  |

| Parameter | Deklaration | Datentyp | Speicherbereich |               | Beschreibung  |
|-----------|-------------|----------|-----------------|---------------|---|
|           |             |          | S7-1200         | S7-1500       |   |
| DONE      | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L | Statusparameter:<br><ul style="list-style-type: none"> <li>0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>1: Auftrag wurde fehlerfrei ausgeführt</li> </ul>                                      |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L | Statusparameter:<br><ul style="list-style-type: none"> <li>BUSY = 1: Der Auftrag ist noch nicht beendet.</li> <li>BUSY = 0: Der Auftrag ist beendet oder noch nicht gestartet.</li> </ul>                                   |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L | Statusparameter:<br><ul style="list-style-type: none"> <li>ERROR=0: Kein Fehler.</li> <li>ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers</li> </ul> |
| STATUS    | Output      | WORD     | E, A, M, D, L   | E, A, M, D, L | Statusparameter:<br>Fehlerinformation (siehe "Parameter ERROR und STATUS")  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

| ERROR | STATUS* (W#16#...) | Erläuterung  |
|-------|--------------------|--|
| 0     | 0000               | Verbindung konnte abgebaut werden.   |
| 0     | 7000               | Keine Auftragsbearbeitung aktiv.   |
| 0     | 7001               | Start der Auftragsbearbeitung, Verbindung wird abgebaut.   |
| 0     | 7002               | Zwischenaufwurf (REQ irrelevant), Verbindung wird abgebaut.  |
| 1     | 8086               | Parameter ID liegt in einem unzulässigen Wertebereich.   |
| 1     | 80A3               | Es wird versucht, eine nicht bestehende Verbindung abzubauen oder die Verbindung ist bereits abgebaut.           |
| 1     | 80C4               | Temporärer Kommunikationsfehler: Schnittstelle wird neu parametrisiert bzw. Verbindung wird gerade eingerichtet. |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

## TSEND: Daten über Kommunikationsverbindung senden

## TSEND: Daten über Kommunikationsverbindung senden

### Beschreibung

Die folgende Beschreibung der Anweisung "TSEND" ist gültig für die CPU S7-1200 bis Version 3.0.

Mit der Anweisung "TSEND" senden Sie Daten über eine bestehende Kommunikationsverbindung. "TSEND" wird asynchron ausgeführt.

Den Sendebereich geben Sie am Parameter DATA an. In diesem sind Adresse und die Länge der zu sendenden Daten enthalten. Für die zu sendenden Daten können alle Datentypen bis auf BOOL und Array of BOOL verwendet werden.

Der Sendeauftrag wird ausgeführt, wenn am Parameter REQ eine steigende Flanke erfasst wird.

Am Parameter LEN legen Sie die maximale Anzahl der Bytes fest, die mit einem Sendeauftrag gesendet werden.

- Bei der Datenübertragung über TCP (Streaming-Protokoll) werden von der Anweisung "TSEND" keine Information über die Länge der gesendeten Daten an "TRCV (Seite 2984)" übermittelt.
- Bei der Datenübertragung über ISO-on-TCP (Nachrichtenorientiertes Protokoll) wird die Länge der gesendeten Daten an "TRCV (Seite 2984)" übermittelt. Die Menge von Daten, die über "TSEND" als Paket gesendet werden, müssen auf der Empfängerseite ("TRCV (Seite 2984)") auch wieder empfangen werden:
  - Ist der Empfangspuffer zu klein für die gesendeten Daten, gibt es auf Empfängerseite einen Fehler.
  - Ist der Empfangspuffer ausreichend groß, kehrt "TRCV" mit DONE=1 dann zurück, sobald das Datenpaket empfangen wurde.

Die zu sendenden Daten dürfen nicht editiert werden, bis der Sendeauftrag vollständig ausgeführt ist. Bei einer erfolgreichen Ausführung des Sendeauftrags wird der Parameter DONE auf "1" gesetzt. Der Signalzustand "1" am Parameter DONE ist dabei keine Bestätigung, dass die gesendeten Daten vom Kommunikationspartner bereits ausgelesen wurden.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------|------------------------------|--|
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | Startet den Sendeauftrag bei einer steigenden Flanke.                                    |
| ID        | Input       | CONN_OUC (WORD) | D, L oder Konstante          | Referenz auf die mit "TCON" aufgebaute Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF |
| LEN       | Input       | UINT            | E, A, M, D, L                | Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden.                          |



| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| DATA      | InOut       | VARIANT  | E, A, M, D      | Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält. Die Adresse verweist auf: <ul style="list-style-type: none"> <li>• Das Prozessabbild der Eingänge</li> <li>• Das Prozessabbild der Ausgänge</li> <li>• Einen Merker</li> <li>• Einen Datenbaustein</li> </ul> |
| DONE      | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>   |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>• 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.</li> </ul>  |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>  |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Status der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter LEN und DATA

- Bei LEN = 0 werden alle über den Parameter DATA angegebenen Daten versendet.
- Ist die Anzahl der Bytes am Parameter LEN größer als die Länge der zu sendenden Daten, die über den Parameter DATA definiert wurden, wird der Fehlercode 8088 am Parameter STATUS ausgegeben (siehe im Folgenden Beschreibung des Parameters STATUS).
- Wird über den Parameter DATA eine Struktur (Struct) referenziert, kann LEN kürzer sein als die Struktur. In diesem Fall werden nur die Daten bis zur Länge des Parameters LEN übertragen.
- Bei den Datentypen STRING und WSTRING werden alle Daten übertragen, wenn der Parameter LEN = 0 ist. Bei LEN > 0 muss die Länge mindestens die maximale Anzahl der Bytes umfassen sowie zwei Byte zusätzlich, welche die Längeninformaton enthalten. Weitere Informationen zum Aufbau der Datentypen finden Sie unter: "Übersicht über die gültigen Datentypen (Seite 1087)".
- Die maximale Anzahl von Bytes, die übertragen werden können, beträgt 65534.
- Bei der Verwendung von strukturierten Variablen aus optimierten DBs sollte die Adresse der strukturierten Variable am Parameter DATA verschaltet und der Parameter LEN = 0 gesetzt werden. Damit kann eine typsichere Übertragung der gesamten Struktur gewährleistet werden, sofern auf Empfängerseite dieselbe Struktur verwendet wird.

### Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TSEND" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| BUSY | DONE | ERROR | Beschreibung  |
|------|------|-------|---|
| 1    | 0    | 0     | Der Auftrag wird bearbeitet.  |
| 0    | 1    | 0     | Der Auftrag wurde erfolgreich durchgeführt.   |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben. |
| 0    | 0    | 0     | Kein neuer Auftrag wurde zugewiesen.  |

#### Hinweis

Wegen der asynchronen Bearbeitung von "TSEND" müssen Sie die Daten im Sendebereich konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert "1" annimmt.

### Parameter ERROR und STATUS

| ERROR | STATUS* (W#16#...) | Beschreibung  |
|-------|--------------------|---|
| 0     | 0000               | Sendeauftrag fehlerfrei beendet.  |
| 0     | 7000               | Keine Auftragsbearbeitung aktiv.  |
| 0     | 7001               | Beginn der Auftragsbearbeitung, Daten werden gesendet.<br>Während dieser Bearbeitung greift das Betriebssystem auf die Daten im Sendebereich DATA zu.   |
| 0     | 7002               | Auftrag wird bearbeitet (REQ irrelevant).<br>Während dieser Bearbeitung greift das Betriebssystem auf die Daten im Sendebereich DATA zu.  |
| 1     | 8085               | <ul style="list-style-type: none"> <li>Parameter LEN ist größer als der größte zulässige Wert (65536).</li> <li>Parameter DATA und LEN haben beide den Wert "0".</li> </ul>   |
| 1     | 8086               | Der Parameter ID liegt außerhalb des zulässigen Adressbereichs (1..0xFF).   |
| 1     | 8088               | Parameter LEN ist größer als der in DATA angegebene Bereich.  |
| 1     | 80A1               | Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die angegebene Verbindung wurde noch nicht aufgebaut.</li> <li>Die angegebene Verbindung wird gerade beendet. Übertragung über diese Verbindung ist nicht möglich.</li> <li>Die Schnittstelle wird neu initialisiert.</li> </ul> |
| 1     | 80B3               | Die parametrisierte Protokollvariante (Parameter ConnectionType in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie bei einer UDP-Verbindung die Anweisung "TUSEND".   |

| ERROR   | STATUS*<br>(W#16#...) | Beschreibung   |
|---|-----------------------|--|
| 1   | 80C3                  | <ul style="list-style-type: none"> <li>Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.</li> <li>Interner Mangel an Ressourcen.</li> </ul>  |
| 1   | 80C4                  | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die Verbindung zum Partner kann derzeit nicht aufgebaut werden.</li> <li>Die Schnittstelle empfängt gerade neue Parametereinstellungen oder die Verbindung wird gerade aufgebaut.</li> </ul> |
| 1   | 80C5                  | Verbindung durch den Kommunikationspartner abgebaut.   |
| 1   | 80C6                  | Netzwerkfehler. Kommunikationspartner kann nicht erreicht werden.  |
| 1   | 80C7                  | Zeitüberschreitung bei Ausführung.   |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                       |  |

## TSEND: Daten über Kommunikationsverbindung senden

### Beschreibung

Die folgende Beschreibung der Anweisung "TSEND" ist gültig für die CPU S7-1500 und S7-1200 V4.0.

Mit der Anweisung "TSEND" senden Sie Daten über eine bestehende Kommunikationsverbindung. "TSEND" wird asynchron ausgeführt.

Den Sendebereich geben Sie am Parameter DATA an. In diesem sind Adresse und die Länge der zu sendenden Daten enthalten. Für die zu sendenden Daten können alle Datentypen bis auf BOOL und Array of BOOL verwendet werden.

Der Sendeauftrag wird ausgeführt, wenn am Parameter REQ eine steigende Flanke erfasst wird.

Am Parameter LEN legen Sie die maximale Anzahl der Bytes fest, die mit einem Sendeauftrag gesendet werden.

- Bei der Datenübertragung über TCP (Streaming-Protokoll) werden von der Anweisung "TSEND" keine Informationen über die Länge der gesendeten Daten an "TRCV (Seite 2984)" übermittelt.
- Bei der Datenübertragung über ISO-on-TCP (Nachrichtenorientiertes Protokoll) wird die Länge der gesendeten Daten an "TRCV (Seite 2984)" übermittelt. Die Menge von Daten, die über "TSEND" als Paket gesendet werden, müssen auf der Empfängerseite ("TRCV (Seite 2984)") auch wieder empfangen werden:
  - Ist der Empfangspuffer zu klein für die gesendeten Daten, gibt es auf Empfängerseite einen Fehler.
  - Ist der Empfangspuffer ausreichend groß, kehrt "TRCV" mit DONE=1 dann zurück, sobald das Datenpaket empfangen wurde.

Die zu sendenden Daten dürfen nicht editiert werden, bis der Sendeauftrag vollständig ausgeführt ist. Bei einer erfolgreichen Ausführung des Sendeauftrags wird der Parameter DONE auf "1" gesetzt. Der Signalzustand "1" am Parameter DONE ist dabei keine Bestätigung, dass die gesendeten Daten vom Kommunikationspartner bereits ausgelesen wurden.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "TSEND":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------|------------------------------|--|
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | Startet den Sendeauftrag bei einer steigenden Flanke.  |
| ID        | Input       | CONN_OUC (WORD) | D, L oder Konstante          | Referenz auf die mit "TCON" aufgebaute Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF   |
| LEN       | Input       | UDINT           | E, A, M, D, L                | Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden.  |
| DATA      | InOut       | VARIANT         | E, A, M, D                   | Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält. Die Adresse verweist auf: <ul style="list-style-type: none"> <li>• Das Prozessabbild der Eingänge</li> <li>• Das Prozessabbild der Ausgänge</li> <li>• Einen Merker</li> <li>• Einen Datenbaustein</li> </ul> |
| DONE      | Output      | BOOL            | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Auftrag fehlerfrei ausgeführt</li> </ul>   |
| BUSY      | Output      | BOOL            | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>• 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.</li> </ul>  |
| ERROR     | Output      | BOOL            | E, A, M, D, L                | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>  |
| STATUS    | Output      | WORD            | E, A, M, D, L                | Status der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter LEN und DATA**

- Bei LEN = 0 werden alle über den Parameter DATA angegebenen Daten versendet.
- Ist die Anzahl der Bytes am Parameter LEN größer als die Länge der zu sendenden Daten, die über den Parameter DATA definiert wurden, wird der Fehlercode 8088 am Parameter STATUS ausgegeben (siehe im Folgenden Beschreibung des Parameters STATUS).
- Wird über den Parameter DATA eine Struktur (Struct) referenziert, kann LEN kürzer sein als die Struktur. In diesem Fall werden nur die Daten bis zur Länge des Parameters LEN übertragen.

- Bei den Datentypen STRING und WSTRING werden alle Daten übertragen, wenn der Parameter LEN = 0 ist. Bei LEN > 0 muss die Länge mindestens die maximale Anzahl der Bytes umfassen sowie zwei Byte zusätzlich, welche die Längeninformaton enthalten. Weitere Informationen zum Aufbau der Datentypen finden Sie unter: "Übersicht über die gültigen Datentypen (Seite 1087)".
- Die maximale Anzahl von Bytes, die übertragen werden können, ist geräteabhängig.
- Bei der Verwendung von strukturierten Variablen aus optimierten DBs sollte die Adresse der strukturierten Variable am Parameter DATA verschaltet und der Parameter LEN = 0 gesetzt werden. Damit kann eine typsichere Übertragung der gesamten Struktur gewährleistet werden, sofern auf Empfängerseite dieselbe Struktur verwendet wird.

### Parameter BUSY, DONE und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "TSEND" auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| BUSY | DONE | ERROR | Beschreibung  |
|------|------|-------|---|
| 1    | 0    | 0     | Der Auftrag wird bearbeitet.  |
| 0    | 1    | 0     | Der Auftrag wurde erfolgreich durchgeführt.   |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird im Parameter STATUS angegeben. |
| 0    | 0    | 0     | Kein neuer Auftrag wurde zugewiesen.  |

#### Hinweis

Wegen der asynchronen Bearbeitung von "TSEND" müssen Sie die Daten im Sendebereich konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert "1" annimmt.

### Parameter ERROR und STATUS

| ERROR | STATUS*<br>(W#16#...) | Beschreibung  |
|-------|-----------------------|---|
| 0     | 0000                  | Sendeauftrag fehlerfrei beendet.  |
| 0     | 7000                  | Keine Auftragsbearbeitung aktiv.  |
| 0     | 7001                  | Beginn der Auftragsbearbeitung, Daten werden gesendet.<br>Während dieser Bearbeitung greift das Betriebssystem auf die Daten im Sendebereich DATA zu. |
| 0     | 7002                  | Auftrag wird bearbeitet (REQ irrelevant).<br>Während dieser Bearbeitung greift das Betriebssystem auf die Daten im Sendebereich DATA zu.              |

| ERROR | STATUS* (W#16#...) | Beschreibung  |
|-------|--------------------|---|
| 1     | 8085               | <ul style="list-style-type: none"> <li>Parameter LEN ist größer als der größte zulässige Wert (65536).</li> <li>Parameter DATA und LEN haben beide den Wert "0".</li> </ul>   |
| 1     | 8086               | Der Parameter ID liegt außerhalb des zulässigen Adressbereichs (1..0xFFFF).   |
| 1     | 8088               | Parameter LEN ist größer als der in DATA angegebene Bereich.  |
| 1     | 80A1               | Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die angegebene Verbindung wurde noch nicht aufgebaut.</li> <li>Die angegebene Verbindung wird gerade beendet. Übertragung über diese Verbindung ist nicht möglich.</li> <li>Die Schnittstelle wird neu initialisiert.</li> </ul> |
| 1     | 80B3               | Die parametrisierte Protokollvariante (Parameter ConnectionType in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie bei einer UDP-Verbindung die Anweisung "TUSEND".   |
| 1     | 80C3               | <ul style="list-style-type: none"> <li>Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.</li> <li>Interner Mangel an Ressourcen.</li> </ul>   |
| 1     | 80C4               | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die Verbindung zum Partner kann derzeit nicht aufgebaut werden.</li> <li>Die Schnittstelle empfängt gerade neue Parametereinstellungen oder die Verbindung wird gerade aufgebaut.</li> </ul>                          |
| 1     | 80C5               | Verbindung durch den Kommunikationspartner abgebaut.  |
| 1     | 80C6               | Netzwerkfehler. Kommunikationspartner kann nicht erreicht werden.   |
| 1     | 80C7               | Zeitüberschreitung bei Ausführung.  |

\* Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### TRCV: Daten über Kommunikationsverbindung empfangen

### TRCV: Daten über Kommunikationsverbindung empfangen

#### Beschreibung

Die folgende Beschreibung der Anweisung "TRCV" ist gültig für die CPU S7-1200 bis Version 3.0.

Mit der Anweisung "TRCV" empfangen Sie Daten über eine bestehende Kommunikationsverbindung. "TRCV" wird asynchron ausgeführt.

Wenn der Parameter EN\_R auf den Wert "1" gesetzt wird, wird der Datenempfang aktiviert. Die empfangenen Daten werden in einen Empfangsbereich eingetragen. Die Länge des Empfangsbereichs legen Sie abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls LEN <> 0) oder die Längenangabe des Parameters DATA (falls LEN = 0) fest.

Während Daten empfangen werden, dürfen an dem Parameter DATA und dem definierten Empfangsbereich keine Änderungen vorgenommen werden, um die Konsistenz der empfangenen Daten sicherzustellen.

Nach einem erfolgreichen Datenempfang wird der Parameter NDR auf den Wert "1" gesetzt. Die tatsächlich empfangene Datenmenge können Sie am Parameter RCVD\_LEN abfragen.

## Empfangsmodi von "TRCV"

Die folgende Tabelle zeigt, wie die empfangenen Daten in den Empfangsbereich eingetragen werden.

| Protokollvariante                                      | Verfügbarkeit der Daten im Empfangsbereich   | Parameter <code>connection_type*</code> der Verbindungsbeschreibung | Parameter LEN   |
|--|--|---|---|
| TCP<br>(Ad-hoc-Modus)                                  | Die Daten sind sofort verfügbar.   | Hexadezimalwert:<br>B#16#11<br>Integerwert: 17                      | 0   |
| TCP<br>(Datenempfang mit angegebener Länge)            | Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde. | Hexadezimalwert:<br>B#16#11<br>Integerwert: 17                      | 1 bis 8192  |
| ISO on TCP<br>(Nachrichtenorientiert Datenübertragung) | Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde. | Hexadezimalwert:<br>B#16#12<br>Integerwert: 18                      | <ul style="list-style-type: none"> <li>• 1 bis 1452, wenn ein CP verwendet wird.</li> <li>• 1 bis 8192, wenn kein CP verwendet wird.</li> </ul> |
| * Siehe "Auto-Hotspot".                                |  |   |   |

### TCP (Ad-hoc-Modus)

Den Ad-hoc-Modus gibt es nur bei der Protokollvariante TCP. Den Ad-hoc-Modus verwenden Sie, um mit der Anweisung "TRCV" Daten mit dynamischen Längen zu empfangen.

Den Ad-hoc-Modus stellen Sie ein, indem Sie dem Parameter LEN den Wert "0" zuweisen. Bei Verwendung des Ad-hoc-Modus können für Datenbausteine mit Standardzugriff alle Datentypen verwendet werden. Für Datenbausteine mit optimiertem Zugriff können als Datentypen nur ARRAY of BYTE oder Datentypen mit einer Länge von 8 Bit verwendet werden (z. B. CHAR, USINT, SINT, etc.). Ist der Ad-hoc-Modus aktiviert, wird der Datenempfang schon nach einem empfangenen Byte am Parameter NDR gemeldet.

### TCP (Datenempfang mit angegebener Länge)

Für einen Datenempfang mit angegebener Länge geben Sie die Länge der Daten am Parameter LEN an. Der Datenempfang wird erst abgeschlossen, wenn die am Parameter LEN angegebene Länge der Daten vollständig empfangen wurde. Erst danach sind die Daten im Empfangsbereich (Parameter DATA) verfügbar. Der Datenempfang wird durch den Ausgangsparameter NDR gemeldet. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD\_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

### ISO on TCP (Nachrichtenorientiert Datenübertragung)

Bei der Protokollvariante ISO on TCP werden über eine Verbindung abgeschlossene Nachrichtenblöcke gesendet, die vom Empfänger auch als solche erkannt werden. Bei Verwendung von ISO on TCP meldet "TRCV" den Datenempfang, sobald der Nachrichtenblock vollständig empfangen wurde. Der Empfangsbereich wird von den Parametern LEN und DATA definiert. Ist der Empfangspuffer (Parameter DATA) zu klein für

die gesendeten Daten, meldet "TRCV" einen Fehler. Der erfolgreiche Datenempfang wird durch den Ausgangsparameter NDR gemeldet. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD\_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

**Parameter**

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------------|------------------------------|---|
| EN_R      | Input       | BOOL            | E, A, M, D, L oder Konstante | Empfangsfreigabe  |
| ID        | Input       | CONN_OUC (WORD) | D, L oder Konstante          | Referenz auf die mit "TCON (Seite 2968)" aufgebaute Verbindung.<br>Wertebereich: W#16#0001 (1) bis W#16#0FFF (4095)   |
| LEN       | Input       | UDINT           | E, A, M, D, L oder Konstante | Länge des Empfangsbereichs in Bytes (ausgeblendet).<br>Wenn Sie einen Speicherbereich mit optimiertem Zugriff am Parameter DATA verwenden, muss der Parameter LEN den Wert "0" haben.   |
| DATA      | InOut       | VARIANT         | E, A, M, D                   | Zeiger auf den Empfangsbereich  |
| NDR       | Output      | BOOL            | E, A, M, D, L                | Statusparameter (New Data Received):<br><ul style="list-style-type: none"> <li>0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>1: Auftrag fehlerfrei ausgeführt</li> </ul>                                    |
| BUSY      | Output      | BOOL            | E, A, M, D, L                | Statusparameter:<br><ul style="list-style-type: none"> <li>0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden</li> </ul>                |
| ERROR     | Output      | BOOL            | E, A, M, D, L                | Statusparameter:<br><ul style="list-style-type: none"> <li>0: Kein Fehler.</li> <li>1: Bei der Ausführung der Anweisung ist ein Fehler aufgetreten.</li> </ul> Detailinformationen werden über den Parameter STATUS ausgegeben. |
| STATUS    | Output      | WORD            | E, A, M, D, L                | Statusparameter:<br>Ausgabe der Status- und Fehlerinformation.  |
| RCVD_LEN  | Output      | UDINT           | E, A, M, D, L                | Tatsächlich empfangene Datenmenge in Bytes  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".



## Parameter LEN, DATA und RCVD\_LEN

- Ist LEN = 0, werden die empfangenen Daten in dem am Parameter DATA angegebenen Empfangsbereich gespeichert. Die Anzahl der empfangenen Bytes werden am Parameter RCVD\_LEN angezeigt.
- Ist die am Parameter LEN angegebene Länge größer als die Länge der am Parameter DATA empfangenen Daten, wird der Fehlercode 8088 am Parameter STATUS ausgegeben (siehe im Folgenden Beschreibung des Parameters STATUS).
- Wird über den Parameter DATA eine Struktur (Struct) referenziert, kann LEN kürzer sein als die Struktur. In diesem Fall werden nur die Daten bis zur Länge des Parameters LEN übertragen.
- Verweist der Parameter DATA auf einen Datenbaustein mit optimiertem Zugriff, muss der Parameter LEN auf "0" gesetzt werden.
- Wird über den Parameter DATA ein Datentyp STRING referenziert, muss die am Parameter LEN angegebene Länge 0 oder  $\geq 2$  sein (LEN = 1 ist nicht erlaubt).
- Wird über den Parameter DATA ein Datentyp WSTRING referenziert, muss die am Parameter LEN angegebene Länge 0 oder  $\geq 5$  sein.

## Parameter BUSY, NDR und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, NDR, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter NDR kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von TRCV auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, NDR und ERROR:

| BUSY | NDR | ERROR | Beschreibung   |
|------|-----|-------|--|
| 1    | -   | -     | Der Auftrag wird bearbeitet.   |
| 0    | 1   | 0     | Der Auftrag wurde erfolgreich durchgeführt.  |
| 0    | 0   | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben. |
| 0    | 0   | 0     | Kein neuer Auftrag wurde zugewiesen.   |

### Hinweis

Aufgrund der asynchronen Bearbeitung von "TRCV" sind die Daten im Empfangsbereich nur dann konsistent, wenn der Parameter NDR auf den Wert "1" gesetzt wird.

**Parameter ERROR und STATUS**

| ERROR  | STATUS* (W#16#...) | Erläuterung  |
|--|--------------------|--|
| 0  | 0000               | Auftrag erfolgreich ausgeführt. Die aktuelle Länge der empfangenen Daten wird am Parameter RCVD_LEN ausgegeben.  |
| 0  | 7000               | Baustein nicht für den Empfang bereit.   |
| 0  | 7001               | Baustein ist für den Empfang bereit, Empfangsauftrag wurde aktiviert.  |
| 0  | 7002               | Zwischenaufwurf, Empfangsauftrag wird bearbeitet.<br>Hinweis: Während dieser Bearbeitung werden Daten in den Empfangsbereich geschrieben. Währenddessen kann der Zugriff auf den Empfangsbereich inkonsistente Daten liefern.  |
| 1  | 8085               | <ul style="list-style-type: none"> <li>Parameter LEN ist größer als der größte zulässige Wert.</li> <li>Der Wert des Parameters LEN oder DATA wurde nach dem ersten Aufruf geändert.</li> <li>Beide LEN-Parameter und der Parameter DATA haben den Wert "0" oder LEN ist länger als der maximal zulässige Wert (65536).</li> </ul> |
| 1  | 8086               | Der Parameter ID liegt außerhalb des zulässigen Adressbereichs (1 .. 0x0FFF).  |
| 1  | 8088               | <ul style="list-style-type: none"> <li>Empfangsbereich ist zu klein.</li> <li>Der Wert am Parameter LEN ist größer als der am Parameter DATA angegebene Empfangsbereich.</li> </ul>  |
| 1  | 80A1               | Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die angegebene Verbindung wurde noch nicht aufgebaut.</li> <li>Die angegebene Verbindung wird gerade beendet. Empfangsauftrag über diese Verbindung ist nicht möglich.</li> <li>Die Verbindung wird gerade neu initialisiert.</li> </ul>                              |
| 1  | 80B3               | Die parametrisierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie bei einer UDP-Verbindung die Anweisung "TURCV".  |
| 1  | 80C3               | <ul style="list-style-type: none"> <li>Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.</li> <li>Interner Mangel an Ressourcen.</li> </ul>  |
| 1  | 80C4               | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die Verbindung zum Partner kann derzeit nicht aufgebaut werden.</li> <li>Die Schnittstelle empfängt neue Parametereinstellungen oder die Verbindung wird aufgebaut.</li> </ul>   |
| 1  | 80C5               | Der remote Partner hat die Verbindung abgebaut.  |
| 1  | 80C6               | Der remote Partner kann nicht erreicht werden (Netzwerkfehler).  |
| 1  | 80C7               | Zeitüberschreitung der Ausführung.   |
| 1  | 80C9               | Die Länge des Empfangsbereichs ist kleiner als die Länge der gesendeten Daten.   |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                    |  |

**TRCV: Daten über Kommunikationsverbindung empfangen**

**Beschreibung**

Die folgende Beschreibung der Anweisung "TRCV" ist gültig für die CPU S7-1500 und S7-1200 V4.0.

Mit der Anweisung "TRCV" empfangen Sie Daten über eine bestehende Kommunikationsverbindung. "TRCV" wird asynchron ausgeführt.

Wenn der Parameter EN\_R auf den Wert "1" gesetzt wird, wird der Datenempfang aktiviert. Die empfangenen Daten werden in einen Empfangsbereich eingetragen. Die Länge des Empfangsbereichs legen Sie abhängig von der verwendeten Protokollvariante entweder durch den Parameter LEN (falls LEN <> 0) oder die Längenangabe des Parameters DATA (falls LEN = 0) fest.

Während Daten empfangen werden, dürfen an dem Parameter DATA und dem definierten Empfangsbereich keine Änderungen vorgenommen werden, um die Konsistenz der empfangenen Daten sicherzustellen.

Nach einem erfolgreichen Datenempfang wird der Parameter NDR auf den Wert "1" gesetzt. Die tatsächlich empfangene Datenmenge können Sie am Parameter RCVD\_LEN abfragen.

### Empfangsmodi von "TRCV"

Die folgende Tabelle zeigt, wie die empfangenen Daten in den Empfangsbereich eingetragen werden.

| Protokollvariante                                       | Parameter ADHOC        | Verfügbarkeit der Daten im Empfangsbereich   | Parameter connection_type der Verbindungsbeschreibung | Parameter LEN   |
|---|------------------------|--|---|---|
| TCP<br>(Ad-hoc-Modus)                                   | 1 (Ad-hoc aktiviert)   | Die Daten sind sofort verfügbar.   | Hexadezimalwert:<br>B#16#11<br>Integerwert: 17        | 1 bis maximale Länge (CPU-abhängig)   |
| TCP<br>(Datenempfang mit angegebener Länge)             | 0 (Ad-hoc deaktiviert) | Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde. | Hexadezimalwert:<br>B#16#11<br>Integerwert: 17        | 1 bis 8192  |
| ISO on TCP<br>(Nachrichtenorientierte Datenübertragung) | -                      | Die Daten sind verfügbar, sobald die am Parameter LEN angegebene Datenlänge vollständig empfangen wurde. | Hexadezimalwert:<br>B#16#12<br>Integerwert: 18        | <ul style="list-style-type: none"> <li>• 1 bis 1452, wenn ein CP verwendet wird.</li> <li>• 1 bis 8192, wenn kein CP verwendet wird.</li> </ul> |

### TCP (Ad-hoc-Modus)

Den Ad-hoc-Modus gibt es nur bei der Protokollvariante TCP. Den Ad-hoc-Modus verwenden Sie, um mit der Anweisung "TRCV" Daten mit dynamischen Längen zu empfangen.

Den Ad-hoc-Modus stellen Sie ein, indem Sie dem Parameter ADHOC den Wert "1" zuweisen. Bei Verwendung des Ad-hoc-Modus können für Datenbausteine mit Standardzugriff alle Datentypen verwendet werden. Für Datenbausteine mit optimiertem Zugriff können als Datentypen nur ARRAY of BYTE oder Datentypen mit einer Länge von 8 Bit verwendet werden (z. B. CHAR, USINT, SINT, etc.). Ist der Ad-hoc-Modus aktiviert, wird der Datenempfang schon nach einem empfangenen Byte am Parameter NDR gemeldet.

### TCP (Datenempfang mit angegebener Länge)

Für einen Datenempfang mit angegebener Länge weisen Sie dem Parameter ADHOC den Wert "0" zu. Ist der Ad-hoc-Modus deaktiviert, wird der Datenempfang erst abgeschlossen, wenn die am Parameter LEN angegebene Länge der Daten vollständig empfangen wurde. Erst danach sind die Daten im Empfangsbereich (Parameter DATA) verfügbar. Der erfolgreiche Datenempfang wird durch den Ausgangsparameter NDR gemeldet. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD\_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

### ISO on TCP (Nachrichtenorientiert Datenübertragung)

Bei der Protokollvariante ISO on TCP werden über eine Verbindung abgeschlossene Nachrichtenblöcke gesendet, die vom Empfänger auch als solche erkannt werden. Bei Verwendung von ISO on TCP meldet "TRCV" den Datenempfang, sobald der Nachrichtenblock vollständig empfangen wurde. Der Empfangsbereich wird von den Parametern LEN und DATA definiert. Ist der Empfangspuffer (Parameter DATA) zu klein für die gesendeten Daten, meldet "TRCV" einen Fehler. Der erfolgreiche Datenempfang wird durch den Ausgangsparameter NDR gemeldet. Die tatsächlich empfangene Datenlänge in Bytes am Parameter RCVD\_LEN entspricht nach dem Empfang der Datenlänge am Parameter LEN.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TRCV":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung  |
|-----------|-------------|-----------------|------------------------------|---|
| EN_R      | Input       | BOOL            | E, A, M, D, L oder Konstante | Empfangsfreigabe  |
| ID        | Input       | CONN_OUC (WORD) | D, L oder Konstante          | Referenz auf die mit "TCON" aufgebaute Verbindung.<br>Wertebereich: W#16#0001 bis W#16#0FFF   |
| LEN       | Input       | UDINT           | E, A, M, D, L oder Konstante | Länge des Empfangsbereichs in Bytes (ausgeblendet).<br>Wenn Sie am Parameter DATA einen Empfangsbereich mit optimiertem Zugriff verwenden, muss der Parameter LEN den Wert "0" haben. |
| ADHOC     | Input       | BOOL            | E, A, M, D, L oder Konstante | Ad-hoc Modus für die Protokollvariante TCP verwenden (ausgeblendet).  |
| DATA      | InOut       | VARIANT         | E, A, M, D                   | Zeiger auf den Empfangsbereich  |
| NDR       | Output      | BOOL            | E, A, M, D, L                | Statusparameter (New Data Received): <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder noch in Bearbeitung</li> <li>• 1: Neue Daten empfangen</li> </ul>  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|-----------|-------------|----------|-----------------|--|
| BUSY      | Output      | BOOL     | E, A, M, D, L   | Statusparameter mit folgenden Werten: <ul style="list-style-type: none"> <li>• 0: Auftrag noch nicht gestartet oder bereits beendet</li> <li>• 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden</li> </ul> |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Statusparameter ERROR: <ul style="list-style-type: none"> <li>• 0: Kein Fehler</li> <li>• 1: Fehler aufgetreten</li> </ul>   |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Status der Anweisung   |
| RCVD_LEN  | Output      | UDINT    | E, A, M, D, L   | Tatsächlich empfangene Datenmenge in Bytes   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter LEN, DATA und RCVD\_LEN

- Ist LEN = 0, werden die empfangenen Daten in dem am Parameter DATA angegebenen Empfangsbereich gespeichert. Die Anzahl der empfangenen Bytes werden am Parameter RCVD\_LEN angezeigt.
- Ist die am Parameter LEN angegebene Länge größer als die Länge der am Parameter DATA empfangenen Daten, wird der Fehlercode 8088 am Parameter STATUS ausgegeben (siehe im Folgenden Beschreibung des Parameters STATUS).
- Wird über den Parameter DATA eine Struktur (Struct) referenziert, kann LEN kürzer sein als die Struktur. In diesem Fall werden nur die Daten bis zur Länge des Parameters LEN übertragen.
- Verweist der Parameter DATA auf einen Datenbaustein mit optimiertem Zugriff, muss der Parameter LEN auf "0" gesetzt werden. Stimmt die Länge der Daten bei elementaren Datentypen nicht überein, werden die Daten nicht empfangen und der Fehlercode 8088 am Parameter STATUS ausgegeben.
- Wird über den Parameter DATA ein Datentyp STRING referenziert, muss die am Parameter LEN angegebene Länge 0 oder  $\geq 2$  sein (LEN = 1 ist nicht erlaubt).
- Wird über den Parameter DATA ein Datentyp WSTRING referenziert, muss die am Parameter LEN angegebene Länge 0 oder  $\geq 5$  sein.

### Parameter BUSY, NDR und ERROR

Den Ausführungsstatus kontrollieren Sie über die Parameter BUSY, NDR, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter NDR kontrollieren Sie, ob ein Auftrag erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von TRCV auftreten. Die Fehlerinformationen werden am Parameter STATUS ausgegeben.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, NDR und ERROR:

| BUSY | NDR | ERROR | Beschreibung   |
|------|-----|-------|--|
| 1    | -   | -     | Der Auftrag wird bearbeitet.   |
| 0    | 1   | 0     | Der Auftrag wurde erfolgreich durchgeführt.  |
| 0    | 0   | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben. |
| 0    | 0   | 0     | Kein neuer Auftrag wurde zugewiesen.   |

**Hinweis**

Aufgrund der asynchronen Bearbeitung von "TRCV" sind die Daten im Empfangsbereich nur dann konsistent, wenn der Parameter NDR auf den Wert "1" gesetzt wird.

**Parameter ERROR und STATUS**

| ERROR | STATUS* (W#16#...) | Erläuterung  |
|-------|--------------------|--|
| 0     | 0000               | Auftrag erfolgreich ausgeführt. Die aktuelle Länge der empfangenen Daten wird am Parameter RCVD_LEN ausgegeben.  |
| 0     | 7000               | Baustein nicht für den Empfang bereit.   |
| 0     | 7001               | Baustein ist für den Empfang bereit, Empfangsauftrag wurde aktiviert.  |
| 0     | 7002               | Zwischenaufruf, Empfangsauftrag wird bearbeitet.<br>Hinweis: Während dieser Bearbeitung werden Daten in den Empfangsbereich geschrieben. Währenddessen kann der Zugriff auf den Empfangsbereich inkonsistente Daten liefern.   |
| 1     | 8085               | <ul style="list-style-type: none"> <li>Parameter LEN ist größer als der größte zulässige Wert.</li> <li>Der Wert des Parameters LEN oder DATA wurde nach dem ersten Aufruf geändert.</li> <li>Beide LEN-Parameter und der Parameter DATA haben den Wert "0" oder LEN ist länger als der maximal zulässige Wert (65536).</li> </ul> |
| 1     | 8086               | Der Parameter ID liegt außerhalb des zulässigen Wertebereichs (1 .. 0x0FFF).   |
| 1     | 8088               | <ul style="list-style-type: none"> <li>Empfangsbereich ist zu klein.</li> <li>Der Wert am Parameter LEN ist größer als der am Parameter DATA angegebene Empfangsbereich.</li> </ul>  |
| 1     | 80A1               | Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die angegebene Verbindung wurde noch nicht aufgebaut.</li> <li>Die angegebene Verbindung wird gerade beendet. Empfangsauftrag über diese Verbindung ist nicht möglich.</li> <li>Die Verbindung wird gerade neu initialisiert.</li> </ul>                              |
| 1     | 80B3               | Die parametrisierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie bei einer UDP-Verbindung die Anweisung "TURCV".  |
| 1     | 80C3               | <ul style="list-style-type: none"> <li>Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsgruppe bearbeitet.</li> <li>Interner Mangel an Ressourcen.</li> </ul>  |
| 1     | 80C4               | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die Verbindung zum Partner kann derzeit nicht aufgebaut werden.</li> <li>Die Schnittstelle empfängt neue Parametereinstellungen oder die Verbindung wird aufgebaut.</li> </ul>   |
| 1     | 80C5               | Der remote Partner hat die Verbindung abgebaut.  |
| 1     | 80C6               | Der remote Partner kann nicht erreicht werden (Netzwerkfehler).  |

| ERROR   | STATUS*<br>(W#16#...) | Erläuterung  |
|---|-----------------------|--|
| 1   | 80C7                  | Zeitüberschreitung der Ausführung.   |
| 1   | 80C9                  | Die Länge des Empfangsbereichs ist kleiner als die Länge der gesendeten Daten. |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                       |  |

## Siehe auch

TRCV: Daten über Kommunikationsverbindung empfangen (Seite 2984)

TCON: Kommunikationsverbindung aufbauen (Seite 2971)

## Struktur der Adressinformation des remoten Partners bei UDP

### Übersicht

- Bei "TUSEND (Seite 2994)" übergeben Sie am Parameter ADDR die Adressinformation des Empfängers. Diese Adressinformation muss die im Folgenden angegebene Struktur haben.
- Bei "TURCV (Seite 2997)" erhalten Sie am Parameter ADDR die Adresse des Absenders der empfangenen Daten. Diese Adressinformation muss die im Folgenden angegebene Struktur haben.

## Datenbaustein für die Adressinformation des remoten Partners

Sie müssen einen DB anlegen, der eine oder mehrere Datenstrukturen gemäß dem PLC-Datentyp "TADDR\_PARAM" enthält.

Im Parameter ADDR von "TUSEND (Seite 2994)" übergeben Sie und am Parameter ADDR von "TURCV (Seite 2997)" erhalten Sie einen Zeiger vom Typ VARIANT auf die Adresse des zugehörigen remoten Partners (z. B. P#DB100.DBX0.0 USINT 8).


### Aufbau der Adressinformation des remoten Partners nach "TADDR\_PARAM"

| Byte    | Parameter   | Datentyp              | Startwert   | Beschreibung   |
|---------|-------------|-----------------------|-------------|--|
| 0 bis 3 | rem_ip_addr | ARRAY [1..4] of USINT | B#16#00 ... | IP-Adresse des remoten Partners, z. B. 192.168.002.003: <ul style="list-style-type: none"> <li>• rem_ip_addr[1] = B#16#C0 (192)</li> <li>• rem_ip_addr[2] = B#16#A8 (168)</li> <li>• rem_ip_addr[3] = B#16#02 (002)</li> <li>• rem_ip_addr[4] = B#16#03 (003)</li> </ul> |
| 4 bis 5 | rem_port_nr | UINT                  | B#16#00 ... | remote Port-Nr. (mögliche Werte siehe: Auto-Hotspot): <ul style="list-style-type: none"> <li>• rem_port_nr[1] = high byte der Port-Nr. in hexadezimaler Darstellung</li> <li>• rem_port_nr[2] = low byte der Port-Nr. in hexadezimaler Darstellung</li> </ul>            |
| 6 bis 7 | reserved    | WORD                  | B#16#00 ... | Nicht verwendet. Belegen Sie diesen Parameter mit "0".   |

### TUSEND: Daten über Ethernet (UDP) senden

#### Beschreibung

Die Anweisung "TUSEND" sendet Daten über UDP an den über den Parameter ADDR adressierten remoten Partner.

|  |
|--|
|  <b>WARNUNG</b>   |
| <b>Datenübertragung über UDP</b>   |
| Bei der Datenübertragung über UDP gemäß RFC 768 werden die Daten unquittiert und damit ungesichert zum remoten Partner übertragen. D. h., es können Daten verloren gehen, ohne dass dies am Baustein angezeigt wird. |

#### Hinweis

Bei aufeinander folgenden Sendevorgängen an verschiedene Partner müssen Sie bei den Aufrufen von "TUSEND" lediglich den Parameter ADDR anpassen. Der erneute Aufruf der Anweisungen "TCON (Seite 2968)" und "TDISCON (Seite 2976)" hingegen entfällt.

#### Arbeitsweise

"TUSEND" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Sendevorgang, indem Sie "TUSEND" mit REQ = 1 aufrufen.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET\_VAL der asynchron arbeitenden Anweisungen.

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).



In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich "TUSEND" aktuell befindet bzw. wann der Sendevorgang beendet ist.

| BUSY  | DONE       | ERROR      | Beschreibung   |
|-------|------------|------------|--|
| TRUE  | irrelevant | irrelevant | Der Auftrag ist in Bearbeitung.  |
| FALSE | TRUE       | FALSE      | Der Auftrag wurde erfolgreich abgeschlossen.   |
| FALSE | FALSE      | TRUE       | Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen. |
| FALSE | FALSE      | FALSE      | Der Anweisung wurde kein (neuer) Auftrag erteilt.  |

#### Hinweis

Durch die asynchrone Arbeitsweise von "TUSEND" müssen Sie die Daten im Sendebereich so lange konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert TRUE annimmt.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TUSEND":

| Parameter | Deklaration | Datentyp | Speicherbereich              |                              | Beschreibung   |
|-----------|-------------|----------|------------------------------|------------------------------|--|
|           |             |          | S7-1200                      | S7-1500                      |  |
| REQ       | Input       | BOOL     | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Steuerparameter REQUEST, startet den Sendeauftrag bei steigender Flanke.<br>Die Daten werden aus dem mit DATA und LEN angegebenen Bereich übergeben.   |
| ID        | Input       | WORD     | M, D oder Konstante          | M, D oder Konstante          | Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung.<br>Wertebereich: W#16#0001 bis W#16#0FFF |
| LEN       | Input       | UINT     | E, A, M, D, L                | E, A, M, D, L                | Anzahl der Bytes, die mit dem Auftrag gesendet werden sollen<br>Wertebereich: 1 bis 1472   |
| DONE      | Output      | BOOL     | E, A, M, D, L                | E, A, M, D, L                | Zustandsparameter DONE: <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>• 1: Auftrag wurde fehlerfrei ausgeführt</li> </ul>   |
| BUSY      | Output      | BOOL     | E, A, M, D, L                | E, A, M, D, L                | <ul style="list-style-type: none"> <li>• BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden.</li> <li>• BUSY = 0: Der Auftrag ist beendet.</li> </ul>  |

| Parameter | Deklaration | Datentyp        | Speicherbereich |               | Beschreibung   |
|-----------|-------------|-----------------|-----------------|---------------|--|
|           |             |                 | S7-1200         | S7-1500       |  |
| ERROR     | Output      | BOOL            | E, A, M, D, L   | E, A, M, D, L | Zustandsparameter ERROR: <ul style="list-style-type: none"> <li>• ERROR = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers</li> </ul>  |
| STATUS    | Output      | WORD            | M, D            | M, D          | Zustandsparameter STATUS: Fehlerinformation  |
| DATA      | InOut       | VARIANT         | E, A, M, D      | E, A, M, D    | Sendebereich, enthält Adresse und Länge<br>Die Adresse verweist auf: <ul style="list-style-type: none"> <li>• das Prozessabbild der Eingänge</li> <li>• das Prozessabbild der Ausgänge</li> <li>• einen Merker</li> <li>• einen Datenbaustein</li> </ul> |
| ADDR      | InOut       | TADDR_P<br>aram | D               | D             | Zeiger auf die Adresse des Empfängers (z. B. P#DB100.DBX0.0 USINT 8)<br>Siehe auch: Struktur der Adressinformation des remoten Partners bei UDP (Seite 2993)   |

**Hinweis**

Erstellen Sie den Datenbaustein für den Parameter ADDR über den Dialog "Neuen Baustein hinzufügen", indem Sie den Type "TADDR\_Param" auswählen.

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

**Parameter ERROR und STATUS**


| ERROR | STATUS* (W#16#...) | Erläuterung   |
|-------|--------------------|---|
| 0     | 0000               | Sendeauftrag wurde ohne Fehler abgeschlossen  |
| 0     | 7000               | Keine Auftragsbearbeitung aktiv   |
| 0     | 7001               | Start der Auftragsbearbeitung, Daten werden versendet<br>Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendebereich DATA zu.        |
| 0     | 7002               | Zwischenaufwurf (REQ irrelevant), Auftrag ist in Bearbeitung<br>Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendebereich DATA zu. |
| 1     | 8085               | Parameter LEN hat den Wert "0" oder ist größer als der größte zulässige Wert.   |
| 1     | 8086               | Parameter ID liegt in einem unzulässigen Wertebereich.  |
| 0     | 8088               | Parameter LEN ist größer als der in DATA angegebene Speicherbereich.  |
| 1     | 8089               | Parameter ADDR zeigt nicht auf einen Datenbaustein mit der Struktur TADDR_Param.  |

| ERROR   | STATUS*<br>(W#16#...)        | Erläuterung  |
|---|------------------------------|--|
| 1   | 80A1                         | Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wurde noch nicht aufgebaut.</li> <li>Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Sendevorgang über diese Verbindung ist nicht möglich.</li> <li>Schnittstelle wird neu initialisiert</li> </ul> |
| 1   | 80A4                         | IP-Adresse (am Parameter ADDR) des remoten Verbindungsendpunkts ist ungültig, evtl. stimmt sie mit der eigenen IP-Adresse überein.   |
| 1   | 80B3                         | <ul style="list-style-type: none"> <li>Die parametrisierte Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie "TSEND (Seite 2981)".</li> <li>Parameter ADDR: ungültige Angaben für Port-Nr.</li> </ul>  |
| 1   | 80C3                         | <ul style="list-style-type: none"> <li>Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet.</li> <li>interner Ressourcenmangel</li> </ul>   |
| 1   | 80C4                         | Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>Die Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems kann momentan nicht aufgebaut werden.</li> <li>Die Schnittstelle wird neu parametrisiert.</li> </ul>  |
| 1   | 80C5                         | Der remote Partner hat die Verbindung abgebaut.  |
| 1   | 80C6                         | Der remote Partner kann nicht erreicht werden (Netzwerkfehler).  |
| 1   | 80C7                         | Zeitüberschreitung der Ausführung.   |
| -   | allgemeine Fehlerinformation | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |                              |  |

## TURCV: Daten über Ethernet (UDP) empfangen

### Beschreibung

Die Anweisung "TURCV" empfängt Daten über UDP. Nach erfolgreichem Abschluss von "TURCV" wird Ihnen am Parameter ADDR die Adresse des remoten Partners, also des Senders, zur Verfügung gestellt.

|  |
|--|
|  <b>WARNUNG</b>   |
| <b>Ungesicherte Datenübertragung</b>   |
| Bei der Datenübertragung über UDP gemäß RFC 768 werden die Daten unquittiert und damit ungesichert zum remoten Partner übertragen. D. h., es können Daten verloren gehen, ohne dass dies am Baustein angezeigt wird. |

### Arbeitsweise

"TURCV" ist eine asynchron arbeitende Anweisung, d. h. die Auftragsbearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Empfangsauftrag, indem Sie "TURCV" mit EN\_R = 1 aufrufen.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt. Dabei entspricht STATUS dem Ausgangsparameter RET\_VAL der asynchron arbeitenden Anweisungen

Siehe auch: Bedeutung von REQ, RET\_VAL und BUSY bei asynchron arbeitenden Anweisungen (Seite 1620).

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, NDR und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich TURCV aktuell befindet bzw. wann der Empfangsvorgang beendet ist.

| BUSY  | NDR        | ERROR      | Beschreibung   |
|-------|------------|------------|--|
| TRUE  | irrelevant | irrelevant | Der Auftrag ist in Bearbeitung.  |
| FALSE | TRUE       | FALSE      | Der Auftrag wurde erfolgreich abgeschlossen.   |
| FALSE | FALSE      | TRUE       | Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen. |
| FALSE | FALSE      | FALSE      | Der Anweisung wurde kein (neuer) Auftrag erteilt.  |

#### Hinweis

Durch die asynchrone Arbeitsweise von "TURCV" sind die Daten im Empfangsbereich erst dann konsistent, wenn der Parameter NDR den Wert TRUE annimmt.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TURCV":

| Parameter | Deklaration | Datentyp | Speicherbereich              |                              | Beschreibung  |
|-----------|-------------|----------|------------------------------|------------------------------|---|
|           |             |          | S7-1200                      | S7-1500                      |   |
| EN_R      | Input       | BOOL     | E, A, M, D, L oder Konstante | E, A, M, D, L oder Konstante | Steuerparameter enabled to receive: Mit EN_R = 1 wird "TURCV" empfangsbereit. Der Empfangsauftrag wird bearbeitet.  |
| ID        | Input       | WORD     | M, D oder Konstante          | M, D oder Konstante          | Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF |
| LEN       | Input       | UINT     | E, A, M, D, L                | E, A, M, D, L                | Länge des Empfangsbereichs in Bytes: 0 (empfohlen) bzw. 1 bis 1472  |

| Parameter | Deklaration | Datentyp    | Speicherbereich |               | Beschreibung   |
|-----------|-------------|-------------|-----------------|---------------|--|
|           |             |             | S7-1200         | S7-1500       |  |
| NDR       | Output      | BOOL        | E, A, M, D, L   | E, A, M, D, L | Zustandsparameter NDR: <ul style="list-style-type: none"> <li>NDR = 0: Auftrag wurde noch nicht gestartet oder läuft noch</li> <li>NDR = 1: Auftrag wurde erfolgreich abgeschlossen</li> </ul>                           |
| ERROR     | Output      | BOOL        | E, A, M, D, L   | E, A, M, D, L | Zustandsparameter ERROR: <ul style="list-style-type: none"> <li>ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers</li> </ul>                        |
| BUSY      | Output      | BOOL        | E, A, M, D, L   | E, A, M, D, L | <ul style="list-style-type: none"> <li>BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden.</li> <li>BUSY = 0: Der Auftrag ist beendet.</li> </ul>                                |
| STATUS    | Output      | WORD        | M, D            | M, D          | Zustandsparameter STATUS:<br>Fehlerinformation   |
| RCVD_LEN  | Output      | UINT        | E, A, M, D, L   | E, A, M, D, L | Anzahl der tatsächlich empfangenen Daten in Bytes  |
| DATA      | InOut       | VARIANT     | E, A, M, D      | E, A, M, D    | Empfangsbereich<br>Die Adresse verweist auf: <ul style="list-style-type: none"> <li>das Prozessabbild der Eingänge</li> <li>das Prozessabbild der Ausgänge</li> <li>einen Merker</li> <li>einen Datenbaustein</li> </ul> |
| ADDR      | InOut       | TADDR_Param | D               | D             | Zeiger auf die Adresse des Senders (z. B. P#DB100.DBX0.0 Byte 8) Siehe auch: Struktur der Adressinformation des remoten Partners bei UDP (Seite 2993)  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

| ERROR | STATUS* (W#16#...) | Erläuterung  |
|-------|--------------------|--|
| 0     | 0000               | Neue Daten wurden übernommen. Die aktuelle Länge der empfangenen Daten wird in RCVD_LEN angezeigt.   |
| 0     | 7000               | Baustein nicht empfangsbereit  |
| 0     | 7001               | Baustein ist empfangsbereit, Empfangsauftrag wurde aktiviert   |
| 0     | 7002               | Zwischenaufwurf, Empfangsauftrag ist in Bearbeitung<br>Hinweis: In dieser Bearbeitungsphase schreibt "TURCV" Daten in den Empfangsbereich. Deshalb können im Fehlerfall inkonsistente Daten im Empfangsbereich stehen. |
| 1     | 8085               | Parameter LEN ist größer als der größte zulässige Wert, oder Sie haben einen der Parameter LEN oder DATA gegenüber dem Erstaufruf geändert   |
| 1     | 8086               | Parameter ID liegt in einem unzulässigen Wertebereich  |

| ERROR | STATUS*<br>(W#16#...)        | Erläuterung  |
|-------|------------------------------|--|
| 1     | 8088                         | <ul style="list-style-type: none"> <li>• Empfangsbereich ist zu klein</li> <li>• Wert in LEN ist größer als der durch DATA vorgegebene Empfangsbereich</li> </ul>  |
| 1     | 8089                         | Parameter ADDR zeigt nicht auf einen Datenbaustein mit der Struktur TADDR_Param.   |
| 1     | 80A1                         | Kommunikationsfehler: <ul style="list-style-type: none"> <li>• Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wurde noch nicht aufgebaut.</li> <li>• Die angegebene Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Empfangsauftrag über diese Verbindung ist nicht möglich.</li> <li>• Die Schnittstelle wird neu parametrier.</li> </ul> |
| 1     | 80A4                         | IP-Adresse (am Parameter ADDR) des remoten Verbindungsendpunkts ist ungültig, evtl. stimmt sie mit der eigenen IP-Adresse überein.   |
| 1     | 80B3                         | Die parametrierete Protokollvariante (Parameter connection_type in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie "TRCV (Seite 2984)".  |
| 1     | 80B7                         | Die Länge am Parameter ADDR entspricht nicht 8 Byte.   |
| 1     | 80C3                         | <ul style="list-style-type: none"> <li>• Ein Baustein mit dieser ID wird bereits in einer anderen Prioritätsklasse bearbeitet.</li> <li>• interner Ressourcenmangel</li> </ul>   |
| 1     | 80C4                         | Temporärer Kommunikationsfehler: Die Schnittstelle wird neu parametrier.   |
| 1     | 80C5                         | Der remote Partner hat die Verbindung abgebaut.  |
| 1     | 80C6                         | Der remote Partner kann nicht erreicht werden (Netzwerkfehler).  |
| 1     | 80C7                         | Zeitüberschreitung der Ausführung.   |
| 1     | 80C9                         | Bei RFC1006 / UDP: Die empfangenen Daten sind länger als erwartet (Größe des Empfangspuffers überschritten).   |
| -     | allgemeine Fehlerinformation | Siehe auch: GET_ERR_ID: Fehler-ID lokal abfragen (Seite 1832)  |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

**Siehe auch**

TCON: Kommunikationsverbindung aufbauen (Seite 2968)

TDISCON: Kommunikationsverbindung abbauen (Seite 2976)

**T\_RESET: Verbindung zurücksetzen**

**Beschreibung**

Die Anweisung "T\_RESET" baut eine bestehende Verbindung ab und wieder auf.

Die lokalen Endpunkte der Verbindung bleiben dabei erhalten. Diese werden automatisch erzeugt:

- Wenn eine Verbindung konfiguriert und in die CPU geladen wurde.
- Wenn eine Verbindung durch das Anwenderprogramm erzeugt wurde, beispielsweise durch den Aufruf der Anweisung "TCON (Seite 2968)".

Die Anweisung "T\_RESET" lässt sich für alle Verbindungsarten (TCP, UDP, ISO-on-TCP,...etc.) durchführen. Hierbei spielt es keine Rolle, ob für die Verbindung die lokale Schnittstelle der CPU oder die Schnittstelle eines CMs/CPs verwendet wurde.

Nach Aufruf der Anweisung "T\_RESET" über den Parameter REQ wird die über den Parameter ID angegebene Verbindung abgebaut und ggf. die Puffer zum Senden und Empfangen von Daten geleert. Mit der Unterbrechung der Verbindung werden auch laufende Datenübertragungen unterbrochen. Dabei besteht die Gefahr, dass bei einer laufenden Übertragung Daten verloren gehen können. Danach versucht die als aktiver Verbindungspartner definierte CPU automatisch, die unterbrochene Kommunikationsverbindung wieder herzustellen. Der Aufruf der Anweisung "TCON (Seite 2968)" zum erneuten Aufbauen der Kommunikationsverbindung ist deshalb nicht notwendig.

Über die Ausgangsparameter DONE, BUSY und STATUS wird der Zustand des Auftrags angezeigt.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T\_RESET":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------|------------------------------|--|
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | Steuerparameter REQUEST, startet den Auftrag zum Abbau der durch ID gegebenen Verbindung. Der Auftragsstart erfolgt bei steigender Flanke.   |
| ID        | Input       | CONN_OUC (WORD) | L, D oder Konstante          | Referenz auf die abzubauenende Verbindung zum passiven Partner. ID muss identisch sein mit dem zugehörigen Parameter ID in der lokalen Verbindungsbeschreibung.<br>Wertebereich: W#16#0001 bis W#16#0FFF |
| DONE      | Output      | BOOL            | E, A, M, D, L                | Zustandsparameter DONE <ul style="list-style-type: none"> <li>• 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>• 1: Auftrag wurde fehlerfrei ausgeführt</li> </ul>            |
| BUSY      | Output      | BOOL            | E, A, M, D, L                | Zustandsparameter BUSY <ul style="list-style-type: none"> <li>• 0: Der Auftrag ist beendet.</li> <li>• 1: Der Auftrag ist noch nicht beendet.</li> </ul>   |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|-----------|-------------|----------|-----------------|---|
| ERROR     | Output      | BOOL     | E, A, M, D, L   | Zustandsparameter ERROR <ul style="list-style-type: none"> <li>• 0: Kein Fehler aufgetreten.</li> <li>• 1: Bei der Bearbeitung ist ein Fehler aufgetreten. Der Parameter STATUS liefert detaillierte Auskunft über die Art des Fehlers</li> </ul> |
| STATUS    | Output      | WORD     | E, A, M, D, L   | Zustandsparameter STATUS<br>Fehlerinformation (siehe Tabelle "Parameter STATUS").   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter STATUS

| STATUS* (W#16#...) | Erläuterung  |
|--------------------|--|
| 0000               | Kein Fehler.   |
| 0001               | Verbindung ist nicht aufgebaut.                      |
| 7001               | Unterbrechung der Verbindung wurde eingeleitet.      |
| 7002               | Unterbrechung der Verbindung wird durchgeführt.      |
| 8081               | Angegebene Verbindung am Parameter ID ist unbekannt. |

\* Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch".

### T\_DIAG: Verbindung überprüfen

#### Beschreibung

Mit der Anweisung "T\_DIAG" überprüfen Sie den Zustand einer Verbindung und lesen weitere Informationen zum lokalen Endpunkt dieser Verbindung.

- Die Verbindung wird referenziert über den Parameter ID. Es können sowohl im Verbindungseditor projektierte, als auch programmierte Verbindungsendpunkten (z. B. mit der Anweisung "TCON") gelesen werden.  
Temporäre Verbindungsendpunkte (die z. B. beim Verbinden einer Engineering-Station entstehen) können nicht diagnostiziert werden, da hierbei auch keine Verbindungs-ID erstellt wird.
- Die gelesenen Verbindungsinformationen werden in einer Struktur abgelegt, die über den Parameter RESULT referenziert ist.
- Der Ausgangsparameter STATUS zeigt an, ob die Verbindungsinformationen gelesen werden konnten. Nur wenn die Anweisung "T\_DIAG" mit STATUS = W#16#0000 und ERROR = FALSE beendet wurde, sind die Verbindungsinformationen in der Struktur am Parameter RESULT gültig.  
Im Fehlerfall können die Verbindungsinformationen nicht ausgewertet werden.



## Mögliche Verbindungsinformationen

Zum Lesen der Verbindungsinformationen können am Parameter RESULT zwei unterschiedliche Strukturen verwendet werden:

- Die Struktur "TDiag\_Status" enthält nur die wichtigsten Informationen zu einem Verbindungsendpunkt, wie das verwendete Protokoll, der Verbindungszustand oder die Anzahl der gesendeten oder empfangenen Daten.
- Die Struktur "TDiag\_StatusExt" liefert neben den wichtigsten Informationen auch die Anzahl der Verbindungsaufbauversuche, den Grund für einen möglichen Verbindungsabbruch, usw.

Der Aufbau und die Parameter der beiden Strukturen sind weiter unten beschrieben (siehe Tabelle "Strukturen TDIAG\_Status und TDIAG\_StatusExt").

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T\_DIAG":

| Parameter | Deklaration | Datentyp        | Speicherbereich              | Beschreibung   |
|-----------|-------------|-----------------|------------------------------|--|
| REQ       | Input       | BOOL            | E, A, M, D, L oder Konstante | Startet bei einer positiven Flanke die Anweisung zum Überprüfen der im Parameter ID angegebenen Verbindung.  |
| ID        | Input       | CONN_OUC (WORD) | L, D oder Konstante          | Referenz auf die zugewiesene Verbindung. Wertebereich: W#16#0001 bis W#16#0FFF   |
| RESULT    | InOut       | VARIANT         | D                            | Zeiger auf die Struktur, in der die Verbindungsinformationen abgelegt werden. An dem Parameter RESULT können die Strukturen TDiag_Status oder TDiag_StatusExt verwendet werden (zur Beschreibung siehe Tabelle "Strukturen TDIAG_Status und TDIAG_StatusExt"). |
| DONE      | Output      | BOOL            | E, A, M, D, L                | Statusparameter: <ul style="list-style-type: none"> <li>• 0: Anweisung noch nicht gestartet oder noch in Bearbeitung.</li> <li>• 1: Anweisung fehlerfrei ausgeführt.</li> </ul>  |
| BUSY      | Output      | BOOL            | E, A, M, D, L                | Statusparameter: <ul style="list-style-type: none"> <li>• 0: Anweisung noch nicht gestartet oder bereits beendet.</li> <li>• 1: Anweisung noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden.</li> </ul>  |
| ERROR     | Output      | BOOL            | E, A, M, D, L                | Statusparameter: <ul style="list-style-type: none"> <li>• 0: Kein Fehler.</li> <li>• 1: Fehler aufgetreten.</li> </ul>   |
| STATUS    | Output      | WORD            | E, A, M, D, L                | Status der Anweisung   |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter BUSY, DONE und ERROR

Den Ausführungsstatus der Anweisung "T\_DIAG" kontrollieren Sie über die Parameter BUSY, DONE, ERROR und STATUS. Der Parameter BUSY zeigt den Bearbeitungsstatus. Mit dem Parameter DONE kontrollieren Sie, ob die Anweisung erfolgreich ausgeführt wurde. Der Parameter ERROR wird gesetzt, wenn Fehler während der Ausführung von "T\_DIAG" auftreten.

Die folgende Tabelle zeigt den Zusammenhang zwischen den Parametern BUSY, DONE und ERROR:

| BUSY | DONE | ERROR | Beschreibung  |
|------|------|-------|---|
| 1    | -    | -     | Die Anweisung wird bearbeitet.  |
| 0    | 1    | 0     | Die Anweisung wurde erfolgreich durchgeführt. Nur in diesem Fall sind die Daten in der mit RESULT referenzierten Struktur gültig. |
| 0    | 0    | 1     | Die Anweisung wurde mit einem Fehler beendet. Die Ursache des Fehlers wird am Parameter STATUS ausgegeben.                        |
| 0    | 0    | 0     | Es wurde keine neue Anweisung zugewiesen.   |

### Parameter STATUS

Die folgende Tabelle zeigt die Bedeutung der Werte am Parameter STATUS:

| STATUS*<br>(W#16#..<br>..)  | Erläuterung   |
|---|---|
| 0000  | Die Anweisung "T_DIAG" wurde erfolgreich ausgeführt. Die Daten innerhalb der am Parameter RESULT referenzierten Struktur können ausgewertet werden.                                 |
| 7000  | Keine Anweisungsbearbeitung aktiv.  |
| 7001  | Anweisungsbearbeitung gestartet.  |
| 7002  | Die Verbindungsinformationen werden gelesen (Parameter REQ irrelevant).   |
| 8086  | Der Wert am Parameter ID ist nicht im erlaubten Bereich (W#16#0001 ... W#16#0FFF).  |
| 8089  | Der Parameter RESULT zeigt auf einen ungültigen Datentyp (nur Strukturen TDIAG_Status und TDIAG_StatusExt erlaubt).   |
| 80A3  | Der Parameter ID referenziert einen Verbindungsendpunkt, der nicht existiert. Dieser Fehler tritt bei programmierten Verbindungen auch nach dem Aufruf der Anweisung "TDISCON" auf. |
| 80C4  | Interner Fehler. Der Zugriff auf den Verbindungsendpunkt ist temporär nicht möglich.  |
| * Die Fehlercodes im Programmierer können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |

## Strukturen TDIAG\_Status und TDIAG\_StatusExt

Die folgende Tabelle beschreibt den Aufbau der Strukturen TDIAG\_Status und TDIAG\_StatusExt:

- Die Struktur TDIAG\_StatusExt ist von dem Parameter InterfaceID bis zum Parameter ReceivedBytes identisch zu der Struktur TDIAG\_Status.
- Die Struktur TDIAG\_StatusExt enthält darüber hinaus die Parameter ConnTrials bis LastDisconnTimeStamp.

Alle Elemente haben nur dann einen gültigen Wert, wenn die Anweisung ohne Fehler ausgeführt werden konnte. Im Fehlerfall bleibt der Inhalt der Parameter unverändert.

| Name   | Datentyp | Beschreibung  |
|--|----------|---|
| Die nachfolgenden Parameter existieren sowohl bei der Struktur TDIAG_Status als auch bei der Struktur TDIAG_StatusExt: |          |   |
| InterfaceID  | HW_ANY   | Schnittstellen-ID (LADDR) der CPU oder des CM/CP.   |
| ID   | CONN_OUT | ID der Verbindung, die diagnostiziert wurde. Nach einem erfolgreichen Aufruf ist der Wert dieses Elements mit dem Parameter ID der Anweisung "T_DIAG" identisch.  |
| ConnectionType   | BYTE     | Bei der Verbindung verwendeter Protokolltyp: <ul style="list-style-type: none"> <li>• 0x01: Nicht verwendet.</li> <li>• ...</li> <li>• 0x0B: TCP-Protokoll (IP_v4)</li> <li>• 0x0C: ISO-on-TCP-Protokoll (RFC1006)</li> <li>• 0x0D: TCP-Protokoll (DNS)</li> <li>• 0x0E: Dial-in-Protokoll</li> <li>• 0x0F: WDC-Protokoll</li> <li>• 0x10: SMTP-Protokoll</li> <li>• 0x11: TCP-Protokoll</li> <li>• 0x12: TCP und ISO-on-TCP Protokoll (RFC1006)</li> <li>• 0x13: UDP-Protokoll</li> <li>• 0x14: reserviert</li> <li>• 0x15: PROFIBUS-Buszugriffsprotokoll (FDL)</li> <li>• 0x16: ISO 8073-Transportprotokoll (ISONative)</li> <li>• ...</li> <li>• 0x20: SMTP- oder SMTPS-Protokoll - basierend auf IPv4</li> <li>• 0x21: SMTP- oder SMTPS-Protokoll - basierend auf IPv6</li> <li>• 0x22: SMTP- oder SMTPS-Protokoll - basierend auf FQDN (Fully Qualified Domain Name)</li> <li>• ...</li> <li>• 0x70: S7-Verbindung</li> <li>• Weitere: reserviert</li> </ul> |
| ActiveEstablished  | BOOL     | <ul style="list-style-type: none"> <li>• FALSE: Es handelt sich lokal um den passiven Verbindungsendpunkt</li> <li>• TRUE: Es handelt sich lokal um den aktiven Verbindungsendpunkt</li> </ul>  |

| Name   | Datentyp | Beschreibung  |
|--|----------|---|
| State  | BYTE     | <p>Aktueller Zustand des Verbindungsendpunkts:</p> <ul style="list-style-type: none"> <li>• 0x00: Nicht verwendet.</li> <li>• 0x01: Verbindung ist abgebaut. Temporärer Zustand, der z. B. nach Aufruf der Anweisung "T_RESET" eingenommen wird. Danach versucht das System selbstständig wieder einen erneuten Verbindungsaufbau.</li> <li>• 0x02: Der aktive Verbindungsendpunkt versucht, die Verbindung zum remoten Kommunikationspartner aufzubauen.</li> <li>• 0x03: Der passive Verbindungsendpunkt wartet auf den Verbindungsaufbau des remoten Kommunikationspartners.</li> <li>• 0x04: Verbindung ist aufgebaut.</li> <li>• 0x05: Die Verbindung wird gerade abgebaut. Grund für den Abbau kann der Aufruf der Anweisung "T_RESET" oder "T_DISCON" sein. Weitere mögliche Gründe können auch ein Protokollfehler oder ein Leitungsbruch sein.</li> <li>• 0x06..0xFF: Nicht verwendet.</li> </ul>                  |
| Kind   | BYTE     | <p>Betriebsart des Verbindungsendpunkts:</p> <ul style="list-style-type: none"> <li>• 0x00: Nicht verwendet.</li> <li>• 0x01: Konfigurierte, statische Verbindung, die projiziert und anschließend auf die CPU geladen wurde.</li> <li>• 0x02: Konfigurierte, dynamische Verbindung, die projiziert und anschließend auf die CPU geladen wurde (wird derzeit nicht unterstützt).</li> <li>• 0x03: Programmierte Verbindung, die aus dem Anwenderprogramm mit der Anweisung "TCON" erzeugt wurde. Durch einen Aufruf der Anweisung "TDISCON" oder ein Übergang in den CPU-Zustand STOPP wurde der Verbindungsendpunkt wieder zerstört.</li> <li>• 0x04: Temporäre, dynamische Verbindung, die z. B. durch die Engineering-Station (ES) oder Operator Station (OS) aufgebaut wird. (diese Verbindungsart kann derzeit wegen der fehlenden ID nicht diagnostiziert werden).</li> <li>• 0x05..0xFF: Nicht verwendet.</li> </ul> |
| SentBytes  | UDINT    | Anzahl der gesendeten Daten.  |
| ReceivedBytes  | UDINT    | Anzahl der empfangenen Daten.   |
| Die nachfolgenden Parameter existieren nur bei der Struktur TDiag_StatusExt: |          |   |
| ConnTrials   | UDINT    | <p>Anzahl der Verbindungsaufbauversuche. Nach einem erfolgreichen Verbindungsaufbau enthält ConnTrials den Wert 0. Ist das Element ungleich 0, deutet dies auf Verbindungsprobleme hin.</p> <p>Hinweis: Bei einem passiven Verbindungsendpunkt ist dieser Wert nie größer als 1.</p>  |
| ConnTrialsSuccesses  | UDINT    | <p>Anzahl der erfolgreichen Verbindungsaufbauversuche. Dieses Element wird während des Lebenszyklus eines Verbindungsendpunkts nie zurückgesetzt und bricht nach dem Wert 0xFFFF FFFF wieder auf 0 um.</p> <p>Hinweis: Wenn es noch nie ein Problem auf dieser Verbindung gab, hat dieser Parameter den Wert 1.</p>   |

| Name                  | Datentyp | Beschreibung  |
|-----------------------|----------|---|
| LastConnErrReason     | UDINT    | <p>Fehlerkennung, die beim letzten fehlerhaften Versuch eines Verbindungsaufbaus auftrat (die Fehlermeldungen sind identisch zu denen am Parameter LastDisconnReason):</p> <ul style="list-style-type: none"> <li>• 0x4F01: Remoter Verbindungsendpunkt nicht erreichbar (dieser Fehler tritt normalerweise während der Phase des Verbindungsaufbaus auf).</li> <li>• 0x4F02: Verbindung wurde lokal abgebaut.</li> <li>• 0x4F03: Die Verbindung wurde durch den remoten Kommunikationspartner abgebaut.</li> <li>• 0x4F04: Die Verbindung wurde durch einen Protokollfehler abgebaut.</li> <li>• 0x4F05: Die Verbindung wurde durch ein lokal erkanntes Netzwerkproblem abgebaut.</li> <li>• 0x4F06: Die Verbindung wurde durch ein remote erkanntes Netzwerkproblem abgebaut.</li> <li>• 0x4F07: Die Verbindung wurde auf Grund einer Zeitüberschreitung im Protokoll abgebaut.</li> <li>• 0x4F08: Falsche Parametrierung: Der Verbindungsaufbau soll zur eigenen Adresse erfolgen.</li> <li>• 0x4F09: Die Verbindung wurde temporär durch den Aufruf der Anweisung "T_RESET" zurückgesetzt.</li> <li>• 0x4F0A: Zu wenig Verbindungsressourcen verfügbar (Mengengerüst überschritten)</li> <li>• 0x4F0B: Interner Fehler: falsche Adressierungsparameter</li> <li>• 0x4F0C: CPU-interner Kommunikationsfehler</li> <li>• 0x4F0D: AS-interner Kommunikationsfehler zwischen CPU und CM/CP</li> <li>• 0x4F0E: Der angegebene lokale TCP/UDP-Port (bzw. RFC1006-TSelector) ist bereits verwendet.</li> </ul> |
| LastConnErrTime Stamp | LDT      | Zeitpunkt des letzten fehlerhaften Verbindungsaufbauversuchs.   |

| Name                 | Datentyp | Beschreibung  |
|----------------------|----------|---|
| LastDisconnReason    | UDINT    | <p>Fehlererkennung, die zum letzten Verbindungsabbau geführt hat (die Fehlermeldungen sind identisch zu denen am Parameter LastConnErrReason):</p> <ul style="list-style-type: none"> <li>• 0x4F01: Remoter Verbindungsendpunkt nicht erreichbar (dieser Fehler tritt normalerweise während der Phase des Verbindungsaufbaus auf).</li> <li>• 0x4F02: Verbindung wurde lokal abgebaut.</li> <li>• 0x4F03: Die Verbindung wurde durch den remoten Kommunikationspartner abgebaut.</li> <li>• 0x4F04: Die Verbindung wurde durch einen Protokollfehler abgebaut.</li> <li>• 0x4F05: Die Verbindung wurde durch ein lokal erkanntes Netzwerkproblem abgebaut.</li> <li>• 0x4F06: Die Verbindung wurde durch ein remote erkanntes Netzwerkproblem abgebaut.</li> <li>• 0x4F07: Die Verbindung wurde auf Grund einer Zeitüberschreitung im Protokoll abgebaut.</li> <li>• 0x4F08: Falsche Parametrierung: Der Verbindungsaufbau soll zur eigenen Adresse erfolgen.</li> <li>• 0x4F09: Die Verbindung wurde temporär durch den Aufruf der Anweisung "T_RESET" zurückgesetzt.</li> <li>• 0x4F0A: Zu wenig Verbindungsressourcen verfügbar (Mengengerüst überschritten)</li> <li>• 0x4F0B: Interner Fehler: falsche Adressierungsparameter</li> <li>• 0x4F0C: CPU-interner Kommunikationsfehler</li> <li>• 0x4F0D: AS-interner Kommunikationsfehler zwischen CPU und CM/CP</li> <li>• 0x4F0E: Der angegebene lokale TCP/UDP-Port (bzw. RFC1006-TSelector) ist bereits verwendet.</li> </ul> |
| LastDisconnTimeStamp | LDT      | Zeitpunkt des letzten Verbindungsabbaus.  |

## T\_CONFIG: Interface konfigurieren

### Beschreibung T\_CONFIG

#### Beschreibung

Die Anweisung "T\_CONFIG" dient zur programmgesteuerten Konfiguration der integrierten PROFINET-Schnittstellen der CPU oder der Schnittstelle eines CPs/CMs. Die bisher gültigen Konfigurationsdaten werden dabei überschrieben.

Sie können die folgenden Einstellungen der Schnittstellen-Konfiguration vornehmen:

- IP-Parameter: IP-Adresse, Subnetzmaske, Router-Adresse
- PROFINET IO-Gerätename (falls die CPU als PROFINET IO Device betrieben wird)

Die Konfigurationsdaten müssen Sie in einem Datenbaustein (Parameter CONF\_DB) hinterlegen.

Die programmgesteuerte Einstellung der IP-Konfiguration mit der Anweisung "T\_CONFIG" können Sie alternativ zur Projektierung in der Gerätekonfiguration vornehmen. Sie wird jedoch nur dann wirksam, wenn Sie in der Hardware-Konfiguration explizit vorgegeben haben, dass die Vergabe von IP-Adressparametern auf anderem Weg bezogen werden soll.

## Arbeitsweise

Die Anweisung "T\_CONFIG" ist eine asynchron arbeitende Anweisung, d. h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Sie starten den Übertragungsvorgang, indem Sie "T\_CONFIG" mit REQ = 1 aufrufen. Zu jedem Zeitpunkt kann nur ein Auftrag aktiv sein.

Der Baustein ist flankengetriggert, d.h. dass nach BUSY= FALSE der Baustein nochmals mit REQ=FALSE aufgerufen werden muss, damit die Instanz freigegeben wird.

Über den Ausgangsparameter BUSY und den Ausgangsparameter STATUS wird der Zustand des Auftrags angezeigt.

In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich die Anweisung aktuell befindet bzw. wann die Übertragung der Konfigurationsdaten beendet ist.

| BUSY  | DONE       | ERROR      | Beschreibung   |
|-------|------------|------------|--|
| TRUE  | irrelevant | irrelevant | Der Auftrag ist in Bearbeitung.  |
| FALSE | TRUE       | FALSE      | Der Auftrag wurde erfolgreich abgeschlossen.   |
| FALSE | FALSE      | TRUE       | Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter STATUS entnehmen. |
| FALSE | FALSE      | FALSE      | Der Anweisung wurde kein (neuer) Auftrag erteilt.  |

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "T\_CONFIG":

| Parameter                        | Deklaration | Datentyp         | Speicherbereich                 |  | Beschreibung  |
|----------------------------------|-------------|------------------|---------------------------------|--|---|
|                                  |             |                  | S7-1200                         | S7-1500                                  |   |
| REQ                              | Input       | BOOL             | E, A, M, D, L<br>oder Konstante | E, A, M, D, L, T,<br>Z oder<br>Konstante | Beim Aufruf der Anweisung mit REQ = 1 wird die Bearbeitung der Anweisung gestartet.   |
| INTERFAC<br>E                    | Input       | HW_INTERFA<br>CE | E, A, M, D, L<br>oder Konstante | E, A, M, D, L<br>oder Konstante          | Hardware-Kennung der Schnittstelle (siehe "Eigenschaften" in dem Inspektorfenster der Gerätekonfiguration). Die Hardware-Kennung ist in den Systemkonstanten der PLC-Variablen hinterlegt.  |
| CONF_DA<br>TA<br>(Seite<br>3011) | Input       | VARIANT          | D                               | D  | Zeiger auf einen Datenbaustein, in dem Sie die Verbindungsdaten hinterlegen. Verweisen Sie mit dem Zeiger auf ein übergeordnetes Struct-Element, das als Unterelemente die Felder Header, Addr und NOS enthält (siehe Beschreibung des Parameters CONF_DATA). |
| DONE                             | Output      | BOOL             | E, A, M, D, L                   | E, A, M, D, L                            | Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde: <ul style="list-style-type: none"> <li>• 0: Bearbeitung noch nicht abgeschlossen</li> <li>• 1: Bearbeitung der Anweisung erfolgreich beendet.</li> </ul>                         |

| Parameter | Deklaration | Datentyp | Speicherbereich |               | Beschreibung  |
|-----------|-------------|----------|-----------------|---------------|---|
|           |             |          | S7-1200         | S7-1500       |   |
| BUSY      | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L | Status der Anweisung:<br><ul style="list-style-type: none"> <li>0: Bearbeitung der Anweisung noch nicht begonnen, abgeschlossen oder abgebrochen</li> <li>1: Bearbeitung der Anweisung läuft</li> </ul> |
| ERROR     | Output      | BOOL     | E, A, M, D, L   | E, A, M, D, L | Fehleranzeige<br><ul style="list-style-type: none"> <li>0: kein Fehler</li> <li>1: Fehler</li> </ul>  |
| STATUS    | Output      | DWORD    | E, A, M, D, L   | E, A, M, D, L | Statusanzeige<br>Zur Bedeutung im Zusammenhang mit den Parametern DONE und ERROR siehe unter Anzeigen der Anweisung.  |
| ERR_LOC   | Output      | DWORD    | E, A, M, D, L   | E, A, M, D, L | Fehlerort (fieldId und id des Teilfelds, in dem an einem Parameter ein Fehler aufgetreten ist)  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

### Parameter ERROR und STATUS

| ERROR | STATUS (DW#16#..) | ERR_LOC* | Erläuterung   |
|-------|-------------------|----------|---|
| 0     | 00000000          | 0        | Auftragsbearbeitung ohne Fehler beendet   |
| 0     | 00700000          | 0        | Keine Auftragsbearbeitung aktiv   |
| 0     | 00700100          | 0        | Start der Auftragsbearbeitung   |
| 0     | 00700200          | 0        | Zwischenaufruf (REQ irrelevant )  |
| 1     | C08xyy00          | 0        | Allgemeine Fehlerinformation<br>Siehe auch: Fehlerauswertung mit dem Ausgangsparameter RET_VAL (Seite 1622)                   |
| 1     | C0808000          | 0        | Hardware-Kennung am Parameter INTERFACE ist ungültig.   |
| 1     | C0808100          | 0        | Hardware-Kennung am Parameter INTERFACE ist der unterstützten PROFINET-Schnittstelle nicht zugewiesen                         |
| 1     | C0808700          | 0        | Falsche Länge des Datenbausteins am Parameter CONF_DATA.  |
| 1     | C0808800          | f, 0     | Field_type hat einen unzulässigen Wert.   |
| 1     | C0808900          | f, 0     | Der Parameter fieldid hat einen unzulässigen Wert oder wurde mehrfach verwendet.  |
| 1     | C0808A00          | f, 0     | Falsche Anzahl bei Parameter subfield_cnt oder falsche Länge am Parameter Length.   |
| 1     | C0808B00          | f, s     | Der Parameter Id eines Teilfelds hat einen unzulässigen Wert.   |
| 1     | C0808C00          | f, s     | Fehler bei der Platzierung der Teilfelder (falsches Teilfeld, falsche Reihenfolge oder mehrfache Verwendung eines Teilfelds). |
| 1     | C0808D00          | f, s     | Der Parameter Lenght eines Teilfelds hat einen falschen oder unzulässigen Wert.   |
| 1     | C0808E00          | f, s     | Der Parameter Mode eines Teilfelds hat einen falschen oder unzulässigen Wert.   |



| ERROR   | STATUS<br>(DW#16#..) | ERR_LOC* | Erläuterung   |
|---|----------------------|----------|---|
| 1   | C0809000             | f, s     | Die Parameter des Teilfelds sind schreibgeschützt. Z. B. erfolgte die Parametervorgabe per Projektierung oder der PNIO-Betrieb ist aktiv. |
| 1   | C0809100             | f, s     | reserviert  |
| 1   | C0809400             | f, s     | Ein Parameterwert im Teilfeld ist nicht definiert oder unzulässig.  |
| 1   | C0809500             | f, s     | Der Wert eines Teilfeld-Parameters ist inkonsistent mit einem anderen Parameterwert.  |
| 1   | C080C200             | 0        | Die Übertragung kann nicht durchgeführt werden (z. B. weil die Schnittstelle nicht erreichbar ist).                                       |
| 1   | C080C300             | 0        | Ungenügende Ressourcen (z. B. mehrfacher Aufruf von "T_CONFIG" mit unterschiedlichen Parametern)  |
| 1   | C080C400             | 0        | Temporärer Kommunikationsfehler   |
| 1   | C080D200             | 0        | Aufruf nicht möglich / keine Unterstützung durch die PROFINET-Schnittstelle   |
| * In der obigen Tabelle ist f die field_id und s die id des Teilfelds, in dem der Fehler aufgetreten ist. |                      |          |   |

## Parameter CONF\_DATA

### Aufbau des DBs der Konfigurationsdaten

Der Parameter CONF\_DATA der Anweisung "T\_CONFIG" zeigt auf einen globalen Datenbaustein (DB), in dem Sie die Konfigurationsdaten hinterlegen.

Der DB setzt sich aus einer Struktur IF\_CONF\_Header und den Strukturen IF\_CONF\_V4 und / oder IF\_CONF\_NOS zusammen:

- Die Struktur IF\_CONF\_Header muss am Anfang des DBs stehen. Über die Struktur legen Sie fest, wie viele Teilfelder Sie verwenden möchten.
- Die Strukturen IF\_CONF\_V4 und IF\_CONF\_NOS sind die verwendbaren Teilfelder in dem DB, welche die eigentlichen Konfigurationsdaten enthalten. Die jeweiligen Parameter der beiden Teilfelder entsprechen weitgehend der Struktur in den Geräteeigenschaften.
- Alle drei Strukturen müssen unterhalb einer übergeordneten Struktur definiert werden (im folgenden Beispiel das Struct-Element "Conf\_Data"). Das folgende Bild zeigt den Aufbau des Datenbausteins.

|    |                  |                        |      |       |  |
|----|------------------|------------------------|------|-------|--|
| 1  | Static           |                        |      |       |  |
| 2  | Conf_data        | Struct                 | 0.0  | false |  |
| 3  | header           | IF_CONF_Header         | 0.0  | false |  |
| 4  | FieldType        | UInt                   | 0.0  | 0     |  |
| 5  | FieldId          | UInt                   | 2.0  | 0     |  |
| 6  | SubfieldCount    | UInt                   | 4.0  | 0     |  |
| 7  | addr             | IF_CONF_v4             | 6.0  | false |  |
| 8  | Id               | UInt                   | 0.0  | 30    |  |
| 9  | Length           | UInt                   | 2.0  | 18    |  |
| 10 | Mode             | UInt                   | 4.0  | 0     |  |
| 11 | InterfaceAddress | IP_V4                  | 6.0  |       |  |
| 12 | ADDR             | array [1..4] of Byte   | 0.0  |       |  |
| 13 | SubnetMask       | IP_V4                  | 10.0 |       |  |
| 14 | ADDR             | array [1..4] of Byte   | 0.0  |       |  |
| 15 | DefaultRouter    | IP_V4                  | 14.0 |       |  |
| 16 | ADDR             | array [1..4] of Byte   | 0.0  |       |  |
| 17 | nos              | IF_CONF_NOS            | 24.0 | false |  |
| 18 | Id               | UInt                   | 0.0  | 40    |  |
| 19 | Length           | UInt                   | 2.0  | 246   |  |
| 20 | Mode             | UInt                   | 4.0  | 0     |  |
| 21 | NOS              | array [1..240] of Byte | 6.0  |       |  |

### Verschaltung des Datenbausteins am Parameter CONF\_DATA

Rufen Sie am Parameter CONF\_DATA das übergeordnete Struct-Element des Datenbausteins auf (im Beispiel oben das Struct-Element "Conf\_Data"; der Aufruf am Parameter erfolgt durch die Angabe des Datenbausteins, gefolgt von dem Namen des Struct-Elements: "Name\_of\_DB".Conf\_data).

### Feld IF\_CONF\_Header

Über das Feld IF\_CONF\_Header wählen Sie aus, wie viele Teilfelder bei der Ausführung von "T\_CONFIG" verwendet werden.

| Byte    | Parameter     | Datentyp | Startwert | Beschreibung  |
|---------|---------------|----------|-----------|---|
| 0 ... 1 | FieldType     | UINT     |           | Feldtyp: Muss immer 0 sein.                             |
| 2 ... 3 | FieldId       | UINT     |           | Feld-ID: Muss immer 0 sein.                             |
| 4 ... 5 | SubfieldCount | UINT     |           | Gesamtanzahl der in der Struktur enthaltenen Teilfelder |

## Allgemeine Parameter der Teilfelder

Die Teilfelder "Addr" und "Nos" enthalten folgende allgemeine Parameter:

- Id  
Dieser Parameter kennzeichnet das jeweilige Feld und darf nicht verändert werden.
- Length  
Dieser Parameter gibt die tatsächliche Länge des Teilfelds an. Wenn ein Feld Parameter vom Datentyp String oder Array enthält, dann kann es sein, dass die maximale Länge des Parameters nicht ausgeschöpft wird. In diesem Fall ist die tatsächliche Länge des Teilfelds geringer als die maximale Länge.
- Mode  
Für diesen Parameter sind die folgenden Werte zulässig:
  - 1: Permanente Gültigkeit der Konfigurationsdaten
  - 2: Temporäre Gültigkeit der Konfigurationsdaten einschließlich Löschen vorhandener permanenter Konfigurationsdaten

## Teilfeld IF\_CONF\_V4

Über das Teilfeld IF\_CONF\_V4 legen Sie die Ethernet-Adressen fest, die Sie für die Schnittstelle der CPU vergeben möchten.

| Byte      | Parameter        | Datentyp | Startwert | Beschreibung   |
|-----------|------------------|----------|-----------|--|
| 0 ... 1   | Id               | UINT     | 30        | Kennung des Teilfelds  |
| 2 ... 3   | Length           | UINT     | 18        | Länge des Teilfelds in Byte  |
| 4 ... 5   | Mode             | UINT     |           | Gültigkeit der Adressierung: <ul style="list-style-type: none"> <li>• 1: permanent</li> <li>• 2: temporär</li> </ul> |
| 6 ... 9   | InterfaceAddress | IP_V4 *  |           | IP-Adresse   |
| 10 ... 12 | SubnetMask       | IP_V4 *  |           | Subnetzmaske   |
| 14 ... 16 | DefaultRouter    | IP_V4 *  |           | Routeradresse  |

\* Der Datentyp IP\_V4 ist eine Struktur von 4 BYTE, welche die jeweilige Adresse des entsprechenden Parameters enthält (z.B. am Parameter SubnetMask die vierstellige Adresse der Subnetzmaske des IP-Protokolls).

## Teilfeld IF\_CONF\_NOS

Über das Teilfeld IF\_CONF\_NOS legen Sie den Stationsnamen fest, der bei der Ausführung der Anweisung "T\_CONFIG" vergeben werden soll.

| Byte    | Parameter | Datentyp | Startwert | Beschreibung                |
|---------|-----------|----------|-----------|-----------------------------|
| 0 ... 1 | Id        | UINT     | 40        | Kennung des Teilfelds       |
| 2 ... 3 | Length    | UINT     | 246       | Länge des Teilfelds in Byte |

| Byte      | Parameter | Datentyp                      | Startwert | Beschreibung   |
|-----------|-----------|-------------------------------|-----------|--|
| 4 ... 5   | Mode      | UINT                          |           | Gültigkeit der Änderung des Stationsnamens: <ul style="list-style-type: none"> <li>• 1: permanent</li> <li>• 2: temporär</li> </ul>  |
| 6 ... 244 | NoS       | ARRAY<br>[1...240] of<br>Byte |           | Stationsname: Sie müssen das ARRAY ab dem ersten Byte belegen. Wenn das ARRAY länger ist als der zu vergebende Stationsname, müssen Sie nach dem eigentlichen Stationsnamen ein Nullbyte eintragen (konform zu IEC 61185-6-10). Andernfalls wird NoS zurückgewiesen, und die Anweisung "T_CONFIG" trägt den Fehlercode DW#16#C0809400 in STATUS ein. Wenn Sie das erste Byte mit Null belegen, wird der Stationsname gelöscht. |

Der Stationsname unterliegt folgenden Einschränkungen:

- Die Namensangabe muss im ASCII-Code erfolgen.
- Beschränkung auf 240 Zeichen insgesamt (Kleinbuchstaben, Ziffern, Bindestrich oder Punkt)
- Ein Namensbestandteil innerhalb des Stationsnamens, d. h. eine Zeichenkette zwischen zwei Punkten, darf maximal 63 Zeichen lang sein.
- Keine Sonderzeichen wie Umlaute, Klammern, Unterstrich, Schrägstrich, Blank etc. Der Bindestrich ist das einzige erlaubte Sonderzeichen.
- Der Stationsname darf nicht mit dem Zeichen "-" beginnen und auch nicht mit diesem Zeichen enden.
- Der Stationsname darf nicht mit Ziffern beginnen.
- Der Stationsname darf nicht die Form n.n.n.n haben (n = 0, ... 999).
- Der Stationsname darf nicht mit der Zeichenfolge "port-xyz" oder "port-xyz-abcde" beginnen (a, b, c, d, e, x, y, z = 0, ... 9).

---

**Hinweis**

Sie können das ARRAY NoS auch kürzer als 240 Bytes anlegen, wobei die Mindestlänge 2 Bytes beträgt. In diesem Fall müssen Sie die Variable Length (Länge des Teilfelds) entsprechend anpassen.

---

### 9.7.5.4 Webserver

#### WWW: Anwenderdefinierte Webseiten synchronisieren

##### Beschreibung

Die Anweisung WWW initialisiert den Webserver der CPU bzw. synchronisiert anwenderdefinierte Webseiten mit dem Anwenderprogramm in der CPU.

Anwenderdefinierten Webseiten bilden zusammen mit dem Webserver der CPU die Möglichkeit, mit einem Webbrowser auf frei gestaltete Webseiten der CPU zuzugreifen.

Mit Hilfe von Scriptanweisungen (z. B. Javascript) und HTML-Code in anwenderdefinierten Webseiten können Sie auch Daten über einen Webbrowser zur Weiterverarbeitung an die CPU übergeben und Daten aus dem Operandenbereich der CPU im Webbrowser anzeigen lassen. Zur Synchronisation zwischen Anwenderprogramm und Webserver, aber auch zur Initialisierung müssen Sie die Anweisung WWW im Anwenderprogramm aufrufen.

## Initialisierung

Anwenderdefinierte Webseiten werden in Datenbausteine "verpackt", damit die CPU sie verarbeiten kann. Aus den Quelldaten (HTML-Dateien, Bilder, Javascript-Dateien, ...) müssen Sie während der Projektierung entsprechende Datenbausteine generieren. Eine besondere Rolle nimmt der Web-Control-DB ein (Default: DB 333), der Status- und Steuerungsinformationen sowie Verweise auf weitere Datenbausteine mit codierten Webseiten enthält. Die Datenbausteine mit codierten Webseiten werden Fragment-DBs genannt.

Wenn die Datenbausteine in die CPU geladen wurden, "weiß" die CPU nicht, dass darin anwenderdefinierte Webseiten codiert sind. Durch die Anweisung "WWW" z. B. im Anlauf-OB wird ihr mitgeteilt, welcher DB der Web-Control-DB ist. Nach dieser Initialisierung sind die anwenderdefinierten Webseiten über einen Webbrowser erreichbar.

## Synchronisierung

Wenn das Anwenderprogramm mit den anwenderdefinierten Webseiten interagieren soll, muss die Anweisung WWW im zyklischen Programmteil verwendet werden.

Beispiele für Interaktion zwischen Anwenderprogramm und Webseite:

- Empfangene Daten überprüfen
- Daten für den anfragenden Webbrowser zusammenstellen und zurücksenden

In diesem Fall muss die Statusinformation aktuell auswertbar sein und Steuerungsinformation z. B. zum Freigeben einer vom Webbrowser angeforderten Webseite dem Webserver mitgeteilt werden.

## Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "WWW":

| Parameter | Deklaration | Datentyp | Beschreibung   |
|-----------|-------------|----------|--|
| CTRL_DB   | Input       | DB_WWW   | Datenbaustein, der die anwenderdefinierten Webseiten beschreibt (Web-Control-DB) |
| RET_VAL   | Output      | INT      | Fehlerinformation  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter Übersicht über die gültigen Datentypen (Seite 1087).

**Parameter RET\_VAL**

| Fehlercode (W#16#...) | Erläuterung  |
|-----------------------|--|
| 0000                  | Es ist kein Fehler aufgetreten. Es stehen keine Webseitenanforderungen an, die vom Anwenderprogramm freigegeben werden müssen.   |
| 00xy                  | x: zeigt an, ob bei der Initialisierung des Web-Control-DBs (CTRL_DB) ein Fehler aufgetreten ist:<br>x=0: keine Fehler aufgetreten.<br>x=1: Fehler aufgetreten. Der Fehler ist im Byte "CTRL_DB.last_error" des Web-Control-DBs kodiert, siehe Beschreibung zum Web-Control-DB.<br>y: Nummer des anstehenden Requests. Mehrere Requests sind möglich (z. B. Request "0" und "1" stehen an: y="3").<br>y="1": Request "0"<br>y="2": Request "1"<br>y="4": Request "2"<br>y="8": Request "3" |
| 803A                  | Der angegebene Web-Control-DB ist auf der CPU nicht vorhanden.   |
| 8081                  | Falsche Version oder falsches Format des Web-Control-DBs   |
| 80C1                  | Es sind keine Ressourcen vorhanden, um die Web-Applikation zu initialisieren   |

**Siehe auch**

Übersicht über die gültigen Datentypen (Seite 1087)

**9.7.5.5 TeleService**

**TM\_MAIL: E-Mail übertragen**

**Beschreibung TM\_MAIL**

**Beschreibung**

Die Anweisung "TM\_MAIL" ist eine asynchron arbeitende Anweisung, d.h. die Bearbeitung erstreckt sich über mehrere Aufrufe. Beim Aufruf der Anweisung "TM\_MAIL" ist die Angabe einer Instanz zwingend erforderlich. In der Instanz darf das Attribut "Remanent" nicht gesetzt werden. Durch dieses Attribut wird gewährleistet, dass die Instanz beim Übergang der CPU von STOP nach RUN initialisiert wird und danach ein neuer Auftrag zum Senden einer E-Mail angestoßen werden kann.

Sie starten das Versenden einer E-Mail mit einem Flankenwechsel von "0" auf "1" am Parameter REQ. Über die Ausgangsparameter "BUSY", "DONE", "ERROR" und die Ausgangsparameter "STATUS" und "SFC\_STATUS" wird der Zustand des Auftrags angezeigt. Dabei entspricht "SFC\_STATUS" dem Ausgangsparameter "STATUS" der aufgerufenen Kommunikations-Bausteine.

Die Ausgangsparameter DONE, ERROR, STATUS und SFC\_STATUS werden jeweils nur einen Zyklus lang angezeigt, wenn der Zustand des Ausgangsparameters BUSY von "1" auf "0" wechselt. In der folgenden Tabelle ist der Zusammenhang zwischen BUSY, DONE und ERROR angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich die Anweisung "TM\_MAIL" aktuell befindet bzw. wann das Senden der E-Mail beendet ist.

| DONE | BUSY | ERROR | Beschreibung  |
|------|------|-------|---|
| 0    | 1    | 0     | Der Auftrag ist in Bearbeitung.   |
| 1    | 0    | 0     | Der Auftrag wurde erfolgreich abgeschlossen.  |
| 0    | 0    | 1     | Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache entnehmen Sie den Parametern STATUS und SFC_STATUS. |
| 0    | 0    | 0     | Der Anweisung "TM_MAIL" wurde kein (neuer) Auftrag erteilt.   |

Wechselt die CPU in den Betriebszustand STOP, während "TM\_MAIL" aktiv ist, so wird die Kommunikationsverbindung zum Mail-Server abgebrochen. Die Kommunikationsverbindung zum Mail-Server geht auch verloren, wenn am Industrial Ethernet-Bus Kommunikationsprobleme auftreten. In diesen Fällen wird das Versenden der E-Mail unterbrochen und sie erreicht ihre Empfänger nicht.

#### ACHTUNG

##### Ändern von Anwenderprogrammen

Sie dürfen die Teile Ihres Anwenderprogramms, die sich unmittelbar auf die Aufrufe von "TM\_MAIL" auswirken, nur ändern:

- wenn die CPU im Betriebszustand "STOP" ist oder
- wenn keine Mail gesendet wird (REQ = 0 und BUSY = 0).

Dies bezieht sich insbesondere auf das Löschen und Austauschen von Programmbausteinen, die Aufrufe von "TM\_MAIL" oder Aufrufe der Instanz von "TM\_MAIL" enthalten.

Bei Nichtbeachtung können Verbindungsressourcen belegt bleiben. Das Automatisierungssystem kann in einen nicht definierten Zustand bei den TPC/IP-Kommunikationsfunktionen über Industrial Ethernet geraten.

Nach dem Übertragen der Änderungen müssen Sie einen Neustart (Warmstart) oder Kaltstart bei der CPU durchführen.

## Datenkonsistenz

Der Eingangsparameter ADDR\_MAIL\_SERVER der Anweisung wird bei jedem Anstoß zum Versenden einer E-Mail von der Anweisung "TM\_MAIL" neu übernommen. Bei einer Änderung während des Betriebs wird der "neue" Wert erst bei einem erneuten Anstoß einer E-Mail wirksam.

Die Parameter WATCH\_DOG\_TIME, TO\_S, CC, FROM, SUBJECT, TEXT, ATTACHMENT, sowie ggf. USERNAME und PASSWORD werden hingegen während des Betriebs der Anweisung "TM\_MAIL" von diesem übernommen und dürfen somit erst geändert werden, wenn der Auftrag beendet wurde (BUSY = 0)

### Parametrierung des TS Adapter IE

Am TS Adapter IE müssen Sie die Parameter für ausgehende Rufe so eingeben, dass der TS Adapter IE eine Verbindung zum Einwahlservers Ihres Internet Service Providers aufbaut.

Wenn Sie für Verbindungsaufbau "bei Bedarf" einstellen, dann wird die Verbindung erst dann aufgebaut, wenn eine Mail versendet werden soll.

Bei einer analogen Modemverbindung kann der Verbindungsaufbau längere Zeit (ca. eine Minute) dauern. Die Zeit für den Verbindungsaufbau müssen Sie bei der Vorgabe des Parameters WATCH\_DOG\_TIME berücksichtigen.

### Parameter

Die folgende Tabelle zeigt die Parameter der Anweisung "TM\_MAIL":

| Parameter            | Deklaration | Datentyp          | Speicherbereich              | Beschreibung   |
|----------------------|-------------|-------------------|------------------------------|--|
| REQ                  | Input       | BOOL              | E, A, M, D, L oder Konstante | Steuerparameter REQUEST: aktiviert das Versenden einer E-Mail bei steigender Flanke.   |
| ID                   | Input       | CONN_OUT C (Word) | D, L oder Konstante          | Referenz auf die aufzubauende Verbindung. Siehe Parameter ID der Anweisungen TCON (Seite 2968), TDISCON (Seite 2976), TSEND (Seite 2981) und TRCV (Seite 2984). Hier muss eine Nummer eingegeben werden, die in keiner weiteren Instanz dieser Anweisungen im Anwenderprogramm verwendet wird. |
| TO_S<br>(Seite 3020) | Input       | STRING            | D                            | Eingangsparemeter Empfängeradressen: STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel).  |
| CC (Seite 3020)      | Input       | STRING            | D                            | Eingangsparemeter CC-Empfängeradressen (optional): STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel).<br>Wird hier ein Leerstring zugewiesen, wird die E-Mail nicht an einen CC-Empfänger gesendet.  |
| SUBJECT              | Input       | STRING            | D                            | Eingangsparemeter Betreff der E-Mail: STRING mit einer maximalen Länge von 240 Zeichen.  |
| TEXT                 | Input       | STRING            | D                            | Eingangsparemeter Text der E-Mail (optional): Referenz auf einen Daten-String mit einer maximalen Länge von 240 Zeichen.<br>Wird an diesem Parameter ein Leerstring zugewiesen, wird die E-Mail ohne Text versendet.   |
| ATTACHMENT           | Input       | VARIANT           | E, A, M, D, L                | Eingangsparemeter Anhang der E-Mail (optional): Referenz auf ein Byte-/Wort-/Doppelwort-Feld mit einer maximalen Länge von 65534 Bytes.<br>Wird kein Wert zugewiesen, wird die E-Mail ohne Anhang versendet.   |
| DONE                 | Output      | BOOL              | E, A, M, D, L                | <ul style="list-style-type: none"> <li>DONE = 0: Auftrag ist noch nicht gestartet oder wird noch ausgeführt.</li> <li>DONE = 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>  |



| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|--|-------------|----------|-----------------|--|
| BUSY   | Output      | BOOL     | E, A, M, D, L   | <ul style="list-style-type: none"> <li>BUSY = 1: Das Senden der E-Mail ist noch nicht abgeschlossen.</li> <li>BUSY = 0: Die Bearbeitung von "TM_MAIL" wurde beendet.</li> </ul>  |
| ERROR  | Output      | BOOL     | E, A, M, D, L   | ERROR = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS und SFC_STATUS liefern detaillierte Auskunft über die Art des Fehlers.   |
| STATUS<br>(Seite 3021)   | Output      | WORD     | E, A, M, D, L   | Ausgangs-/Zustandsparameter STATUS: Rückgabewert bzw. Fehlerinformation der Anweisung "TM_MAIL".   |
| ADDR_MAIL_SERVER   | Static*     | DWORD    | E, A, M, D, L   | <p>Eingangsparameter IP-Adresse des Mail-Servers: Anzugeben als Datenwort im HEX-Format, z.B.: IP-Adresse = 192.168.0.200.</p> <p>ADDR_MAIL_SERVER = DW#16#C0A800C8, wobei:</p> <ul style="list-style-type: none"> <li>192 = 16#C0,</li> <li>168 = 16#A8</li> <li>0 = 16#00 und</li> <li>200 = 16#C8 entspricht.</li> </ul>  |
| WATCH_DOG_TIME   | Static*     | TIME     | E, A, M, D, L   | <p>Eingangsparameter max. Zeitspanne:</p> <p>In der vorgegebenen Zeit (WATCH_DOG_TIME) sollte die Anweisung "TM_MAIL" eine Verbindung aufbauen. Wird diese Zeit überschritten, so wird der Baustein mit einem Fehler beendet. Die Zeit, bis der Baustein beendet und der Fehler ausgegeben wird, kann die WATCH_DOG_TIME überschreiten, da der Verbindungsabbau ebenso Zeit beansprucht. Zu Anfang sollten Sie eine Zeit von 2 Minuten einstellen. Bei der ISDN-Telefonverbindung kann diese Zeit deutlich kleiner gewählt werden.</p> |
| USERNAME   | Static*     | STRING   | D               | Eingangsparameter Benutzername: STRING mit einer maximalen Länge von 180 Zeichen. Für Authentisierungsverfahren ist ein Benutzername zwingend erforderlich.  |
| PASSWORD   | Static*     | STRING   | D               | Eingangsparameter Passwort: STRING mit einer maximalen Länge von 180 Zeichen. Für Authentisierungsverfahren ist ein Passwort zwingend erforderlich.  |
| FROM<br>(Seite 3020)   | Static*     | STRING   | D               | Eingangsparameter Absenderadresse: STRING mit einer maximalen Länge von 240 Zeichen (siehe Aufrufbeispiel).  |
| SFC_STATUS<br>(Seite 3021)   | Static*     | WORD     | E, A, M, D, L   | Ausgangs-/Zustandsparameter "SFC_STATUS": Fehlerinformation der aufgerufenen Kommunikations-Bausteine.   |
| * Die Werte der Parameter werden nicht bei jedem Aufruf der Anweisung "TM_MAIL" geändert. Die Werte liegen in den statischen Parametern der Instanz und werden nur einmal vor dem ersten Aufruf der Anweisung beschrieben. |             |          |                 |  |

Weitere Informationen zu den gültigen Datentypen finden Sie unter "Übersicht über die gültigen Datentypen (Seite 1087)".

---

#### **Hinweis**

##### **Optionale Parameter**

Die optionalen Parameter CC, TEXT und ATTACHMENT werden nur mit der E-Mail versendet, wenn die die entsprechenden Parameter einen String der Länge > 0 enthalten.

---

## **SMTP-Authentisierung**

Unter Authentisierung versteht man ein Verfahren zur Sicherstellung einer Identität z.B. durch eine Passwortabfrage.

Die Anweisung "TM\_MAIL" unterstützt das SMTP-Authentisierungsverfahren AUTH-LOGIN, das von den meisten Mail-Servern gefordert wird. Informationen über die Authentisierungsverfahren Ihres Mail-Servers entnehmen Sie bitte dem Handbuch des Mail-Servers oder der Website Ihres Internet Service Providers.

Um das Authentisierungsverfahren AUTH-LOGIN zu nutzen benötigt die Anweisung "TM\_MAIL" den Benutzernamen, mit dem er sich beim Mail-Server anmelden kann. Dieser Benutzername entspricht dem Nutzernamen mit dem Sie ein Mailkonto auf Ihrem Mail-Server eingerichtet haben. Er wird über den Parameter USERNAME der Anweisung "TM\_MAIL" bekannt gegeben.

Des Weiteren benötigt die Anweisung "TM\_MAIL" zur Anmeldung das dazugehörige Passwort. Dieses Passwort entspricht dem Passwort, das Sie beim Einrichten Ihres Mailkontos vergeben haben. Es wird über den Parameter PASSWORD der Anweisung "TM\_MAIL" bekannt gegeben.

Der Benutzername und das Passwort werden jeweils unverschlüsselt (BASE64-kodiert) an den Mailserver übertragen.

Wird im DB kein Benutzername angegeben, so wird das Authentisierungsverfahren AUTH-LOGIN auch nicht verwendet. Die E-Mail wird dann ohne Authentisierung versendet.

## **Parameter TO\_S, CC und FROM**

### **Beschreibung**

Die Parameter TO\_S, CC und FROM sind Strings mit z. B. folgendem Inhalt:

- TO: <wenna@mydomain.com>, <ruby@mydomain.com> ,
- CC: <admin@mydomain.com>, <judy@mydomain.com> ,
- FROM: <admin@mydomain.com>

Folgende Regeln sind bei der Eingabe der Parameter zu beachten:

- Die Zeichen "TO:", "CC:" und "FROM:" müssen eingegeben werden.
- Vor jeder Adresse muss ein Leerzeichen und eine spitze Klammer auf "<" eingegeben werden.

- Nach jeder Adresse muss eine spitze Klammer zu ">" eingegeben werden.
- Nach jeder Adresse in TO und CC muss ein Komma eingegeben werden.
- Unter FROM darf nur eine E-Mail Adresse eingetragen werden, bei dieser darf kein Komma am Ende stehen

Aus Gründen der Laufzeit und des Speicherplatzes führt die Anweisung "TM\_MAIL" keine Syntaxprüfung der Parameter TO\_S, CC und FROM durch.

## Parameter STATUS und SFB\_STATUS

### Beschreibung

Die Rückgabewerte der Anweisung "TM\_MAIL" können folgendermaßen klassifiziert werden:

- W#16#0000: "TM\_MAIL" wurde erfolgreich beendet
- W#16#7xxx: Status von "TM\_MAIL"
- W#16#8xxx: beim internen Aufruf eines Kommunikations-Bausteins oder vom Mail-Server wurde ein Fehler gemeldet.

Die folgende Tabelle zeigt die Rückgabewerte von "TM\_MAIL" mit Ausnahme der Fehlercodes der intern aufgerufenen Kommunikations-Bausteine.

| Rückgabewert STATUS* (W#16#...): | Rückgabewert SFB_STATUS (W#16#...): | Erläuterung  | Hinweise   |
|----------------------------------|-------------------------------------|--|--|
| 0000                             | -                                   | Die Bearbeitung von "TM_MAIL" wurde fehlerfrei abgeschlossen.  | Ein fehlerfreies Beenden von "TM_MAIL" bedeutet nicht, dass die versendete E-Mail ankommt (siehe unten - Hinweis Punkt 1)                                  |
| 7001                             |                                     | "TM_MAIL" ist aktiv (BUSY = 1).  | Erstaufwurf; Auftrag angestoßen  |
| 7002                             | 7002                                | "TM_MAIL" ist aktiv (BUSY = 1).  | Zwischenaufwurf; Auftrag ist bereits aktiv   |
| 8xxx                             | xxxx                                | Die Bearbeitung von "TM_MAIL" wurde mit einem Fehlercode der intern aufgerufenen Kommunikations-Anweisungen abgeschlossen. | Ausführliche Informationen zur Auswertung des Parameters SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Kommunikations-Anweisungen. |
| 8010                             | xxxx                                | Fehler beim Verbindungsaufbau.   | Weitere Informationen zur Auswertung des Parameters SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TCON (Seite 2968)".   |
| 8011                             | xxxx                                | Fehler beim Senden der Daten.  | Weitere Informationen zur Auswertung des SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TSEND (Seite 2981)".             |

9.7 Referenzen

| Rückgabewert STATUS* (W#16#...): | Rückgabewert SFB_STATUS (W#16#...): | Erläuterung   | Hinweise   |
|----------------------------------|-------------------------------------|---|--|
| 8012                             | xxxx                                | Fehler beim Empfangen der Daten.  | Weitere Informationen zur Auswertung des SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TRCV (Seite 2984)".  |
| 8013                             | xxxx                                | Fehler beim Verbindungsaufbau.  | Weitere Informationen zur Auswertung des SFB_STATUS finden Sie in den Beschreibungen des Parameters STATUS der Anweisung "TCON (Seite 2968)" und "TDISCON (Seite 2976)".   |
| 8014                             | -                                   | Aufbau einer Verbindung nicht möglich.  | Möglicherweise haben Sie eine fehlerhafte Mail-Server-IP-Adresse (ADDR_MAIL_SERVER) oder eine zu kleine Zeitspanne (WATCH_DOG_TIME) für den Verbindungsaufbau eingegeben. Es besteht auch die Möglichkeit, dass die CPU keine Verbindung zum Netzwerk hat oder die CPU-Konfiguration fehlerhaft ist. |
| 8016                             | -                                   | Fehler beim Kopieren des Anhangs  | -  |
| 82xx, 84xx, bzw. 85xx            | -                                   | Die Fehlermeldung stammt vom Mail-Server und entspricht bis auf die "8" der Fehlernummer des SMTP-Protokolls. In den nachfolgenden Spalten sind einige Fehlercodes aufgeführt, die auftreten können:      | Siehe Punkt 2 im Hinweis.  |
| 8450                             | -                                   | Aktion nicht ausgeführt: Mailbox nicht verfügbar/nicht erreichbar.  | Versuchen Sie es später noch einmal.   |
| 8451                             | -                                   | Aktion abgebrochen: Lokaler Fehler in der Verarbeitung  | Versuchen Sie es später noch einmal.   |
| 8500                             | -                                   | Syntax-Fehler: Fehler nicht erkannt. Das schließt auch den Fehler einer zu langen Befehlskette ein. Ursache kann auch sein, dass der E-Mail-Server das Authentisierungsverfahren LOGIN nicht unterstützt. | Überprüfen Sie die Parameter des "TM_MAIL". Versuchen Sie, eine E-Mail ohne Authentisierung zu versenden. Ersetzen Sie dazu den Parameter USERNAME durch einen leeren String.  |
| 8501                             | -                                   | Syntax-Fehler: Parameter oder Argument falsch   | Möglicherweise haben Sie eine fehlerhafte Adresse in TO_S oder CC eingegeben.  |
| 8502                             | -                                   | Befehl unbekannt bzw. nicht implementiert.  | Überprüfen Sie Ihre Eingaben, insbesondere den Parameter FROM. Möglicherweise ist dieser unvollständig und Sie haben "@" oder "." vergessen.   |
| 8535                             | -                                   | SMTP-Authentisierung unvollständig.   | Eventuell haben Sie einen fehlerhaften Benutzernamen oder ein fehlerhaftes Passwort eingegeben.  |

| Rückgabewert<br>STATUS*<br>(W#16#...):   | Rückgabewert<br>SFB_STATUS<br>(W#16#...): | Erläuterung   | Hinweise  |
|--|---|---|---|
| 8550   | -   | Mail-Server kann nicht erreicht werden, Sie haben keine Zugriffsrechte. | Möglicherweise haben Sie einen fehlerhaften Benutzernamen bzw. Passwort eingegeben oder der Mail-Server unterstützt Ihr LOGIN nicht. Eine weitere Fehlerursache könnte eine fehlerhafte Eingabe des Domainnamens nach dem "@" in TO_S bzw. CC sein. |
| 8552   | -   | Aktion abgebrochen: Überschreitung der zugewiesenen Speichergröße       | Versuchen Sie es später noch einmal.  |
| 8554   | -   | Übertragung misslungen.   | Versuchen Sie es später noch einmal.  |
| * Die Fehlercodes im Programmeditor können als Integer- oder Hexadezimalwert dargestellt werden. Informationen zur Umschaltung der Anzeigeformate finden Sie unter "Siehe auch". |   |   |   |

---

### Hinweis

#### Statusfehler

1. Eine fehlerhafte Eingabe der Empfängeradressen erzeugt keinen Statusfehler der Anweisung "TM\_MAIL". Das Versenden der E-Mail an weitere Empfänger, auch wenn diese richtig eingegeben wurden, ist in diesem Fall nicht sichergestellt.
  2. Weitere Informationen zum SMTP-Fehlercode und weitere Fehlercodes finden Sie im SMTP-Protokoll im Internet bzw. in der Fehlerdokumentation des Mail-Servers. Darüber hinaus können Sie die letzte, vom Mail-Server gesendete Fehlermeldung in Textform in Ihrem Instanz-DB im Parameter BUFFER1 einsehen. Dort finden Sie unter "Daten" die zuletzt von der Anweisung "TM\_MAIL" versendeten Daten.
-



## Prozesse visualisieren (Basic)

### 10.1 Bilder erstellen

#### 10.1.1 Grundlagen

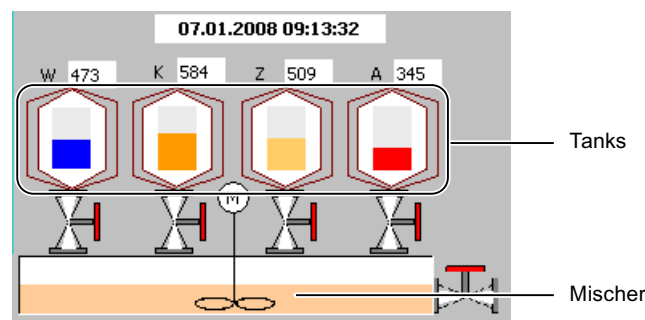
##### 10.1.1.1 Grundlagen zu Bildern

#### Einleitung

Mit WinCC erstellen Sie Bilder zum Bedienen und Beobachten von Maschinen und Anlagen. Um Bilder zu erstellen, stehen Ihnen vordefinierte Objekte zur Verfügung, mit denen Sie Ihre Anlage nachbilden, Prozessabläufe anzeigen und Prozesswerte vorgeben.

#### Anwendungsbeispiel

Diese Abbildung zeigt ein Bild, das mit WinCC erstellt wurde. Die Anlagenbediener beobachten und bedienen mit diesem Bild eine Mischstation zur Herstellung verschiedener Fruchtsäfte. Aus verschiedenen Tanks werden Fruchtsaft-Anteile in einen Mischer gefüllt und gemischt. Anzeigt wird die Füllhöhe in den Tanks.



#### Aufbau eines Bildes

Sie fügen die Objekte in Ihr Bild ein, die Sie für die Darstellung des Prozesses benötigen. Sie konfigurieren die Objekte entsprechend den Anforderungen Ihres Prozesses.

Ein Bild kann aus statischen und dynamischen Elementen bestehen.

## 10.1 Bilder erstellen

- Statische Elemente, z. B. Text und Grafik, ändern sich in Runtime nicht. Im Beispiel der Mischanlage sind z. B. die Beschriftungen der Tanks (W, K, Z, A) statisch.
- Dynamische Elemente ändern sich abhängig vom Prozess. Sie visualisieren aktuelle Prozesswerte wie folgt:
  - Aus dem Speicher der Steuerung
  - Aus dem Speicher des Bediengeräts in Form alphanumerischer Anzeigen, Kurven und Balken.

Zu den dynamischen Objekten gehören auch Eingabefelder am Bediengerät. Im Beispiel der Mischanlage sind die Füllstände der Tanks dynamische Objekte.

Prozesswerte und Eingaben werden zwischen Steuerung und Bediengerät über Variablen ausgetauscht.

### Bildeigenschaften

Die Darstellung des Bildes ist abhängig vom Bediengerät, für das Sie projektieren. Die Darstellung entspricht dem Aussehen der Bedienoberfläche des Geräts. Die Bildeigenschaften, z. B. die Bildauflösung und die verfügbaren Schriftarten und Farben, hängen ebenfalls vom eingestellten Bediengerät ab. Wenn das eingestellte Bediengerät Funktionstasten besitzt, zeigt das Bild diese Funktionstasten.

Eine Funktionstaste ist eine Taste am Bediengerät, die Sie in WinCC mit einer oder mehreren Funktionen belegen können. Die Funktionen werden getriggert, sobald der Bediener die Taste am Bediengerät drückt.

Eine Funktionstaste belegen Sie global oder lokal:

- Funktionstasten mit globaler Belegung lösen stets dieselbe Aktion aus, unabhängig vom aktuell angezeigten Bild.
- Funktionstasten mit lokaler Belegung lösen abhängig vom angezeigten Bild am Bediengerät unterschiedliche Aktionen aus. Die Belegung gilt nur für das Bild, in dem Sie die Funktionstaste definiert haben.

### Aufruf der Bilder

Damit der Bediener ein Bild in Runtime am Bediengerät aufrufen kann, müssen Sie jedes projektierte Bild in den Bedienablauf einbinden. Hierzu stehen Ihnen mehrere Möglichkeiten zur Verfügung:

- Sie nutzen den Editor "Bilder", um in den Bildern Schaltflächen und Funktionstasten zu projektieren, mit denen andere Bilder aufgerufen werden.
- Sie nutzen den Editor "Globales Bild", um global belegte Funktionstasten zu projektieren.

#### 10.1.1.2 Geräteabhängiger Funktionsumfang von Bildern

### Einleitung

Die Funktionen eines Bediengeräts bestimmen die Darstellung des Geräts in WinCC und den Funktionsumfang der Editoren.



Folgende Eigenschaften von Bildern sind abhängig vom eingestellten Bediengerät:

- Gerätelayout
- Bildauflösung
- Anzahl der Farben
- Schriftarten
- Verwendbare Objekte

## Gerätelayout

Das Gerätelayout eines Bildes bildet das Bediengerät ab, für das Sie projektieren. Wenn das Bediengerät z. B. Funktionstasten besitzt, erscheinen diese im Gerätelayout des Bildes.

## Bildauflösung

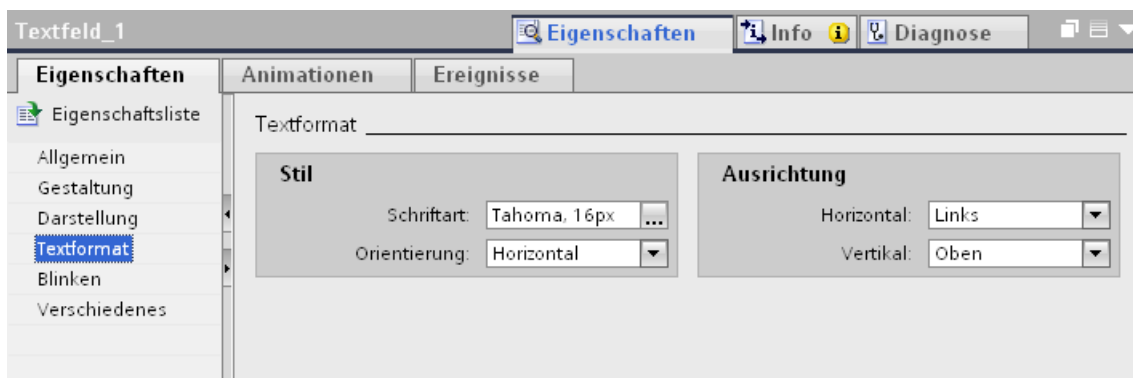
Da unterschiedliche Bediengeräte unterschiedliche Displaygrößen besitzen, hängt die Auflösung eines Bildes vom eingestellten Bediengerät ab. Die Bildauflösung können Sie nur verändern, wenn Sie das Bediengerät "WinCC Runtime Advanced" bzw. "WinCC Runtime Professional" projektieren.

## Anzahl der Farben

Den Objekten eines Bildes können Sie Farben zuordnen. Die Anzahl der möglichen Farben hängt davon ab, wie viele und welche Farben das eingestellte Bediengerät unterstützt.

## Schriftarten

In allen Bildobjekten, die statischen oder dynamischen Text enthalten, können Sie das Erscheinungsbild der Texte anpassen. So heben Sie einzelner Texte in einem Bild hervor. Sie wählen z. B. Schriftart, Schriftschnitt und Schriftgrad aus und stellen zusätzliche Effekte ein, z. B. Unterstreichung.



Die Einstellungen für Textauszeichnungen wie Schriftschnitt und Effekte beziehen sich immer auf den gesamten Text eines Bildobjekts. So können Sie z. B. eine Überschrift insgesamt fett darstellen, nicht aber einzelne Buchstaben oder Wörter dieser Überschrift.

## Verwendbare Objekte

Einige Bildobjekte können nicht für alle Bediengeräte projiziert werden. In der Task Card "Werkzeuge" werden diese Bildobjekte nicht angezeigt. So können Sie z. B. für das Bediengerät KTP1000 keinen Schieberegler projizieren.

### 10.1.1.3 Grundlagen

## Task Cards

### Einleitung

Im Editor "Bilder" stehen folgende Task Cards zur Verfügung:

- Werkzeuge: Anzeige- und Bedienobjekte
- Animationen: Vorlagen zur dynamischen Projektierung
- Layout: Hilfsmittel zur Anpassung der Darstellung
- Bibliotheken: Verwaltung der Projektbibliothek und der Globalen Bibliotheken

---

#### Hinweis

#### WinCC Basic

Die Task Card "Animationen" steht in WinCC Basic nicht zur Verfügung.



---

## Werkzeuge

Die Task Card "Werkzeuge" enthält Objekte in unterschiedlichen Paletten:

- Basisobjekte
- Elemente
- Controls
- Eigene Controls (optional)
- Grafiken

Aus den Paletten fügen Sie die Objekte per Drag&Drop oder Doppelklick in Ihre Bilder ein. Die Auswahl an Objekten ist abhängig vom Bediengerät, das Sie gerade projektieren. Mit den folgenden Symbolen wechseln Sie den Anzeigemodus:

| Symbol  | Bedeutung                          |
|---|------------------------------------|
|  | Stellt die Objekte als Liste dar.  |
|  | Stellt die Objekte als Grafik dar. |

## Animationen

Die Task Card "Animationen" enthält in den Paletten die möglichen Dynamisierungen eines Bildobjekts. Aus den Paletten "Bewegungen", "Anzeige" und "Variablenanbindung" fügen Sie per Drag&Drop oder Doppelklick die Animation einem Bildobjekt hinzu.

## Layout

Die Task Card "Layout" enthält folgende Paletten zum Anzeigen von Objekten und Elementen:

- Ebenen: Dient der Verwaltung der Ebenen von Bildobjekten. Die Ebenen werden in einer Baumansicht gezeigt und enthalten Informationen zur aktiven Ebene und zur Sichtbarkeit aller Ebene.
- Raster: Sie geben an, ob Sie die Objekte an einem Raster oder an anderen Objekten ausrichten wollen und stellen für ein Raster die Rastergröße ein.
- Objekte außerhalb des Bereichs: Objekte mit Namen, Position und Typ werden angezeigt, die sich außerhalb des sichtbaren Bereichs befinden


## Bibliotheken

Die Task Card "Bibliotheken" zeigt folgende Bibliotheken in getrennten Paletten:

- Projektbibliothek: Die Projektbibliothek wird zusammen mit dem Projekt gespeichert.
- Globale Bibliothek: Die Globale Bibliothek wird als separate Datei im angegebenen Pfad auf dem Projektierungsrechner gespeichert.

## Ansicht verschieben

## Einleitung


Um im Arbeitsbereich nur ein Ausschnitt des gesamten Bildes darzustellen, verwenden Sie das Symbol  des Editors "Bilder".

## Voraussetzung

- Ein Bild ist geöffnet.
- Im Arbeitsbereich wird nur ein Ausschnitt des gesamten Bildes dargestellt.

## Vorgehen

Um eine Ansicht zu verschieben:

1. Klicken Sie in der unteren rechten Ecke des Arbeitsbereichs auf das Symbol  und drücken Sie die linke Maustaste.  
Eine Miniaturansicht des gesamten Bildes wird aufgeblendet. Ein orangefarbener Rahmen zeigt den aktuellen Ausschnitt.
2. Halten Sie die Maustaste gedrückt und verschieben Sie den Rahmen in den gewünschten Ausschnitt.

---

### Hinweis

Wenn Sie ein Bildobjekt aus dem sichtbaren Ausschnitt zu einem in der aktuellen Ansicht unsichtbaren Ausschnitt ziehen, scrollt das Bild mit.

---

## Ansicht zoomen

### Einleitung

Wenn Sie einen kleinen Ausschnitt des Bildes detaillierter sehen wollen, vergrößern Sie die Ansicht des Bildes im Arbeitsbereich. Maximal zoomen Sie auf 800 %.

Sie zoomen über die Symbolleiste im Arbeitsbereich oder über die Task Card "Layout > Zoom".


Sie haben verschiedene Methoden für die Vergrößerung des Bildes, z. B. über den Vergrößerungsfaktor oder Sie passen den Arbeitsbereich an Höhe des Bildes an.

### Voraussetzung

Das Bild ist geöffnet.

## Vorgehen

Um eine Ansicht mit Auswahlrahmen zu zoomen, gehen Sie wie folgt vor:

1. Klicken Sie in der Symbolleiste auf das Symbol .
2. Ziehen Sie mit der Maus im Bild einen Auswahlrahmen auf.

Nachdem Sie die Maustaste losgelassen haben, wird der Ausschnitt auf die Größe des gesamten Arbeitsbereichs vergrößert, der sich innerhalb des Auswahlrahmens befindet.

Alternativ verwenden Sie in der unteren rechten Ecke des Bildes den Schieberegler.

## Ergebnis

Der gewählte Bildausschnitt wird vergrößert angezeigt.

#### 10.1.1.4 Mit Bildern arbeiten

##### Arbeitsschritte

##### Arbeitsschritte

Zum Erstellen von Bildern sind folgende grundlegende Schritte nötig:

- Planen Sie die Struktur der Prozessdarstellung: Anzahl der Bilder und deren Anordnung.  
Beispiel: Teilprozesse werden in einzelnen Bildern dargestellt und in einem Hauptbild zusammengeführt.
- Planen Sie die Navigation innerhalb eines Bildes und zwischen den einzelnen Bildern.
- Passen Sie die Vorlagen und das Globale Bild an.  
Sie definieren Objekte zentral und belegen z. B. Funktionstasten.
- Erstellen Sie die Bilder. Nutzen Sie dabei folgende Möglichkeiten zum effizienten Erstellen:
  - Arbeiten mit Bibliotheken
  - Arbeiten mit Ebenen
  - Arbeiten mit Bildbausteinen

##### Neues Bild anlegen

##### Einleitung

Um Prozesse in Ihrer Anlage darzustellen, legen Sie Bilder an.

##### Voraussetzung

- Das Projekt ist angelegt.
- Das Inspektorfenster ist geöffnet.

## Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Bilder > Neues Bild hinzufügen". Das Bild wird im Projekt angelegt und im Arbeitsbereich angezeigt. Die Eigenschaften des Bildes werden im Inspektorfenster angezeigt.
2. Geben Sie einen aussagekräftigen Namen für das Bild ein.
3. Konfigurieren Sie die Bildeigenschaften im Inspektorfenster:
  - Legen Sie fest, ob und auf welcher Vorlage das Bild beruhen soll.
  - Legen Sie "Hintergrundfarbe" und "Bildnummer" fest.
  - Hinterlegen Sie unter "Tooltip" einen dokumentierenden Text.
  - Legen Sie unter "Ebenen" fest, welche Ebenen im Engineering System angezeigt werden sollen.
  - Dynamisieren Sie die Anzeige des Bildes unter "Animationen".
  - Legen Sie unter "Ereignisse" fest, welche Funktionen beim Aufrufen und Verlassen des Bildes oder bei anderen Ereignissen in Runtime abgearbeitet werden sollen.

---

### Hinweis

Nicht alle Bediengeräte unterstützen die Animation "Sichtbarkeit".

---

## Ergebnis

Das Bild ist in Ihrem Projekt angelegt. Sie können in weiteren Arbeitsschritten Objekte und Bedienelemente aus der Task Card "Werkzeuge" einfügen und Funktionstasten belegen.

## Bilder verwalten

### Einleitung

In der Projektnavigation können Sie Bilder innerhalb eines Projekts in andere Gruppen verschieben, kopieren, umbenennen oder löschen.

### Bild in Gruppe verschieben

1. Selektieren Sie in der Projektnavigation den Ordner "Bilder".
2. Wählen Sie im Kontextmenü den Befehl "Gruppe hinzufügen". Ein Ordner mit dem Namen "Gruppe\_x" wird eingefügt.
3. Selektieren Sie in der Projektnavigation das Bild.
4. Ziehen Sie das Bild mit Drag&Drop auf die gewünschte Gruppe. Das Bild wird in diese Gruppe verschoben.

## Bild kopieren

1. Selektieren Sie in der Projektnavigation das Bild.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren", um das Bild in die Zwischenablage zu kopieren.
3. Markieren Sie in der Projektnavigation die Stelle, an der Sie das Bild einfügen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen", um das Bild einzufügen.  
Eine Kopie des Bildes wird eingefügt. Der Name des Originals wird bei der Kopie um eine laufende Nummer erweitert.

Alternativ halten Sie die Taste <Strg> gedrückt und ziehen Sie das Bild an die gewünschte Stelle.

---

### Hinweis

Wenn Sie ein Bild mit verknüpfter Vorlage geräteübergreifend oder projektübergreifend kopieren, wird die Vorlage mitkopiert. Eine eventuell vorhandene, passende Vorlage wird nicht genutzt. Dies gilt insbesondere auch, wenn Sie Bilder über Drag&Drop kopieren.

---

## Bild umbenennen

1. Selektieren Sie in der Projektnavigation das Bild.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie einen neuen Namen ein.
4. Drücken Sie die Taste <Enter>.

Alternativ verwenden Sie zum Umbenennen des Bildes die Taste <F2>.

## Bild löschen

1. Selektieren Sie in der Projektnavigation das Bild.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".  
Das Bild mit allen enthaltenen Objekten wird aus dem aktuellen Projekt gelöscht.

## Startbild des Projekts definieren

### Einleitung

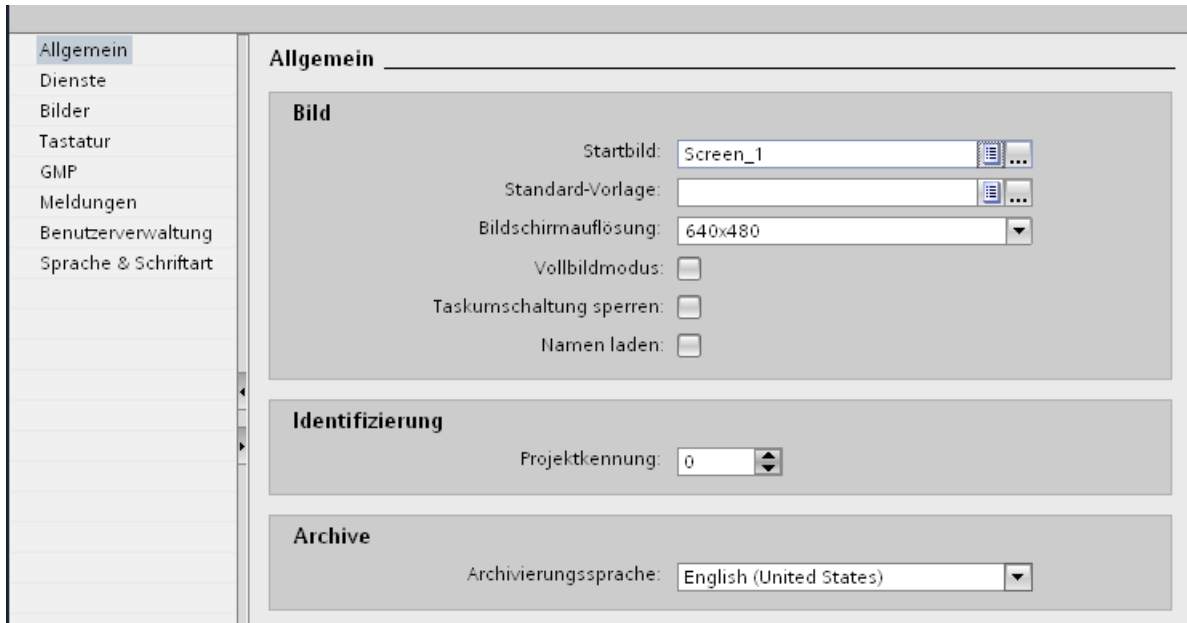
Das Startbild ist das erste Bild, das beim Start des Projekts in Runtime angezeigt wird. Sie können für jedes Bediengerät ein eigenes Startbild definieren. Vom Startbild aus ruft der Bediener die weiteren Bilder auf.

### Voraussetzung

Das Bild, das Startbild werden soll, ist im Projekt angelegt.

## Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen > Allgemein".



2. Wählen Sie als "Startbild" das gewünschte Bild aus.

Alternativ selektieren Sie ein Bild in der Projektnavigation und wählen im Kontextmenü "Als Startbild verwenden".

## Ergebnis

Wenn Sie Runtime starten, wird das Startbild am Bediengerät angezeigt.

### 10.1.1.5 Arbeiten mit Vorlagen

## Grundlagen zum Arbeiten mit Vorlagen

### Einleitung

In einer Vorlage projektieren Sie Objekte, die in allen Bildern dargestellt werden, die auf dieser Vorlage basieren.

Dabei gelten folgende Regeln:

- Ein Bild muss nicht auf einer Vorlage basieren.
- Ein Bild beruht nur auf einer Vorlage.
- Für ein Gerät können Sie mehrere Vorlagen erstellen.
- Eine Vorlage kann nicht auf einer anderen Vorlage basieren.



## Objekte für eine Vorlage

Innerhalb der Vorlage legen Sie Funktionen und Objekte fest, die für alle Bilder auf der Grundlage dieser Vorlage gelten sollen:

- **Belegung von Funktionstasten:** Bei Bediengeräten mit Funktionstasten belegen Sie auch Funktionstasten in der Vorlage. Diese Belegung überschreibt eine mögliche globale Belegung.
- **Permanentfenster:** Einige Geräte, unterstützen im oberen Bereich des Bildes ein Permanentfenster für alle Bilder. Im Unterschied zur Vorlage belegt das Permanentfenster einen Bereich des Bildes nur für sich.
- **Bedienobjekte:** Sie können alle Bildobjekte in eine Vorlage einfügen, die Sie auch für ein Bild nutzen.

## Anwendungsbeispiele

- In der Vorlage belegen Sie z. B. eine Funktionstaste mit der Funktion "AktiviereBild". Mit dieser Taste wechselt der Bediener in Runtime zu einem anderen Bild. Diese Projektierung gilt für alle Bilder, die auf dieser Vorlage basieren.
- In der Vorlage fügen Sie eine Grafik mit Ihrem Firmenlogo ein. Das Firmenlogo wird in allen Bildern angezeigt, die auf der Vorlage basieren.

---

### Hinweis

Wenn sich ein Objekt aus der Vorlage an derselben Position befindet wie ein Objekt im Bild, wird das Vorlagenobjekt verdeckt.

---

## Siehe auch

Neue Vorlage anlegen (Seite 3037)

Vorlagen verwalten (Seite 3038)

Globales Bild (Seite 3035)

Vorlage im Bild verwenden (Seite 3039)

## Globales Bild

### Einleitung

Für alle Bilder eines Bediengeräts legen Sie globale Elemente fest, unabhängig von der verwendeten Vorlage.

### Funktionstasten

Bei Bediengeräten mit Funktionstasten belegen Sie die Funktionstasten global innerhalb des Editors "Globales Bild". Diese globale Belegung gilt für alle Bilder des Bediengeräts.

Um Funktionstasten lokal in Bildern oder Vorlagen zu belegen, gehen Sie wie folgt vor:



- Eingabedialoge
- Meldungen vom Betriebssystem
- bei Touch Panels die Direkttasten

## Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 3034)

## Neue Vorlage anlegen

### Einleitung

In einer Vorlage bearbeiten Sie zentral Objekte und Funktionstasten. Wenn Sie in der Vorlage ein Objekt oder die Belegung einer Funktionstaste ändern, ändert sich das Objekt in allen Bildern, die auf der Vorlage basieren.

---

#### Hinweis

##### **Bediengeräteabhängigkeit**

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

---

### Voraussetzung

- Das Projekt ist angelegt.
- Das Inspektorfenster ist geöffnet.

### Vorgehen

1. Doppelklicken Sie in der Projektnavigation unter "Bilderverwaltung > Vorlagen > Neue Vorlage hinzufügen".  
Die Vorlage wird im Projekt angelegt und im Arbeitsbereich angezeigt.  
Die Eigenschaften der Vorlage werden im Inspektorfenster angezeigt.
2. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Namen der Vorlage fest
3. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Ebenen" fest, welche Ebenen im Engineering System angezeigt werden.
4. Fügen sie die benötigten Objekte aus der Task Card "Werkzeuge" hinzu.
5. Projektieren Sie die Funktionstasten.

### Ergebnis

Die Vorlage ist in Ihrem Projekt angelegt.

## Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 3034)

## Vorlagen verwalten

### Einleitung

Im Projektfenster verschieben, kopieren, umbenennen oder löschen Sie Vorlagen innerhalb eines Projekts.

### Vorlage in Gruppe verschieben

1. Selektieren Sie in der Projektnavigation "Bildverwaltung > Vorlagen".
2. Wählen Sie im Kontextmenü "Gruppe hinzufügen".  
Ein Ordner mit dem Namen "Gruppe\_x" wird eingefügt.
3. Selektieren Sie in der Projektnavigation die Vorlage.
4. Ziehen Sie die Vorlage mit Drag&Drop auf die gewünschte Gruppe.  
Die Vorlage wird in diese Gruppe verschoben.

### Vorlage kopieren

1. Selektieren Sie in der Projektnavigation die Vorlage
2. Wählen Sie im Kontextmenü "Kopieren".
3. Markieren Sie in der Projektnavigation die Stelle, an der Sie die Vorlage einfügen wollen.
4. Um die Vorlage einzufügen, wählen Sie im Kontextmenü "Einfügen".  
Die Kopie erhält automatisch einen eindeutigen Namen.

Alternativ ziehen Sie die Vorlage mit gedrückter Taste <Strg> an die gewünschte Stelle.

### Vorlage löschen

1. Selektieren Sie in der Projektnavigation die zu löschende Vorlage.
2. Wählen Sie im Kontextmenü "Löschen".  
Die Vorlage mit allen enthaltenen Objekten wird aus dem aktuellen Projekt gelöscht.

### Vorlage einem Bild zuweisen

1. Selektieren Sie in der Projektnavigation das Bild, dem Sie die Vorlage zuweisen wollen.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Wählen Sie unter "Vorlage" die gewünschte Vorlage aus.  
Die ausgewählte Vorlage mit allen enthaltenen Objekten wird dem Bild zugeordnet.

## Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 3034)

## Vorlage im Bild verwenden

### Einleitung

Sie verwenden eine Vorlage im Bild. Alle in der Vorlage projizierten Objekte sind auch im Bild verfügbar.

### Voraussetzung

Eine Vorlage ist angelegt.

Ein Bild ist angelegt.

### Vorgehensweise

Um eine Vorlage in einem Bild zu verwenden, gehen Sie wie folgt vor:

1. Doppelklicken Sie in der Projektnavigation auf ein Bild. Das Bild wird im Arbeitsbereich geöffnet.
2. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Wählen Sie unter "Vorlage" eine Vorlage, die auf das Bild angewendet wird.

## Vorlage im Bild einblenden

Wenn Sie ein Bild editieren, können Sie eine vorhandene Vorlage im Bild einblenden.

Um eine Vorlage im Bild einzublenden, gehen Sie wie folgt vor:

1. Aktivieren Sie im Menü "Extras > Einstellungen > Visualisierung > Vorlage in Bildern anzeigen".

## Ergebnis

Das Bild basiert auf der gewählten Vorlage. Alle Objekte, die Sie in der Vorlage projiziert haben, sind im Bild vorhanden. Die Vorlage wird im Bild angezeigt.

## Siehe auch

Grundlagen zum Arbeiten mit Vorlagen (Seite 3034)

## 10.1.2 Arbeiten mit Objekten

### 10.1.2.1 Übersicht über die Objekte

#### Einleitung

Objekte sind grafische Elemente, mit denen Sie die Bilder Ihres Projekts gestalten.

Die Task Card "Werkzeuge" enthält alle Objekte, die für das Bediengerät verfügbar sind. Sie blenden die Task Card mit dem Menübefehl "Ansicht" ein, indem Sie die Option "Task Card" aktivieren.

Abhängig vom aktuell geöffneten Editor enthält das Werkzeugfenster unterschiedliche Paletten. Wenn der Editor "Bilder" geöffnet ist, werden im Werkzeugfenster folgende Paletten bereitgestellt:

- "Basisobjekte"  
Zu den Basisobjekten gehören grundlegende grafische Objekte, z. B. "Linie", "Kreis", "Textfeld" oder "Grafikanzeige".
- "Elemente"  
Zu den Elementen gehören grundlegende Bedienelemente, z. B. "E/A-Feld", "Schaltfläche" oder "Zeigerinstrument".
- "Controls"  
Die Controls haben einen erweiterten Funktionsumfang. Sie stellen auch Prozessabläufe dynamisch dar, z. B. Kurvenanzeige und Rezepturanzeige.

- "Grafiken"
 

Grafiken sind thematisch in Form eines Verzeichnisbaums gegliedert. Die verschiedenen Ordner enthalten z. B. folgende grafische Darstellungen:

  - Maschinen- und Anlagenbereiche
  - Messgeräte
  - Bedienelemente
  - Flaggen
  - Gebäude

Sie erstellen Verknüpfungen zu Ihren eigenen Grafikordnern. Die externen Grafiken liegen in diesen Ordnern und Unterordnern. Sie werden im Werkzeugfenster angezeigt und über die Verknüpfung in das Projekt eingebunden.
- Task Card "Bibliotheken"
 

Zusätzlich zu den Anzeige- und Bedienobjekten stehen Ihnen die Bibliotheksobjekte zur Verfügung. Sie befinden sich innerhalb der Paletten der Task Card "Bibliotheken". Eine Bibliothek enthält fertig konfigurierte Objekte, z. B. Grafiken von Rohren, Pumpen, vorkonfigurierten Schaltflächen. Sie binden Bibliotheksobjekte auch mehrfach in Ihr Projekt ein, ohne sie jeweils neu zu konfigurieren.

Mit WinCC werden Bibliotheken mitgeliefert z. B. "HMI Buttons & Switches".







Sie hinterlegen auch benutzerdefinierte Objekte und Bildbausteine in eigenen Bibliotheken. Bildbausteine sind Objekte, die Sie aus vorhandenen Bildobjekten zusammenstellen und für die Sie projektierbare Eigenschaften festlegen.

**Hinweis**








**Bediengeräteabhängigkeit**

Abhängig vom Bediengerät, das Sie projektieren, sind einige Objekte aus dem Werkzeugfenster nicht oder nur mit eingeschränkter Funktionsweise verfügbar. Nicht verfügbare Eigenschaften eines Objekts werden deaktiviert dargestellt und sind nicht auswählbar.






**Basisobjekte**

| Symbol  | Objekt          | Hinweise   |
|---|-----------------|--|
|  | "Linie"         | -  |
|  | "Ellipse"       | -  |
|  | "Kreis"         | -  |
|  | "Rechteck"      | -  |
|  | "Textfeld"      | Stellt einzeiligen oder mehrzeiligen Text dar. Schrift und Darstellung sind einstellbar.   |
|  | "Grafikanzeige" | Zeigt Grafiken aus externen Grafikprogrammen an und fügt OLE-Objekte ein. Folgende Grafikformate können verwendet werden: <code>*.emf</code> , <code>*.wmf</code> , <code>*.dib</code> , <code>*.bmp</code> , <code>*.jpg</code> , <code>*.jpeg</code> , <code>*.gif</code> und <code>*.tif</code> . |

Elemente

| Symbol  | Objekt                  | Hinweise  |
|---|-------------------------|---|
|    | "E/A-Feld"              | Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable.<br>Sie legen Grenzwerte für die im E/A-Feld dargestellten Variablenwerte fest. Für eine unsichtbare Eingabe des Bedieners in Runtime projektieren Sie "Verdeckte Eingabe". |
|    | "Schaltfläche"          | Führt abhängig von der Projektierung eine Funktionsliste oder ein Skript aus.   |
|    | "Symbolisches E/A-Feld" | Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Abhängig vom Variablenwert wird ein Text aus einer Textliste angezeigt.  |
|    | "Grafisches E/A-Feld"   | Gibt Werte einer Variablen aus und/oder schreibt Werte in eine Variable. Abhängig vom Variablenwert wird eine Grafik aus einer Grafikliste angezeigt.   |
|    | "Datum/Uhrzeit-Feld"    | Gibt Datum und Uhrzeit der Systemzeit oder aus einer Variablen aus. Ermöglicht dem Bediener die Eingabe neuer Werte. Die Anzeigeform ist einstellbar.   |
|    | "Balken"                | Stellt einen Wert aus der Steuerung in Form einer Säule dar, die mit einer Skala versehen ist.  |
|  | "Schalter"              | Schaltet zwischen zwei definierten Zuständen um. Sie können einen Schalter mit Text oder mit einer Grafik beschriften.  |

Controls

| Symbol  | Objekt                   | Beschreibung  |
|---|--------------------------|---|
|  | "Meldeanzeige"           | Zeigt aktuell anstehende Meldungen oder Meldeereignisse aus dem Meldepuffer oder Meldearchiv an.  |
|  | "Kurvenanzeige"          | Stellt mehrere Kurven mit Werteverläufen aus der Steuerung oder aus einem Archiv dar.   |
|  | "Benutzeranzeige"        | Ermöglicht einem Administrator die Benutzerverwaltung am Bediengerät.<br>Ermöglicht einem Bediener ohne Administratorrechte, sein Kennwort zu ändern. |
|  | "Rezepturanzeige"        | Zeigt Datensätze an und ermöglicht die Bearbeitung der Datensätze.  |
|  | "System-Diagnoseanzeige" | Ermöglicht eine Übersicht über alle diagnosefähigen Geräte.<br>Zeigt fehler in der Anlage an.   |



## Siehe auch

- Objekt einfügen (Seite 3044)
- Objekt positionieren (Seite 3047)
- Größe eines Objekts ändern (Seite 3048)
- Mehrere Objekte auswählen (Seite 3050)
- Objekte ausrichten (Seite 3052)
- Objekt nach vorn oder hinten schieben (Seite 3054)
- Objekte außerhalb des Bildbereichs einblenden (Seite 3055)
- Objekt drehen (Seite 3055)
- Objekt spiegeln (Seite 3057)
- Mehrere Objekte eines Typs einfügen (stempeln) (Seite 3058)
- Position und Größe mehrerer Objekte verändern (Seite 3060)
- Externe Grafiken (Seite 3060)
- Externe Grafiken verwalten (Seite 3062)
- Externe Grafik in der Grafiksammlung hinterlegen (Seite 3063)
- Grundlagen zu Gruppen (Seite 3065)
- Übersicht zum Tastaturzugriff (Seite 3072)
- Beispiel: Rechteck einfügen und konfigurieren (Seite 3075)
- Objekte für Basic Panels (Seite 3127)

### 10.1.2.2 Bearbeitungsmöglichkeiten von Objekten

#### Einleitung

Objekte sind grafische Elemente, mit denen Sie die Bilder Ihres Projekts gestalten.

Sie haben folgende Möglichkeiten zum Bearbeiten von Objekten:

- Objekte über das Kontextmenü kopieren, einfügen oder löschen. Wenn Sie ein Objekt in ein Bild kopieren und das Bild bereits ein gleichnamiges Objekt enthält, wird der Name des Objekts geändert.
- Objekte in der Standardgröße einfügen oder beim Einfügen die Größe der Objekte bestimmen
- Objekt vor oder hinter andere Objekte schieben
- Objekt drehen
- Objekte spiegeln
- Tab-Reihenfolge für Objekte festlegen
- Stempeln: Mehrere Objekte gleichen Typs einfügen
- Mehrere Objekte gleichzeitig markieren

## 10.1 Bilder erstellen

- Position und Größe mehrerer Objekte verändern
- Objekten weisen Sie externe Grafiken zu, z. B. in die Grafikanzeige.  
Nur die Grafiken werden angezeigt, die vorher in der Grafiksammlung des WinCC Projekts hinterlegt wurden.  
In der Grafiksammlung hinterlegen Sie Grafiken:
  - Per Drag&Drop aus der Objektgruppe "Grafiken" in den Arbeitsbereich
  - Als Grafik-Datei folgender Formate: \*.bmp, \*.dib, \*.ico, \*.emf, \*.wmf, \*.gif, \*.tif, \*.jpeg oder \*.jpg
  - Als OLE-Objekt  
Das OLE-Objekt erstellen Sie entweder neu oder Sie hinterlegen eine vorhandene Grafik-Datei als OLE-Objekt. Um OLE-Objekte zu hinterlegen, muss auf dem Projektierungsrechner ein OLE-fähiges Grafikprogramm installiert sein.

### Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.3 Objekt einfügen

#### Einleitung

Im Editor "Bilder" oder "Protokolle" fügen Sie die Objekte der Task Card "Werkzeuge" ein. Sie ziehen die Objekte mit der Maus in den Arbeitsbereich. Sie fügen die Objekte in der Standardgröße ein oder bestimmen die Größe beim Einfügen.

Darüber hinaus können Sie Objekte über die Zwischenablage von einem Editor in einen anderen Editor kopieren oder verschieben, z. B. um ein Bildobjekt in ein Protokoll zu übernehmen. Alternativ können Sie zum Kopieren und Verschieben statt der Zwischenablage auch die Maus verwenden:

- Kopieren: <Strg + Drag&Drop>
- Verschieben: Drag&Drop

---

#### Hinweis

##### Basic Panels

Der Editor "Protokolle" steht für Basic Panels nicht zur Verfügung.

---

#### Voraussetzung

Die Task Card "Werkzeuge" ist geöffnet.

## Objekt in Standardgröße einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt oder die gewünschte Grafik im WinCC Grafikordner.  
Wenn Sie den Mauszeiger im Arbeitsbereich bewegen, wird er zu einem Fadenkreuz mit angehängtem Objektsymbol.
2. Klicken Sie auf die Stelle im Arbeitsbereich, an der Sie das Objekt oder die Grafik einfügen wollen.  
Das Objekt wird in der Standardgröße an der gewünschten Position im Arbeitsbereich eingefügt.

Alternativ doppelklicken Sie in der Task Card "Werkzeuge" auf ein Objekt.

## Objekt kopieren

1. Markieren Sie das gewünschte Objekt.
2. Wählen Sie im Kontextmenü "Kopieren".
3. Klicken Sie an die gewünschte Stelle und wählen Sie im Kontextmenü "Einfügen".

WinCC fügt eine Kopie des Objekts an der gewünschten Stelle ein. Sie können nur die Eigenschaften ändern, die in dem jeweiligen Kontext angemessen sind.

Beispiel: Im Editor "Bilder" legen Sie für E/A-Felder fest, legen Sie den Modus für Ein- und Ausgabe fest. Im Editor "Protokolle" ist der Modus auf "Ausgabe" festgelegt.

Original und Kopie sind nicht miteinander verbunden und werden unabhängig voneinander konfiguriert.

## Linien einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt.
2. Klicken Sie an einer Stelle im Arbeitsbereich. Eine Linie wird in der Standardgröße eingefügt.

## Polygon oder Polygonzug einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das gewünschte Objekt "Polygonzug" oder "Polygon".
2. Klicken Sie an einer Stelle im Arbeitsbereich. Der Anfangspunkt des Objekts ist festgelegt.
3. Klicken Sie an eine andere Stelle im Arbeitsbereich. Ein Eckpunkt ist festgelegt.
4. Für jeden weiteren Eckpunkt klicken Sie im Arbeitsbereich an die entsprechende Stelle.
5. Doppelklicken Sie an einer Stelle im Arbeitsbereich. Der letzte Eckpunkt ist festgelegt. Alle Punkte des Polygons oder Polygonzugs sind festgelegt.

---

### Hinweis


#### Basic Panels

Die Objekte "Polygonzug" und "Polygon" stehen für Basic Panels nicht zur Verfügung.

---

---

### Hinweis

Um nacheinander mehrere Objekte gleichen Typs einzufügen, nutzen Sie die Funktion "Stempel". Sie vermeiden dadurch, das Objekt vor jedem Einfügen erneut in der Task Card "Werkzeuge" zu markieren. Wählen Sie dazu in der Symbolleiste der Task Card "Werkzeuge" das Symbol .

---

### Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.4 Objekt löschen

#### Einleitung

Sie löschen Objekte einzeln oder über eine Mehrfachauswahl.

#### Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

#### Vorgehen

1. Selektieren Sie das Objekt, das Sie löschen wollen.  
Um mehrere Objekte zu löschen, halten Sie die Taste <Shift> gedrückt und markieren Sie nacheinander die zu löschenden Objekte. Alternativ ziehen Sie mit der Maus einen Bereich um die gewünschten Objekte auf.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

#### Ergebnis

Die selektierten Objekte werden gelöscht.

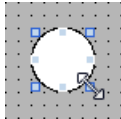
### Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.5 Objekt positionieren

#### Einleitung

Wenn Sie ein Objekt markieren, wird das Objekt von einem Rechteck mit Anfassern umrahmt. Dieses Rechteck ist das Objektumfassendes Rechteck. Die Position eines Objekts ist definiert durch die Koordinaten der linken oberen Ecke des Objektumfassenden Rechtecks.



---

#### Hinweis

Wenn die Position außerhalb des Arbeitsbereichs liegt, wird das Objekt in Runtime nicht dargestellt.

---

#### Positionieren und Ausrichten

Sie haben die Möglichkeit im Arbeitsbereich ein Raster anzeigen zu lassen. Um Objekte einfacher zu positionieren, stehen Ihnen drei Optionen zur Verfügung:

- "Am Raster ausrichten": Wenn Sie Objekte bewegen, werden sie automatisch am Raster ausgerichtet und eingefügt. Wenn Sie gleichzeitig die Taste <Alt> gedrückt halten, wird das Objekt nicht am Raster ausgerichtet.
- "An Objekten ausrichten": Wenn Sie Objekte bewegen, werden Ihnen Hilfslinien angezeigt. Beim Positionieren können Sie sich an anderen Objekten orientieren.
- "Kein": Sie positionieren die Objekte an beliebiger Position.

Sie aktivieren und deaktivieren das Raster und die Optionen wie folgt:

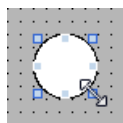
- Im Menü "Extras > Einstellungen > Visualisierung > Bilder"
- In der Task Card "Layout > Raster"

#### Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

#### Vorgehen

1. Selektieren Sie das Objekt, das Sie verschieben wollen.  
Das markierte Objekt wird von einem Rechteck mit Anfassern umrahmt.



2. Klicken Sie auf das Objekt und halten Sie die linke Maustaste gedrückt.

3. Bewegen Sie den Mauszeiger auf die neue Position.  
Der Umriss des Objekts bewegt sich mit der Maus und zeigt die neue Position für das Objekt an.



Das Objekt selbst behält zunächst seine ursprüngliche Position.

4. Lassen Sie die Maustaste los.  
Das Objekt wird auf die Position verschoben, die vorher durch den Objektumriss angezeigt wurde.

### Alternatives Vorgehen

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Position & Größe" die X-Werte und Y-Werte der Position ein.

### Ergebnis

Das Objekt wird an der neuen Position angezeigt.

### Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.6 Größe eines Objekts ändern

#### Einleitung

Wenn Sie ein Objekt markieren, wird das Objekt von einem objektumfassenden Rechteck mit Anfassern umrahmt. Sie haben folgende Möglichkeiten, die Größe eines Objekts zu ändern:

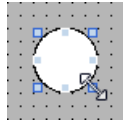
- Die Anfasser mit der Maus ziehen
- Eigenschaft "Größe" im Inspektorfenster ändern

#### Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

## Vorgehen

1. Wählen Sie das Objekt, dessen Größe Sie ändern wollen.  
Das objektumfassende Rechteck wird angezeigt. Das folgende Bild zeigt ein markiertes Objekt:






2. Ziehen Sie einen Anfasser des Rechtecks an eine neue Position.  
Die Größe des Objekts ändert sich.
  - Wenn die Funktion "Am Raster ausrichten" aktiviert ist, ändert sich die Objektgröße entsprechend den Rasterpunkten.
  - Wenn Sie beim Ziehen des Anfassers die Taste <Alt> gedrückt halten, wird diese Funktion nicht verwendet.  
Um das Objekt proportional zu skalieren, halten Sie die Taste <Shift> gedrückt, während Sie mit der Maus die Größe ändern.

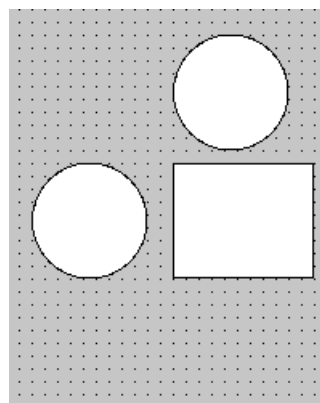
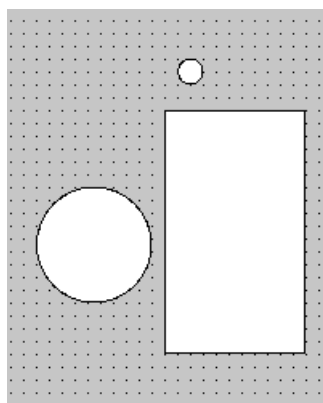
## Alternatives Vorgehen




1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Position & Größe" die Größe des Objekts ein.

## Objektgröße vereinheitlichen

1. Wählen Sie die Objekte aus.
2. Klicken Sie auf eine der folgenden Schaltflächen:  oder  oder   
Die Größe der ausgewählten Objekte wird aneinander angepasst.

Das folgende Bild zeigt, wie Sie die ausgewählten Objekte an die Höhe des Referenzobjekts anpassen:



| Symbol  | Beschreibung  |
|---|---|
|  | Passt die ausgewählten Objekte an die Breite des Referenzobjekts an.              |
|  | Passt die ausgewählten Objekte an die Höhe des Referenzobjekts an.                |
|  | Passt die ausgewählten Objekte an die Breite und die Höhe des Referenzobjekts an. |

## Ergebnis

Das Objekt wird in der neuen Größe angezeigt.

## Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.7 Mehrere Objekte auswählen

## Einleitung

Um die Eigenschaften von mehreren Objekten zugleich zu ändern oder die Objekte aneinander auszurichten, wählen Sie alle betreffenden Objekte aus. Dieser Vorgang heißt "Mehrfachauswahl".

Im Inspektorfenster erscheinen alle Eigenschaften der markierten Objekte.

Sie haben folgende Möglichkeiten, mehrere Objekte zu markieren:

- Sie ziehen einen Auswahlrahmen um die gewünschten Objekte.
- Sie klicken mit gedrückter Taste <Shift> auf die gewünschten Objekte.

## Auswahlrahmen einer Mehrfachauswahl

Der Auswahlrahmen umschließt alle Objekte einer Mehrfachauswahl. Der Auswahlrahmen ist vergleichbar mit dem objektumfassenden Rechteck eines einzelnen Objekts.

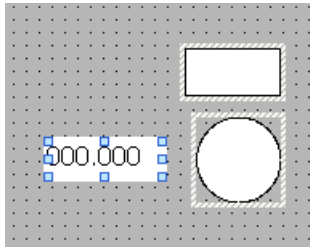
Der Auswahlrahmen ist nicht sichtbar. Nach Abschluss der Mehrfachauswahl werden folgende Rahmen angezeigt:

- Das Referenzobjekt wird mit dem objektumfassenden Rechteck angezeigt.
- Die anderen ausgewählten Objekte werden mit einem gestrichelten Rahmen angezeigt.



## Referenzobjekt festlegen

Das Referenzobjekt ist das Objekt, an dem die anderen Objekte ausgerichtet werden. Das Referenzobjekt ist von einem Rechteck mit Anfassern umrahmt. Das folgende Bild zeigt ein Referenzobjekt mit zwei weiteren ausgewählten Objekten:



Um das Referenzobjekt festzulegen, haben Sie folgende Möglichkeiten:

- Sie wählen die Objekte über Mehrfachauswahl aus. Das zuerst ausgewählte Objekt ist dann das Referenzobjekt.
- Sie ziehen einen Auswahlrahmen um die gewünschten Objekte. Das Referenzobjekt wird automatisch festgelegt. Wenn Sie ein anderes Objekt innerhalb der Auswahl als Referenzobjekt festlegen wollen, klicken Sie auf das gewünschte Objekt. Die Mehrfachauswahl wird dadurch nicht aufgehoben.

## Voraussetzung

Der Arbeitsbereich mit mindestens zwei Objekten ist geöffnet.

## Mehrere Objekte mit einem Auswahlrahmen auswählen

1. Positionieren Sie den Mauszeiger im Arbeitsbereich in der Nähe eines der auszuwählenden Objekte.
2. Ziehen Sie bei gedrückter Maustaste einen Auswahlrahmen um die auszuwählenden Objekte.

Oder:

1. Halten Sie die Taste <Shift> gedrückt.
2. Klicken Sie mit der Maus nacheinander auf die gewünschten Objekte. Alle ausgewählten Objekte werden durch Rahmen markiert. Das zuerst markierte Objekt wird als Referenzobjekt gekennzeichnet.

---

### Hinweis

Sie entfernen ein Objekt aus der Mehrfachauswahl, indem Sie die Taste <Shift> gedrückt halten und das Objekt erneut anklicken.

---

## Ergebnis

Mehrere Objekte sind markiert. Eines davon ist als Referenzobjekt gekennzeichnet. Sie können jetzt folgende Schritte ausführen:

- Die Objekteigenschaften aller Objekte verändern
- Die Größe aller Objekte um einen gleichen Faktor ändern, indem Sie mit der Maus den Auswahlrahmen entsprechend größer oder kleiner ziehen
- Alle Objekte gemeinsam verschieben
- Die Objekte am Referenzobjekt ausrichten

## Siehe auch

Übersicht über die Objekte (Seite 3040)









### 10.1.2.8 Objekte ausrichten

#### Vorgehen

1. Wählen Sie die gewünschten Objekte über Mehrfachauswahl aus.
2. Legen Sie ein Objekt als Referenzobjekt fest.
3. Wählen Sie in der Symbolleiste oder im Kontextmenü den gewünschten Befehl - siehe Tabelle unten.  
Die ausgewählten Objekte werden ausgerichtet.

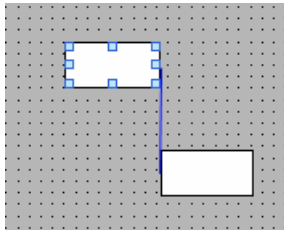
#### Objekte bündig ausrichten

Die ausgewählten Objekte werden bündig am Referenzobjekt ausgerichtet.

| Symbol  | Beschreibung  |
|---|---|
|  | Richtet die ausgewählten Objekte an der linken Kante des Referenzobjekts aus.             |
|  | Richtet die ausgewählten Objekte an der vertikalen Mittelachse des Referenzobjekts aus.   |
|  | Richtet die ausgewählten Objekte an der rechten Kante des Referenzobjekts aus.            |
|  | Richtet die ausgewählten Objekte an der Oberkante des Referenzobjekts aus.                |
|  | Richtet die ausgewählten Objekte an der horizontalen Mittelachse des Referenzobjekts aus. |
|  | Richtet die ausgewählten Objekte an der Unterkante des Referenzobjekts aus.               |
|  | Zentriert die ausgewählten Objekte auf dem Mittelpunkt des Referenzobjekts.               |
|  | Zentriert die ausgewählten Objekte vertikal im Bild.                                      |

### An einem Objekt ausrichten

Wenn Sie Objekte bewegen, werden Ihnen Hilfslinien angezeigt. Beim Positionieren können Sie sich an anderen Objekten orientieren.



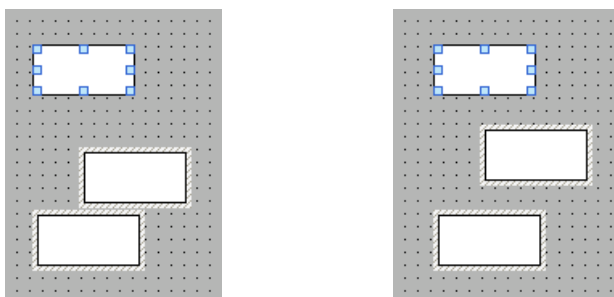
Wenn Sie mit der Tastatur arbeiten, drücken Sie die Alt-Taste. Wenn Sie das gewählte Objekt mit den Pfeiltasten bewegen, wird Ihnen der nächste Ankerpunkt angezeigt.



### Objekte gleichmäßig verteilen

Sie benötigen mindestens drei ausgewählte Objekte. Ein Referenzobjekt wird nicht benötigt.

1. Wählen Sie die Objekte aus.
2. Klicken Sie auf eine der Schaltflächen "Horizontal gleichmäßig verteilen" oder "Vertikal gleichmäßig verteilen".  
Die ausgewählten Objekte werden in gleichmäßigen Abständen verteilt.

Das folgende Bild zeigt, wie Sie den vertikalen Abstand zwischen den ausgewählten Objekten vereinheitlichen:



| Symbol  | Beschreibung   |
|---|--|
|  | Vereinheitlicht den horizontalen Abstand zwischen den Objekten.<br>Die Lage der äußersten Objekte (rechts und links) bleibt unverändert. Die übrigen Objekte werden in gleichmäßigen Abständen dazwischen verteilt.            |
|  | Vereinheitlicht den vertikalen Abstand zwischen den Objekten.<br>Die Lage des obersten und untersten Objekts (rechts und links) bleibt unverändert. Die übrigen Objekte werden in gleichmäßigen Abständen dazwischen verteilt. |

### Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.9 Objekt nach vorn oder hinten schieben

#### Einleitung

Mit den Funktionen "Reihenfolge" im Kontextmenü zu einem ausgewählten Objekt oder in der Symbolleiste schieben Sie das Objekt innerhalb einer Ebene vor oder hinter andere Objekte.

---

#### Hinweis

ActiveX-Controls liegen immer vor anderen Objekten einer Ebene (.NET-Eigenschaft).





---

#### Voraussetzung

Ein Bild mit mehreren Objekten auf einer Ebene ist geöffnet.

#### Vorgehen

1. Wählen Sie das Objekt, das Sie nach vorn oder nach hinten schieben wollen.
2. Wählen Sie im Kontextmenü den Befehl "Reihenfolge" und einen der folgenden Befehle:

| Symbol  | Beschreibung   |
|---|--|
|  | Schiebt das markierte Objekt vor alle anderen Objekte derselben Ebene    |
|  | Schiebt das markierte Objekt hinter alle anderen Objekte derselben Ebene |
|  | Schiebt das markierte Objekt um eine Position nach vorn                  |
|  | Schiebt das markierte Objekt um eine Position nach hinten                |

#### Alternatives Vorgehen

1. Öffnen Sie die Palette "Ebenen" der Task Card "Layout".
2. Navigieren Sie zum gewünschten Objekt.
3. Ziehen Sie das Objekt in der Baumstruktur mit gedrückter Maustaste an die gewünschte Position innerhalb der Ebene.
4. Lassen Sie die Maustaste los.

#### Ergebnis

Das Objekt wird nach vorn oder nach hinten geschoben.

#### Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.10 Objekte außerhalb des Bildbereichs einblenden

#### Einleitung


Wenn Sie Objekten Positionen zuweisen, die sich außerhalb des projektierbaren Bereichs befinden, werden diese Objekte ausgeblendet. Mit den Funktionen der Palette "Objekte außerhalb des sichtbaren Bereichs" in der Task Card "Layout" schieben Sie diese Objekte zurück in das Bild.

#### Voraussetzung

- Ein Bild mit Objekten außerhalb des projektierbaren Bereichs ist geöffnet.
- Die Task Card "Layout" ist geöffnet.

#### Vorgehen

1. Öffnen Sie die Task Card "Layout > Objekte außerhalb des Bereichs". Die Liste der Objekte, die sich außerhalb des projektierbaren Bereichs befinden, wird angezeigt.
2. Selektieren Sie in der Liste das Objekt, das Sie zurück in das Bild schieben wollen.
3. Wählen Sie im Kontextmenü des Objekts "Ins Bild ziehen".

Alternativ öffnen Sie die Task Card "Layout > Ebene". Objekte außerhalb des Bereichs werden mit dem Symbol  dargestellt. Wenn Sie auf dieses Symbol klicken, wird das Objekt wieder in das Bild verschoben.

#### Ergebnis

Die Objekte werden in den projektierbaren Bereich gezogen.

#### Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.11 Objekt drehen

#### Einleitung

Sie können ein geeignetes Objekt in Schritten von 90° im oder gegen den Uhrzeigersinn um sich selbst drehen.

---

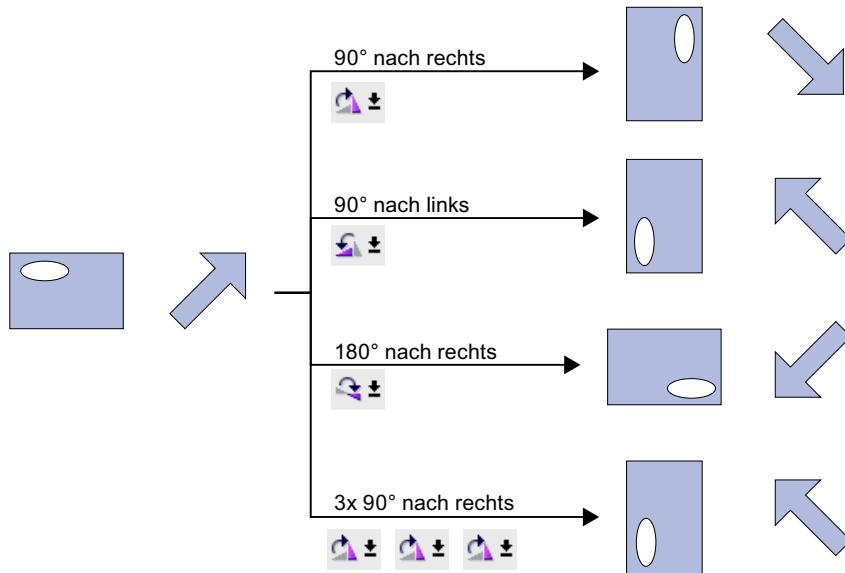
#### Hinweis

Nicht alle Objekte sind drehbar. Einige Objekte, die in Bildern drehbar sind, können in Protokollen nicht gedreht werden.

---

Mithilfe der Mehrfachauswahl drehen Sie auch mehrere Objekte zugleich. Bestimmte WinCC Objekte, z. B. Schaltflächen, lassen sich nicht drehen.

In einem gedrehten Objekt ändert sich die Ausrichtung der im Objekt enthaltenen Elemente. In der folgenden Abbildung sehen Sie, wie sich ein Rechteck und eine Ellipse mit den verschiedenen Befehlen zum Drehen eines Objekts verhalten:



### Voraussetzung

Der Arbeitsbereich mit mindestens einem Objekt ist geöffnet.

### Vorgehen

1. Selektieren Sie das Objekt, das Sie drehen wollen.
2. Klicken Sie in der Symbolleiste auf eines der folgenden Symbole:
  - , um das Objekt im Uhrzeigersinn um seinen Mittelpunkt zu drehen. Der Drehwinkel beträgt 90°.
  - , um das Objekt gegen den Uhrzeigersinn um seinen Mittelpunkt zu drehen. Der Drehwinkel beträgt 90°.
  - , um das Objekt im Uhrzeigersinn um 180° zu drehen.

### Ergebnis

Das Objekt wird gedreht angezeigt.

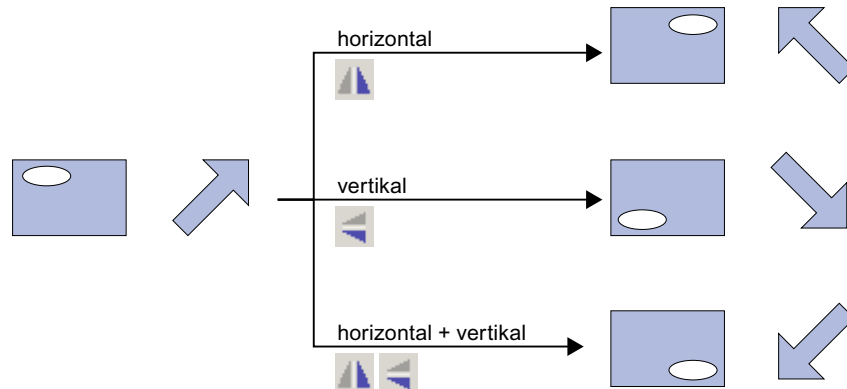
### Siehe auch

- Übersicht über die Objekte (Seite 3040)
- Ellipse (Seite 3133)

### 10.1.2.12 Objekt spiegeln

#### Einleitung



Sie können ein Objekt an seiner vertikalen oder horizontalen Mittelachse spiegeln. Wenn Sie ein Objekt spiegeln, ändert sich auch die Ausrichtung von im Objekt enthaltenen Elementen. In der folgenden Abbildung sehen Sie, wie sich ein Rechteck und eine Ellipse mit den verschiedenen Befehlen zum Spiegeln eines Objekts verhalten.



#### Voraussetzung

Ein Bild mit mindestens einem Objekt ist geöffnet.

#### Vorgehen

1. Wählen Sie das Objekt, das Sie spiegeln wollen.
2. Klicken Sie im Kontextmenü auf den Befehl "Spiegeln" und wählen Sie eine der angebotenen Möglichkeiten:
  - , um das markierte Objekt an seiner vertikalen Mittelachse zu spiegeln.
  - , um das markierte Objekt an seiner horizontalen Mittelachse zu spiegeln.

#### Ergebnis

Das Objekt wird gespiegelt angezeigt.

#### Siehe auch

Übersicht über die Objekte (Seite 3040)

Ellipse (Seite 3133)

### 10.1.2.13 Objekt gestalten

#### Einleitung

Sie gestalten Rahmen und Hintergrund von einem Objekt.

#### Voraussetzung

Eine Linie ist in einem Bild angelegt.

#### Vorgehen

1. Markieren Sie die Linie im Bild.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
3. Wählen Sie als Stil "Strich".
4. Um die gestrichelte Linie zweifarbig darzustellen, wählen Sie für die Linienbreite "1".
5. Wählen Sie im Bereich "Linienenden" die Einstellung "Pfeil".

#### Ergebnis

Die Line wird zweifarbig und gestrichelt dargestellt. Das Linienende ist ein Pfeil.

### 10.1.2.14 Mehrere Objekte eines Typs einfügen (stempeln)


#### Einleitung

WinCC bietet die Möglichkeit, mehrere Objekte des gleichen Typs direkt hintereinander zu "stempeln" d. h. einzufügen, ohne dabei jedesmal das Objekt neu auszuwählen. Außerdem haben Sie die Möglichkeit, ein bereits eingefügtes Objekt beliebig zu vervielfachen.


#### Voraussetzung

Die Task Card "Werkzeuge" ist geöffnet.

#### Mehrere Objekte eines Typs einfügen

1. Wählen Sie in der Task Card "Werkzeuge" das Objekt, das Sie einfügen wollen.
2. Klicken Sie in der Symbolleiste der Task Card "Werkzeuge" auf das Symbol . Die Funktion "Stempel" ist aktiviert.



3. Wenn Sie das Objekt in Standardgröße einfügen wollen, klicken Sie auf die gewünschte Position im Arbeitsbereich.  
Wenn Sie das Objekt in einer anderen Größe einfügen wollen, positionieren Sie den Mauszeiger auf die gewünschte Position im Arbeitsbereich. Ziehen Sie das Objekt bei gedrückter linker Maustaste auf die gewünschte Größe.  
Sobald Sie die Maustaste loslassen, ist das Objekt in den Arbeitsbereich eingefügt.
4. Um weitere Objekte desselben Typs einzufügen, wiederholen Sie Schritt 3.
5. Klicken Sie erneut auf das Symbol .  
Die Funktion "Stempel" ist deaktiviert.

---

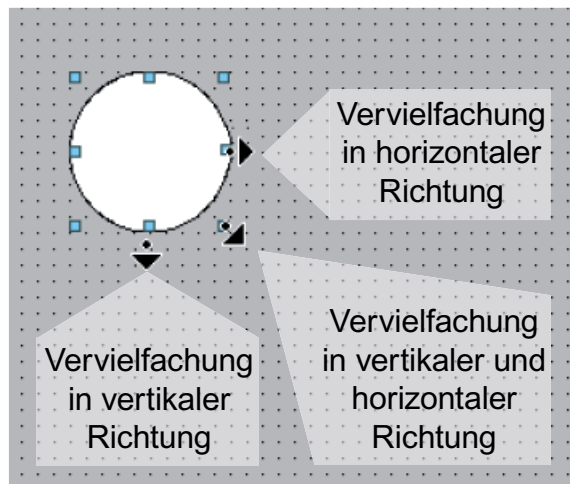
### Hinweis

Vorhandene Objekte kopieren Sie, indem Sie Drag&Drop nutzen und gleichzeitig die Taste <Strg> drücken. Das vorhandene Objekt wird dann nicht verschoben. Stattdessen wird eine Kopie davon an der neuen Position eingefügt.

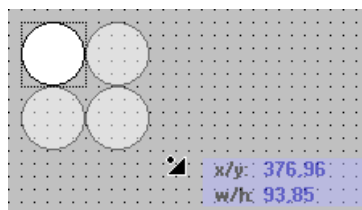
---

### Objekt einfügen und vervielfachen

1. Fügen Sie das gewünschte Objekt aus der Task Card "Werkzeuge" ein.
2. Drücken Sie die Taste <Strg> und positionieren Sie den Mauszeiger auf einen der im folgenden Bild dargestellten Anfasser.



3. Ziehen Sie die Anfasser bei gedrückter linker Maustaste nach rechts und/oder unten.
4. Das Objekt wird dem verfügbaren Platz entsprechend vervielfacht, je weiter Sie den Mauszeiger bewegen.



## Ergebnis

Sie haben ein Objekt in einem Bild eingefügt und vervielfältigt.

## Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.15 Position und Größe mehrerer Objekte verändern

#### Mögliche Änderungen

Nachdem Sie mehrere Objekte ausgewählt haben, bearbeiten Sie diese:

- Mit der Maus verschieben
  - Um die absolute Lage der markierten Objekte zu verändern, positionieren Sie den Mauszeiger über ein Objekt und verschieben bei gedrückter Maustaste die Mehrfachauswahl.
  - Um die Größe aller Objekte um den gleichen Faktor zu ändern, ziehen Sie mit der Maus an den Anfassern des Referenzobjekts.
- Mit den Symbolen der Symbolleiste über dem Arbeitsbereich verschieben
  - Die Lage der markierten Objekte zueinander ändern
  - Die Höhe und Breite der markierten Objekte angleichen
- Mit den Befehlen im Kontextmenü des Arbeitsbereichs verschieben
  - Die Lage der markierten Objekte zueinander ändern
  - Die Höhe und Breite der markierten Objekte angleichen

## Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.16 Externe Grafiken

#### Einleitung

In WinCC können Sie Grafiken verwenden, die mit einem externen Grafikprogramm erstellt wurden. Um diese Grafiken zu verwenden, hinterlegen Sie diese in der Grafiksammlung des WinCC Projekts.

In der Grafiksammlung hinterlegen Sie Grafiken:

- Wenn Sie grafische Objekte per Drag&Drop aus der Palette "Grafiken" in den Arbeitsbereich ziehen, werden diese automatisch in der Grafiksammlung hinterlegt. Die Grafiknamen werden in der Reihenfolge ihrer Erstellung nummeriert, z. B. "Grafik\_1". Um die Grafik umzubenennen, verwenden Sie die Taste <F2>.
- Als Grafikdatei mit folgenden Formaten:  
\*.bmp, \*.ico, \*.emf, \*.wmf, \*.gif, \*.tif, \*.png, \*.jpeg oder \*.jpg
- Als OLE-Objekt, das mit einem externen Grafikprogramm verknüpft ist und in WinCC eingebettet wird. Bei einer OLE-Verknüpfung öffnen Sie von WinCC aus das externe Grafikprogramm. Mit dem Grafikprogramm bearbeiten Sie das verknüpfte Objekt. Nur wenn das externe Grafikprogramm OLE unterstützt und wenn das Grafikprogramm auf dem Projektierungsrechner installiert ist, funktioniert eine OLE-Verknüpfung.

## Verwendung von Grafiken aus der Grafiksammlung

Grafiken aus der Grafiksammlung verwenden Sie in Ihren Bildern:

- In einer Grafikanzeige
- In einer Grafikliste
- Als Beschriftung für eine Schaltfläche/Funktionstaste

## Transparente Grafiken

In WinCC verwenden Sie auch Grafiken mit transparentem Hintergrund. Wenn Sie eine Grafik mit transparentem Hintergrund in einem Grafikobjekt von WinCC einfügen, wird die Transparenz durch die im Grafikobjekt festgelegte Hintergrundfarbe ersetzt. Die gewählte Hintergrundfarbe wird fest mit der Grafik verbunden. Wenn Sie die Grafik in einem weiteren Grafikobjekt von WinCC verwenden, wird diese mit der gleichen Hintergrundfarbe dargestellt wie an dem zuerst projektierten Grafikobjekt. Wenn Sie die Grafik mit unterschiedlichen Hintergrundfarben verwenden wollen, nehmen Sie diese Grafik unter einem anderen Namen erneut in die Grafiksammlung auf. Die zusätzliche Hintergrundfarbe projizieren Sie bei der Verwendung der Grafik am entsprechenden Grafikobjekt von WinCC.

## Grafiken verwalten

Mit WinCC wird eine umfangreiche Sammlung von Grafiken und Symbolen installiert, z. B.:

Im Werkzeugfenster in der Palette "Grafik" sind die grafischen Objekte im Ordner "WinCC Grafikordner" thematisch gegliedert. Die Verknüpfung zum Grafikordner von WinCC lässt sich nicht entfernen, bearbeiten oder umbenennen.

Über die Palette "Grafiken" verwalten Sie auch ihre externen Grafiken. Folgende Möglichkeiten stehen zur Verfügung:

- Verknüpfung zu Grafikordnern anlegen  
Die externen Grafiken, die in diesem Ordner und in den Unterordner liegen, werden im Werkzeugfenster angezeigt und so in das Projekt eingebunden.
- Ordnerverknüpfungen bearbeiten
- Das Programm für die Bearbeitung der externen Grafik öffnen Sie von WinCC aus.

## Siehe auch

Übersicht über die Objekte (Seite 3040)

### 10.1.2.17 Externe Grafiken verwalten

#### Einleitung

Externe Grafiken, die Sie in WinCC verwenden wollen, verwalten Sie im Editor "Bilder" über die Task Card "Werkzeuge" in der Palette "Grafiken".

#### Voraussetzung

- Der Editor "Bilder" ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.
- Die Grafiken liegen vor.
- Die Grafiken besitzen folgende Formate:  
\*.bmp, \*.ico, \*.emf, \*.wmf, \*.gif, \*.tif, \*.jpeg oder \*.jpg

#### Ordnerverknüpfung anlegen

1. Klicken Sie auf "Eigene Grafikordner".
2. Wählen Sie im Kontextmenü "Verknüpfung...".  
Der Dialog "Verknüpfung zu Ordner erstellen" wird geöffnet. Für die neue Ordnerverknüpfung wird ein Name vorgeschlagen.
3. Ändern Sie bei Bedarf den Namen. Wählen Sie den Pfad aus, in dem die Grafiken abgelegt sind.
4. Bestätigen Sie die Eingabe mit "OK".  
Die neue Ordnerverknüpfung wird in der Objektgruppe "Grafiken" hinzugefügt. Im Werkzeugfenster werden die externen Grafiken angezeigt, die im Zielordner und in Unterordner liegen.

#### Ordnerverknüpfungen bearbeiten

1. Markieren Sie die Ordnerverknüpfung, die Sie bearbeiten wollen.
2. Wählen Sie im Kontextmenü den Befehl "Verknüpfung bearbeiten...".  
Der Dialog "Verknüpfung zu Ordner bearbeiten" wird geöffnet.
3. Ändern Sie bei Bedarf den Namen der Ordnerverknüpfung und den Pfad.
4. Bestätigen Sie die Eingabe mit "OK".

### Ordnerverknüpfungen umbenennen

1. Markieren Sie die Ordnerverknüpfung, die Sie umbenennen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie der neuen Ordnerverknüpfung einen Namen.

### Ordnerverknüpfungen entfernen

1. Markieren Sie die Ordnerverknüpfung, die Sie löschen wollen.
2. Wählen Sie im Kontextmenü den Befehl "Entfernen".

### Externe Grafiken bearbeiten

1. Markieren Sie die Grafik, die Sie bearbeiten wollen.
2. Wählen Sie im Kontextmenü den Befehl "Grafik bearbeiten".  
Das mit der Grafik-Datei verknüpfte Bildbearbeitungs-Programm wird geöffnet.

### Grafikordner von WinCC aus bearbeiten

1. Markieren Sie die Grafik, die Sie bearbeiten wollen.
2. Wählen Sie im Kontextmenü den Befehl "Übergeordneten Ordner öffnen".  
Der Windows-Explorer wird geöffnet.

### Siehe auch

Übersicht über die Objekte (Seite 3040)

## 10.1.2.18 Externe Grafik in der Grafiksammlung hinterlegen

### Einleitung

Um in einem externen Grafikprogramm erstellte Grafiken in Ihren Bildern anzuzeigen, hinterlegen Sie diese vorher in der Grafiksammlung des WinCC Projekts.

### Voraussetzung

- Ein Bild ist angelegt.
- In dem Bild ist eine Grafikanzeige eingefügt.
- Das Inspektorfenster der Grafikanzeige ist geöffnet.

Um eine externe Grafik in der Grafiksammlung zu hinterlegen:

- Eine Grafik liegt vor.


Um ein OLE-Objekt in der Sammlung zu hinterlegen:

- Ein OLE-fähiges Grafikprogramm ist auf dem Projektierungsrechner installiert.

### Grafikdatei hinterlegen

1. Öffnen Sie den Windows-Explorer.
2. Selektieren Sie die Grafik, die Sie hinterlegen wollen.
3. Ziehen Sie per Drag&Drop die Grafik in die Grafiksammlung.

### Neue Grafik als OLE-Objekt erstellen und hinterlegen

1. Markieren Sie die Grafikanzeige im Bild.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Öffnen Sie die Grafikauswahlliste.
4. Klicken Sie auf das Symbol .
5. Der Dialog "Objekt einfügen" wird geöffnet.

---

#### Hinweis

Zusätzlich wird der Dialog "Externe Applikation läuft..." geöffnet. Der Dialog wird erst geschlossen, wenn die externe Applikation beendet wurde.

---

6. Wählen Sie im Dialog "Objekt einfügen" die Option "Neu erstellen" und einen Objekttyp. Welche Objekttypen angezeigt werden, ist abhängig von den Einstellungen unter "Einstellungen > OLE-Einstellungen".
7. Klicken Sie auf "OK". Das dazugehörige Grafikprogramm wird geöffnet. Wenn Sie die Grafik erstellt haben, dann beenden Sie das Grafikprogramm über "Datei > Beenden" oder "Datei > Beenden & zurück zu WinCC". Die Grafik wird im Standardformat des Grafikprogramms abgespeichert und in der Grafiksammlung angezeigt.

### Erstellte Grafik in WinCC einfügen

---


#### Hinweis

Wenn Sie eine neue Grafik als OLE-Objekt erstellen, wird die Grafik nach dem Speichern in dem externen Grafikprogramm eventuell nicht direkt in WinCC übernommen.

---

1. Öffnen Sie den Dialog für das Einfügen einer Grafik erneut.
2. Wählen Sie im Dialog "Objekt einfügen" die Option "Aus Datei erstellen".
3. Klicken Sie auf die Schaltfläche "Durchsuchen".
4. Navigieren Sie zu der erstellten Grafik und wählen Sie diese aus.

## Vorhandene Grafik als OLE-Objekt hinterlegen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie die Grafikauswahlliste.
3. Klicken Sie auf das Symbol .
4. Der Dialog "Objekt einfügen" wird geöffnet.

---

### Hinweis

Zusätzlich wird der Dialog "Externe Applikation läuft..." geöffnet. Der Dialog wird erst geschlossen, wenn die externe Applikation beendet wurde.

---

5. Wählen Sie im Dialog "Objekt einfügen" die Option "Aus Datei erstellen".
6. Klicken Sie auf die Schaltfläche "Durchsuchen".
7. Navigieren Sie mithilfe des Dialogs zu dem Ordner, in dem die Grafikdatei gespeichert ist.

---

### Hinweis

Um Grafikdateien zu importieren, beachten Sie folgende Größenbeschränkungen:

\*.bmp, \*.tif, \*.emf, \*.wmf ≤4 MByte

\*.jpg, \*.jpeg, \*.ico, \*.gif "\*"≤1 MByte

---

## Ergebnis

Die Grafik ist in der Grafiksammlung hinterlegt. In einem Bild wird sie mithilfe einer Grafikanzeige dargestellt oder als Listenelement in einer Grafikliste aufgenommen.

Die in der Grafiksammlung hinterlegten OLE-Objekte öffnen Sie durch Doppelklicken mit dem dazugehörigen Grafikprogramm, um sie zu bearbeiten. Wenn Sie die Grafik bearbeitet haben, dann beenden Sie das Grafikprogramm über "Datei > Beenden" oder "Datei > Beenden & zurück zu WinCC". Die Änderungen werden in WinCC übernommen.

## Siehe auch

Übersicht über die Objekte (Seite 3040)

## 10.1.2.19 Arbeiten mit Objektgruppen

### Grundlagen zu Gruppen

### Einleitung

Gruppen sind mehrere Objekte, die mit der Funktion "Gruppieren" zusammengefügt sind. Sie bearbeiten eine Gruppe wie ein einzelnes Objekt.

## Übersicht

WinCC bietet Folgendes, um mehrere Objekte gemeinsam zu bearbeiten.

- Mehrfachauswahl
- Objekte gruppieren

## Bearbeitungsmodus

Um ein Objekt einer Gruppe einzeln zu bearbeiten, selektieren Sie innerhalb der Task Card "Layout > Ebenen" das Objekt.

Alternativ wählen Sie im Kontextmenü der Objektgruppe "Gruppieren > Gruppe bearbeiten".

## Hierarchische Gruppen

Um eine Gruppe zu erweitern, fügen Sie weitere Objekte oder Gruppen hinzu. Die Gruppe vergrößert sich um die neuen Objekte und baut sich hierarchisch in Haupt- und Untergruppen oder -objekte auf. Sie lösen eine solche hierarchische Gruppe nur stufenweise wieder auf. In der Reihenfolge, wie Sie die Objekte oder Gruppen gruppiert haben, lösen Sie die Gruppe auch wieder auf. Um eine solche hierarchische Gruppe aufzulösen, sind genauso viele Schritte erforderlich wie zum Gruppieren.

## Objektumfassendes Rechteck

Bei einer Gruppe wird nur noch ein Objektumfassendes Rechteck für die gesamte Gruppe angezeigt. Bei der Mehrfachauswahl dagegen werden die Objektumfassenden Rechtecke aller Objekte angezeigt.

## Ebenen

Alle Objekte einer Gruppe befinden sich in der gleichen Ebene.

## Siehe auch

Übersicht über die Objekte (Seite 3040)

## Objekte gruppieren

## Einleitung

Mit dem Menübefehl "Gruppieren" fassen Sie mehrere Objekte zu einer Gruppe zusammen.

Sie können die Größe und Position der Gruppe ändern. Dafür gelten folgende Regeln:



- Wenn Sie die Position einer Gruppe ändern, wird die Position der gruppierten Objekte an die neuen Koordinaten angepasst. Relativ zur Gruppe ändert sich die Position der gruppierten Objekte nicht.
- Wenn Sie die Größe einer Gruppe ändern, werden Höhe und Breite der gruppierten Objekte entsprechend skaliert.
- Um die Größe der Gruppe proportional zu ändern, ziehen Sie mit der Maus das objektumfassende Rechteck mit gedrückter Taste <Shift> in die gewünschte Größe.

---

#### **Hinweis**

Um eine hierarchische Gruppe aufzubauen, gruppieren Sie einzelne Gruppen wie Objekte.

---

#### **Voraussetzung**

- Ein Bild mit mindestens zwei Objekten ist geöffnet.

#### **Objekte gruppieren**

1. Selektieren Sie alle Objekte, die Sie gruppieren wollen.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppieren".

Die Objekte der Gruppe werden mit einem objektumfassenden Rechteck dargestellt.

#### **Objekte innerhalb einer Gruppe gruppieren**

1. Selektieren Sie die Gruppe, die Sie bearbeiten möchten.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppe bearbeiten".  
Die Gruppe, die Sie bearbeiten, wird durch einen roten Rahmen hervorgehoben.
3. Selektieren Sie alle Objekte, in der Gruppe, die Sie zu einer Untergruppe zusammenfassen wollen.
4. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppieren".

Eine Untergruppe mit den Objekten wird angelegt.

#### **Objekte in eine bestehende Gruppe aufnehmen**

1. Selektieren Sie die Gruppe, zu der Sie Objekte hinzufügen möchten.
2. Drücken Sie die <Shift> Taste und selektieren Sie das Objekt, das Sie in die Gruppe aufnehmen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Zu Gruppe hinzufügen".

Das Objekt wird in die Gruppe eingefügt.

### Alternatives Vorgehen

Gruppen können Sie auch in der Task Card "Layout" bearbeiten. Mit Drag&Drop können Sie in der Palette "Ebenen" auch hierarchische Gruppen einfach bearbeiten.

### Ergebnis

Die markierten Objekte sind in einer Gruppe enthalten. Das objektumfassende Rechteck der Mehrfachauswahl wird zum objektumfassenden Rechteck der Gruppe. Die Anfassers sind nur noch für die Gruppe eingeblendet. Die Gruppe befindet sich in der aktiven Ebene.

### Gruppe aufheben

#### Einleitung

Mit dem Befehl "Gruppierung aufheben" lösen Sie eine Gruppe wieder in einzelne Objekte auf.

#### Voraussetzung

- Ein Bild mit einer Gruppe ist geöffnet.

### Gruppe aufheben

1. Markieren Sie die Gruppe.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppierung aufheben".

### Gruppierung innerhalb einer Gruppe auflösen

1. Selektieren Sie die übergeordnete Gruppe.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppe bearbeiten".  
Die Gruppe, die Sie bearbeiten, wird durch einen roten Rahmen hervorgehoben.
3. Selektieren Sie die untergeordnete Gruppe.
4. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppierung aufheben".

### Ergebnis

Die untergeordnete Gruppe wird aufgelöst. Die Objekte werden der nächsthöheren Gruppe zugeordnet.

### Alternatives Vorgehen

Gruppen können Sie auch in der Task Card "Layout" bearbeiten. Mit Drag&Drop können Sie in der Palette "Ebenen" auch hierarchische Gruppen einfach bearbeiten.

## Objekte zu einer Gruppe hinzufügen

### Einleitung

Mit dem Befehl "Objekte zur Gruppe hinzufügen" fügen Sie Objekte zu einer Gruppe hinzu, ohne diese vorher aufzulösen.

### Voraussetzungen

Ein Bild mit einer Gruppe und mindestens einem weiteren Objekt ist geöffnet.

### Vorgehen

1. Selektieren Sie die Gruppe.
2. Drücken Sie die <Shift> Taste und selektieren Sie das Objekt, das Sie in die Gruppe aufnehmen möchten.
3. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Zu Gruppe hinzufügen".

### Ergebnis

Die Gruppe besteht aus den ursprünglichen Objekten und den neu hinzugefügten Objekten. Die hinzugefügten Objekte sind in der Tiefenstaffelung der Gruppe vorne angeordnet.

### Alternatives Vorgehen

Gruppen können Sie auch in der Task Card "Layout" bearbeiten. Mit Drag&Drop können Sie in der Palette "Ebenen" auch hierarchische Gruppen einfach bearbeiten.

## Objekte aus der Gruppe entfernen

### Einleitung

Mit dem Befehl "Objekte aus Gruppe entfernen" entfernen Sie einzelne Objekte aus einer Gruppe, ohne diese vorher aufzulösen.

Zur Bearbeitung eines Objekts in einer Gruppe müssen Sie das Objekt nicht aus der Gruppe entfernen. Objekte einer Gruppen können Sie einzeln bearbeiten.

## Voraussetzung

- Ein Bild mit einer Gruppe ist geöffnet.

## Objekte aus einer Gruppe entfernen

Um ein Objekt aus einer Gruppe herauszulösen:

1. Selektieren Sie die Gruppe.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppe bearbeiten".  
Die Gruppe, die Sie bearbeiten wird durch einen roten Rahmen hervorgehoben.
3. Selektieren Sie in der Gruppe alle Objekte, die Sie aus der Gruppe entfernen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Aus Gruppe entfernen".

Die Objekte werden aus der Gruppe herausgelöst.

---

### Hinweis

Wenn nur noch zwei Objekte in der Gruppe sind, dann steht der Menübefehl "Aus Gruppe entfernen" nicht zur Verfügung.

---

## Objekte aus einer Gruppe löschen

Um ein Objekt aus der Gruppe und aus dem Bild zu entfernen:

1. Selektieren Sie die Gruppe.
2. Wählen Sie im Kontextmenü den Befehl "Gruppieren > Gruppe bearbeiten".  
Die Gruppe, die Sie bearbeiten wird durch einen roten Rahmen hervorgehoben.
3. Selektieren Sie in der Gruppe alle Objekte, die Sie löschen möchten.
4. Wählen Sie im Kontextmenü den Befehl "Löschen".

---

### Hinweis

Wenn nur noch zwei Objekte in der Gruppe sind, dann steht der Menübefehl "Löschen" nicht zur Verfügung.

---

## Alternatives Vorgehen

Gruppen können Sie auch in der Task Card "Layout" bearbeiten. Mit Drag&Drop können Sie in der Palette "Ebenen" auch hierarchische Gruppen einfach bearbeiten.

## Objekt in einer Gruppe bearbeiten

## Einleitung

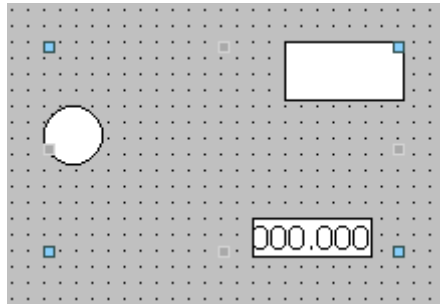
Objekte einer Gruppe können einzeln bearbeitet werden.

## Voraussetzung

Ein Bild mit einer Gruppe ist geöffnet.

## Gruppierte Objekte bearbeiten

1. Selektieren Sie die Gruppe.

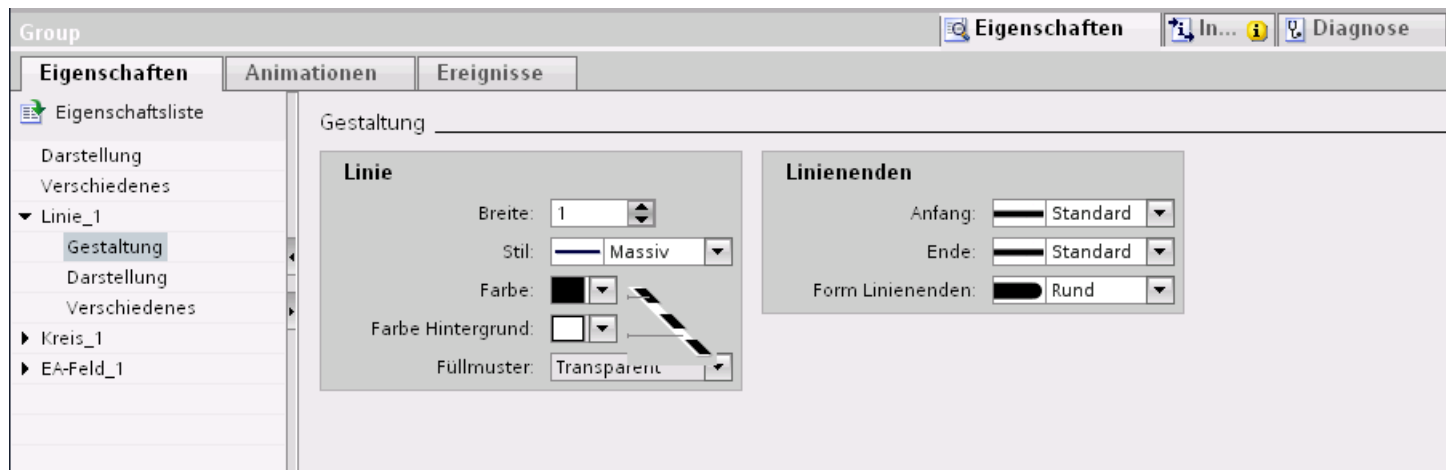


Die Eigenschaften der Gruppe werden im Inspektorfenster angezeigt.

2. Ändern Sie die Position und Größe der gruppierten Objekte unter "Eigenschaften > Eigenschaften > Darstellung".
3. Ändern Sie den Namen der Gruppe unter "Eigenschaften > Eigenschaften > Verschiedenes".

## Eigenschaften eines Objekts in einer Gruppe ändern

1. Selektieren Sie die Gruppe.
2. Wählen Sie im Inspektorfenster das Objekt, dessen Eigenschaften Sie ändern wollen.



Die Eigenschaften des Objektes werden angezeigt.

3. Ändern Sie die Objekteigenschaften.

## Ergebnis

Das Objekt wurde geändert, bleibt aber weiterhin Bestandteil der Gruppe. Die anderen Objekte der Gruppe bleiben unverändert.

### 10.1.2.20 Konfigurieren des Tastenzugriffs

## Übersicht zum Tastaturzugriff

### Einleitung

An Tastengeräten ohne Maus aktiviert der Bediener Bedienobjekte mit der Taste <Tab>. Um komfortabel zu bedienen und um sicherzustellen, dass der Bediener alle nötigen Werte eingibt, richten Sie für ihn die Eingabe über die Tastatur ein. Wenn Sie die Tastatur bedienen, aktivieren Sie mit der Taste <Tab> die Objekte in einer bestimmten Reihenfolge und geben die erforderlichen Werte ein.

Bei Bediengeräten ohne Tasten simulieren Sie die Taste <Tab>, indem Sie die Systemfunktion "SimuliereSystemtaste" an eine Funktionstaste projektieren.

### Bedienberechtigung und Bedienfreigabe

Wenn Sie ein Objekt für die Bedienung mit der Taste <Tab> projektieren, muss das Objekt eine Bedienberechtigung und eine Bedienfreigabe haben.

### Tab-Reihenfolge bearbeiten

Die Tab-Reihenfolge wird beim Anlegen der Bedienobjekte automatisch bestimmt. Die Nummern der Tab-Reihenfolge werden in der Reihenfolge vergeben, in der die Bildobjekte angelegt werden.

In folgenden Fällen ist die Änderung der Tab-Reihenfolge sinnvoll:

- Der Bediener wechselt direkt auf ein bestimmtes Bedienobjekt
- Das Bild erfordert eine bestimmte Reihenfolge

Um die Tab-Reihenfolge zu ändern, wechseln Sie in den Tab-Reihenfolgemodus. In diesem Modus wird an den Bedienobjekten links oben die Tab-Reihenfolge-Nummer eingeblendet. Sie sehen auch die Tab-Reihenfolge-Nummern von verdeckten Objekten. Die Verteilung dieser Nummern bearbeiten Sie mit der Maus.

---

#### Hinweis

Im Tab-Reihenfolge-Modus sind keine anderen Funktionen verfügbar.

---

### Siehe auch

Beispiel: Rechteck einfügen und konfigurieren (Seite 3075)

Übersicht über die Objekte (Seite 3040)

## Bedienberechtigung und Bedienfreigabe für ein Objekt festlegen

### Einleitung

Wenn Sie ein Objekt für die Bedienung mit der Taste <Tab> projektieren, muss das Objekt eine Bedienberechtigung und eine Bedienfreigabe haben.

### Voraussetzung

Ein Bild mit mindestens einem Objekt ist geöffnet.

### Vorgehen

1. Wählen Sie das Objekt.
2. Wählen Sie in im Inspektorfenster "Eigenschaften > Eigenschaften > Sicherheit".
3. Wählen Sie unter "Berechtigung" die Bedienberechtigung aus.
4. Aktivieren Sie die Berechtigung zur Bedienung.

### Ergebnis

In Runtime kann der Bediener mit der Taste <Tab> das Objekt anwählen.

### Siehe auch

Beispiel: Rechteck einfügen und konfigurieren (Seite 3075)

## Tab-Reihenfolge festlegen

### Einleitung

In Runtime sind alle bedienbaren Objekte per Taste <Tab> erreichbar. Sie legen mit dem Befehl "Tab-Reihenfolge" die Reihenfolge fest, in der die Objekte in Runtime aktiviert werden.

---

#### Hinweis

Objekte mit dem Modus "Ausgabe" oder "Zwei Zustände" sind in Runtime nicht per Taste <Tab> erreichbar.

---

In Runtime bedienen Sie das Bild:

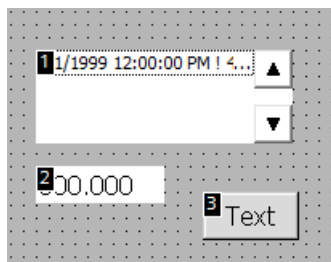
- Per Taste <Tab>
- Per Maus
- Per vorher eingestellten Hotkey

## Voraussetzung

- Das aktive Bild enthält bedienbare Objekte.
- Kein Objekt ist markiert.
- Die Objekte sind zur Bedienung in Runtime freigegeben und haben eine Bedienberechtigung.

## Vorgehen

1. Wählen Sie im Menü "Bearbeiten > Tab-Reihenfolge bearbeiten".  
Der Tab-Reihenfolgemodus wird aktiviert. Die Tab-Reihenfolge-Nummer wird bei allen bedienbaren Objekten angezeigt. Die Tab-Reihenfolge-Nummer wird auch bei verdeckten Objekten angezeigt.
2. Um den Tab-Reihenfolgemodus zu bearbeiten, klicken Sie die bedienbaren Objekte in der Reihenfolge an, in der die Objekte in Runtime mit <Tab> aktiviert werden sollen.  
Das folgende Bild zeigt, wie im Bild die Tab-Reihenfolge festgelegt wird. In Runtime aktiviert die Taste <Tab> zuerst die Meldeanzeige (Nummer 1), dann das E/A-Feld (Nummer 2) und dann die Schaltfläche (Nummer 3):



3. Um ein Bildobjekt aus der Tab-Reihenfolge auszuschließen, halten Sie die Tastenkombination <Shift+Strg> gedrückt und klicken Sie auf das gewünschte Objekt.  
Die Tab-Reihenfolge-Nummer wird nicht mehr im Bildobjekt angezeigt. Das Bildobjekt ist jetzt aus der Tab-Reihenfolge ausgeschlossen. Die verbleibenden Tab-Reihenfolge-Nummern werden automatisch um 1 erniedrigt.
4. Um ein ausgeschlossenes Bildobjekt wieder in die Tab-Reihenfolge aufzunehmen, wiederholen Sie Schritt 3.  
Das Bildobjekt wird als erstes Objekt in die Tab-Reihenfolge aufgenommen.

## Ergebnis

In Runtime selektiert der Bediener mit der Taste <Tab> in der festgelegten Reihenfolge die Objekte.

## Siehe auch

Beispiel: Rechteck einfügen und konfigurieren (Seite 3075)



### 10.1.2.21 Beispiele

#### Beispiel: Rechteck einfügen und konfigurieren

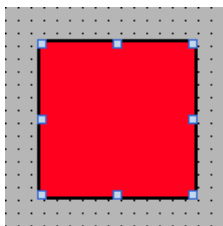
##### Aufgabe

In diesem Beispiel fügen Sie ein Rechteck in ein Bild ein. Dabei konfigurieren Sie folgende Eigenschaften:

- Name = "MyRectangle"
- Position = (20, 20)
- Größe = (100,100)
- Farbe = rot
- 2 Pixel breiter schwarzer Rahmen

##### Prinzip

Das Rechteck ist ein geschlossenes Objekt, das Sie mit einer Farbe oder einem Muster füllen können. Die Höhe und Breite eines Rechtecks lassen sich beliebig verändern, sodass eine horizontale oder vertikale Ausrichtung möglich ist.



##### Übersicht

Um ein Rechteck zu erstellen, sind folgende Schritte notwendig:

- Rechteck einfügen
- Rechteck konfigurieren

##### Siehe auch

Grundlagen zu Gruppen (Seite 3065)

Übersicht über die Objekte (Seite 3040)

## Beispiel: Rechteck einfügen

### Aufgabe

In diesem Beispiel fügen Sie ein Rechteck ein und benennen es um. Verwenden Sie für den Namen nicht die Sonderzeichen ?, ", /, \, \*, <, >.

### Voraussetzung

- Ein Bild ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.

### Vorgehen

1. Klicken Sie in der Task Card "Werkzeuge" auf die Palette "Basisobjekte".
2. Ziehen Sie das Objekt "Rechteck" in das Bild.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verschiedenes".
4. Geben Sie als neuen Namen "MyRectangle" ein.

### Ergebnis

Das Rechteck ist eingefügt und "MyRectangle" benannt. Das Rechteck besitzt die Standard-Eigenschaften des Objekts "Rechteck".

## Beispiel: Rechteck konfigurieren

### Aufgabe

In diesem Beispiel konfigurieren Sie das Rechteck:

- Farbe = rot
- 2 Pixel breiter schwarzer Rahmen
- Position = (20, 20)
- Größe = (100,100)

### Farbe des Rechtecks ändern

Um die Farbe des Rechtecks zu ändern:

1. Selektieren Sie das Rechteck.
2. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Hintergrund > Farbe " die Hintergrundfarbe fest.
3. Wählen Sie "Massiv" als Füllmuster.

4. Legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Rahmen > Farbe " die Farbe für den Rahmen fest.
5. Geben Sie für "Breite" den Wert "2" ein.
6. Wählen Sie "Massiv" als "Stil".

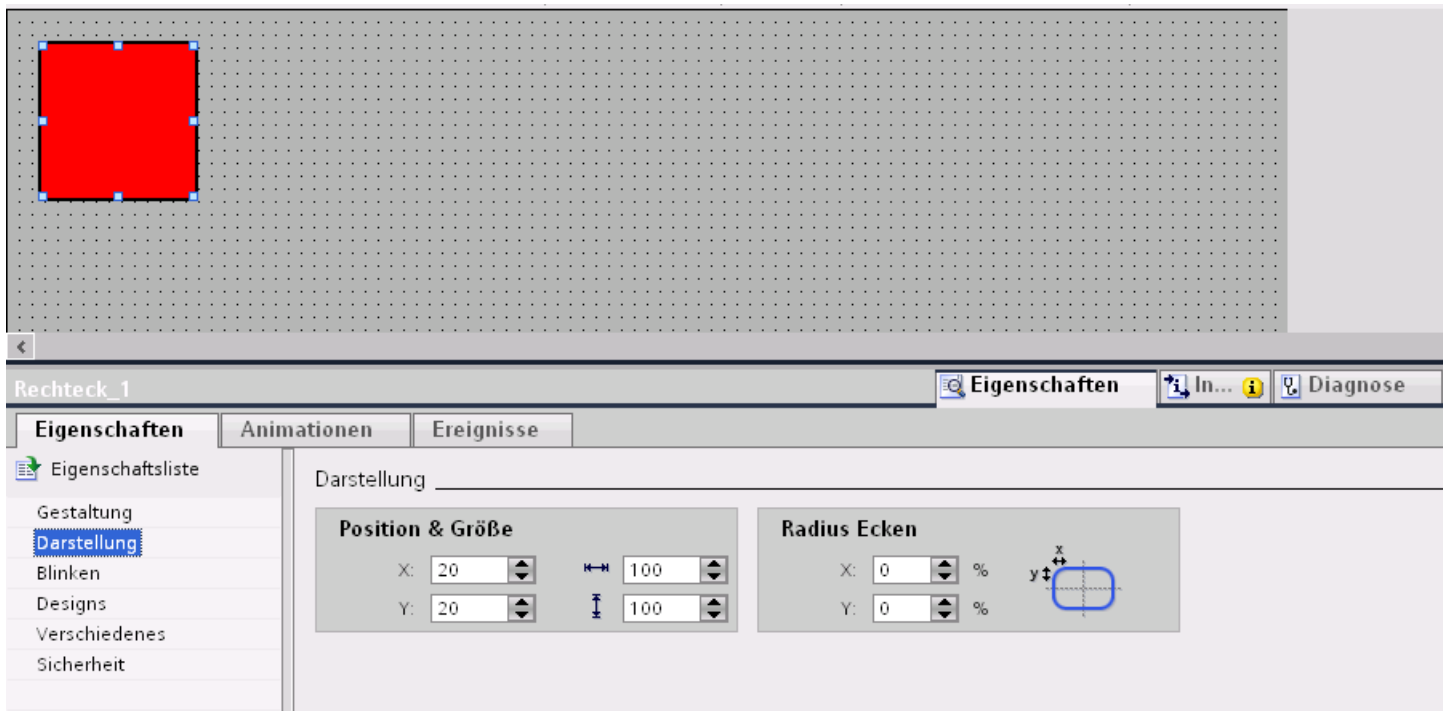
### Zwischenergebnis

Das Rechteck ist rot und hat einen 2 Pixel breiten schwarzen Rahmen.

### Position und Größe des Rechtecks ändern

Um die Position und Größe des Rechtecks zu ändern:

1. Selektieren Sie das Rechteck.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".



3. Geben Sie unter "Position & Größe" für X und Y jeweils den Wert "20" ein.
4. Geben Sie als Breite und Höhe jeweils den Wert "100" ein.

### Ergebnis

Das Rechteck ist an den Koordinaten (20, 20) positioniert und ist je 100 Pixel breit und hoch.

## 10.1.3 Arbeiten mit Textlisten und Grafiklisten

### 10.1.3.1 Arbeiten mit Textlisten

#### Grundlagen zu Textlisten

##### Einleitung

In einer Textliste sind Texte den Werten einer Variablen zugeordnet. Bei der Projektierung ordnen Sie die Textliste z. B. einem Symbolischen E/A-Feld zu. Damit liefern Sie dem Objekt die anzuzeigenden Texte. Im Editor "Textlisten" erstellen Sie die Textlisten. Die Anbindung der Textliste an eine Variable projektieren Sie an dem Objekt, das die Textliste verwendet.

Die Auswahl der Objekte, die mit einer Textliste versehen werden können, hängt von der Runtime ab.

##### Verwendung

Sie verwenden die Textliste z. B. für die Anzeige einer Auswahlliste in einem Symbolischen E/A-Feld.

Wenn das Symbolische E/A-Feld ein Anzeigefeld ist, werden die zugehörigen Texte abhängig vom Wert der projektierten Variablen angezeigt. Wenn das Symbolische E/A-Feld ein Eingabefeld ist, nimmt die projektierte Variable den zugehörigen Wert an, sobald der Bediener in Runtime den entsprechenden Text auswählt.

---

##### Hinweis

##### Anzeige von Variablenwerten ohne Text

Die Darstellung von Variablenwerten, denen kein Text zugeordnet wurde, ist von der Runtime abhängig:

- Anzeige- und Bedienobjekt bleibt leer.
  - Drei Sternchen \*\*\* werden angezeigt.
- 

##### Mehrsprachige Texte

Sie können die Texte in einer Textliste mehrsprachig projektieren. Die Texte werden in Runtime in der eingestellten Runtime-Sprache angezeigt. Dazu stellen Sie die Sprachen im Projektfenster unter "Sprachunterstützung > Projektsprachen" ein.

## Projektierungsschritte

Zur Anzeige von Texten z. B. in einem Symbolischen E/A-Feld sind folgende Schritte erforderlich:

1. Erstellen der Textliste
2. Zuordnung der Texte zu Werten bzw. Wertebereichen einer Textliste
3. Zuordnung einer Textliste im Anzeigeobjekt, z. B. dem Symbolischen E/A-Feld

## Textliste erstellen

### Einleitung

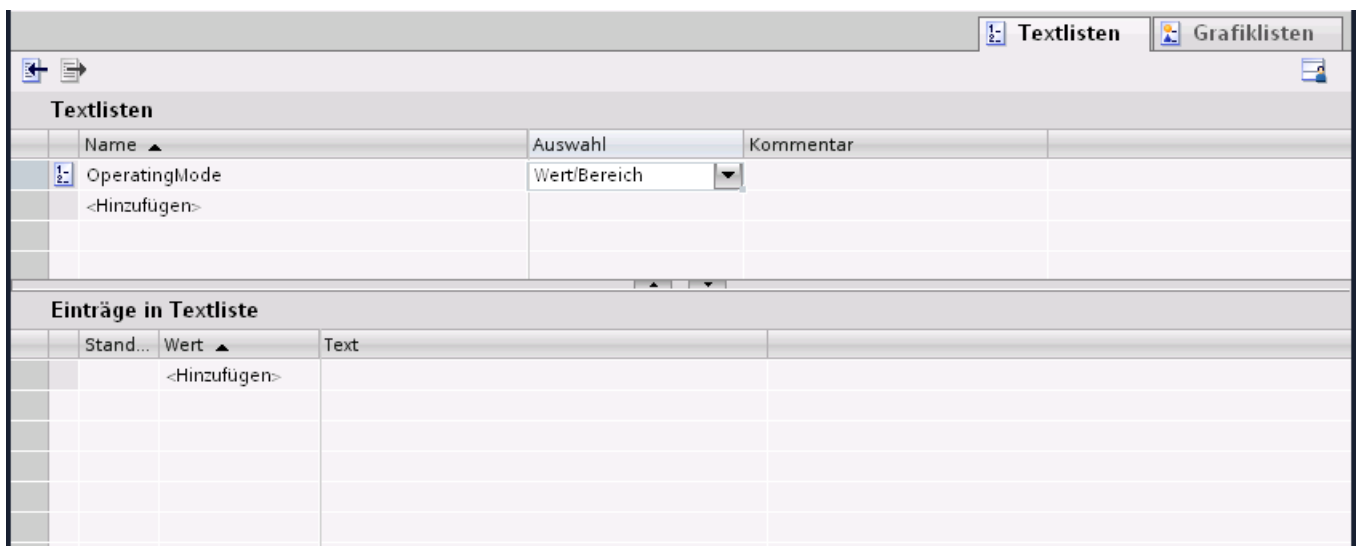
Die Textliste bietet die Möglichkeit, Werten spezifische Texte zuzuordnen und diese in Runtime, z.B. in einem Symbolischen E/A-Feld, auszugeben. Dabei kann der Typ des Symbolischen E/A-Feldes angegeben werden, z.B. als reines Eingabefeld.

Folgende Listenarten stehen zur Verfügung:

- Wert/Bereich
- Bit
- Bitnummer

### Vorgehensweise

1. Doppelklicken Sie im Projektfenster auf "Text- und Grafiklisten".
2. Öffnen Sie die Registerkarte "Textlisten".



3. Klicken Sie in der Tabelle "Textlisten" auf "Hinzufügen". Das Inspektorfenster zur Textliste wird geöffnet.
4. Geben Sie der Textliste einen aussagekräftigen Namen.

### 10.1 Bilder erstellen

5. Wählen Sie unter "Auswahl" den Textlistentyp:
  - Wert/Bereich: Text aus Textliste wird angezeigt, wenn die Variable einen Wert im angegebenen Bereich angenommen hat.
  - Bit (0,1): Ein Text aus der Textliste wird angezeigt, wenn die Variable den Wert 0 angenommen hat, ein anderer Text aus der Textliste wird angezeigt, wenn die Variable den Wert 1 angenommen hat.
  - Bitnummer (0-31): Text aus der Textliste wird angezeigt, wenn die Variable die zugeordnete Bitnummer angenommen hat.
6. Geben Sie einen Kommentar für die Textliste ein.

---

#### Hinweis

In den Texten einer Textliste können Sie kein Semikolon verwenden. Das Semikolon ist ein Steuerzeichen und wird in einem Text automatisch gelöscht.

---

#### Ergebnis

Eine Textliste ist erstellt.

#### Texte und Werte einer Bereich-Textliste zuordnen

#### Einleitung

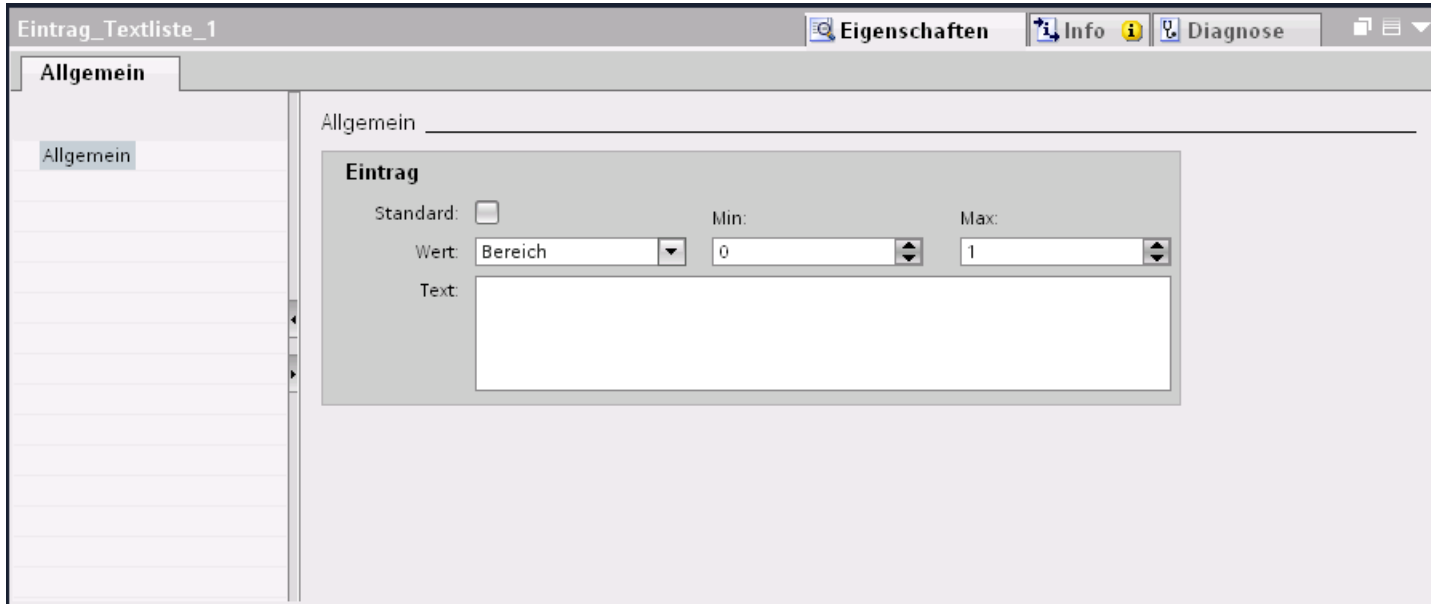
Zu jeder Bereich-Textliste geben Sie an, bei welchem Wertebereich welche Texte angezeigt werden.

#### Voraussetzung

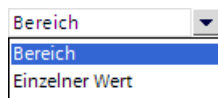
- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Textlisten" ist geöffnet.
- Eine Bereich-Textliste ist angelegt und selektiert.

## Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen".  
Das Inspektorfenster für diesen Listeneintrag wird geöffnet.



2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Bereich".



- Geben Sie für "Min" z. B. den Wert "1" ein.
- Geben Sie für "Max" z. B. den Wert "20" ein.
- Geben Sie für "Text" den Text ein, der in Runtime angezeigt wird, wenn die Variable sich innerhalb des angegebenen Wertebereichs befindet.

---

### Hinweis

Verwenden Sie kein Semikolon und maximal 255 Zeichen für den Text.

---

3. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen". Ein zweiter Listeneintrag wird erstellt.

### 10.1 Bilder erstellen

4. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Bereich".
  - Geben Sie für "Min" z. B. den Wert "21" ein.
  - Geben Sie für "Max" z. B. den Wert "40" ein.
  - Geben Sie für "Text" den Text ein, der in Runtime angezeigt wird, wenn die Variable sich innerhalb des angegebenen Wertebereichs befindet
5. Bei Bedarf aktivieren Sie "Standardeintrag".  
Der eingegebene Text erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt. Pro Liste ist nur ein Standardeintrag möglich.

#### Ergebnis

Eine Bereich-Textliste ist erstellt. Den möglichen Wertebereichen sind Texte zugeordnet.

#### Texte und Werte einer Bit-Textliste zuordnen

#### Einleitung

Zu jeder Textliste geben Sie an, bei welchem Bit-Wert welcher Text angezeigt wird.

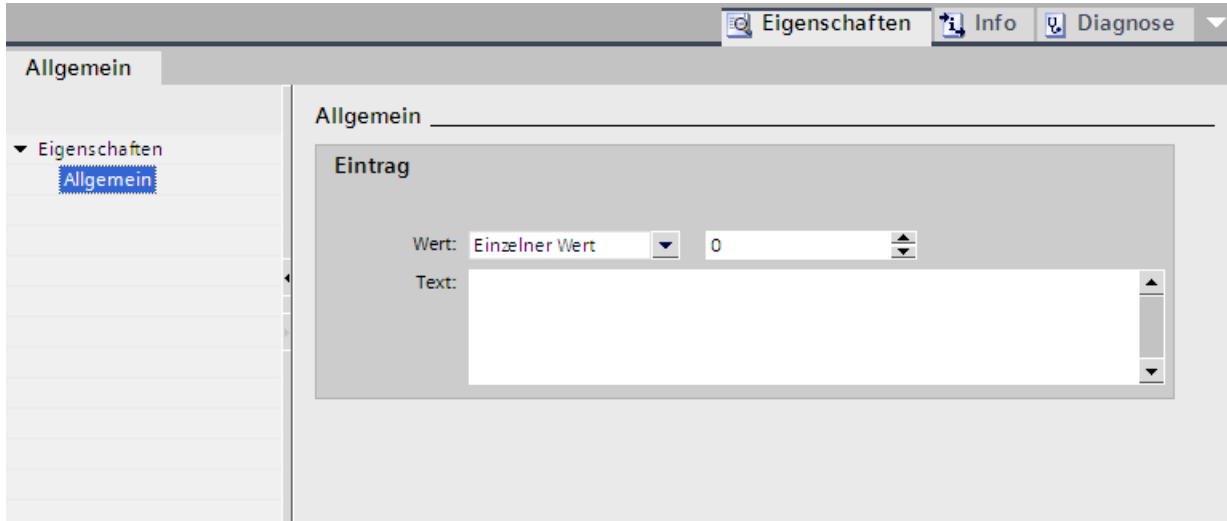
#### Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Textlisten" ist geöffnet.
- Eine Bit-Textliste ist angelegt und selektiert.



## Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen".  
Das Inspektorfenster für diesen Listeneintrag wird geöffnet



2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Einzelner Wert".
  - Geben Sie für "Wert" "0" ein.
  - Geben Sie unter "Text" den Text ein, der in Runtime angezeigt wird, wenn die Bitvariable auf "0" gesetzt wird.

---

### Hinweis

Verwenden Sie kein Semikolon und maximal 255 Zeichen für den Text.

---

3. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen". Ein zweiter Listeneintrag wird erstellt.
4. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Einzelner Wert".
  - Geben Sie unter "Wert" "1" ein.
  - Geben Sie unter "Text" den Text ein, der in Runtime angezeigt wird, wenn die Bitvariable auf "1" gesetzt wird.

## Ergebnis

Eine Bit-Textliste ist erstellt. Den möglichen Werten "0" und "1" sind Texte zugeordnet, die in Runtime erscheinen.

## Texte und Werte einer Bitnummer-Textliste zuordnen

### Einleitung

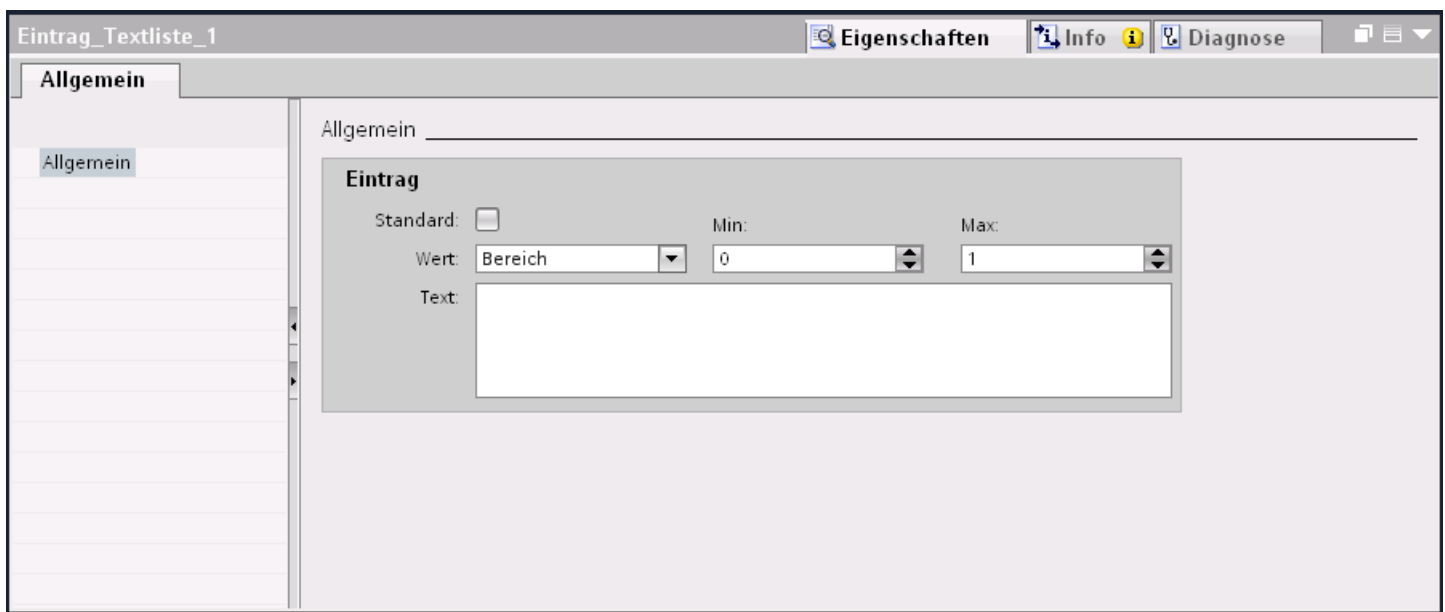
Zu jeder Bitnummer-Textliste geben Sie an, bei welcher Bitnummer welche Texte angezeigt werden.

### Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Textlisten" ist geöffnet.
- Eine Bitnummer-Textliste ist angelegt und selektiert.

### Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Textliste" auf "Hinzufügen".  
Das Inspektorfenster für diesen Listeneintrag wird geöffnet.



2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellung "Einzelner Wert".
  - Geben Sie für "Wert" z. B. "10" ein.
  - Geben Sie unter "Text" den Text ein, der in Runtime angezeigt wird, wenn die Variable den Wert "10" angenommen hat.

---

#### Hinweis

Verwenden Sie kein Semikolon und maximal 255 Zeichen für den Text.

---

3. Bei Bedarf aktivieren Sie "Standardeintrag".  
Der eingegebene Text erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt. Pro Liste ist nur ein Standardeintrag möglich.
4. Legen Sie für weitere Bitnummern derselben Textliste weitere Listeneinträge an.

## Ergebnis

Eine Bitnummer-Textliste ist erstellt. Den angegebenen Bitnummern sind Texte zugeordnet, die in Runtime erscheinen.

## Objekt mit Textliste projektieren

### Einleitung

Ausgabewert und Wertübernahme für Textlisten werden in dem Anzeige- und Bedienobjekt festgelegt, das die Texte der Textliste in Runtime anzeigt. Die Eigenschaften dieser Objekte werden nach Bedarf konfiguriert.

### Voraussetzung

- Eine Textliste ist erstellt.
- Eine Variable ist angelegt.
- Der Editor "Bilder" ist geöffnet.
- Ein Bild mit einem Symbolischen E/A-Feld ist geöffnet. Das Objekt ist selektiert.

## Vorgehen

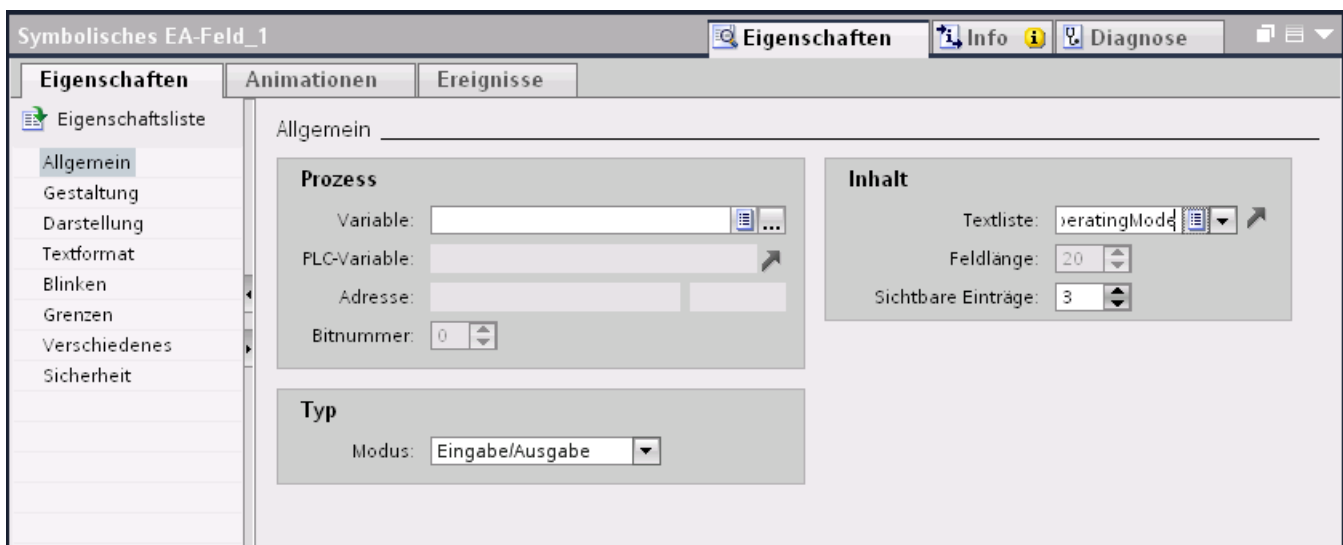
1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Textliste" die Textliste aus, die Sie in Runtime anzeigen lassen wollen.
2. Wählen Sie als "Modus" die Einstellung "Ausgabe".

### Hinweis

#### Runtime-Abhängigkeit

Je nach Runtime stehen unterschiedliche Feldtypen für ein Symbolisches E/A-Feld zur Verfügung.

3. Wählen Sie als "Variable" die Variable, deren Werte die Anzeige im Symbolischen E/A-Feld bestimmt.



## Ergebnis

Wenn die Variable den angegebenen Wert annimmt, werden in Runtime die definierten Texte der Textliste im Symbolischen E/A-Feld angezeigt.

### 10.1.3.2 Arbeiten mit Grafiklisten

## Grundlagen zu Grafiklisten

### Einleitung

In einer Grafikliste sind bestimmte Grafiken den möglichen Werten einer Variablen zugeordnet. Bei der Projektierung ordnen Sie die Grafikliste einer Schaltfläche oder einem Grafischen E/A-Feld zu. Damit liefern Sie dem Objekt die anzuzeigenden Grafiken.

Sie erstellen die Grafiklisten mit dem Editor "Text- und Grafiklisten". Die Anbindung der Grafikliste an eine Variable projektieren Sie an dem Objekt, das die Grafikliste verwendet. Die Verfügbarkeit der Grafikliste hängt vom verwendeten Bediengerät ab.

## Verwendung

Sie projektieren die Grafikliste für folgende Anwendungsfälle:

- Auswahlliste mit einem Grafischen E/A-Feld
- Zustandsabhängige Grafik auf einer Schaltfläche

Sie können die Grafiken in einer Grafikliste mehrsprachig projektieren. In Runtime werden die Grafiken der eingestellten Runtime-Sprache angezeigt.

## Grafikquellen

Die Grafiken können aus folgenden Quellen in die Grafikliste eingefügt werden:

- Auswahl aus der Grafiksammlung
- Auswahl einer vorhandenen Datei  
Sie können folgende Dateitypen verwenden:  
\*.bmp, \*.ico, \*.emf, \*.wmf, \*.gif, \*.tiff, \*.png, \*.jpeg und \*.jpg.
- Erstellen einer neuen Datei

## Funktion

Wenn das Grafische E/A-Feld ein Anzeigefeld ist, werden abhängig vom Wert der projektierten Variablen die zugehörigen Grafiken angezeigt. Wenn das Grafische E/A-Feld ein Eingabefeld ist, nimmt die projektierte Variable den zugehörigen Wert an, sobald der Bediener in Runtime eine Grafik auswählt.

## Projektierungsschritte

Zur Anzeige von Grafiken z.B. in einem Grafischen E/A-Feld sind folgende Schritte erforderlich:

1. Erstellen der Grafikliste
2. Zuordnung der Grafiken zu Werten bzw. Wertebereichen einer Grafikliste
3. Zuordnung einer Grafikliste im Anzeigeobjekt, z.B. dem Grafischen E/A-Feld

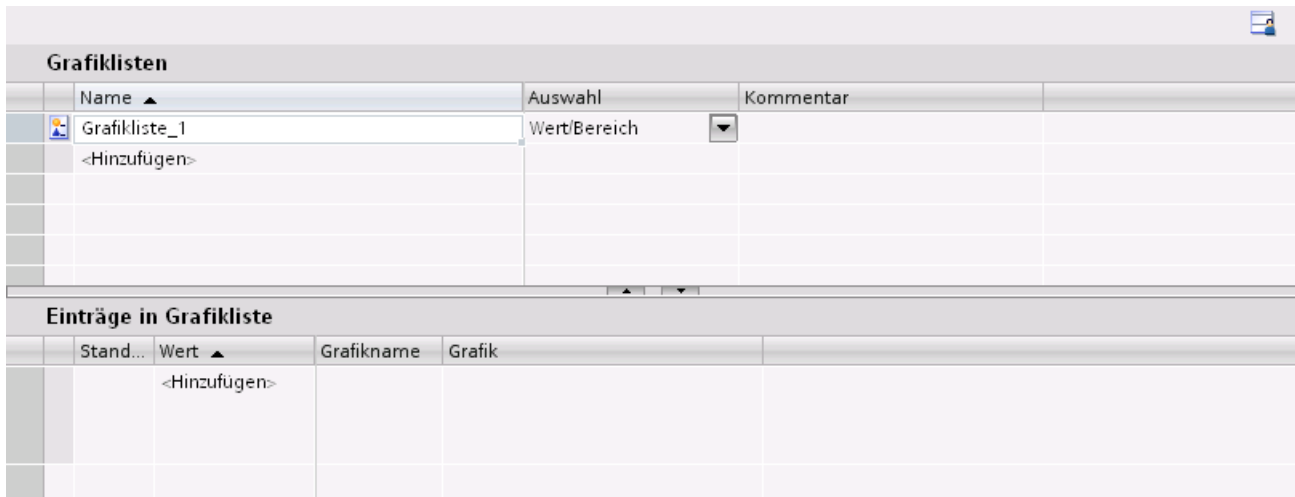
## Grafikliste erstellen

### Einleitung

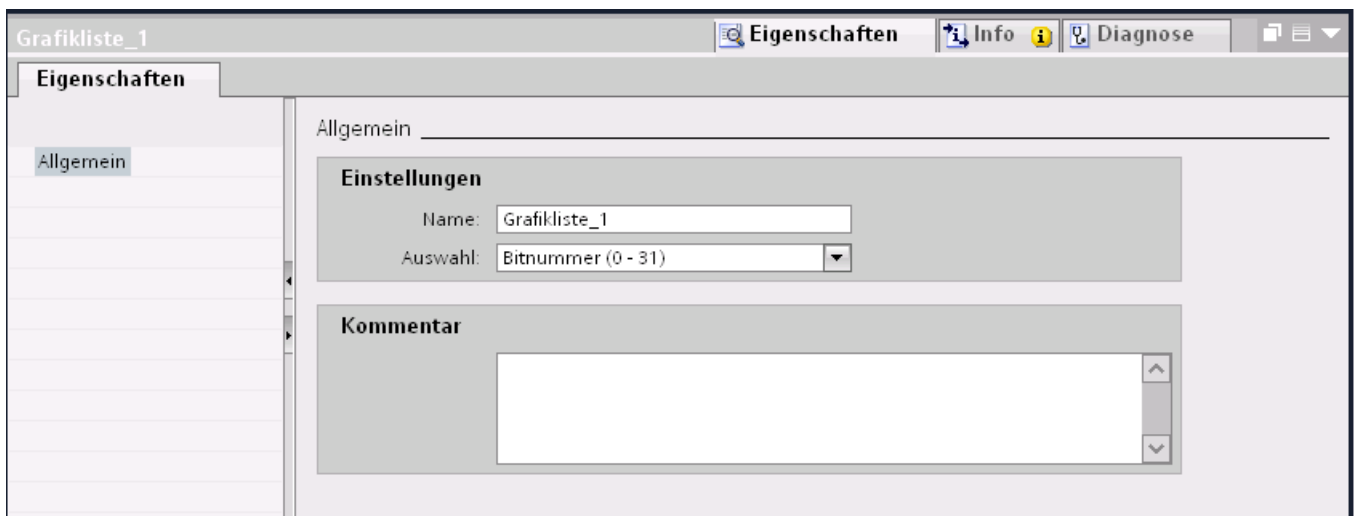
Die Grafikliste bietet die Möglichkeit, variablen Werten spezifische Grafiken zuzuordnen und diese in Runtime in einem Grafische E/A-Feld auszugeben. Dabei kann der Typ des Grafischen E/A-Feldes angegeben werden, z.B. als reines Ausgabefeld.

### Vorgehen

1. Doppelklicken Sie Sie im Projektfenster auf "Text- und Grafiklisten"
2. Öffnen Sie die Registerkarte "Grafiklisten".



3. Klicken Sie in der Tabelle "Grafiklisten" auf "Hinzufügen". Das Inspektorfenster zur Grafikliste wird geöffnet.



4. Geben Sie der Grafikliste einen aussagekräftigen Namen.
5. Wählen Sie unter "Auswahl" z. B. den Grafiklistentyp "Bitnummer (0 - 31)"
6. Geben Sie einen Kommentar für die Grafikliste ein.

### Ergebnis

Eine Grafikliste vom Typ "Bereich (0 - 31)" ist erstellt.

## Grafik und Werte einer Bereich-Grafikliste zuordnen

### Einleitung

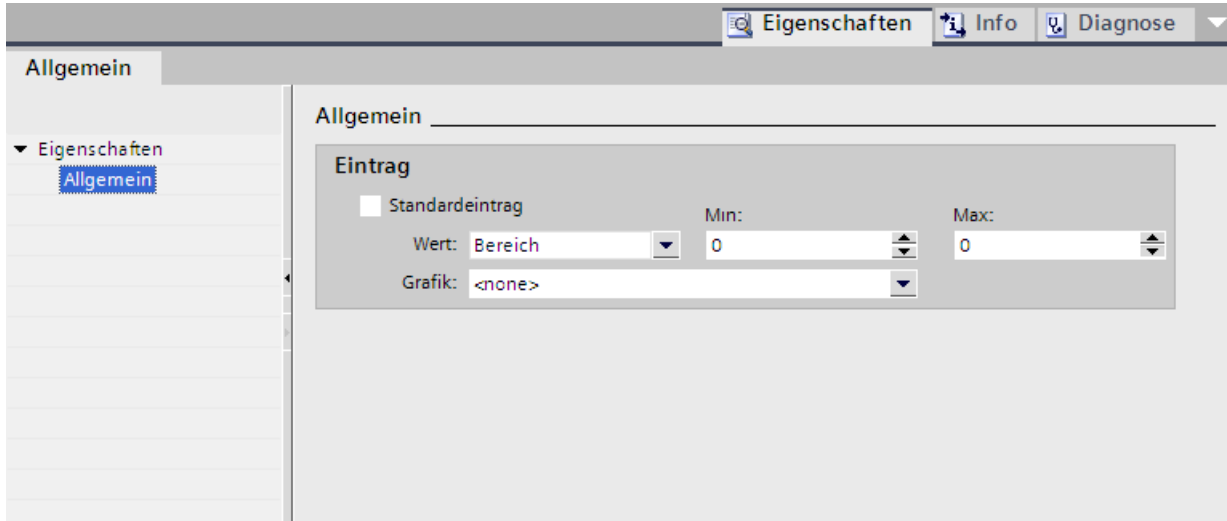
Zu jeder Bereich-Grafikliste geben Sie an, bei welchem Wertebereich welche Grafiken angezeigt werden.

### Voraussetzung

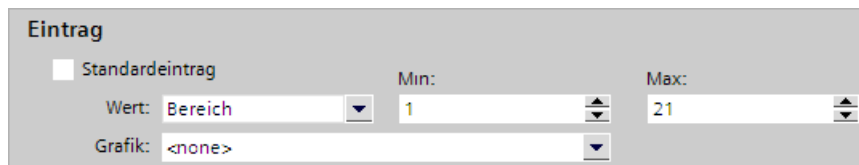
- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Grafiklisten" ist geöffnet.
- Eine Bereich-Grafikliste ist angelegt und selektiert.

## Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen".  
Das Inspektorfenster für diesen Listeneintrag wird geöffnet.



2. Wählen Sie im Inspektorfenster ""Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Bereich":
  - Geben Sie für "Min" z. B. den Wert "1" ein.
  - Geben Sie für "Max" z. B. den Wert "20" ein.
  - Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn die Variable sich innerhalb des angegebenen Wertebereichs befindet.



---

### Hinweis

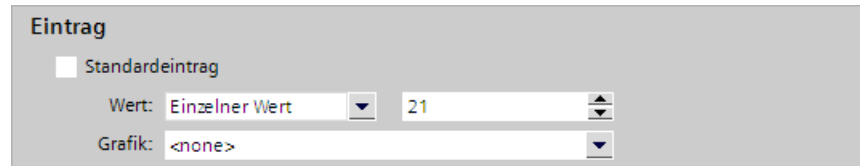
Alternativ zur Aufklappliste können Sie Grafiken aus Bibliotheken oder aus Ihrem Dateisystem einfügen:

1. Selektieren Sie eine Grafik in der Bibliothek oder in Ihrem Dateisystem.
2. Ziehen Sie per Drag&Drop die Grafik in die Tabelle "Einträge in Grafikliste > Grafik".

3. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen". Ein weiterer Listeneintrag wird erstellt.



4. Wählen Sie im Inspektorfenster ""Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Einzelner Wert":
  - Geben Sie z. B. als Wert "21" ein.
  - Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn in der Variablen das Bit "21" gesetzt wird.



Eintrag

Standardeintrag

Wert: Einzelner Wert 21

Grafik: <none>

5. Bei Bedarf aktivieren Sie "Standardeintrag".  
Die Grafik erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt.  
Pro Liste ist nur ein Standardeintrag möglich.

## Ergebnis

Eine Bereich-Grafikliste ist erstellt. Den möglichen Werten sind Grafiken zugeordnet, die in Runtime erscheinen.

## Grafiken und Werte einer Bit-Grafikliste zuordnen

### Einleitung

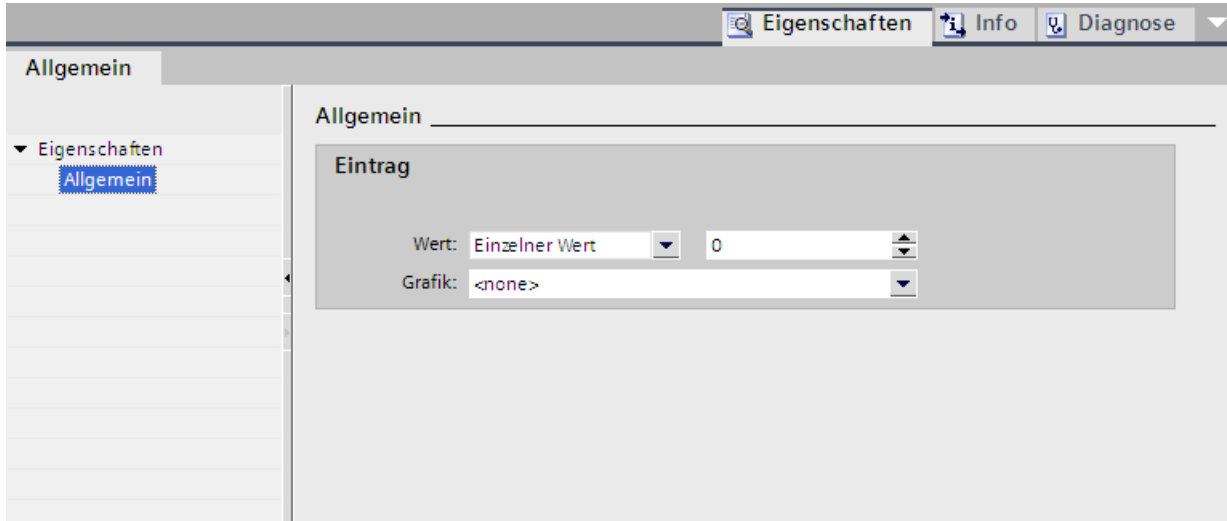
Zu jeder Grafikliste geben Sie an, bei welchem Bit-Wert welche Grafik angezeigt wird.

### Voraussetzung

- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Grafikliste" ist geöffnet.
- Eine Bit-Grafikliste ist angelegt und selektiert.

## Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen".  
Das Inspektorfenster für diesen Listeneintrag wird geöffnet.



2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Einzelner Wert":
  - Geben Sie als Wert "0" ein.
  - Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn in der Variablen das Bit "0" gesetzt wird.

---

### Hinweis

Alternativ zur Aufklappliste können Sie Grafiken aus Bibliotheken oder aus Ihrem Dateisystem einfügen:

1. Selektieren Sie eine Grafik in der Bibliothek oder in Ihrem Dateisystem.
  2. Ziehen Sie per Drag&Drop die Grafik in die Tabelle "Einträge in Grafikliste > Grafik".
- 

3. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen". Ein neuer Listeneintrag wird erstellt.
4. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Wert > Einzelner Wert":
  - Geben Sie als Wert "1" ein.
  - Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn in der Variablen das Bit "1" gesetzt wird.

## Ergebnis

Eine Bit-Grafikliste ist erstellt. Den Werten "0" und "1" sind Grafiken zugeordnet, die in Runtime erscheinen.

## Grafiken und Werte einer Bitnummer-Grafikliste zuordnen

### Einleitung

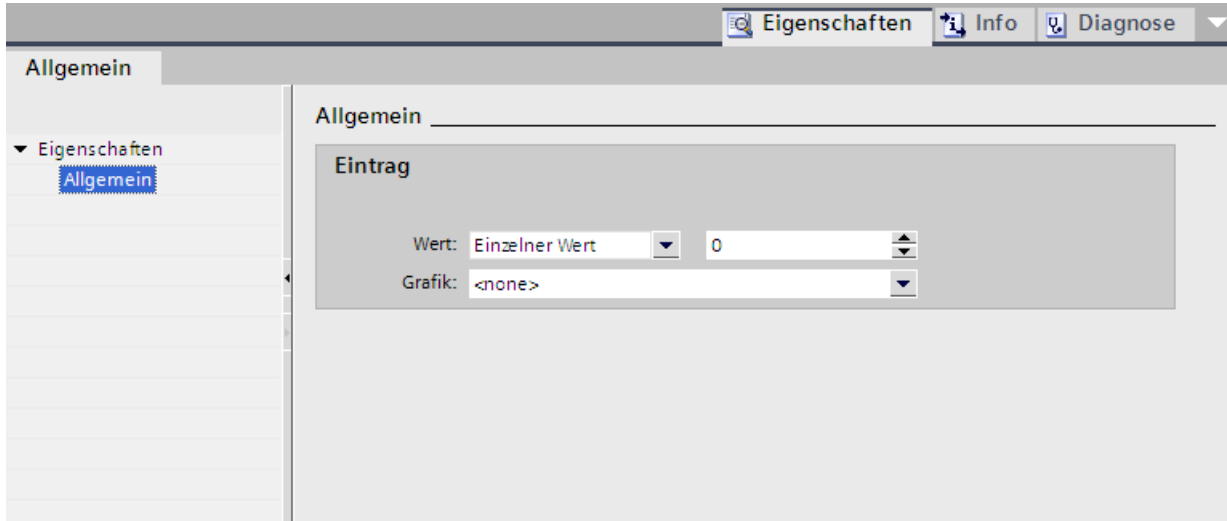
Zu jeder Bitnummer-Grafikliste geben Sie an, bei welcher Bitnummer welche Grafiken angezeigt werden sollen.

### Voraussetzung

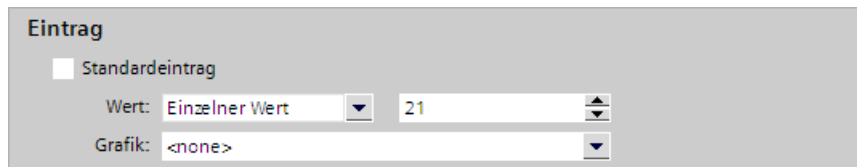
- Der Editor "Text- und Grafiklisten" ist geöffnet.
- Die Registerkarte "Grafiklisten" ist geöffnet.
- Eine Bitnummer-Grafikliste ist angelegt und selektiert.

## Vorgehen

1. Klicken Sie in der Tabelle "Einträge in Grafikliste" auf "Hinzufügen".  
Das Inspektorfenster für diesen Listeneintrag wird geöffnet.



2. Wählen Sie im Inspektorfenster ""Eigenschaften > Eigenschaften > Allgemein > Wert" die Einstellungen "Einzelner Wert":
  - Geben Sie als Wert z. B. "0" ein.
  - Wählen Sie eine Grafik aus, die in Runtime angezeigt wird, wenn in der Variablen das Bit "0" gesetzt wird



### Hinweis

Alternativ zur Aufklappliste können Sie Grafiken aus Bibliotheken oder aus Ihrem Dateisystem einfügen:

1. Selektieren Sie eine Grafik in der Bibliothek oder in Ihrem Dateisystem.
2. Ziehen Sie per Drag&Drop die Grafik in die Tabelle "Einträge in Grafikliste > Grafik".

3. Bei Bedarf aktivieren Sie "Standardeintrag".  
Die Grafik erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt.  
Pro Liste ist nur ein Standardeintrag möglich.
4. Legen Sie für weitere Bitnummern derselben Grafikliste weitere Listeneinträge an.

## Ergebnis

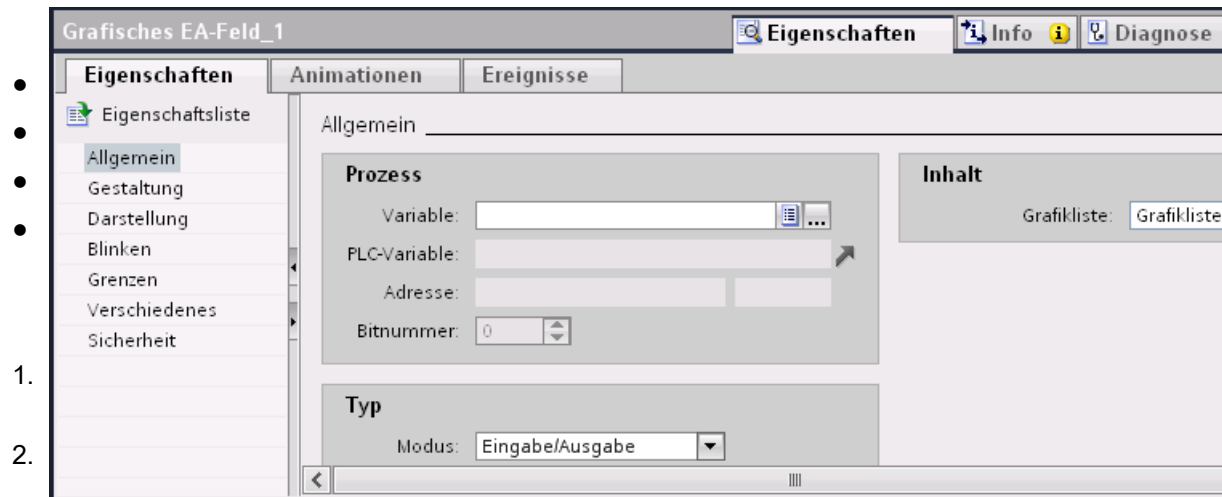
Eine Bitnummer-Grafikliste ist erstellt. Den angegebenen Bitnummern sind Grafiken zugeordnet, die in Runtime erscheinen.

## Objekt mit Grafikliste projektieren

### Einleitung

Ausgabewert und Wertübernahme für Grafiklisten werden in dem Anzeige- und Bedienobjekt festgelegt, das die Grafiken der Grafikliste in Runtime anzeigt. Die Eigenschaften dieser Objekte werden nach Bedarf konfiguriert.

### Voraussetzung



### Vorgehen

- 1.
- 2.

#### Hinweis

#### Runtime-Abhängigkeit

Je nach Runtime stehen unterschiedliche Feldtypen für ein Grafisches E/A-Feld zur Verfügung.

3. Wählen Sie als "Variable" die Variable, deren Werte die Anzeige im Grafischen E/A-Feld bestimmt.

### Ergebnis

In Runtime werden die definierten Grafiken der Grafikliste im Grafischen E/A-Feld angezeigt, wenn die Variable den angegebenen Wert annimmt.

## 10.1.4 Dynamisieren von Bildern

### 10.1.4.1 Grundlagen zur Dynamisierung

#### Dynamisierung von Objekten

In WinCC dynamisieren Sie Objekte, um Ihre Anlage auf Bediengeräten abzubilden und Prozessabläufe anzuzeigen.

Sie realisieren Dynamisierungen durch

- Animationen
- Variablen
- Systemfunktionen

Ein Beispiel ist die Abbildung eines Tanks, dessen Flüssigkeitspegel abhängig von einem Prozesswert steigt oder fällt.

Welche Möglichkeiten der Dynamisierung verfügbar sind, hängt vom Objekt ab. Wenn Sie ein Objekt kopieren, werden auch die Dynamisierungen mitkopiert.

#### Siehe auch

Dynamisierung im Inspektorfenster (Seite 3096)

Neue Animation projektieren (Seite 3098)

Grundlagen zu Ereignissen (Seite 3106)

### 10.1.4.2 Dynamisierung im Inspektorfenster

#### Einleitung

Grundsätzlich können Sie alle Bildobjekte, die Sie in einem Bild projiziert haben, dynamisieren. Vom Gerät und dem selektierten Objekt hängt ab, welche Möglichkeiten der Dynamisierung und welche Ereignisse verfügbar sind.

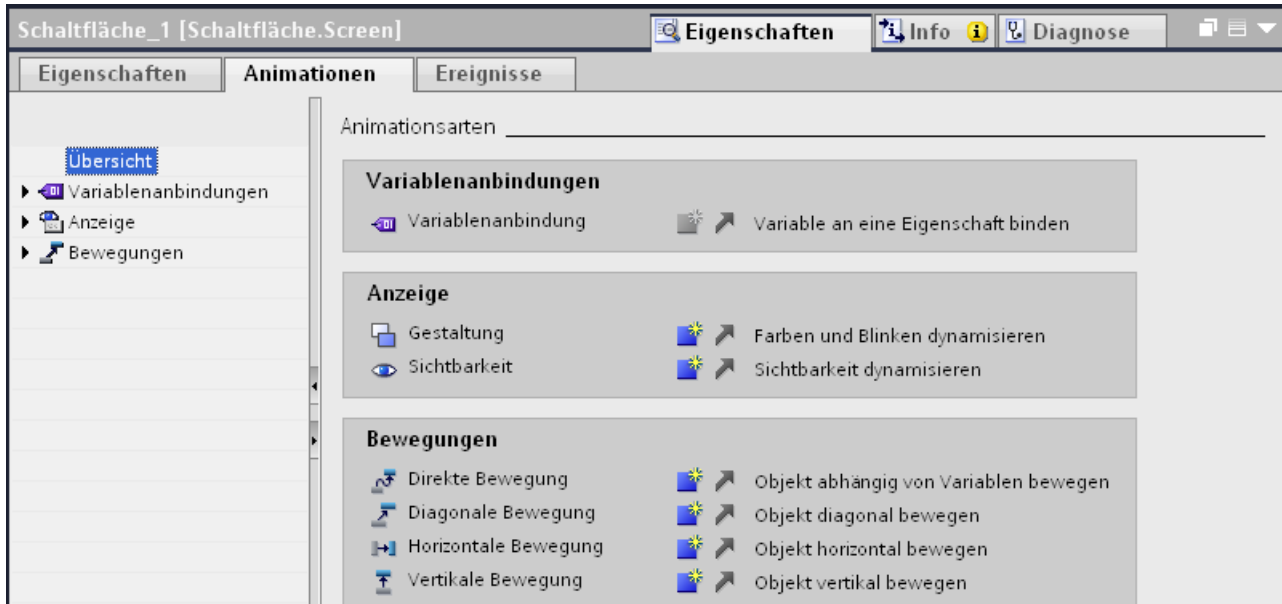
#### Animationen

WinCC unterstützt Sie bei der Dynamisierung durch bereits vordefinierte Animationen. Wenn Sie ein Objekt animieren wollen, projektieren Sie zunächst die gewünschte Animation im Inspektorfenster des Objekts. Anschließend passen Sie die Animation an die Erfordernisse ihres Projekts an.

Die Auswahl der unterstützten Animationen ist abhängig vom Bediengerät und vom selektierten Objekt. Sie wählen zwischen folgenden Animationsarten:

- Darstellung: Gestaltung, Sichtbarkeit
- Bewegungen: direkte, diagonale, horizontale und vertikale Bewegung
- Variablenanbindung

Die Animationsart "Variablenanbindung" können Sie für ein Objekt mehrfach projektieren. Animationen projektieren Sie im Inspektorfenster "Eigenschaften > Animationen".

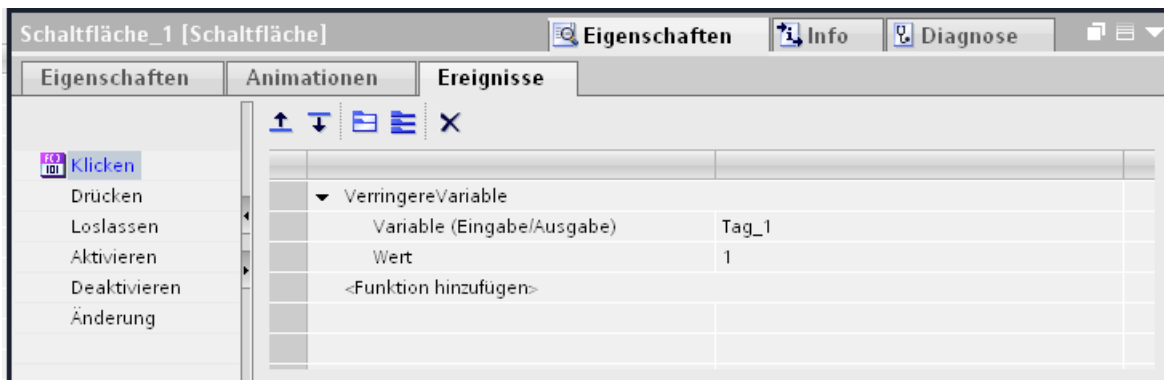


## Ereignisse

Bedienbare Objekte reagieren zusätzlich auf Ereignisse, z. B. einen Mausklick.

An ein Ereignis projektieren Sie eine Funktionsliste mit Systemfunktionen. Als Reaktion auf das ausgelöste Ereignis werden die Systemfunktionen abgearbeitet.

Ereignisse projektieren Sie im Inspektorfenster "Eigenschaften > Ereignisse".



Nähere Hinweise hierzu finden Sie unter "Arbeiten mit Funktionslisten".

## Siehe auch

Grundlagen zur Dynamisierung (Seite 3096)

### 10.1.4.3 Dynamisieren mit Animationen

#### Neue Animation projektieren


##### Einleitung

Um Bildobjekte zu dynamisieren, verwenden Sie vordefinierte Animationen.

##### Voraussetzung

- Ein Bild mit mindestens einem dynamisierbaren Objekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

##### Vorgehen im Inspektorfenster

1. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen".
2. Wählen Sie die gewünschte Animation aus.
3. Klicken Sie auf die Schaltfläche .

##### Vorgehen in der Task Card "Animationen"

1. Öffnen Sie in der Task Card "Animationen" die Objektgruppe, welche die gewünschte Animation enthält.
2. Ziehen Sie die gewünschte Animation auf das Objekt, das Sie dynamisieren wollen.

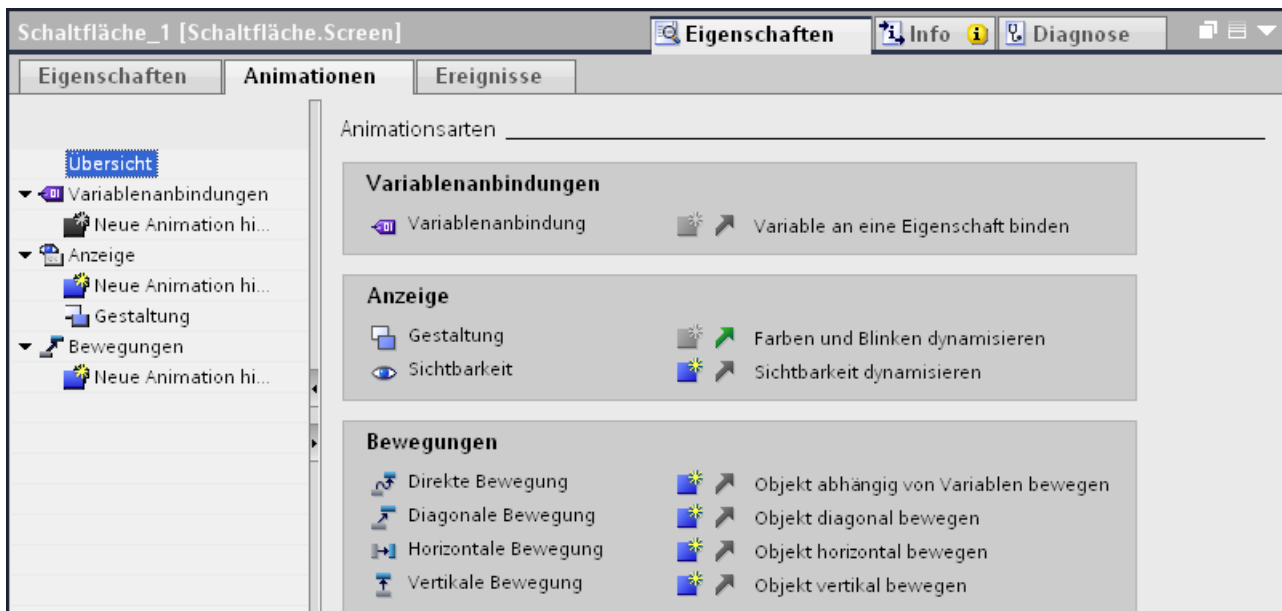
Alternativ selektieren Sie das Objekt im Bild und doppelklicken in der Task Card "Animation" auf die gewünschte Animation.

##### Ergebnis

Die Animation wird im Inspektorfenster des Objekts eingetragen. In den folgenden Schritten konfigurieren Sie die Animation.

In der Übersicht der Animationen symbolisiert Ihnen der grünen Pfeil, welche Animation bereits projiziert ist. Wenn Sie auf den grünen Pfeil klicken, dann öffnet sich die projizierte Animation im Inspektorfenster.





## Siehe auch

Sichtbarkeit eines Objekts dynamisieren (Seite 3103)

Grundlagen zur Dynamisierung (Seite 3096)

## Gestaltung eines Objekts dynamisieren

### Einleitung

In Runtime wird das Erscheinungsbild eines Bildobjekts durch die Wertänderung einer Variablen gesteuert. Wenn die Variable einen bestimmten Wert annimmt, ändert das Bildobjekt entsprechend der Projektierung seine Farbe oder sein Blinkverhalten.


### Typ

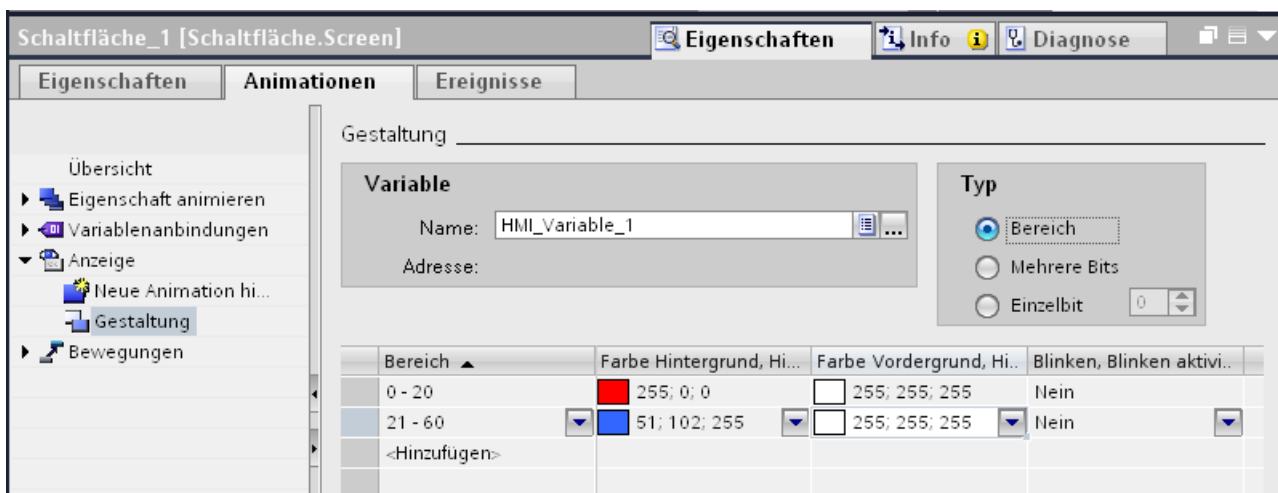
Je nach Auswahl werden Bereiche oder Einzelwerte der Variablen in Runtime betrachtet. Das Erscheinungsbild des Objekts ändert sich entsprechend der Projektierung.

### Voraussetzung

- Ein Bild ist geöffnet.
- Im Bild ist ein dynamisierbares Objekt enthalten und markiert.
- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

## Vorgehen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen". Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
2. Wählen Sie die Animation "Gestaltung" und klicken Sie auf die Schaltfläche . Die Parameter der Animation werden angezeigt.
3. Wählen Sie "Variable > Name" eine Variable aus.
4. Wählen Sie z. B. "Typ > Bereich".
5. Klicken Sie in der Tabelle auf "Hinzufügen".
6. Geben Sie in die Spalte "Bereich" z. B. das Variablenintervall "0 - 20" ein.
7. Wählen Sie für "Vordergrundfarbe" und "Hintergrundfarbe" die Farbe aus, die das Objekt annimmt, sobald in Runtime die Variable das Intervall erreicht.
8. Wählen Sie aus der Liste "Blinken" ein Blinkverhalten für das Objekt.
9. Um ein weiteres Variablenintervall z. B. "21 - 60" anzulegen, wiederholen Sie die Schritte 5-8.



## Ergebnis

Abhängig vom Prozesswert der Variablen ändern sich in Runtime das Blinkverhalten und die Farbe des Objekts.

## Bewegung projektieren

### Einleitung


Dynamische Objekte können Sie so projektieren, dass sie sich auf einer bestimmten Bahn bewegen. Die Bewegung wird über Variablen gesteuert. Bei jeder Aktualisierung der Variablen wird das Objekt bewegt.

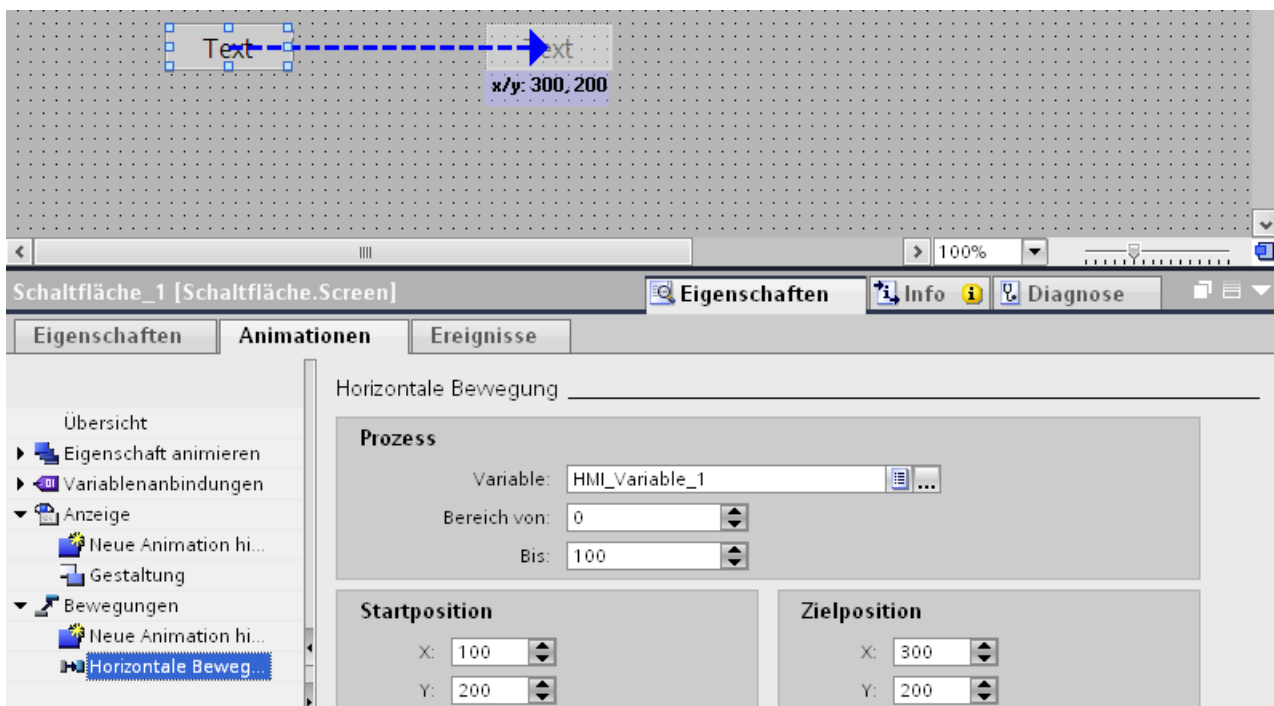
Für jedes Objekt können Sie nur eine einzige Bewegungsart projektieren.

## Voraussetzung

- Eine Variable ist angelegt.
- Ein Bild mit mindestens einem dynamisierbaren Objekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

## Vorgehen

1. Markieren Sie im Bild das Objekt, das Sie dynamisieren wollen. Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen". Die für das markierte Objekt verfügbaren Animationen werden angezeigt.
3. Wählen Sie "Horizontale Bewegung" und klicken Sie auf die Schaltfläche . Die Parameter der Animation werden angezeigt. Im Arbeitsbereich wird eine transparente Kopie des Objekts eingeblendet, die durch einen Pfeil mit dem Originalobjekt verbunden ist.
4. Wählen Sie eine Variable zur Bewegungssteuerung aus.
5. Schieben Sie die Objektkopie an die gewünschte Zielposition. Im Inspektorfenster werden die Pixelwerte der Endposition automatisch eingetragen.
6. Passen Sie bei Bedarf den Wertebereich der Variablen an.



## Ergebnis

In Runtime bewegt sich das Objekt bei jeder Wertänderung der Variablen, die zur Steuerung der Bewegung dient. Die Bewegungsrichtung entspricht der projizierten Bewegungsart "horizontal".

---

### Hinweis

Vertikale und diagonale Bewegung projizieren Sie auf die gleiche Art wie die horizontale Bewegung.

---

## Direkte Bewegung projizieren


### Einleitung

Bei der "Direkten Bewegung" wird das Objekt jeweils in x-Richtung und in y-Richtung verschoben. Zwei Variablen legen die Anzahl der Pixel fest, um die sich das Objekt von seiner ursprünglichen statischen Anfangsposition bewegt.

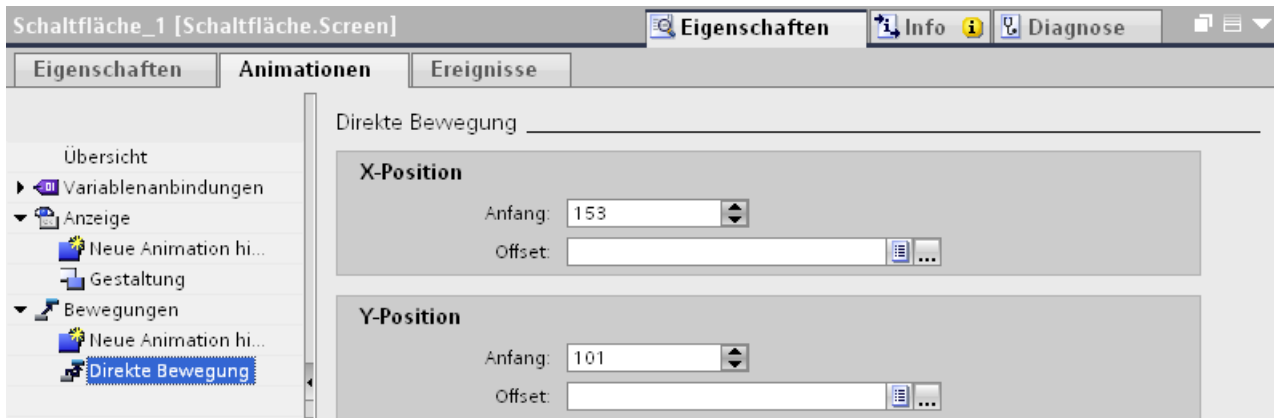
### Voraussetzung

- Zwei Variable sind angelegt.
- Ein Bild mit mindestens einem dynamisierbaren Objekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Das Werkzeugfenster wird angezeigt.

### "Direkte Bewegung" projizieren

1. Markieren Sie im Bild das Objekt, das Sie dynamisieren wollen. Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen".
3. Wählen Sie "Direkte Bewegung" und klicken Sie auf die Schaltfläche . Die Parameter der Animation werden angezeigt.

4. Wählen Sie für die X-Position eine Variable, mit welcher die Bewegung in x-Richtung gesteuert wird.
5. Wählen Sie für die Y-Position eine Variable, mit welcher die Bewegung in y-Richtung gesteuert wird.



## Ergebnis

In Runtime bewegt sich das Objekt bei jeder Wertänderung der Variablen, die zur Steuerung der Bewegung dient.

## Sichtbarkeit eines Objekts dynamisieren

### Einleitung

Mit dem Dynamisieren der Eigenschaft "Sichtbarkeit" können Sie z. B. eine Warnung im Bild einblenden, wenn eine Variable einen kritischen Wert überschreitet. Wenn der Wert der Variablen wieder im unkritischen Bereich liegt, wird die Warnung ausgeblendet.


Die Objekte "Einfache Rezeptanzeige" und "Einfache Meldeanzeige" sind immer sichtbar.

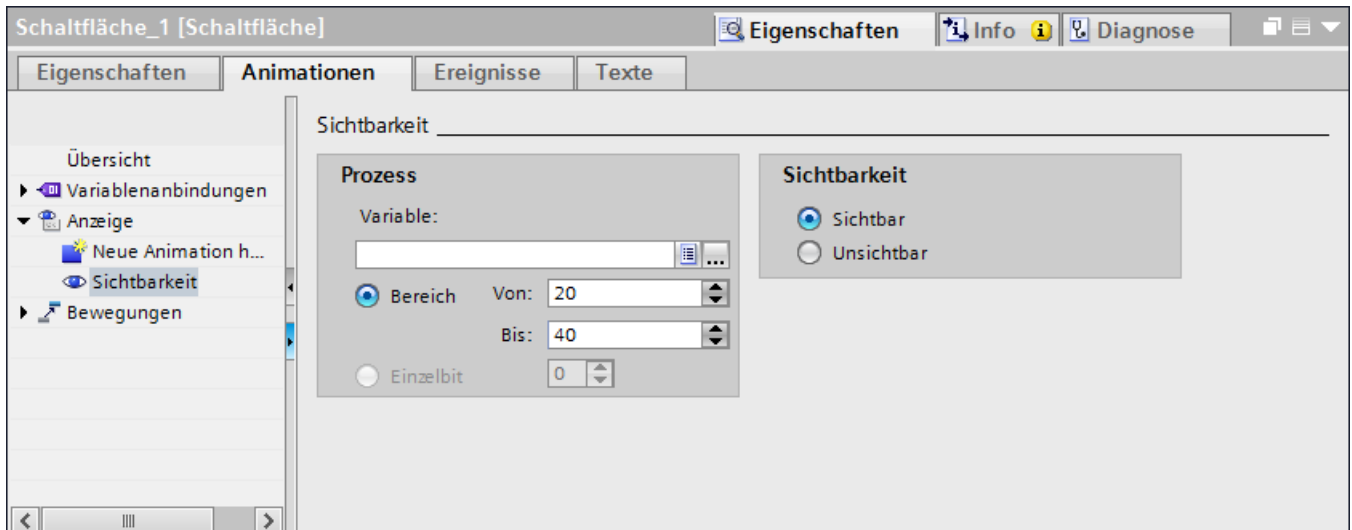
### Voraussetzung

- Eine Variable ist angelegt.
- Ein Bild mit einem Objekt, das Sie in Runtime einblenden oder ausblenden wollen, ist geöffnet.
- Das Inspektorfenster ist geöffnet.

### Vorgehen

1. Markieren Sie im Bild das Objekt, das Sie dynamisieren wollen. Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Animationen". Die für das markierte Objekt verfügbaren Animationen werden angezeigt.

3. Wählen Sie "Sichtbarkeit" und klicken Sie auf die Schaltfläche . Die Parameter der Animation werden angezeigt.
4. Wählen Sie eine Variable.
5. Aktivieren Sie "Bereich".
6. Wählen Sie z.B. "Von 20 und "Bis 40".
7. Aktivieren Sie "Sichtbar".



## Ergebnis

Das Bildobjekt wird in Runtime abhängig vom Variablenwert eingeblendet oder ausgeblendet:

- Wenn der Variablenwert mit dem projizierten Bereich zwischen 20 und 40 übereinstimmt, wird das Bildobjekt eingeblendet.
- Wenn der Variablenwert ausserhalb des projizierten Bereichs liegt, wird das Bildobjekt ausgeblendet.

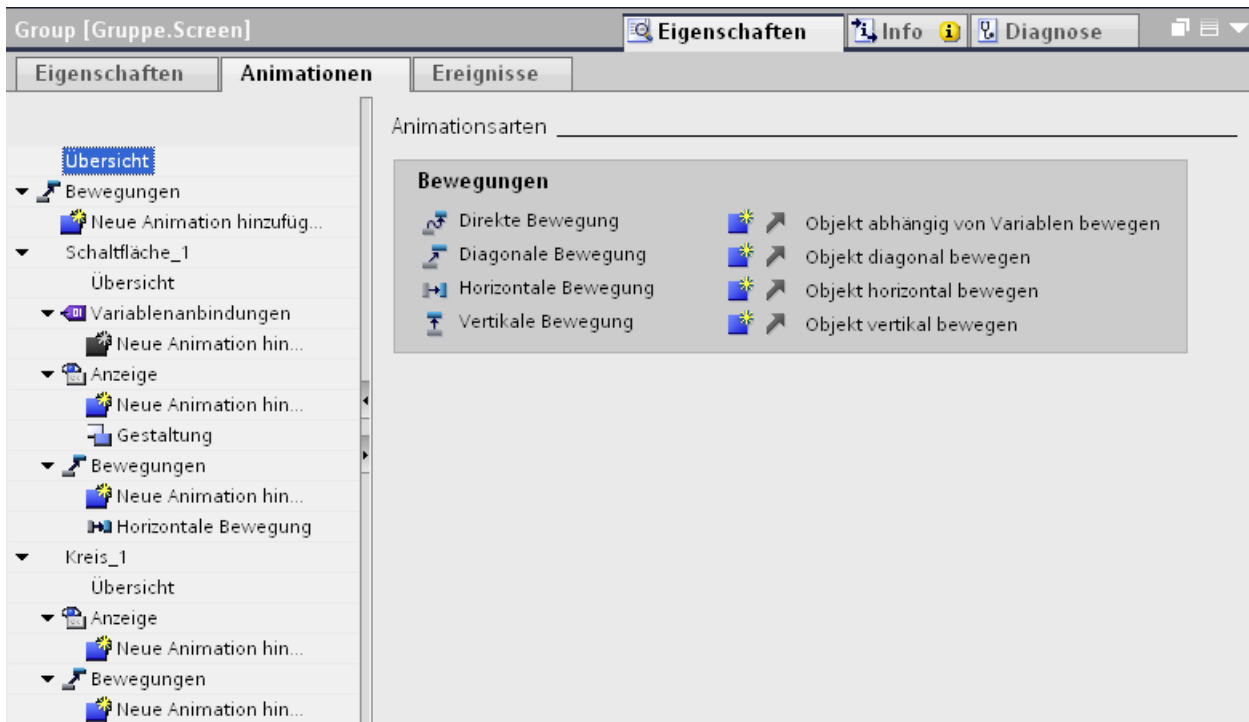
## Siehe auch

Neue Animation projektieren (Seite 3098)

## Animationen von Objektgruppen

### Animationen auf Objektgruppen anwenden

Das Inspektorfenster zeigt alle Objekte einer Gruppe und ihre möglichen Animationen. Zusätzlich werden die Animationsarten separat aufgeführt, die von allen Objekten der Gruppe unterstützt werden.



Wenn Sie eine Animation für eine Objektgruppe projektieren, gilt diese Animation für alle Einzelobjekte, die diese Animation unterstützen.

### Anwendungsbeispiel

Für das Objekt einer Objektgruppe ist die Animation "horizontale Bewegung" projektiert. Für die gesamte Objektgruppe ist die Animation "direkte Bewegung" projektiert. In Runtime wird nur die Animation der Objektgruppe d. h. "direkte Bewegung" ausgeführt. Dies gilt auch für Objektgruppen innerhalb von Objektgruppen. Es wird nur die Animation der Gruppe der obersten Ebene aufgeführt.

### Animationen bei Objektgruppen und bei Mehrfachselektion

#### Animationen von mehreren Objekten ändern

Bei einer Mehrfachauswahl werden im Inspektorfenster die Animationen angezeigt, die für das Referenzobjekt projektiert sind. Die Animationen können Sie wie gewohnt ändern. Die

Änderungen gelten für alle Objekte der Mehrfachauswahl, welche die projektierte Animation unterstützen. Somit erhalten auch Objekte die Animation des Referenzobjekts, für die Sie bisher keine Animation projektiert haben.

### Anwendungsbeispiel

Sie markieren gleichzeitig eine Schaltfläche und einen Kreis. Die Schaltfläche ist das Referenzobjekt. Für die Schaltfläche ist bereits die Animation "Gestaltung" projektiert und wird daher im Inspektorfenster der Mehrfachauswahl angezeigt. Wenn Sie in im Inspektorfenster "Eigenschaften > Animationen > Gestaltung > Blinken" aktivieren, gelten danach die Einstellungen der Animation "Gestaltung" für die Schaltfläche und für den Kreis.

### Neue Animationen für mehrere Objekte projektieren

Wenn Sie für die Objekte einer Mehrfachauswahl eine neue Animation projektieren, gilt diese Animation für alle markierten Objekte, welche die projektierte Animation unterstützen. Wenn die neue Animation eine bereits bestehende Animation ersetzen würde, wird eine Sicherheitsabfrage angezeigt.

### Anwendungsbeispiel

Sie markieren einen Kreis und ein Rechteck. Für den Kreis ist bereits die Animation "diagonale Bewegung" projektiert. Sie projektieren für die Mehrfachauswahl die Animation "horizontale Bewegung". Diese Animation gilt für das Rechteck, da dafür noch keine Animation des Typs Bewegung projektiert ist. Für den Kreis werden Sie gefragt, ob Sie die bestehende Animation "diagonale Bewegung" durch die neue Animation "horizontale Bewegung" ersetzen wollen.

## 10.1.4.4 Dynamisieren mit Systemfunktionen

### Grundlagen zu Ereignissen

#### Einleitung

Bildobjekte reagieren auf Ereignisse. An die Ereignisse eines Objekts projektieren Sie eine Funktionsliste mit Systemfunktionen.

#### Ereignisse

Welche Ereignisse und Systemfunktionen verfügbar sind, hängt vom verwendeten Objekt ab.

Wenn der Bediener z. B. ein Bildobjekt aktiviert, wird die projektierte Systemfunktion ausgeführt.

Weitere Informationen finden Sie unter "Grundlagen zur Funktionsliste (Seite 3341)".



## Siehe auch

Grundlagen zur Dynamisierung (Seite 3096)

Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 3107)

Systemfunktion an das Ereignis "Klicken" projektieren (Seite 3107)

## Systemfunktion an das Ereignis "Klicken" projektieren

### Einleitung

Sie projektieren eine Funktionsliste an ein Ereignis eines Objekts. Wenn das Ereignis in Runtime eintritt, wird die verknüpfte Systemfunktion ausgeführt.

### Voraussetzungen

Ein Bild ist geöffnet.

Eine Schaltfläche ist im Bild angelegt.

Das Inspektorfenster ist geöffnet.

### Vorgehen

1. Selektieren Sie die Schaltfläche.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
3. Wählen Sie das Ereignis "Klicken".
4. Klicken Sie in der Tabelle auf "Funktion hinzufügen"
5. Wählen Sie die Systemfunktion "ZeigeMeldefenster".

### Ergebnis

Wenn der Bediener in Runtime auf die Schaltfläche klickt, öffnet sich das Meldefenster im Bild.

## Siehe auch

Grundlagen zu Ereignissen (Seite 3106)

## Beispiel: Schaltfläche zur Sprachumschaltung projektieren

### Einleitung

In diesem Beispiel projektieren Sie eine Schaltfläche, mit der Sie in Runtime zwischen mehreren Runtime-Sprachen umschalten.

## Voraussetzungen

- Sie haben das Beispiel "Mehrsprachige Beschriftung einer Schaltfläche projektieren" ausgeführt.
- Das Bild "Screen\_1" ist geöffnet.
- Die Schaltfläche ist im Bild selektiert.

## Vorgehen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Drücken".
2. Klicken Sie in der Tabelle auf "Funktion hinzufügen".
3. Wählen Sie die Systemfunktion "SetzeSprache" und die Einstellung "Umschalten".

## Ergebnis

Sie haben der Schaltfläche die Funktion "SetzeSprache" zugewiesen. Wenn Sie in Runtime die Schaltfläche drücken, dann schaltet die Runtime-Sprache um. Die Reihenfolge der Runtime-Sprachen wird so umgesetzt, wie sie im Editor "Sprache & Schriftart" durch die Nummerierung der Sprachen vorgegeben ist.

## Siehe auch

Grundlagen zu Ereignissen (Seite 3106)

Schaltfläche (Seite 3151)

## 10.1.5 Arbeiten mit Funktionstasten

### 10.1.5.1 Arbeiten mit Funktionstasten

#### Einleitung

Eine Funktionstaste ist eine Taste am Bediengerät mit projektierbarer Funktionsbelegung. An die Ereignisse "Taste gedrückt" und "Taste loslassen" ist eine Funktionsliste projektierbar.

Sie können eine Funktionstaste global oder lokal belegen.

---

#### Hinweis

##### **Bediengeräteabhängigkeit**

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

---

## Globale Funktionstasten

Funktionstasten mit globaler Belegung lösen immer die gleiche Aktion aus, unabhängig vom angezeigten Bild.

Sie projektieren globale Funktionstasten im Editor "Globales Bild". Die globale Belegung gilt für alle Bilder des eingestellten Bediengeräts.

Global belegte Funktionstasten reduzieren den Projektierungsaufwand deutlich, weil Sie den globalen Tasten nicht in jedem einzelnen Bild Funktionen zuweisen.

## Lokale Funktionstasten in Bildern

Lokale Funktionstasten in Bildern können in jedem Bild eine andere Aktion triggern. Die Belegung gilt nur für das Bild, in dem Sie die Funktionstaste definiert haben.

Innerhalb eines Bildes hat eine Funktionstaste nur eine Funktionsbelegung, entweder global oder lokal. Welche Belegung den Vorrang hat, legt der Projekteur fest.

---

### Hinweis

Wenn ein Bild mit lokalen Funktionstasten von einer Meldeanzeige oder einem Meldefenster überdeckt wird, dann sind die Funktionstasten in Runtime trotzdem aktiv. Dieser Fall kann besonders bei Bediengeräten mit kleinem Display eintreten.

---

## Lokale Funktionstasten in Vorlagen

Lokale Funktionstasten, die in Vorlagen belegt sind, gelten für alle Bilder, die auf dieser Vorlage basieren. Sie können in jedem Bild eine andere Aktion triggern. Funktionstasten in Vorlagen belegen Sie in der Vorlage im Editor "Bilder". Innerhalb einer Vorlage hat eine Funktionstaste nur eine Funktionsbelegung, entweder global oder lokal. Welche Belegung den Vorrang hat, legt der Projekteur fest.

## Belegung mit Hotkeys

Sie können Bedienobjekte mit Hotkeys belegen, z. B. Schaltflächen. Welche Hotkeys zur Verfügung stehen, ist vom Bediengerät abhängig.

---

### Hinweis

Die Funktionstaste ist mit einer lokalen oder globalen Aktion belegt. Wenn Sie die Funktionstaste zusätzlich mit einem Hotkey belegen, wird in Runtime die Funktion des Hotkeys ausgeführt.








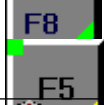

---

## Grafiken

Wenn eine Funktionstaste unmittelbar neben dem Display angeordnet ist, können Sie der Funktionstaste eine Grafik zuordnen, welche die Funktion der Funktionstaste verdeutlicht.

### Anzeige der Belegung

Tabelle 10-1 Die folgende Tabelle zeigt, welche Symbole die Belegung der Funktionstasten anzeigen:

| Funktionstaste  | Beschreibung   |
|---|--|
|    | Nicht belegt   |
|    | Global belegt  |
|    | Lokal belegt in der Vorlage  |
|    | Lokal belegt   |
|   | Lokal belegt (lokale Belegung der Vorlage überschreibt die globale Belegung)   |
|  | Lokal belegt (lokale Belegung überschreibt die globale Belegung)   |
|  | Lokal belegt (lokale Belegung überschreibt lokale Belegung der Vorlage)  |
|  | Lokal belegt (lokale Belegung überschreibt lokale Belegung der Vorlage, die bereits die globale Belegung überschreibt) |
|  | Schaltfläche mit Bildnavigation belegt   |

**Hinweis**

**Basic Panels**

Der Editor "Bildnavigation" steht für Basic Panels nicht zur Verfügung.

### 10.1.5.2 Funktionstasten global belegen

#### Einleitung

Sie definieren die globale Belegung einer Funktionstaste im Editor "Globales Bild". Die globale Belegung gilt in allen Bildern des eingestellten Bediengeräts.

---

#### Hinweis

#### Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

---

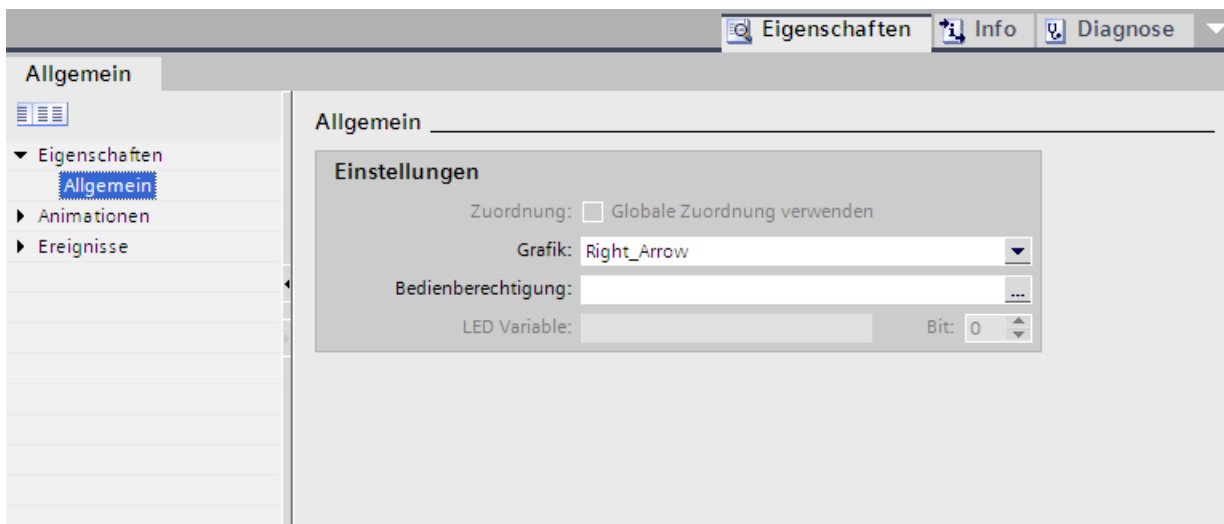
#### Voraussetzung

- Das Projekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.

#### Vorgehen

Um einer Funktionstaste eine bildunabhängige Funktion zuzuweisen, gehen Sie folgendermaßen vor:

1. Um den Editor "Globales Bild" zu öffnen, doppelklicken Sie im Projektfenster in der Gruppe "Bildverwaltung" auf "Globales Bild".
2. Selektieren Sie die gewünschte Funktionstaste.  
Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.



3. Konfigurieren Sie unter "Allgemein" eine Grafik und eine Bedienberechtigung für die Funktionstaste.
4. Projektieren Sie unter "Ereignisse" eine Funktionsliste an das gewünschte Ereignis.

## Ergebnis

Wenn nicht eine lokale Belegung die globale Belegung überschreibt, dann ändert sich in allen Bildern des eingestellten Bediengeräts die Belegung der Funktionstaste entsprechend Ihrer Eingabe.

### 10.1.5.3 Funktionstasten lokal belegen

#### Einleitung

Funktionstasten werden global und lokal belegt. Eine lokale Belegung der Funktionstasten gilt nur für das Bild oder die Vorlage, in der sie definiert wurde. Folgende lokale Funktionstasten gibt es:

- Lokale Funktionstasten eines Bildes  
Für jedes Bild belegen Sie die Funktionstaste unterschiedlich mit Funktionen. Die Belegung gilt nur für das Bild, in dem Sie die Funktionstaste definiert haben.
- Lokale Funktionstasten einer Vorlage  
Sie belegen die Funktionstasten in einer Vorlage. Die Belegung gilt für alle Bilder, die auf dieser Vorlage basieren und nicht durch eine lokale Belegung in einem Bild überschrieben sind.

Eine lokale Belegung überschreibt die globale Belegung einer Funktionstaste.

---

#### Hinweis

##### Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

---

#### Verwendung vorhandener Belegungen

Die Option zur Verwendung von vorhandenen Belegungen ist im Inspektorfenster folgendermaßen benannt:

- In einer Vorlage: "Globale Zuordnung verwenden"
- In einem Bild:
  - Bild basiert auf einer Vorlage: "Lokale Vorlage verwenden"
  - Bild basiert nicht auf einer Vorlage: "Globale Zuordnung verwenden"

Die Option "Lokale Vorlage verwenden" beinhaltet die Verwendung der lokalen Belegung in der Vorlage und die globale Belegung.

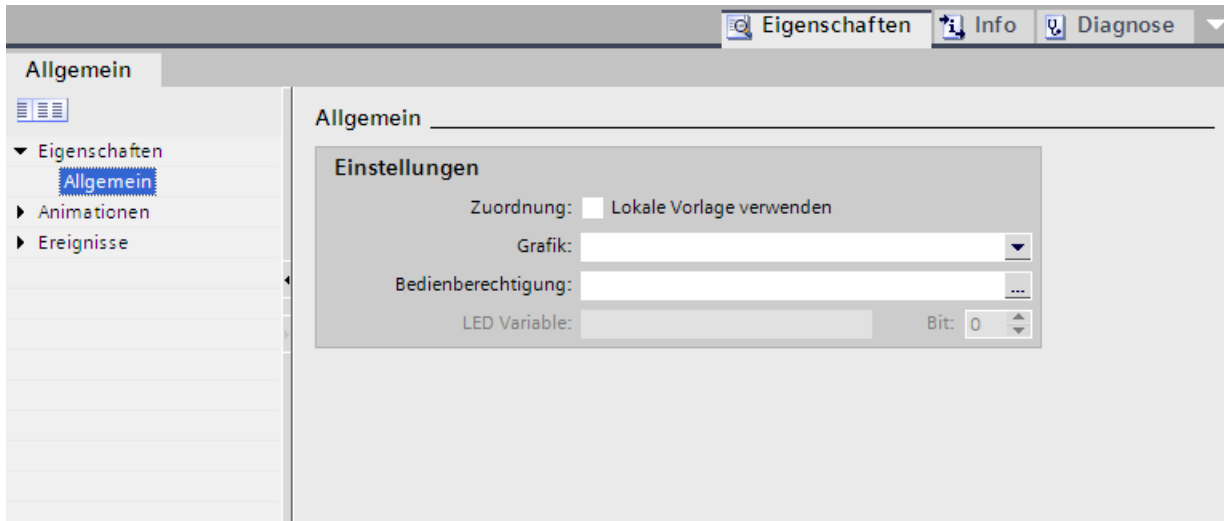
#### Voraussetzung

- Das Bild oder die Vorlage, in der Sie eine Funktionstaste lokal belegen wollen, ist geöffnet.
- Das Inspektorfenster ist geöffnet.

## Vorgehen

Gehen Sie folgendermaßen vor:

1. Selektieren Sie im Bild oder in der Vorlage die gewünschte Funktionstaste. Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf "Allgemein".



3. Deaktivieren Sie die Option "Lokale Vorlage verwenden" oder "Globale Zuordnung verwenden".
4. Konfigurieren Sie unter "Allgemein" eine Grafik und eine Bedienberechtigung für die Funktionstaste.
5. Projektieren Sie unter "Ereignisse" eine Funktionsliste an das gewünschte Ereignis.

## Ergebnis

Im Bild oder in der Vorlage ist die Funktionstaste mit den projektierten Funktionen belegt.

### 10.1.5.4 Einer Funktionstaste eine Funktion zuweisen

#### Einleitung

Eine Funktionstaste kann zwei Zustände haben:

- Gedrückt: Definiert durch das Ereignis "Taste gedrückt"
- Losgelassen: Definiert durch das Ereignis "Taste loslassen"

Beide Ereignisse projektieren Sie im Inspektorfenster der Funktionstaste. Sie können jedem Ereignis eine Funktionsliste zuweisen, die Systemfunktionen oder Skripte enthält. In Runtime wird die Funktionsliste beim Auftreten des Ereignisses abgearbeitet.

---

**Hinweis**

**Bediengeräteabhängigkeit**

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

---

**Hinweis**

**Basic Panels**

Skripte stehen für Basic Panels nicht zur Verfügung.

---

## Voraussetzung

Wenn Sie eine Funktionstaste global belegen wollen:

- Der Editor "Globales Bild" ist geöffnet.

Wenn Sie eine Funktionstaste lokal belegen wollen:

- Das Bild, in dem Sie eine Funktionstaste belegen wollen, ist geöffnet.

Wenn Sie eine Funktionstaste in einer Vorlage lokal belegen wollen:

- Die Vorlage, in der Sie eine Funktionstaste belegen wollen, ist geöffnet.
- Das Inspektorfenster ist geöffnet.

## Vorgehen

Gehen Sie folgendermaßen vor:

1. Markieren Sie die Funktionstaste, die Sie definieren wollen.  
Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Projektieren Sie im Inspektorfenster unter "Eigenschaften" in der Gruppe "Allgemein" die Funktionsliste an das gewünschte Ereignis.

## Ergebnis

Wenn der Bediener die Funktionstaste drückt oder loslässt, dann wird in Runtime die Funktionsliste abgearbeitet.

### 10.1.5.5 Bedienberechtigung für eine Funktionstaste vergeben

## Einleitung

In WinCC können Sie in Runtime für eine Funktionstaste eine Bedienberechtigung vergeben. Damit schränken Sie bereits beim Erstellen Ihres Projekts den Benutzerkreis für



Funktionstasten auf spezielle Personen oder Bedienergruppen ein. Nur autorisiertes Personal verändert dann in Runtime wichtige Parameter und Einstellungen.

Wenn Sie einen Zugriffsschutz projektieren, verhindern Sie Fehlbedienungen und erhöhen die Sicherheit der Anlage oder Maschine.

### Hinweis

#### Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

### Voraussetzung

- Die Benutzergruppen sind definiert.

Wenn Sie eine globale Funktionstaste schützen wollen:

- Der Editor "Globales Bild" ist geöffnet.

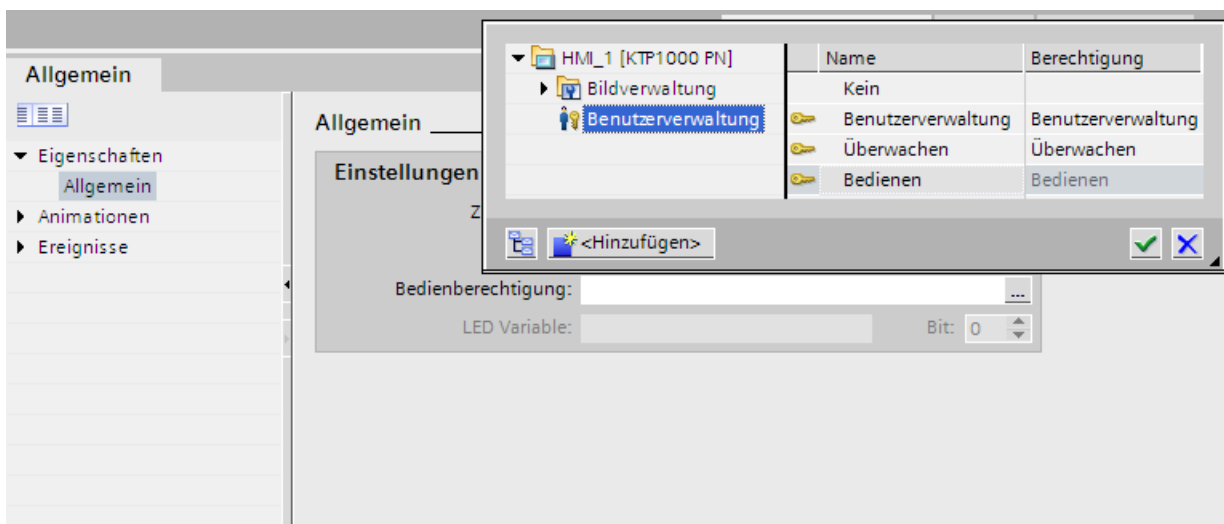
Wenn Sie eine lokale Funktionstaste eines Bildes oder einer Vorlage schützen wollen:

- Das Bild oder die Vorlage mit der Funktionstaste ist geöffnet.
- Das Inspektorfenster ist geöffnet.

### Vorgehen

Gehen Sie folgendermaßen vor:

1. Markieren Sie die gewünschte Funktionstaste.  
Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf "Allgemein".



3. Wählen Sie aus der Liste "Berechtigung" die Benutzergruppe, welche die Funktionstaste in Runtime bedienen darf.

## Ergebnis

Die Bedienberechtigung ist projiziert.

### 10.1.5.6 Einer Funktionstaste eine Grafik zuweisen

## Einleitung

Um die Funktion einer Taste zu verdeutlichen, können Sie in einem Bild eine Grafik neben der Funktionstaste einfügen. Grafiken weisen Sie nur solchen Funktionstasten zu, die an den Bildschirm des Bediengeräts angrenzen.

---

### Hinweis

#### Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

---

## Voraussetzung

Wenn Sie einer globalen Funktionstaste eine Grafik zuweisen wollen:

- Der Editor "Globales Bild" ist geöffnet.

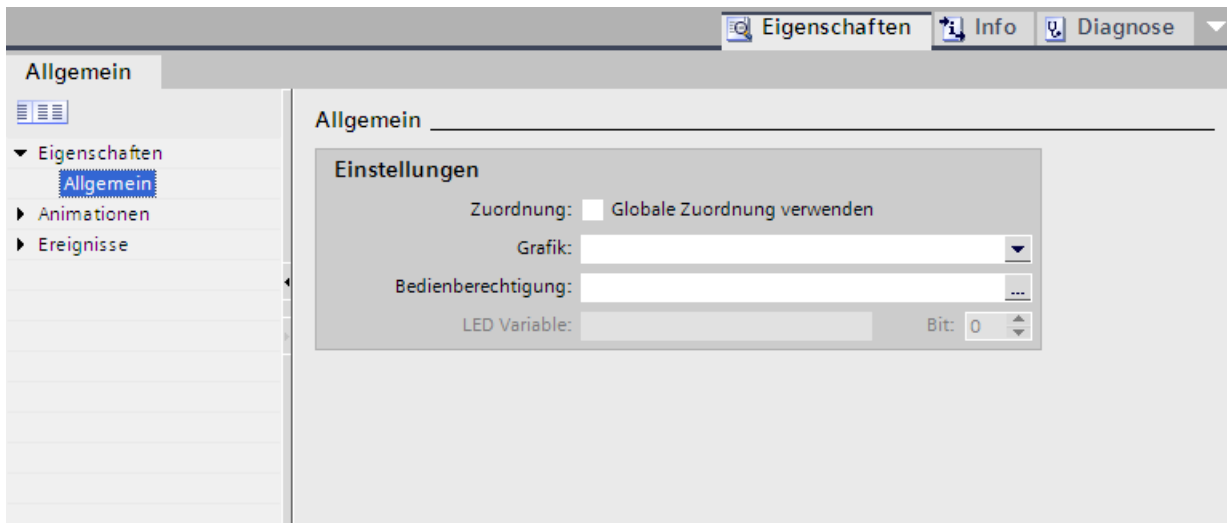
Wenn Sie einer lokalen Funktionstaste in einem Bild oder einer Vorlage eine Grafik zuweisen wollen:

- Das Bild oder die Vorlage mit der betreffenden Funktionstaste ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Die Grafik für die Funktionstaste ist erstellt.

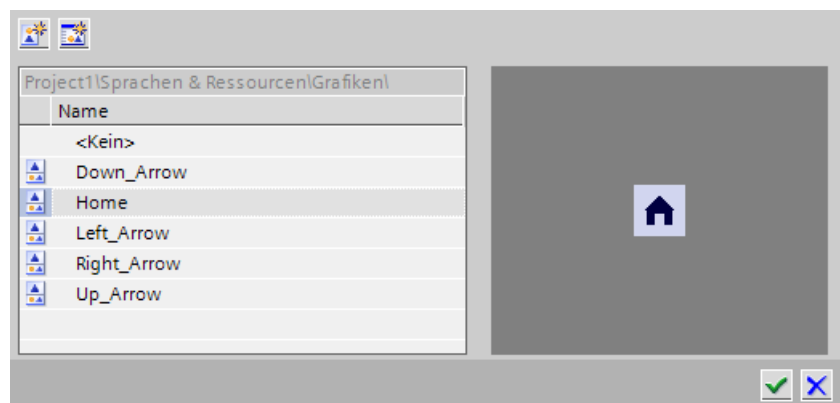
## Vorgehen



Gehen Sie folgendermaßen vor:

1. Markieren Sie die gewünschte Funktionstaste.  
Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf "Allgemein".



3. Klicken Sie unter "Grafik" in die Liste.  
Die Grafiksammlung des WinCC-Projekts wird angezeigt. Links werden Ihnen die externen Grafiken angezeigt, die bereits in der Sammlung hinterlegt sind. Im rechten Anzeigebereich sehen Sie eine Vorschau der Grafik, die in der Sammlung markiert ist.



Mit den Symbolen  und  stellen Sie die Sammlung als Thumbnails oder als Liste dar.

Um OLE-Objekte mit dem dazugehörigen Grafikprogramm zu öffnen und zu bearbeiten, doppelklicken Sie auf das Objekt.

4. Klicken Sie in der Sammlung auf die gewünschte Grafik oder hinterlegen Sie die Grafik in der Grafiksammlung.  
Die Grafik wird rechts in der Vorschau angezeigt.
5. Um die Grafik in das Bild einzufügen, klicken Sie auf "Auswählen".  
Mit "Abwählen" entfernen Sie eine Grafik aus dem Bild.

## **Ergebnis**

Die Grafik wird neben der Funktionstaste angezeigt.

### **10.1.5.7 LED-Variablen projektieren**

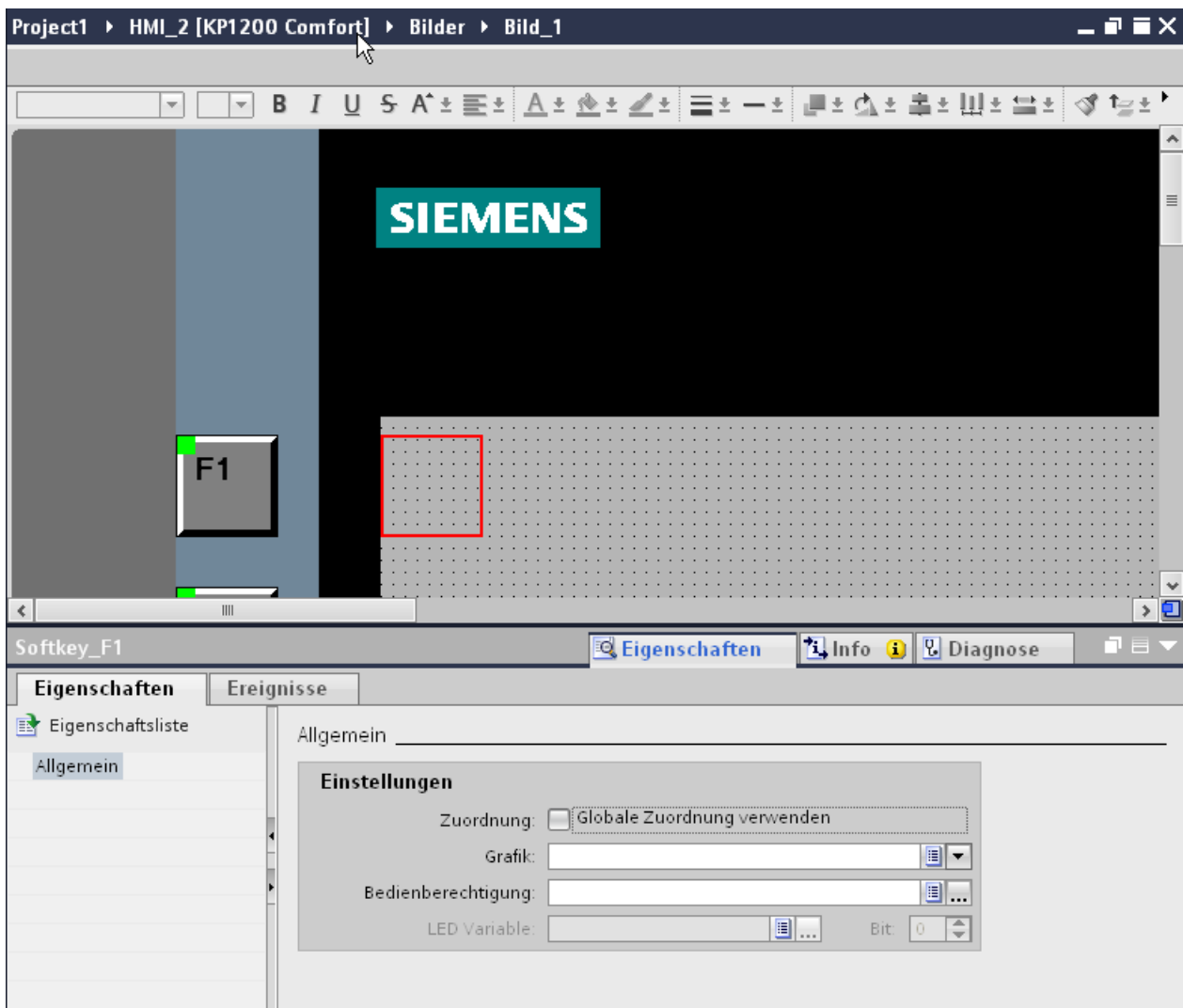
#### **Voraussetzungen**

- Bediengerät mit Tastenbedienung ist angelegt.
- LED Variable ist angelegt.

#### **Vorgehensweise**

1. Legen Sie ein neues Bild an.
2. Klicken Sie auf eine F-Taste des Bediengeräts.

3. Klicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften" auf "Allgemein".



4. Wählen Sie im Bereich "Allgemein > Einstellungen" unter "LED Variable" eine Variable aus.

5. Geben Sie unter "Bit" die passende Bit-Nummer ein.

Die passende Bitnummer ist abhängig vom Bediengerät und der Belege der Ein- und Ausgänge am Bediengerät.

### Belegung der Ein- und Ausgänge

Die genaue Belegung der Ein- und Ausgänge finden Sie unter:

- PROFINET IO-Direktasten: Auto-Hotspot
- PROFIBUS DP-Direktasten: Auto-Hotspot

### 10.1.5.8 Beispiel: Funktionstasten zur Bildnavigation verwenden

#### Aufgabe

In diesem Beispiel belegen Sie eine Funktionstaste lokal in einem Bild. Wenn der Bediener diese Funktionstaste drückt, wird ein Bildwechsel zu einem vordefinierten Bild z. B. mit dem Namen "Kessel2" getriggert.

---

#### Hinweis

##### Bediengeräteabhängigkeit

Funktionstasten stehen nicht bei allen Bediengeräten zur Verfügung.

---

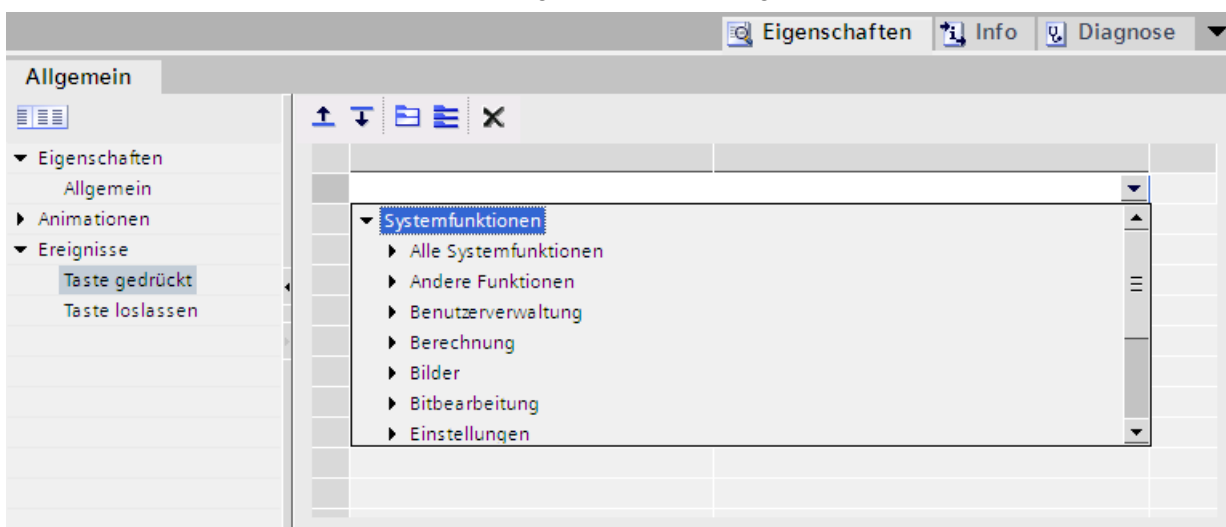
#### Voraussetzung

- Das Bild, in dem Sie die Funktionstaste belegen wollen, ist geöffnet.
- Das Bild "Kessel2" ist angelegt.
- Das Inspektorfenster ist geöffnet.

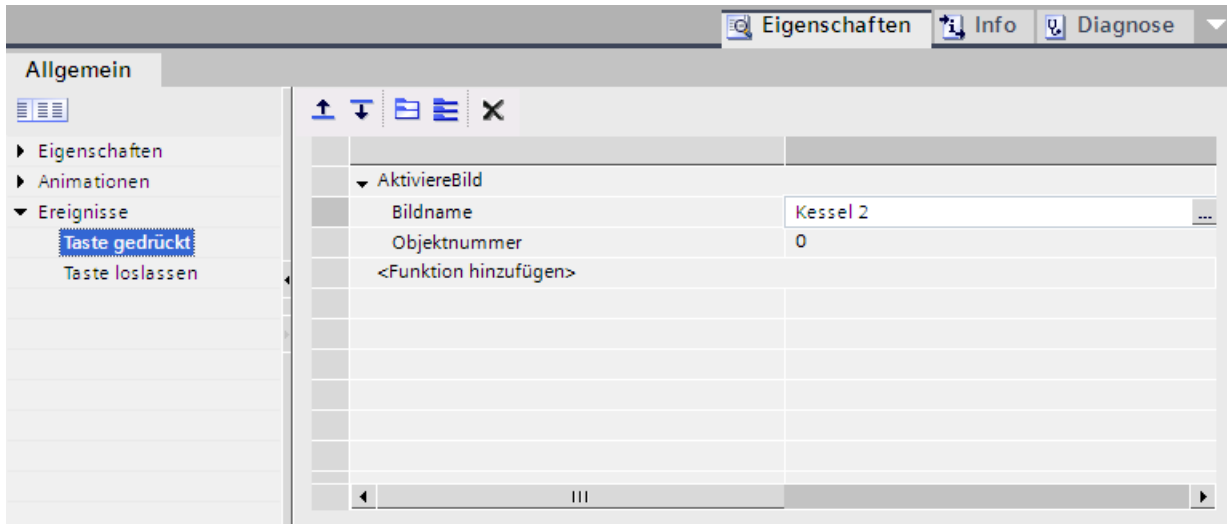
#### Vorgehen

Um die Funktion "AktiviereBild" zu verwenden, gehen Sie folgendermaßen vor:

1. Selektieren Sie die gewünschte Funktionstaste.  
Die Eigenschaften der Funktionstaste werden im Inspektorfenster angezeigt.
2. Klicken Sie auf "Allgemein".
3. Um eine globale Belegung zu überschreiben, deaktivieren Sie die Option "Lokale Vorlage verwenden".
4. Klicken Sie unter "Ereignisse" auf "Taste gedrückt".



5. Wählen Sie aus der Liste die Systemfunktion "AktiviereBild" aus.  
Die Funktion "AktiviereBild" mit den Parametern "Bildname" und "Objektnummer" wird unter der Gruppe "Funktionsliste" angezeigt.



6. Wählen Sie aus der Liste "Bildname" das Bild "Kessel 2" aus.

## Ergebnis

Wenn der Bediener in Runtime die ausgewählte Funktionstaste drückt, wechselt er zum Bild "Kessel 2".

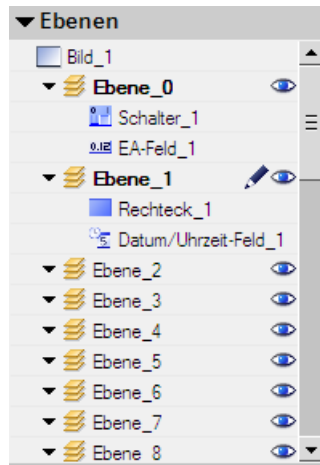
## 10.1.6 Arbeiten mit Ebenen

### 10.1.6.1 Grundlagen zum Arbeiten mit Ebenen

#### Ebenen

Um Objekte eines Bildes differenziert zu bearbeiten, verwenden Sie Ebenen. Ein Bild besitzt 32 Ebenen, die Sie beliebig benennen können. Wenn Sie den Ebenen Objekte zuordnen, dann

legen Sie damit die Tiefenstaffelung des Bildes fest. Objekte der Ebene 0 liegen im Hintergrund des Bildes, Objekte der Ebene 31 im Vordergrund.



Die Objekte einer einzelnen Ebene sind auch hierarchisch angeordnet. Wenn Sie ein Bild erstellen, wird das zuerst eingefügte Objekt innerhalb der Ebene hinten angeordnet. Jedes weitere Objekt wird jeweils eine Position weiter vorn eingefügt. Innerhalb einer Ebene können Sie die Objekte nach vorne und hinten verschieben.

## Prinzip der Ebenentechnik

Von den 32 Ebenen ist immer eine Ebene aktiv. Wenn Sie Objekte in ein Bild einfügen, werden diese standardmäßig der aktiven Ebene zugeordnet. Die Nummer der aktiven Ebene wird im Inspektorfenster des Bildes angezeigt und in der Task Card "Layout > Ebenen".

Wenn Sie ein Bild öffnen, werden immer alle 32 Ebenen des Bildes angezeigt. Im Inspektorfenster des Bildes und in der Task Card "Layout > Ebenen" können Sie alle Ebenen außer der aktiven Ebene ausblenden. Sie bearbeiten dann gezielt die Objekte der aktiven Ebene.

In der Baumansicht der Palette "Ebenen" in der Task Card "Layout" verwalten Sie Ebenen und Objekte per Drag&Drop und über das Kontextmenü.

## Anwendungsbeispiele

Ebenen verwenden Sie z. B. in folgenden Fällen:

- Um die Beschriftung von Objekten auszublenden, während Sie editieren
- Um Objekte, z. B. Meldefenster, auszublenden, während Sie weitere Objekte projektieren

### 10.1.6.2 Objekte auf Ebenen verschieben

#### Einleitung

Standardmäßig liegen neu eingefügte Objekte auf der aktiven Ebene. Sie können ein Objekt nachträglich einer anderen Ebene zuordnen.



## Voraussetzung



- Ein Bild mit einem Objekt ist geöffnet.
- Das Inspektorfenster ist geöffnet.

## Vorgehen

1. Selektieren Sie das Objekt im Bild.  
Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verschiedenes > Ebene" die Ebene ein, auf die Sie das Objekt schieben wollen.

Alternativ wählen Sie in der Task Card "Layout" das Objekt und ziehen Sie es per Drag&Drop auf die gewünschte Ebene.

## Reihenfolge von Objekten ändern

1. Selektieren Sie das Objekt im Bild.  
Die Objekteigenschaften werden im Inspektorfenster angezeigt.
2. Um das Objekt in der Reihenfolge nach vorne oder nach hinten zu verschieben, wählen Sie im Kontextmenü den Befehl "Reihenfolge > Eins nach hinten" oder "Eins nach vorne".  
Alternativ verwenden Sie die Schaltfläche  oder  in der Symbolleiste.


## Ergebnis

Das Objekt wird der ausgewählten Ebene zugewiesen und ganz im Vordergrund eingeordnet.

### 10.1.6.3 Aktive Ebene einstellen

## Einleitung

In einem Bild sind die Bildobjekte immer einer von 32 Ebenen zugeordnet. Im Bild ist immer eine Ebene aktiv. Wenn Sie Objekte in ein Bild einfügen, werden diese standardmäßig der aktiven Ebene zugeordnet.

Die Nummer der aktiven Ebene wird in der Symbolleiste "Ebene" angezeigt. Die aktive Ebene wird in der Task Card "Layout > Ebenen" mit dem Symbol  gekennzeichnet.

Zu Beginn der Projektierung ist die Ebene 0 die aktive Ebene. Während des Projektierens aktivieren Sie bei Bedarf eine andere Ebene.

## Voraussetzung

- Ein Bild mit mindestens einem Objekt ist geöffnet.
- Das Inspektorfenster des aktiven Bildes ist geöffnet.

## Vorgehen

1. Klicken Sie im Inspektorfenster des aktuellen Bildes "Eigenschaften > Eigenschaften > Ebenen".
2. Geben Sie unter "Einstellungen > Aktive Ebene" die Nummer der Ebene ein.

## Alternatives Vorgehen

1. Wählen Sie in der Task Card "Layout > Ebenen".
2. Wählen Sie im Kontextmenü einer Ebene den Befehl "Auf aktiviert setzen".

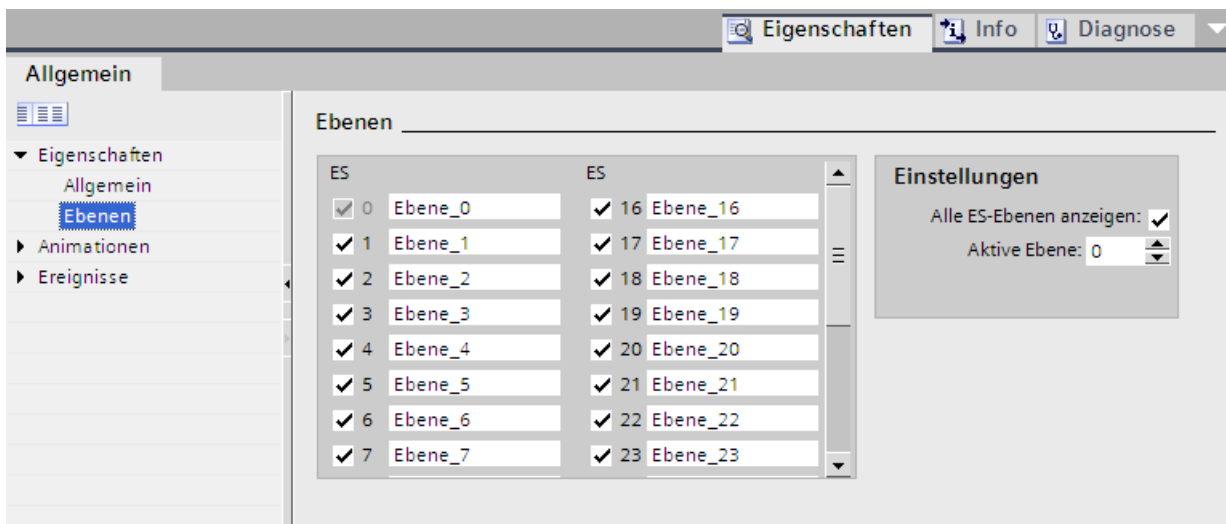
## Ergebnis

Die Ebene mit der angegebenen Nummer wird zur aktiven Ebene.

### 10.1.6.4 Ebenen ein- und ausblenden

## Einleitung



Sie blenden die Ebenen eines Bildes bei Bedarf ein oder aus. Sie legen fest, welche Ebenen im Engineering System eingeblendet werden. Wenn Sie ein Bild öffnen, sind immer alle Ebenen eingeblendet.



## Voraussetzung

- Das Bild ist geöffnet.
- Die Task Card "Layout" ist geöffnet.

## Vorgehen

1. Wählen Sie in der Task Card "Layout > Ebenen" die Ebene, die Sie aus- oder einblenden wollen.
2. Klicken Sie auf eines der Symbole neben der entsprechenden Ebene:
  -  ausgeblendete Ebene wird eingeblendet
  -  eingeblendete Ebene wird ausgeblendet

---

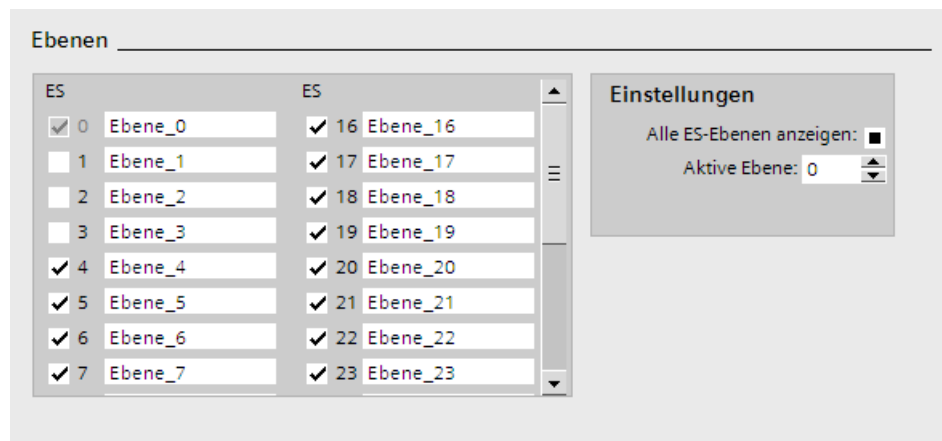
### Hinweis

Die aktive Ebene können Sie nicht ausblenden.

---

## Alternatives Vorgehen

1. Klicken Sie im Bild auf eine Stelle, an der kein Objekt vorhanden ist. Die Eigenschaften des Bildes werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Ebenen".



3. Deaktivieren Sie in der Liste die Ebenen, die Sie ausblenden wollen. Wenn Sie für eine Ebene "Alle ES-Ebenen" aktivieren, werden die Objekte dieser Ebene im Engineering System angezeigt.

## Ergebnis

Die Ebenen werden entsprechend Ihren Einstellungen angezeigt.

### 10.1.6.5 Ebenen umbenennen

## Einleitung

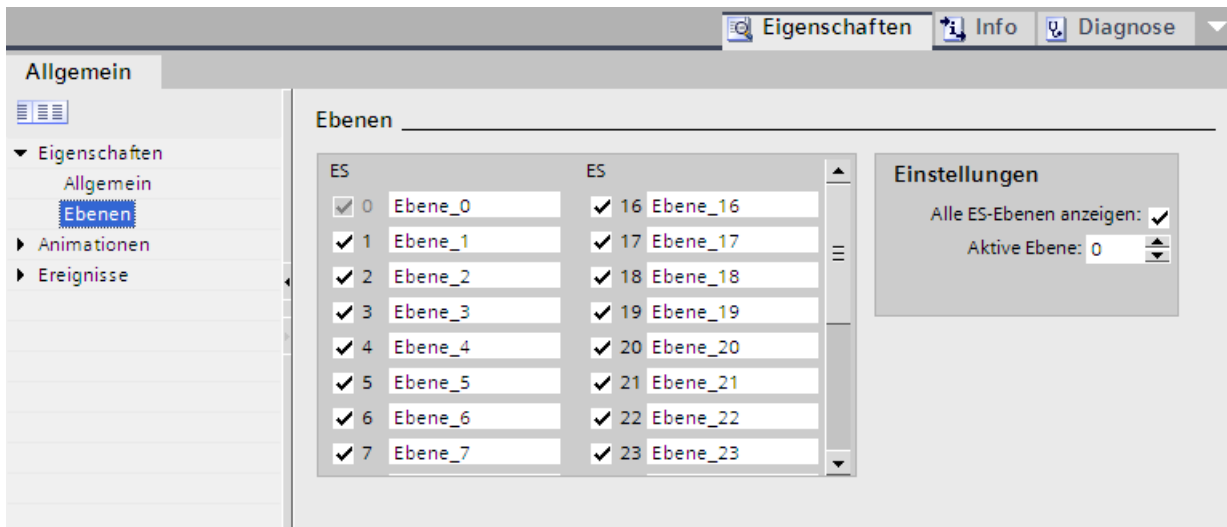
Wenn Sie ein Bild anlegen, sind die 32 Ebenen standardmäßig durchnummeriert. Zur besseren Übersicht benennen Sie die Ebenen nach Ihren Anforderungen um.

### Voraussetzung

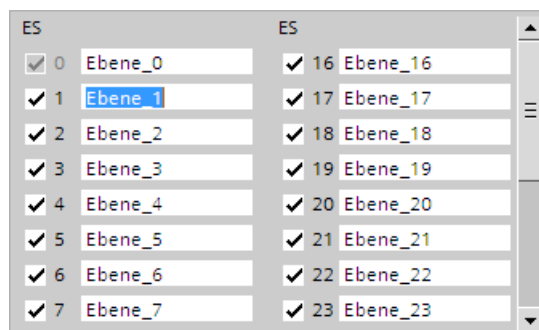
- Das Bild ist geöffnet.

### Vorgehen

1. Klicken Sie im Bild auf eine Stelle, an der kein Objekt vorhanden ist. Die Eigenschaften des Bildes werden im Inspektorfenster angezeigt.
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Ebenen".



3. Geben Sie den neuen Ebenennamen ein.



### Ergebnis

Die Ebene wird mit neuem Namen angezeigt.

## 10.1.7 Anzeige- und Bedienobjekte

### 10.1.7.1 Geräteabhängigkeit der Objekte

#### Objekte für Basic Panels

#### Verfügbarkeit der Anzeige- und Bedienobjekte für Basic Panels

Im Objektfenster werden nur die Objekte angezeigt, die auf dem Gerät, für das Sie projektieren verwendet werden können. Die nachfolgende Tabelle zeigt die Verfügbarkeit der Anzeige- und Bedienobjekte auf Basic Panels.

#### Übersicht

|                              | KP300 Basic<br>KP400 Basic | KTP400 Basic<br>KTP600 Basic<br>KTP1000 Basic<br>TP1500 Basic |
|------------------------------|----------------------------|---|
| Balken                       | ja                         | ja  |
| Benutzeranzeige              | ja                         | ja  |
| Datum/Uhrzeit-Feld           | ja                         | ja  |
| E/A-Feld                     | ja                         | ja  |
| Ellipse                      | ja                         | ja  |
| Grafikanzeige                | ja                         | ja  |
| Grafisches E/A-Feld          | ja                         | ja  |
| Hilfeindikator               | ja                         | nein  |
| Kreis                        | ja                         | ja  |
| Kurvenanzeige                | ja                         | ja  |
| Linie                        | ja                         | ja  |
| Meldeanzeige<br>Meldefenster | ja                         | ja  |
| Meldeindikator               | ja                         | ja  |
| Rechteck                     | ja                         | ja  |
| Rezepturanzeige              | ja                         | ja  |
| Schaltfläche                 | ja                         | ja  |
| Schalter                     | ja                         | ja  |
| Symbolisches E/A-Feld        | ja                         | ja  |
| System-Diagnoseanzeige       | ja                         | ja  |
| Textfeld                     | ja                         | ja  |

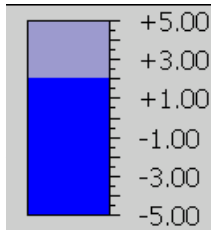
**Siehe auch**

Übersicht über die Objekte (Seite 3040)

**10.1.7.2 Objekte**

**Balken**

**Verwendung**



"Balken" stellen Sie Variablen grafisch dar. Die Balkenanzeige kann mit einer Skala beschriftet werden.

**Darstellung**

Sie ändern im Inspektorfenster die Einstellung zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- **Farbverlauf:** Legt die Darstellung des Farbumschlags beim Überschreiten von Grenzwerten fest.
- **Grenzwertlinie, Grenzwertmarkierung anzeigen:** Zeigt den projizierten Grenzwert als Linie oder Markierung an.
- **Balkensegmente definieren:** Legt die Einteilung der Balkenskala fest.
- **Skaleneinteilung festlegen:** Legt die Unterteilungen, Teilstriche und Intervalle einer Balkenskala fest.

**Farbverlauf**

Sie legen im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung" die Darstellung des Farbumschlags fest.

| Farbverlauf     | Beschreibung  |
|-----------------|---|
| "Segmentweise"  | Wenn eine bestimmte Grenze erreicht wurde, färbt sich der Balken segmentweise. Mit segmentweiser Darstellung visualisieren Sie z.B. welche Grenzen von dem angezeigten Wert überschritten werden. |
| "Ganzer Balken" | Wenn eine bestimmte Grenze erreicht wurde, färbt sich der gesamte Balken.   |

**Grenzwertlinien und Grenzwertmarkierung anzeigen**

Sie zeigen mit der Eigenschaft "Linien" und "Markierungen" den projizierten Grenzwert im Balken in Runtime als Linie oder als Markierung an:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
2. Aktivieren Sie "Linien" und "Markierungen".

## Balkensegmente definieren

Sie legen mit der Eigenschaft "Unterteilungen" die Anzahl der Segmente fest, in die der Balken durch die Hauptteilstriche der Skala unterteilt wird.

Sie definieren mit der Eigenschaft "Intervall" den Abstand zwischen den Hauptteilstrichen. Der Wert wird als Wertdifferenz zwischen zwei benachbarten Hauptteilstrichen angegeben:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Skalen".
2. Aktivieren Sie "Skala anzeigen".
3. Wählen Sie für "Einstellungen > Unterteilungen" den entsprechenden Wert.
4. Wählen Sie für "Einstellungen > Beschriftung Teilstriche" den entsprechenden Wert.
5. Wählen Sie für "Großes Intervall > Intervall" den entsprechenden Wert.

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

## Benutzeranzeige

### Verwendung

Das Objekt "Benutzeranzeige" dient der Einrichtung und Verwaltung von Benutzern und Berechtigungen.

|                   |                     |
|-------------------|---------------------|
| Administrator     | Administrator group |
| Meister           | Group_1             |
| Operator_1        | Group_1             |
| PLC User          | Unberechtigt        |
| <Neuer Benutzer > |                     |

---

### Hinweis

Verwenden Sie die einfache Benutzeranzeige nicht in einer Gruppe.

---

## Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Anzahl Zeilen: Legt die maximale Anzahl der sichtbaren Einträge fest.

## Anzahl Zeilen

Die Anzahl der Zeilen in der Benutzeranzeige, die in Runtime dargestellt werden, wird im Inspektorfenster festgelegt. Die Einstellung der Anzahl der Zeilen wirkt nur, wenn die Eigenschaft "Objekt an Inhalt anpassen" aktiviert ist.

1. Klicken Sie im Inspektorfenster auf die Gruppe "Eigenschaften > Eigenschaften > Anzeige".
2. Geben Sie unter "Anzahl Zeilen" einen ganzzahligen Zahlenwert ein.
3. Aktivieren Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
4. Aktivieren Sie "Objekt an Inhalt anpassen".

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

Einfache Benutzeranzeige (Seite 3994)

Benutzeranzeige projektieren (Seite 3321)

## Datum/Uhrzeit-Feld

### Verwendung

Das Objekt "Datum/Uhrzeit-Feld" zeigt die Systemzeit und das Systemdatum an. Die Darstellung im "Datum/Uhrzeit-Feld" ist abhängig von der am Bediengerät eingestellten Sprache.

31.12.2000 10:59:59

## Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Systemzeit anzeigen: Legt fest, dass die Systemzeit angezeigt wird.
- Variable verwenden: Legt fest, dass die Uhrzeit der verbundenen Variable angezeigt wird.
- Langes Datum/Uhrzeit-Format: Legt fest, in welchem Format Datum und Uhrzeit angezeigt werden.



## Systemzeit anzeigen

Die Uhrzeit, die im "Datum/Uhrzeit-Feld" auf dem Bediengerät angezeigt wird, wird im Inspektorfenster festgelegt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Aktivieren Sie "Format > Systemzeit".

## Variable verwenden

Im Datum/Uhrzeit-Feld wird die Uhrzeit der verbundenen Variable angezeigt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie im Bereich "Format" eine Variable vom Datentyp "DateTime", z. B. eine interne Variable.

## Langes Datum/Uhrzeit-Format

Die Darstellung von Datum und Uhrzeit wird im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Format" festgelegt.

| Option        | Beschreibung  |
|---------------|---|
| "Aktiviert"   | Datum und Uhrzeit werden vollständig angezeigt, z. B. "Sunday, December 31, 2000 10:59:59 AM" |
| "Deaktiviert" | Datum und Uhrzeit werden verkürzt angezeigt, z. B. "12/31/2000 10:59:59 AM"                   |

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

## E/A-Feld

## Verwendung

Das Objekt "E/A-Feld" dient der Eingabe und Anzeige von Prozesswerten.



## Darstellung

Im Inspektorfenster ändern Sie die Einstellung zu Position, Geometrie, Stil, Farben und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Modus: Legt das Verhalten des Objekts in Runtime fest.
- Anzeigeformat: Legt das Anzeigeformat im E/A-Feld für die Eingabe und Ausgabe von Werten fest.
- Verdeckte Eingabe: Legt fest, ob der Eingabewert während der Eingabe normal oder verschlüsselt angezeigt wird.

---

### Hinweis

#### Protokolle

In Protokollen geben E/A-Felder ausschließlich Daten aus. Der Modus "Ausgabe" ist voreingestellt. Eigenschaften zum Konfigurieren der Eingabe stehen nicht zur Verfügung, z. B. "Verdeckte Eingabe".

---

## Modus

Das Verhalten des E/A-Feldes legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Typ" fest.

| Modus             | Beschreibung  |
|-------------------|---|
| "Eingabe"         | In das E/A-Feld können in Runtime nur Werte eingegeben werden.            |
| "Eingabe/Ausgabe" | In das E/A-Feld können in Runtime Werte eingegeben und ausgegeben werden. |
| "Ausgabe"         | Das E/A-Feld wird nur zur Ausgabe von Werten eingesetzt.                  |

## Darstellung

Das "Anzeigeformat" für die Eingabe und Ausgabe von Werten legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Format" fest.

| Darstellung     |  |
|-----------------|--|
| "Binär"         | Eingabe und Ausgabe von Werten in binärer Schreibweise.  |
| "Datum"         | Eingabe und Ausgabe von Datumsangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache.                 |
| "Datum/Uhrzeit" | Eingabe und Ausgabe von Datumsangaben und Zeitangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache. |
| "Dezimal"       | Eingabe und Ausgabe von Werten in dezimaler Schreibweise.  |
| "Hexadezimal"   | Eingabe und Ausgabe von Werten in hexadezimaler Schreibweise.  |
| "Uhrzeit"       | Eingabe und Ausgabe von Zeitangaben. Das Format ist abhängig von der am Bediengerät eingestellten Sprache.                   |
| "Zeichenkette"  | Eingabe und Ausgaben von Zeichenketten.  |

---

**Hinweis****Datenformate**

Für Runtime Professional stehen nicht alle Datenformate zur Auswahl.

---

**Verdeckte Eingabe**

In Runtime kann die Eingabe normal oder verschlüsselt angezeigt werden, z. B. zur verdeckten Eingabe eines Kennworts. Bei der verdeckten Eingabe wird für jedes Zeichen ein "\*" angezeigt. Das Datenformat des Eingabewerts ist nicht erkennbar.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Verhalten".
2. Aktivieren Sie "Verdeckte Eingabe".

**Überschneidungen vermeiden bei Ausgabefeldern**

Wenn in einem Bild mehrere E/A-Felder als Ausgabefelder mit transparentem Hintergrund projiziert sind, überschneiden sich eventuell diese E/A-Felder. Der transparente Teil des einen Feldes überdeckt die Ziffern des anderen Feldes. Dabei kommt es möglicherweise zu Darstellungsproblemen in Runtime. Um solche Überschneidungen zu vermeiden, setzen Sie in den Objekteigenschaften unter "Eigenschaften > Eigenschaften > Gestaltung" die Rahmen der E/A-Felder auf null. Aktivieren Sie "Eigenschaften > Eigenschaften > Darstellung > Objekt an Inhalt anpassen".

**Siehe auch**

Geräteabhängigkeit der Objekte (Seite 3127)

**Ellipse****Verwendung**

Das Objekt "Ellipse" ist ein geschlossenes Objekt, das mit einer Farbe oder einem Muster gefüllt werden kann.

**Darstellung**

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radius horizontal: Legt den horizontalen Radius des elliptischen Objekts fest.
- Radius vertikal: Legt den vertikalen Radius des elliptischen Objekts fest.

### Radius horizontal

Den horizontalen Radius des Objekts "Ellipse" legen Sie im Inspektorfenster fest. Der Wert wird in der Einheit Pixel eingegeben.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie unter "Horizontal" einen Wert zwischen 0 und 2500 ein.

### Radius vertikal

Den vertikalen Radius des Objekts "Ellipse" legen Sie im Inspektorfenster fest. Der Wert wird in der Einheit Pixel eingegeben.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie bei "Vertikal" einen Wert zwischen 0 und 2500 ein.

### Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

Objekt drehen (Seite 3055)

Objekt spiegeln (Seite 3057)

### Grafikanzeige

### Verwendung

Das Objekt "Grafikanzeige" dient der Anzeige von Grafiken.



### Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Grafik: Legt die Grafikdatei fest, die in dem Objekt angezeigt wird.
- Grafik anpassen: Legt die automatische Größenanpassung für Objekte mit Grafiken fest.
- Transparente Farbe: Legt fest, ob die transparente Farbe für die Grafik verwendet wird.

## Grafik einfügen

Im Objekt "Grafikanzeige" verwenden Sie die folgenden Grafikformate: \*.bmp, \*.tif, \*.png, \*.ico, \*.emf, \*.wmf, \*.gif, \*.jpg oder \*.jpeg. Sie können Grafiken auch als OLE-Objekte in der Grafikanzeige verwenden.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie die Grafik, die Sie einfügen wollen.  
Die Grafik wird rechts in der Vorschau angezeigt.
3. Um die Grafik in die Grafikanzeige einzufügen, klicken Sie auf "Übernehmen".

## Grafik anpassen

Im Inspektorfenster wird festgelegt, ob die in einer Grafikanzeige dargestellte Grafik sich in Runtime an die Größe der Grafikanzeige anpasst oder nicht.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie die gewünschte Größenanpassung für die Grafik.

## Transparente Farbe

Mit dieser Eigenschaft legen Sie fest, ob die Transparentfarbe für die anzuzeigende Grafik verwendet wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung".
2. Aktivieren Sie "Hintergrund > Transparent".
3. Wählen Sie eine transparente Farbe.

---

### Hinweis

Bei der Verwendung von Bitmaps in WinCC Bildern erfordert die Einstellung "Transparente Farbe" eine hohe Zeichenperformance bei der Darstellung auf Panels. Um eine bessere Darstellungsperformance zu erreichen, deaktivieren Sie die Einstellung "Transparent" in den Eigenschaften des jeweiligen Anzeigeobjekts. Diese Einschränkung gilt insbesondere bei der Verwendung von Bitmaps als Hintergrundbilder.

---

### Hinweis

#### Basic Panels

Die Eigenschaft "Transparent" steht für Basic Panels nicht zur Verfügung.

---

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

Externe Grafik in der Grafiksammlung hinterlegen (Seite 3063)

Bearbeitungsmöglichkeiten von Objekten (Seite 3043)

Objekte für Basic Panels (Seite 3127)

## Grafisches E/A-Feld

### Verwendung

Mit dem Objekt "Grafisches E/A-Feld" projektieren Sie eine Liste, die der Anzeige und der Auswahl von Grafikdateien dient.



### Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Modus: Legt das Verhalten des Objekts in Runtime fest.
- Typ Bildlaufleiste: Legt die grafische Darstellung der Bildlaufleiste fest.

---

#### Hinweis

##### Basic Panels

Die Bildlaufleiste steht für Basic Panels nicht zur Verfügung.

---

#### Hinweis

##### Protokolle

In Protokollen geben Grafische E/A-Felder ausschließlich Grafiken aus. Der Modus "Ausgabe" ist voreingestellt. Eigenschaften zum Konfigurieren der Auswahl von Grafiken stehen nicht zur Verfügung, z. B. "Bildlaufleiste".

---

### Modus

Das Verhalten des Objekts "Grafisches E/A-Feld" wird im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Typ > Modus" festgelegt.

| Modus             | Beschreibung   |
|-------------------|--|
| "Eingabe"         | Das Objekt "Grafisches E/A-Feld" wird nur zur Auswahl von Grafiken eingesetzt.   |
| "Eingabe/Ausgabe" | Das Objekt "Grafisches E/A-Feld" wird zur Auswahl und zur Anzeige von Grafiken eingesetzt.   |
| "Ausgabe"         | Das Objekt "Grafisches E/A-Feld" wird nur zur Anzeige von Grafiken eingesetzt.   |
| "Zwei Zustände"   | Das Objekt "Grafisches E/A-Feld" wird nur zur Anzeige von Grafiken eingesetzt und kann maximal zwei Zustände besitzen.<br>Sie verwenden keine Grafikliste, sondern fügen je eine Grafik für den Zustand "EIN" und "AUS" ein. |

## Grafik anpassen

Im Inspektorfenster wird festgelegt, ob die in einem Grafischen E/A-Feld dargestellte Grafik sich in Runtime an die Größe der Anzeige anpasst oder nicht.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Wählen Sie die gewünschte Größenanpassung für die Grafik.

## Typ Bildlaufleiste

Das Verhalten für die grafische Darstellung der Bildlaufleiste legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Gestaltung > Bildlaufleiste > Typ" fest.

| Typ                         | Beschreibung                                      |
|-----------------------------|---|
| "Dauerhaft"                 | Die Bildlaufleiste ist immer sichtbar.            |
| "Kein Rollbalken"           | Die Bildlaufleiste ist nicht sichtbar.            |
| "Sichtbar nach dem Klicken" | Die Bildlaufleiste wird durch Mausclick sichtbar. |

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

Symbolisches E/A-Feld (Seite 3153)

## Hilfeindikator

## Verwendung

Das Objekt "Hilfeindikator" ist für die Bediengeräte OP 73 und KP300 Basic verfügbar. Wenn zum angewählten Objekt ein Tooltip existiert, dann wird der Hilfeindikator in Runtime eingeblendet. Wenn für das geöffnete Bild ein Tooltip projiziert wurde, ist der Hilfeindikator immer sichtbar.



Das Objekt "Hilfeindikator" projizieren Sie ausschließlich im Globalen Bild.

## Darstellung

Im Inspektorfenster passen Sie folgende Eigenschaft an:

- Position: Legt die Position des Objekts "Hilfeindikator" fest.

## Position

Mit dieser Eigenschaft legen Sie die Position des Objekts "Hilfeindikator" fest.

1. Markieren Sie in der Vorlage das Objekt "Hilfeindikator".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
3. Geben Sie bei X und Y die Werte ein. Alternativ können Sie das markierte Objekt mithilfe der Pfeiltasten positionieren.

Wenn Sie in einem Bild an diese Position ein Bildobjekt projiziert haben, überdeckt der eingblendete Hilfeindikator das Bildobjekt. Der Hilfeindikator wird nur von auftretenden Systemmeldungen und Dialogen überdeckt.

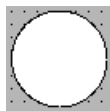
## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

## Kreis

## Verwendung

Das Objekt "Kreis" ist ein geschlossenes Objekt, das mit einer Farbe oder einem Muster gefüllt werden kann.



## Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radius: Legt die Größe des Kreises fest.

## Radius

Den Radius des Objekts "Kreis" wird im Inspektorfenster festgelegt. Der Wert wird in der Einheit Pixel eingegeben.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Radius" einen Wert zwischen 0 und 2500 ein.



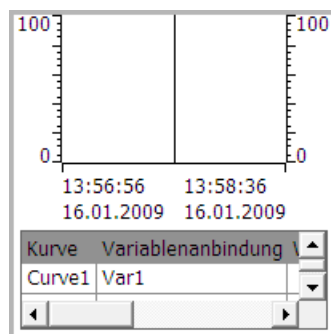
## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

## Kurvenanzeige

### Verwendung

Die Kurvenanzeige dient zur grafischen Darstellung von Variablenwerten aus dem aktuellen Prozess oder aus einem Archiv in Form von Kurven.



### Darstellung

Im Inspektionsfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie folgende Eigenschaften an:

- Wertetabelle, Lineal, Raster anzeigen: Legt fest, ob zusätzlich zum Koordinatensystem eine Wertetabelle, ein Lineal oder Raster zur leichteren Ablesbarkeit eingeblendet wird.
- Symbolleisten: Legt die Anzeige der Bedienelemente fest.

### Wertetabelle, Lineal und Raster anzeigen

Zur leichteren Ablesbarkeit zeigen Sie in Runtime eine Wertetabelle, ein Lineal und ein Raster an.

1. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Gestaltung > Lineal anzeigen".
2. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Tabelle > Tabelle anzeigen".
3. Aktivieren Sie unter "Eigenschaften > Eigenschaften > Tabelle > Raster anzeigen".










## Symbolleisten

Die Darstellung der Bedienelemente legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Symbolleiste" fest.

### Hinweis

#### Basic Panels

Da für Basic Panels keine Archivierung möglich ist, stehen die Bedienelemente nicht zur Verfügung.

| Schaltfläche<br>Symbolleiste  | Kurzbeschreibung   | Beschreibung   |
|---|--------------------|--|
|    | "Gehe zu Start"    | Blättert zum Beginn der Kurvenaufzeichnung zurück. Die Startwerte werden angezeigt, mit denen die Kurvenaufzeichnung begonnen hat. |
|    | "Vergrößern"       | Verkleinert den dargestellten Zeitabschnitt  |
|    | "Verkleinern"      | Vergrößert den dargestellten Zeitabschnitt.  |
|  | "Lineal rückwärts" | Bewegt das Lineal zurück.  |
|  | "Lineal vorwärts"  | Bewegt das Lineal vorwärts.  |
|  | "Rückwärts"        | Blättert um eine Anzeigenbreite zurück.  |
|  | "Vorwärts"         | Blättert um eine Anzeigenbreite vorwärts.  |
|  | "Lineal"           | Blendet das Lineal ein oder aus. Das Lineal zeigt den zu einem X-Wert gehörenden Y-Wert an.  |
|  | "Start/Stopp "     | Hält die Kurvenaufzeichnung an oder setzt die Kurvenaufzeichnung fort.   |

## Verhalten bei Projektierung

### Anzeige von Spaltenüberschriften

Die Darstellung der Tabelle in der Kurvenanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften in der Tabelle abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

### Konsistenzprüfung

Wenn bei einer Konsistenzprüfung an Kurvenanzeigen Warnungen oder Fehler im Ausgabefenster angezeigt werden, wird nicht in allen Fällen mit dem Befehl "Gehe zu Fehler/ Variable" an die exakte Fehlerstelle gesprungen. Eventuell wird nur die Kurvenanzeige als Fehlerursache angezeigt.

## Kurven hinzufügen, konfigurieren und entfernen

Die Kurven der Kurvenanzeige verwalten Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Kurve". Sie können Kurven zwischen verschiedenen Kurvenanzeigen kopieren.

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 3201)

Touch- und Tasten-Bedienung (Seite 3988)

Überblick (Seite 3987)

## Linie

### Verwendung

Das Objekt "Linie" ist ein offenes Objekt. Länge und Neigung der Linie bestimmen Sie durch Höhe und Breite des objektumfassenden Rechtecks.



### Darstellung

Sie ändern im Inspektorfenster die Einstellungen zu Position, Geometrie, Stil und Farbe des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Liniensstil
- Linienanfang und Linienende

### Liniensstil

Die Darstellung der Linie legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung" fest. Wenn Sie z. B. "Massiv" wählen, wird die Linie durchgezogen dargestellt.

---

#### Hinweis

Die verfügbaren Liniensstile sind abhängig vom gewählten Bediengerät.

---

### Linienanfang und Linienende

Den Anfangspunkt und den Endpunkt der Linie legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Gestaltung > Linienenden" fest.

Als Anfangs- bzw. Endpunkte verwenden Sie z. B. Pfeilspitzen. Welche Anfangs- und Endpunkte Ihnen zur Verfügung stehen, ist geräteabhängig.

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

## Meldeanzeige

## Verwendung

Meldungen werden am Bediengerät in der Meldeanzeige oder im Meldefenster dargestellt.

Das folgende Bild zeigt eine einfache Meldeanzeige:



## Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts.

---

### Hinweis

Die Auswahl an Schriftarten ist abhängig davon, welche Schriftarten Sie in den Runtime-Einstellungen unter "Sprache & Schriftart" projiziert haben und welche Schriftarten Ihr Bediengerät unterstützt.

---

Insbesondere passen Sie die folgenden Eigenschaften an:

- Bedienelemente: Legt die Bedienelemente der Meldeanzeige fest.
- Meldeklassen: Legt fest, welche Meldeklassen in der Meldeanzeige angezeigt werden.
- Spalten: Legt die in Runtime darzustellenden Spalten fest.

---




**Hinweis**

Wenn Sie verschiedene Meldeklassen ausgeben lassen, wird in Runtime zunächst nach Meldeklassen und danach nach Eintritt der Meldung sortiert.

---

**Bedienelemente**

Die Bedienelemente, mit denen die Meldeanzeige in Runtime bedient werden kann, werden im Inspektorfenster unter "Anzeige > Einstellungen" festgelegt. Die folgende Tabelle zeigt die Bedienelemente der Meldeanzeige und ihre Funktion:

| Schaltfläche    |   | Funktion  |
|-----------------|---|---|
| "Hilfetext"     |    | Zeigt Hilfetext für eine Meldung an.  |
| "Quittieren"    |   | Quittiert eine Meldung.   |
| "Loop-In-Alarm" |  | Wenn ein Bildwechsel projiziert ist, wechselt die Anzeige zu einem Bild, das Informationen zur Meldung enthält. |

**Meldeklassen auswählen**

1. Klicken Sie im Inspektorfenster auf "Eigenschaften".
2. Aktivieren Sie unter "Meldeklassen" die Meldeklassen, die in der Meldeanzeige in Runtime dargestellt werden sollen.

**Spalten festlegen**

Die Spalten, die in Runtime in der Meldeanzeige angezeigt werden, legen Sie im Inspektorfenster fest.

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Spalten".
2. Aktivieren Sie unter "Spalten" die Spalten, die in Runtime dargestellt werden sollen.

**Anzeige von Spaltenüberschriften**

Die Darstellung der Meldeanzeige ist abhängig von der Einstellung der Anzeige in der Systemsteuerung. Je nach Einstellung werden die Spaltenüberschriften abgeschnitten. Die Einstellung finden Sie in der Systemsteuerung unter "Anzeige > Registerkarte Darstellung". Zur korrekten Anzeige der Spaltenüberschriften stellen Sie bei "Fenster und Schaltflächen" die Darstellung "Windows - klassisch" ein.

Dieses Verhalten tritt nur beim Projektieren auf. In Runtime werden die Spaltenüberschriften korrekt angezeigt.

---

#### **Hinweis**

Im Engineering System können Sie im Inspektorfenster in der Gruppe "Animationen", z. B. die Sichtbarkeit eines Objekts dynamisieren. Das Objekt "Einfache Meldeanzeige" unterstützt in Runtime keine Animation. Wenn Sie eine Animation projiziert haben und z. B. eine Konsistenzprüfung des Projekts durchführen, dann wird im Ausgabe-Fenster eine Fehlermeldung ausgegeben.

---

#### **Siehe auch**

Geräteabhängigkeit der Objekte (Seite 3127)

Meldefenster (Seite 3144)

Meldeindikator (Seite 3147)

Meldeanzeige projektieren (Seite 3229)

#### **Meldefenster**

#### **Verwendung**

Meldungen werden am Bediengerät in der Meldeanzeige oder im Meldefenster dargestellt. Aussehen und Bedienung des Meldefensters sind ähnlich der Meldeanzeige. Das Meldefenster hat die folgenden gleichen Ausprägungen wie die Meldeanzeige:

- Einfaches Meldefenster
- Erweitertes Meldefenster
- Meldezeile

---

#### **Hinweis**

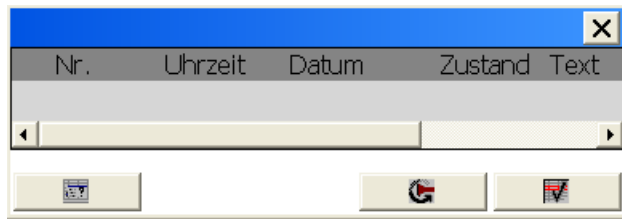
##### **Basic Panels**

Für Basic Panels steht ausschließlich das einfache Meldefenster zur Verfügung.

---

Das Meldefenster projektieren Sie im Editor "Globales Bild".

Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört. Je nach Projektierung wird es erst wieder geschlossen, wenn die Meldung quittiert wurde.



**Hinweis**

Im Engineering System dynamisieren Sie im Inspektorfenster "Eigenschaften > Animationen", z. B. die Sichtbarkeit eines Objekts. Das Objekt "Einfaches Meldefenster" unterstützt in Runtime keine Animation. Wenn Sie eine Animation projiziert haben und z. B. eine Konsistenzprüfung des Projekts durchführen, dann wird im Ausgabe-Fenster eine Fehlermeldung ausgegeben.

**Darstellung**

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Sie projizieren das Meldefenster wie die Meldeanzeige. Abweichend davon passen Sie die folgenden Eigenschaften an:




- Gebundene Meldefenster: Legt fest, dass das Meldefenster beim Bildwechsel den Fokus behält.
- Fenster: Sie legen Bedienung und Verhalten des Meldefensters in Runtime fest.

**Hinweis**

Wenn Sie verschiedene Meldeklassen ausgeben lassen, wird in Runtime zunächst nach Meldeklassen und danach nach Eintritt der Meldung sortiert.

**Bedienelemente**

Die Bedienelemente, mit denen die Meldeanzeige in Runtime bedient werden kann, legen Sie im Inspektorfenster "Eigenschaften > Anzeige > Einstellungen" fest. Die folgende Tabelle zeigt die Bedienelemente des Meldefensters und ihre Funktion:

| Schaltfläche    |   | Funktion  |
|-----------------|---|---|
| "Tooltip"       |  | Zeigt Tooltip für eine Meldung an.  |
| "Quittieren"    |  | Quittiert eine Meldung.   |
| "Loop-In-Alarm" |  | Wenn ein Bildwechsel projiziert ist, wechselt die Anzeige zu einem Bild, das Informationen zur Meldung enthält. |

## Zugriffsschutz in Runtime

Im Inspektorfenster der Meldeanzeige projizieren Sie "Eigenschaften > Eigenschaften > Sicherheit" den Zugriffsschutz. Wenn ein angemeldeter Benutzer die erforderliche Berechtigung besitzt, kann er mit Hilfe der Bedienelemente in der Meldeanzeige Meldungen quittieren und editieren. Wenn der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt oder kein Benutzer angemeldet ist, wird beim Betätigen der Schaltflächen "Quittieren", "Editieren" oder durch Doppelklick auf eine Meldezeile der Anmelde-dialog eingeblendet.

---

### Hinweis

#### Basic Panels

Der Zugriffsschutz steht für Basic Panels nicht zur Verfügung.

---

## Fokus für das Meldefenster aktivieren

Damit auch beim Bildwechsel das Meldefenster nicht den Fokus verliert, wählen Sie folgende Option:

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Modus".
2. Aktivieren Sie "Beschriftung".

## Fenster

Das Verhalten des Meldefensters wird im Inspektorfenster "Eigenschaften > Eigenschaften > Modus > Fenster" festgelegt. Die folgende Tabelle zeigt die möglichen Eigenschaften:

| Option               | Funktion  |
|----------------------|---|
| Automatisch anzeigen | Meldefenster wird automatisch eingeblendet, z. B. beim Auftreten einer Systemmeldung.   |
| Schließbar           | Nach Ablauf einer bestimmten Zeit wird das Fenster wieder geschlossen. Die Anzeigedauer legen Sie in den Melde-einstellungen fest.  |
| Gebunden             | Meldefenster ist an eine Bestätigung gebunden, z. B: Meldung muss quittiert werden. Wenn das gebundene Meldefenster den Fokus besitzt, dann lassen sich die Schaltflächen im dahinterliegenden Bild nicht bedienen. Die an eine Funktionstaste projizierten Funktionen werden ausgeführt. |
| Größe veränderbar    | In Runtime kann die Größe des Meldefensters geändert werden.  |

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

Meldeanzeige (Seite 3142)

Meldefenster projizieren (Seite 3231)



## Meldeindikator

### Verwendung

Der Meldeindikator ist ein grafisches Symbol, das je nach Projektierung auf anstehende oder zu quittierende Meldungen hinweist. Den Meldeindikator projektieren Sie im Editor "Globales Bild". Das folgende Bild zeigt einen Meldeindikator:



### Meldeindikator OP 73

Für das Bediengerät OP 73 steht ein "einfacher" Meldeindikator zur Verfügung. Das folgende Bild zeigt den Meldeindikator für das Bediengeräte OP 73:



Der "einfache" Meldeindikator weist auf zu quittierende Meldungen oder auf bereits quittierte und noch nicht gegangene Meldungen hin. Beim "einfachen" Meldeindikator lässt sich nur die Position festlegen. Der Meldeindikator wird auf dem Gerät an der gewählten Position eingeblendet. Haben Sie in einem Bild an diese Position ein Bildobjekt projektiert, dann überdeckt der eingeblendete Meldeindikator das Bildobjekt. Der Meldeindikator wird von Systemdialogen, z. B. Anmeldedialog, Hilfedialog und von Meldefenstern überdeckt.

### Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Meldeklassen: Legt die Meldeklassen fest, bei denen der Meldeindikator eingeblendet wird.
- Bedienung in Runtime: Legt fest, bei welcher Bedienaktion in Runtime das Meldefenster angezeigt wird.

### Meldeklassen

Welche Meldeklassen mit Meldeindikator eingeblendet werden, legen Sie im Inspektorfenster "Allgemein > Meldeklassen" fest. Meldeklassen sind z. B. "Warnings" oder "Errors".

### Bedienung in Runtime festlegen

1. Markieren Sie den Meldeindikator im Bild.
2. Klicken Sie im Inspektorfenster "Ereignisse > Klicken" oder "Klicken bei Blinken".
3. Die "Funktionsliste" wird geöffnet. Klicken Sie auf die erste Zeile der Funktionsliste. Die Liste der im Projekt verfügbaren Systemfunktionen und Skripte wird geöffnet.

4. Wählen Sie unter "Meldungen" die Systemfunktion "ZeigeMeldefenster".
5. Wählen Sie bei "Objektnamen" den Namen des Meldefensters aus der Auswahlliste. Legen Sie unter "Darstellung" fest, ob das Meldefenster eingeblendet, ausgeblendet oder zwischen beiden Zuständen hin und hergeschaltet wird.

### Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

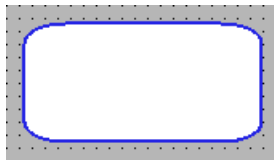
Meldeanzeige (Seite 3142)

Meldeindikator projektieren (Seite 3232)

### Rechteck

### Verwendung

Das Objekt "Rechteck" ist ein geschlossenes Objekt, das Sie mit einer Farbe füllen können.



### Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Radius Ecken: Legt den horizontalen bzw. vertikalen Abstand zwischen der Ecke des objektumfassenden Rechtecks und dem Startpunkt einer Eckrundung fest.

### Radius Ecken

Die Ecken des Objekts "Rechteck" können beliebig abgerundet werden. Wenn die Eigenschaften "X" und "Y" auf den Wert 100 % gesetzt werden, wird das Rechteck als Ellipse dargestellt. Sobald eine der beiden Eigenschaften den Wert 0 % hat, wird ein normales Rechteck ohne Eckrundung dargestellt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Geben Sie im Bereich "Radius Ecken" einen Wert für "X" ein.  
Der Werteingabe ist der prozentuale Anteil an der halben Breite des Rechtecks.
3. Geben Sie im Bereich "Radius Ecken" einen Wert für "Y" ein.  
Der Werteingabe ist der prozentuale Anteil an der halben Höhe des Rechtecks.

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

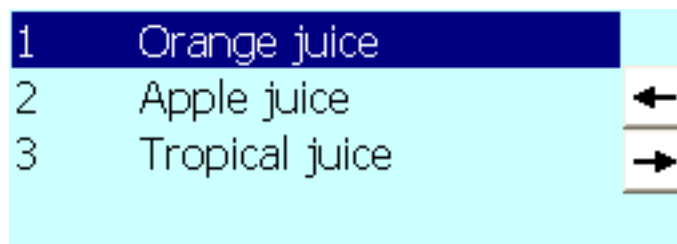
Objekt drehen (Seite 3055)

Objekt spiegeln (Seite 3057)

## Rezepturanzeige

### Verwendung

Das Objekt "Rezepturanzeige" dient der Anzeige und Änderung von Rezepturen.



### Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Symbolleiste: Legt die Menüeinträge der Rezepturanzeige fest.

### Symbolleiste

Die Menüeinträge, mit denen die Rezepturanzeige in Runtime bedient wird, projizieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Symbolleiste".

| Menüeintrag            | Beschreibung  |
|------------------------|---|
| "Tooltip"              | Ruft den projizierten Tooltip zur ausgewählten Rezeptur auf.  |
| "Datensatz hinzufügen" | Legt einen neuen Rezepturdatsatz in der Rezeptur an.          |
| "Datensatz löschen"    | Löscht den markierten Datensatz.                              |
| "Speichern"            | Speichert den geänderten Datensatz unter dem aktuellen Namen. |
| "Speichern unter"      | Speichert den geänderten Datensatz unter einem neuen Namen.   |
| "In PLC schreiben"     | Überträgt den aktuellen Wert in die Steuerung.                |
| "Von PLC lesen"        | Liest den aktuellen Wert aus der Steuerung.                   |
| "Umbenennen"           | Legt fest, dass die Schaltfläche "Umbenennen" angezeigt wird. |
| "Vorwärts"             | Legt fest, dass die Schaltflächen des Menüs sichtbar sind.    |
| "Zurück"               | Legt fest, die Schaltfläche "Zurück" angezeigt wird.          |

### Siehe auch

- Geräteabhängigkeit der Objekte (Seite 3127)
- Einfache Rezepturanzeige (Seite 3282)
- Anzeige der Rezepturen (Seite 3276)
- Einfache Rezepturanzeige projektieren (Seite 3293)
- Beschreibung Einfache Rezepturanzeige (Seite 3296)

### Schalter

#### Verwendung

Mit dem Objekt "Schalter" projektieren Sie einen Schalter, mit dem Sie in Runtime zwischen zwei vordefinierten Zuständen umschalten. Den aktuellen Zustand des Objekts "Schalter" visualisieren Sie durch eine Beschriftung oder durch eine Grafik.

Das folgende Bild zeigt einen Schalter vom Typ "Schalter".



#### Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgende Eigenschaft an:

- Typ: Legt die grafische Darstellung des Objekts fest.

#### Typ

Die Darstellung des Schalters wird im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Typ" festgelegt.

| Typ                   | Beschreibung  |
|-----------------------|---|
| "Schalter"            | Die beiden Zustände des Schalters werden in Form eines Schalters dargestellt. Die Stellung des Schalters zeigt den aktuellen Zustand an. Durch Verschieben des Schalters wird in Runtime umgeschaltet.<br>Bei diesem Typ legen Sie unter "Schaltrichtung" die Bewegungsrichtung des Schalters fest. |
| "Schalter mit Text"   | Der Schalter wird als Schaltfläche dargestellt. Der aktuelle Zustand wird durch eine Beschriftung visualisiert. In Runtime wird durch Klicken auf die Schaltfläche umgeschaltet.  |
| "Schalter mit Grafik" | Der Schalter wird als Schaltfläche dargestellt. Der aktuelle Zustand wird durch eine Grafik visualisiert. In Runtime wird durch Klicken auf die Schaltfläche umgeschaltet.  |

---

**Hinweis**

**Basic Panels**

Der Typ "Schalter" steht bei Basic Panels nicht zur Verfügung.

---

**Siehe auch**

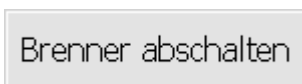
Geräteabhängigkeit der Objekte (Seite 3127)

Übersicht über die Objekte (Seite 3040)

**Schaltfläche**

**Verwendung**

Mit dem Objekt "Schaltfläche" projizieren Sie ein Objekt, mit dem der Bediener in Runtime eine beliebig projektierbare Funktion ausführt.



**Darstellung**

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Modus: Legt die grafische Darstellung des Objekts fest.
- Text / Grafik: Legt fest, ob die grafische Darstellung statisch oder dynamisch ist.
- Hotkey festlegen: Legt eine Taste oder Tastenkombination fest, mit der der Bediener die Schaltfläche bedient.

---

**Hinweis**

Die Festlegung eines Hotkey ist nur bei Bediengeräten mit Tasten möglich.

---

## Modus

Die Darstellung der Schaltfläche wird im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Modus" festgelegt.

| Modus        | Beschreibung   |
|--------------|--|
| "Unsichtbar" | Die Schaltfläche ist in Runtime nicht sichtbar.  |
| "Text"       | Die Schaltfläche wird mit Text angezeigt. Der Text dient dazu, die Funktion der Schaltfläche ablesbar zu machen.     |
| "Grafik"     | Die Schaltfläche wird mit einer Grafik angezeigt. Die Grafik dient dazu, die Funktion der Schaltfläche darzustellen. |

Abhängig vom Gerät stehen Ihnen weitere Optionen zur Verfügung.

## Text / Grafik

Abhängig von der Eigenschaft "Modus" können Sie die Anzeige statisch oder dynamisch festlegen. Die Anzeige wird im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Text" oder "Grafik" festgelegt.

Vom Typ "Grafik" stehen Ihnen z. B. folgende Optionen zur Auswahl.

| Typ      | Option        | Beschreibung   |
|----------|---------------|--|
| "Grafik" | "Grafik"      | Bei "Grafik AUS" legen Sie eine Grafik fest, die in der Schaltfläche für den Zustand "AUS" angezeigt wird.<br>Wenn Sie "Grafik EIN" aktivieren, können Sie eine Grafik für den Zustand "EIN" eingeben. |
|          | "Grafikliste" | Die Grafik in der Schaltfläche ist abhängig vom Zustand. Je nach Zustand wird der entsprechende Eintrag aus der Grafik angezeigt.  |

## Hotkey festlegen

Im Inspektorfenster wird eine Taste oder Tastenkombination festgelegt, mit der der Bediener die Schaltfläche in Runtime bedienen kann.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie im Bereich "Hotkey" in der Auswahlliste eine Taste oder eine Tastenkombination.

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 3107)

Beispiel: Schaltfläche mit Anmeldedialog projektieren (Seite 3329)

Beispiel: Schaltfläche mit Zugriffsschutz projektieren (Seite 3333)

## Symbolisches E/A-Feld

### Verwendung

Mit dem Objekt "Symbolisches E/A-Feld" projektieren Sie eine Auswahlliste, die der Eingabe und Ausgabe von Texten in Runtime dient.



### Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Modus: Legt das Verhalten des Objekts in Runtime fest.
- Textliste: Legt die Textliste fest, die mit dem Objekt verknüpft wird.
- Schaltfläche für Auswahlliste: Legt fest, dass das Objekt eine Schaltfläche zum Öffnen der Auswahlliste besitzt.

---

#### Hinweis

#### Protokolle

In Protokollen geben symbolische E/A-Felder ausschließlich Daten aus. Der Modus "Ausgabe" ist voreingestellt. Eigenschaften zum Konfigurieren der Auswahl von Grafiken stehen nicht zur Verfügung, z. B. "Schaltfläche für Auswahlliste".

---

### Modus

Das Verhalten des Symbolischen E/A-Feldes legen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Typ" fest.

| Modus             | Beschreibung   |
|-------------------|--|
| "Ausgabe"         | Das Symbolische E/A-Feld dient zur Ausgabe von Werten.   |
| "Eingabe"         | Das Symbolische E/A-Feld dient zur Eingabe von Werten.   |
| "Eingabe/Ausgabe" | Das Symbolische E/A-Feld dient zur Eingabe und Ausgabe von Werten.   |
| "Zwei Zustände"   | Das Symbolische E/A-Feld wird nur zur Ausgabe von Werten eingesetzt und besitzt maximal zwei Zustände. Das Feld schaltet zwischen zwei vordefinierten Texten um. Damit werden z. B. die beiden Zustände eines Ventils, geschlossen oder offen, visualisiert. |

---

#### Hinweis

Welches Verhalten für das Symbolische E/A-Feld möglich ist, hängt von der Runtime ab.

---

## Textliste

Im Inspektorfenster legen Sie fest, welche Textliste mit dem Symbolischen E/A-Feld verknüpft wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Öffnen Sie unter "Inhalt" die Auswahlliste für "Textliste".
3. Wählen Sie eine Textliste aus.

## Schaltfläche für Auswahlliste

Mit der Eigenschaft "Schaltfläche für Auswahlliste" wird eine Schaltfläche zum Öffnen der Auswahlliste angezeigt.

1. Klicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Verhalten > Schaltfläche für Auswahlliste".

---

### Hinweis

#### Basic Panels

Die Option "Schaltfläche für Auswahlliste" steht für Basic Panels nicht zur Verfügung.

---

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

Grafisches E/A-Feld (Seite 3136)

## System-Diagnoseanzeige

### Einleitung

Die System-Diagnoseanzeige bietet Ihnen einen Überblick über alle verfügbaren Geräte in Ihrer Anlage. Sie navigieren direkt zur Fehlerursache und zum zugehörigen Gerät. Sie haben Zugriff zu allen diagnosefähigen Geräten, die Sie im Editor "Geräte & Netze" konfiguriert haben.

### Verwendung

Durch die System-Diagnoseanzeige erreichen Sie den höchstmöglichen Detaillierungsgrad der Diagnosedaten. Eine präzise Diagnose ist möglich, da alle verfügbaren Daten angezeigt werden. Sie haben den Systemzustand der gesamten Anlage auf einen Blick.

Basic Panels unterstützen nur die "Einfache System-Diagnoseanzeige".





|   |                  |
|---|------------------|
| 1 | HMI_connection   |
| 2 | HMI_connection_1 |
| 3 | HMI_connection_3 |
| 4 | HMI_connection_4 |

Drei verschiedene Ansichten stehen Ihnen in der System-Diagnoseanzeige zur Verfügung.

- Geräteansicht
- Diagnosepuffer-Ansicht
- Detailansicht

### Geräteansicht

Die Geräteansicht der System-Diagnoseanzeige zeigt in tabellarischer Sicht alle verfügbaren Verbindungen. Durch einen Doppelklick auf eine Verbindung öffnet die Detailansicht. Die Geräteansicht wird nur angezeigt, wenn mehr als eine Verbindung im Editor "Geräte & Netze" angelegt wurde.

### Diagnosepuffer-Ansicht

In der Diagnosepuffer-Ansicht werden die aktuellen Daten aus dem Diagnosepuffer angezeigt.

### Detailansicht

Die Detailansicht zeigt ausführliche Informationen über die selektierte Verbindung Fehlertexte können Sie in der Detailansicht nicht sortieren. Die Detailansicht ist nur verfügbar, wenn eine integrierte Verbindung zu einer S7 1200 oder S7 1500.

## Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zur Position, Geometrie, Stil, Farbe und Schriftart des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Zeilen pro Eintrag: Legt die Anzahl der Zeilen fest, die für einen Eintrag angezeigt werden.

## System-Diagnoseanzeige projektieren

1. Ziehen Sie per Drag&Drop die System-Diagnoseanzeige aus dem Werkzeugfenster.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Darstellung".
3. Wählen Sie für "Zeilen pro Eintrag" z. B. 5.
4. Wählen Sie "Eigenschaften > Eigenschaften > Sicherheit" eine Berechtigung für die Bedienung aus.

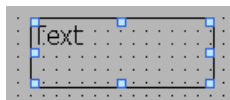
## Siehe auch

Ansichten in der Systemdiagnose (Seite 3268)

## Textfeld

## Verwendung

Das Objekt "Textfeld" ist ein geschlossenes Objekt, das Sie mit einer Farbe füllen können.



## Darstellung

Im Inspektorfenster ändern Sie die Einstellungen zu Position, Geometrie, Stil, Farbe und Schriftarten des Objekts. Insbesondere passen Sie die folgenden Eigenschaften an:

- Text: Legt den Text für das Textfeld fest.
- Größe des Textfelds: Legt fest, ob die Größe des Objekts an den Platzbedarf des größten Listeneintrags angepasst wird.

## Text

Der Text für das Textfeld wird im Inspektorfenster festgelegt.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie einen Text ein.  
Für mehrzeiligen Text können Sie einen Zeilenumbruch durch Drücken der Tastenkombination <Umschalttaste + Enter> setzen.

## Größe des Textfelds

Im Inspektorfenster wird festgelegt, ob die Größe des Objekts an den Platzbedarf des größten Listeneintrags angepasst wird.

1. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Darstellung".
2. Aktivieren Sie "Größenanpassung > Größe an Inhalt anpassen".

## Siehe auch

Geräteabhängigkeit der Objekte (Seite 3127)

## 10.1.8 Bildnavigation projektieren

### 10.1.8.1 Grundlagen zur Bildnavigation

#### Navigationsarten für den Bildwechsel

Für einen Fertigungsprozess aus mehreren Teilprozessen projektieren Sie mehrere Bilder. Damit der Bediener in Runtime von einem Bild zum Nächsten schalten kann, haben Sie folgende Möglichkeiten:

- Bildwechsel an Schaltflächen projektieren
- Bildwechsel an lokale Funktionstaste projektieren

## Vorgehen

Bevor Sie einen Bildwechsel erstellen, legen Sie die Anlagenstruktur fest und leiten Sie daraus die Bildwechsel ab, die Sie projektieren wollen.

Das Startbild legen Sie unter "Runtime-Einstellungen > Allgemein > Startbild" an.

## Siehe auch

Schaltfläche mit Bildwechsel belegen (Seite 3158)

### 10.1.8.2 Schaltfläche mit Bildwechsel belegen

#### Einleitung

Um im laufenden Betrieb am Bediengerät zwischen den Bildern umzuschalten, projizieren Sie im Bild eine Schaltfläche.

---

#### Hinweis

Wenn Sie im Inspektorfenster eines Bildes die Animation "Sichtbarkeit" auf "ausgeblendet" eingestellt haben, kann dieses Bild in Runtime nicht aufgerufen werden.

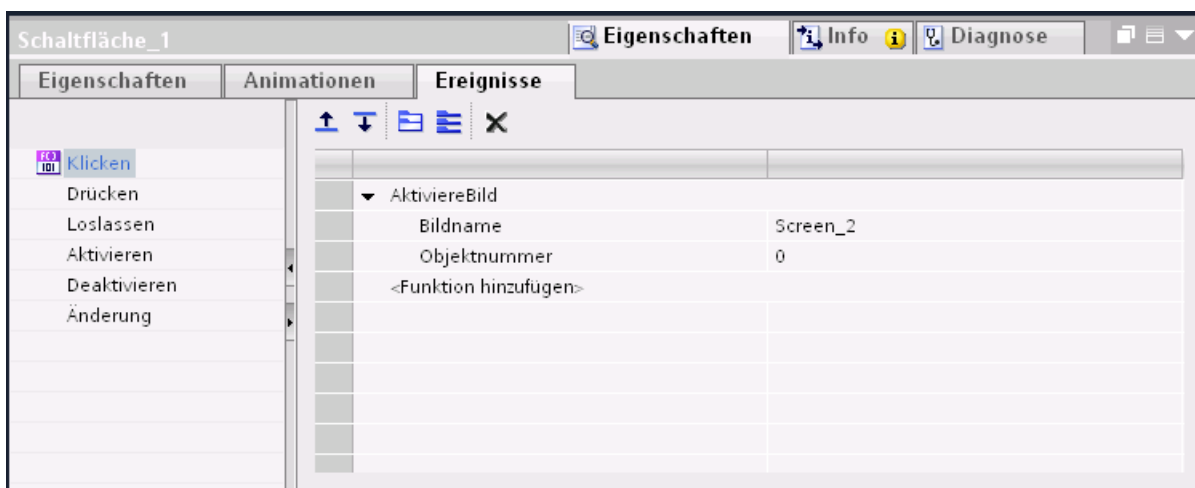
---

#### Voraussetzungen

- Ein Projekt ist angelegt.
- Das Bild "Screen\_2" ist angelegt.
- Das Bild "Screen\_1" ist angelegt.

#### Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Screen\_1". Das Bild wird im Arbeitsbereich angezeigt.
2. Ziehen Sie per Drag&Drop aus dem Projektnavigation "Screen\_2" in das geöffnete Bild. Eine Schaltfläche, die mit dem Namen "Screen\_1" wird eingefügt.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken". In der "Funktionsliste" wird die Systemfunktion "AktiviereBild" angezeigt.



4. Geben Sie bei Bedarf als "Objektnummer" die Tab-Reihenfolge-Nummer des Objekts ein, das nach dem Bildwechsel den Fokus erhalten soll. Sie können auch eine Variable angeben, welche die Objektnummer enthält.

## Alternatives Vorgehen

1. Ziehen Sie per Drag&Drop eine Schaltfläche aus der Task Card "Werkzeuge" in das Bild "Screen2".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken".
3. Wählen Sie die Systemfunktion "AktiviereBild".
4. Wählen Sie für "Bildnummer" das Bild "Screen\_2".

## Ergebnis

In Runtime wechselt der Bediener mit der Schaltfläche zum Bild "Screen\_1". Wenn Sie eine Objektnummer angegeben haben, erhält das Objekt mit dieser Objektnummer nach dem Bildwechsel den Fokus.

## Siehe auch

Grundlagen zur Bildnavigation (Seite 3157)

### 10.1.8.3 Funktionstaste mit einem Bildwechsel belegen

## Einleitung

Um im laufenden Betrieb am Bediengerät zwischen den Bildern umzuschalten, projizieren Sie im Bild eine Funktionstaste zum Bildwechsel.

---

### Hinweis

Wenn Sie im Inspektorfenster eines Bildes die Animation "Sichtbarkeit" auf "ausgeblendet" eingestellt haben, kann dieses Bild in Runtime nicht aufgerufen werden.

---

## Voraussetzungen

- Ein Projekt ist angelegt.
- Das Bild "Screen\_2" ist angelegt.
- Das Bild "Screen\_1" ist angelegt.

## Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Screen\_1". Das Bild wird im Arbeitsbereich angezeigt.
2. Ziehen Sie aus dem Projektnavigation "Screen\_2" auf eine Funktionstaste z. B. "F2". Die projizierte Funktionstaste zeigt ein gelbes Dreieck an.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Taste drücken". Die Systemfunktion wird "AktiviereBild" angezeigt.

## Ergebnis

In Runtime wechselt der Bediener mit der Funktionstaste "F2" zum angegebenen Bild "Screen\_2".

## 10.2 Mit Variablen arbeiten

### 10.2.1 Grundlagen

#### 10.2.1.1 Grundlagen zu Variablen

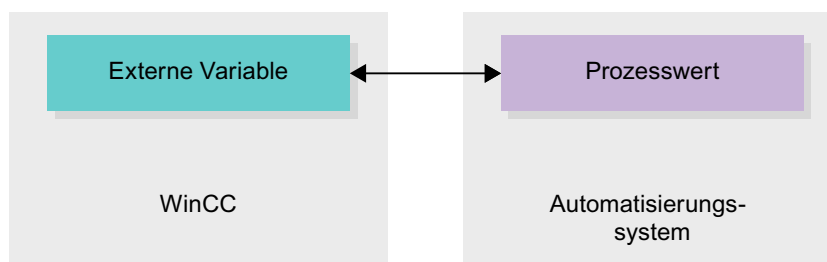
## Einleitung

Prozesswerte werden in Runtime durch Variablen weitergegeben. Prozesswerte sind Daten, die im Speicher eines der angeschlossenen Automatisierungssysteme abgelegt werden. Sie stellen den Zustand einer Anlage dar, z. B. als Temperaturen, Füllstände oder Schaltzustände. Für die Verarbeitung der Prozesswerte in WinCC definieren Sie externe Variablen.

WinCC arbeitet mit zwei Arten von Variablen:

- Externe Variablen
- Interne Variablen

Die externen Variablen sind das Bindeglied zwischen WinCC und den Automatisierungssystemen. Die Werte von externen Variablen entsprechen den Prozesswerten aus dem Speicher eines Automatisierungssystems. Der Wert einer externen Variable wird ermittelt, indem der Prozesswert aus dem Speicher des Automatisierungssystems ausgelesen wird. Umgekehrt können Sie einen Prozesswert auch in den Speicher des Automatisierungssystems zurückschreiben.



Interne Variablen besitzen keine Prozessanbindung und transportieren nur Werte innerhalb von WinCC. Die Variablenwerte stehen nur so lange zur Verfügung wie Runtime läuft.

## Variablen in WinCC

Bei externen Variablen wird in den Eigenschaften der jeweiligen Variable festgelegt, über welche Verbindung WinCC mit dem Automatisierungssystem kommuniziert und wie der Datenaustausch erfolgt.

Variablen ohne Wertversorgung durch den Prozess, die so genannten internen Variablen, werden nicht mit dem Automatisierungssystem verbunden. In der Eigenschaft "Verbindung" der Variable erkennen Sie dies am Eintrag "Interne Variable".

Zur besseren Übersicht legen Sie die Variablen in verschiedenen Variablen Tabellen an. In der Projektnavigation greifen Sie dann im Knoten "HMI-Variablen" direkt auf die einzelnen Variablen Tabellen zu. Mithilfe der Tabelle "Zeige alle Variablen" lassen Sie sich die Variablen aus allen Variablen Tabellen anzeigen.

## Siehe auch

Übersicht über HMI-Variablen Tabellen (Seite 3161)

Interne Variablen (Seite 3166)

Externe Variablen (Seite 3162)

Adressierung externer Variablen (Seite 3164)

Externe Variable anlegen (Seite 3167)

Grundlagen zu Arrays (Seite 3196)

Grundlagen zu Zyklen (Seite 3200)

### 10.2.1.2 Übersicht über HMI-Variablen Tabellen

#### Einleitung

HMI-Variablen Tabellen enthalten die Definitionen der geräteweit gültigen HMI-Variablen. Für jedes im Projekt angelegte HMI-Gerät wird automatisch eine Variablen Tabelle angelegt.

In der Projektnavigation gibt es für jedes HMI-Gerät einen Ordner "HMI-Variablen". Darin können folgende Tabellen enthalten sein:

- Standard-Variablen Tabelle
- Anwenderdefinierte Variablen Tabellen
- Alle Variablen

In einer HMI-Variablen Tabelle stehen Ihnen zusätzlich folgende Tabellen zur Verfügung:

- Bitmeldungen
- Analogmeldungen

Mithilfe dieser Tabellen projektieren Sie Meldungen für die aktuell selektierte HMI-Variablen.

In der Projektnavigation können Sie im Ordner HMI-Variablen weitere Variablen Tabellen anlegen, um Variablen und Konstanten darin zu sortieren und zu gruppieren. Die Variablen können Sie per Drag&Drop oder mithilfe des Felds "Variablen Tabelle" in eine andere Variablen Tabelle verschieben. Das Feld "Variablen Tabelle aktivieren Sie über das Kontextmenü der Spaltenüberschriften.

## Standard-Variablentabelle

Für jedes HMI-Gerät des Projekts gibt es eine Standard-Variablentabelle. Sie kann nicht gelöscht, umbenannt oder verschoben werden. Die Standard-Variablentabelle enthält HMI-Variablen und abhängig vom HMI-Gerät auch Systemvariablen. Sie können alle HMI-Variablen in der Standard-Variablentabelle deklarieren, oder nach Bedarf weitere, anwenderdefinierte Variablentabellen anlegen.

## Anwenderdefinierte Variablentabellen

Sie können mehrere anwenderdefinierte Variablentabellen für jedes HMI-Gerät anlegen, um Variablen nach Ihren Bedürfnissen zu gruppieren. Anwenderdefinierte Variablentabellen können Sie umbenennen, in Gruppen zusammenfassen oder löschen. Zur Gruppierung von Variablentabellen legen Sie im Ordner HMI-Variablen weitere Unterordner an.

## Alle Variablen

Die Tabelle "Alle Variablen" zeigt eine Übersicht über alle HMI-Variablen und Systemvariablen des jeweiligen HMI-Geräts. Diese Tabelle kann nicht gelöscht, umbenannt oder verschoben werden.

## Tabelle Bitmeldungen

In der Tabelle "Bitmeldungen" projektieren Sie Bitmeldungen an die in der HMI-Variablentabelle selektierte HMI-Variable. Wenn Sie eine Bitmeldung projektieren, ist eine Mehrfachselektion in der HMI-Variablentabelle nicht möglich. Projektieren Sie die Bitmeldungen für jede HMI-Variable separat.

## Tabelle Analogmeldungen

In der Tabelle "Analogmeldungen" projektieren Sie Analogmeldungen an die in der HMI-Variablentabelle selektierte HMI-Variable. Wenn Sie eine Analogmeldung projektieren, ist eine Mehrfachselektion in der HMI-Variablentabelle nicht möglich. Projektieren Sie die Analogmeldungen für jede HMI-Variable separat.

## Siehe auch

Grundlagen zu Variablen (Seite 3160)

### 10.2.1.3 Externe Variablen

#### Einleitung

Externe Variablen ermöglichen die Kommunikation, also den Datenaustausch, zwischen den Komponenten eines Automatisierungssystems, z. B. zwischen Bediengerät und Steuerung.



## Prinzip

Eine externe Variable ist das Abbild eines definierten Speicherplatzes in der Steuerung. Auf diesen Speicherplatz können Sie sowohl vom Bediengerät als auch von der Steuerung lesend und schreibend zugreifen.

Da externe Variablen das Abbild eines Speicherplatzes in der Steuerung sind, hängen die anwendbaren Datentypen von der Steuerung ab, mit der das Bediengerät verbunden ist.

Wenn Sie in STEP 7 ein PLC-Steuerungsprogramm schreiben, werden die im Steuerungsprogramm erstellten PLC-Variablen in die PLC-Variablen-tabelle eingetragen. Wenn Sie eine externe Variable mit einer PLC-Variable verbinden wollen, greifen Sie über die PLC-Variablen-tabelle direkt auf die PLC-Variablen zu und verbinden sie mit der externen Variable.

## Datentypen

An einer externen Variable stehen Ihnen in WinCC alle Datentypen zur Verfügung, die an der verbundenen Steuerung verfügbar sind. Informationen über Datentypen, die bei einer Verbindung zu anderen Steuerungen zur Verfügung stehen, finden Sie in der Dokumentation zu den entsprechenden Kommunikationstreibern.

Nähere Hinweise hierzu finden Sie unter "Grundlagen zur Kommunikation (Seite 3441)".

---

### Hinweis

Für die Kommunikation zwischen Bediengerät und Steuerung stehen Ihnen neben den externen Variablen auch Bereichszeiger zur Verfügung. Die Bereichszeiger können Sie im Editor "Verbindungen" einrichten und aktivieren.

---

## Aktualisierung von Variablenwerten

Bei externen Variablen werden in Runtime die aktuellen Variablenwerte über die Kommunikationsverbindung zwischen WinCC und den verbundenen Automatisierungssystemen übertragen und im Speicher der Runtime abgelegt. Danach wird der Variablenwert mit der eingestellten Zykluszeit aktualisiert. Für die Verwendung im Runtime-Projekt greift WinCC auf Variablenwerte im Speicher der Runtime zu, die zum vorherigen Zykluszeitpunkt aus der Steuerung gelesen wurden. Dadurch kann sich der Wert in der Steuerung bereits ändern, während der Wert aus dem Speicher der Runtime verarbeitet wird.

## Siehe auch

Adressierung externer Variablen (Seite 3164)

Grundlagen zur Kommunikation (Seite 3441)

Grundlagen zu Variablen (Seite 3160)



### 10.2.1.4 Adressierung externer Variablen

#### Einleitung

Die Möglichkeiten zur Adressierung externer Variablen sind abhängig von der Verbindungsart zwischen WinCC und der jeweiligen Steuerung. Folgende Verbindungsarten müssen Sie unterscheiden:

- **Integrierte Verbindung**  
Verbindungen von Geräten, die sich innerhalb eines Projekts befinden und mit dem Editor "Geräte & Netze" erstellt wurden, werden als integrierte Verbindungen bezeichnet.
- **Nicht integrierte Verbindung**  
Verbindungen von Geräten, die mit dem Editor "Verbindungen" erstellt wurden, werden als nicht integrierte Verbindungen bezeichnet. Die Geräte müssen sich nicht alle innerhalb eines Projekts befinden.

Die Art der Verbindung erkennen Sie auch an ihrem Symbol.

|   |                              |
|---|------------------------------|
|  | Integrierte Verbindung       |
|  | Nicht integrierte Verbindung |

Nähere Hinweise hierzu finden Sie im Kapitel Grundlagen zur Kommunikation (Seite 3441).

#### Adressierung bei integrierter Verbindung

Eine integrierte Verbindung bietet den Vorteil, dass Sie eine Variable sowohl symbolisch als auch absolut adressieren können.

Bei der symbolischen Adressierung wählen Sie die PLC-Variable über ihren Namen aus und verbinden Sie mit der HMI-Variable. Der gültige Datentyp für die HMI-Variable wird vom System automatisch gewählt. Bei der Adressierung von Elementen in Datenbausteinen müssen Sie folgende Fälle unterscheiden:

##### **Symbolische Adressierung von Datenbausteinen mit optimiertem Zugriff und mit Standardzugriff:**

Bei der symbolischen Adressierung eines Datenbausteins mit optimiertem Zugriff und mit Standardzugriff wird die Adresse eines Elements im Datenbaustein dynamisch vergeben und bei einer Änderung automatisch in die HMI-Variable übernommen. Ein Übersetzen des verbundenen Datenbausteins oder des WinCC-Projekts ist dafür nicht notwendig.

Bei Datenbausteinen mit optimiertem Zugriff steht Ihnen nur die symbolische Adressierung zur Verfügung.

Bei der symbolischen Adressierung von Elementen eines Datenbausteins muss das WinCC-Projekt lediglich bei folgenden Änderungen neu übersetzt und geladen werden:

- Wenn sich der Name oder der Datentyp des verbundenen Datenbaustein-Elements oder der globalen PLC-Variable geändert hat.
- Wenn sich der Name oder der Datentyp eines im Datenbaustein-Element oder der globalen PLC-Variable übergeordneten Strukturknotens des verbundenen Elements geändert hat.
- Wenn sich die Nummer des verbundenen Datenbausteins geändert hat.

Die symbolische Adressierung steht Ihnen derzeit mit folgenden Steuerungen zur Verfügung:

- SIMATIC S7 1200
- SIMATIC S7 1500

Die symbolische Adressierung steht Ihnen außerdem bei einer integrierten Verbindung zur Verfügung.

---

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" verwenden, dann ist eine symbolische Adressierung nicht möglich.

---

Die absolute Adressierung können Sie auch bei einer integrierten Verbindung verwenden. PLC-Variablen aus einer SIMATIC S7 300/400 Steuerung können Sie nur absolut adressieren. Wenn Sie eine HMI-Variable mit einer PLC-Variable verbunden haben, und sich die Adresse der PLC-Variablen ändert, müssen Sie das Steuerungsprogramm neu übersetzen, damit die neue Adresse in WinCC aktualisiert wird. Übersetzen Sie anschließend auch das WinCC-Projekt neu und laden Sie es auf das Bediengerät.

In WinCC ist die symbolische Adressierung als Standard voreingestellt. Um die Standardeinstellung zu ändern, wählen Sie den Menübefehl "Extras > Einstellungen". Im Dialog "Einstellungen" wählen Sie "Visualisierung > Variablen". Deaktivieren Sie bei Bedarf die Option "Symbolischer Zugriff".

Die Verfügbarkeit einer integrierten Verbindung ist abhängig von der verwendeten Steuerung. Die folgende Tabelle zeigt die Verfügbarkeit:

| Steuerung  | Integrierte Verbindung | Bemerkungen   |
|------------|------------------------|---|
| S7 300/400 | Ja                     | Die Verknüpfung der Variablen wird in Runtime nicht geprüft. Wenn sich in der Steuerung die Variablenadresse ändert, und das HMI-Gerät nicht neu übersetzt und geladen wurde, wird die Änderung in Runtime nicht registriert.   |
| S7 1200    | Ja                     | Bei der symbolischen Adressierung erfolgt eine Gültigkeitsprüfung der Variablenverbindung in Runtime. Bei einer Adressänderung in der Steuerung wird die Änderung registriert und eine Fehlermeldung ausgegeben. Bei der absoluten Adressierung gilt das für die S7 300/400 beschriebene Verhalten. |
| S7 1500    | Ja                     | Bei der symbolischen Adressierung erfolgt eine Gültigkeitsprüfung der Variablenverbindung in Runtime. Bei einer Adressänderung in der Steuerung wird die Änderung registriert und eine Fehlermeldung ausgegeben. Bei der absoluten Adressierung gilt das für die S7 300/400 beschriebene Verhalten. |

Eine integrierte Verbindung erstellen Sie im Editor "Geräte & Netze". Wenn die Steuerung im Projekt enthalten ist und integrierte Verbindungen unterstützt, dann können Sie sich die Verbindung auch automatisch erstellen lassen. Wählen Sie dazu bei der Projektierung der HMI-Variable einfach eine bestehende PLC-Variable aus, mit der Sie die HMI-Variable verbinden wollen. Die integrierte Verbindung wird dann vom System automatisch erstellt.

### Adressierung bei nicht integrierter Verbindung

Bei einem Projekt mit nicht integrierter Verbindung projektieren Sie eine Variablenverbindung immer mit absoluter Adressierung. Den gültigen Datentyp müssen Sie selbst wählen. Wenn sich bei einem Projekt mit nicht integrierter Verbindung im Projektverlauf die Adresse einer PLC-Variablen ändert, müssen Sie die Änderung in WinCC ebenfalls ändern. Die Variablenverbindung kann in Runtime nicht auf Gültigkeit geprüft werden, es erfolgt keine Fehlermeldung.

Eine nicht integrierte Verbindung ist für alle unterstützten Steuerungen verfügbar.

Die symbolische Adressierung ist bei nicht integrierter Verbindung nicht verfügbar.

Bei einer nicht integrierten Verbindung muss das Steuerungsprogramm nicht Bestandteil des WinCC Projekts sein. Die Projektierung der Steuerung und des WinCC-Projekts können Sie unabhängig voneinander durchführen. Für die Projektierung in WinCC müssen lediglich die verwendeten Adressen in der Steuerung und deren Funktion bekannt sein.

### Siehe auch

Externe Variablen (Seite 3162)

Grundlagen zu Variablen (Seite 3160)

Grundlagen zur Kommunikation (Seite 3441)

### 10.2.1.5 Interne Variablen

#### Einleitung

Interne Variablen haben keine Verbindung zur Steuerung. Interne Variablen transportieren Werte innerhalb des Bediengeräts. Die Variablenwerte stehen nur so lange zur Verfügung wie Runtime läuft.

#### Prinzip

Interne Variablen werden im Speicher des Bediengeräts abgelegt. Daher kann nur dieses Bediengerät lesend und schreibend auf die internen Variablen zugreifen. Interne Variablen legen Sie z. B. an, um lokale Berechnungen durchzuführen.

Für interne Variablen stehen Ihnen die HMI Datentypen zur Verfügung.

Folgende HMI-Datentypen sind verfügbar:

| HMI-Datentyp | Datenformat                      |
|--------------|----------------------------------|
| Array        | eindimensionales Array           |
| Bool         | Binäre Variable                  |
| DateTime     | Datum/ Uhrzeit Format            |
| DInt         | Vorzeichenbehafteter 32-Bit Wert |
| Int          | Vorzeichenbehafteter 16-Bit Wert |
| LReal        | Gleitkommazahl 64-Bit IEEE 754   |

| HMI-Datentyp | Datenformat                     |
|--------------|---------------------------------|
| Real         | Gleitkommazahl 32-Bit IEEE 754  |
| SInt         | Vorzeichenbehafteter 8-Bit Wert |
| UDInt        | Vorzeichenloser 32-Bit Wert     |
| UInt         | Vorzeichenloser 16-Bit Wert     |
| USInt        | Vorzeichenloser 8-Bit Wert      |
| WString      | Textvariable 16-Bit Zeichensatz |

## Siehe auch

Grundlagen zu Variablen (Seite 3160)

## 10.2.2 Arbeiten mit Variablen

### 10.2.2.1 Anlegen von Variablen

#### Externe Variable anlegen

#### Einleitung

Mit einer externen Variable greifen Sie über eine PLC-Variablen auf eine Adresse in der Steuerung zu. Für die Adressierung stehen folgende Möglichkeiten zur Verfügung:

- Symbolische Adressierung
- Absolute Adressierung



Nähere Hinweise zur symbolischen Adressierung finden Sie im Kapitel "Adressierung externer Variablen (Seite 3164)". Verwenden Sie möglichst die symbolische Adressierung bei der Projektierung einer Variablen. Variablen legen Sie entweder in der Standard-Variablen-Tabelle oder in einer selbst definierten Variablen-Tabelle an.

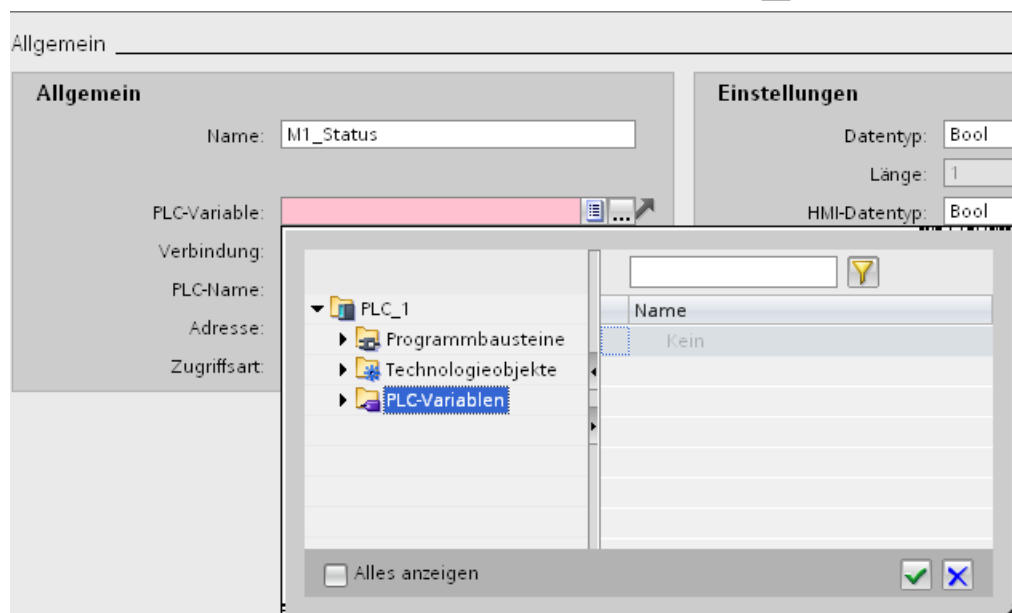
#### Voraussetzung

- Das Projekt ist geöffnet.
- Eine Verbindung zur Steuerung ist projektiert.
- Das Inspektorfenster ist geöffnet.

### Vorgehensweise

Um eine externe Variable anzulegen, gehen Sie folgendermaßen vor:

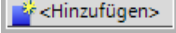
1. Öffnen Sie in der Projektnavigation den Ordner "HMI-Variablen" und doppelklicken Sie die Standard-Variablentabelle. Die Variablentabelle wird geöffnet.  
Alternativ legen Sie eine neue Variablentabelle an und öffnen diese anschließend.
2. Doppelklicken Sie in der Variablentabelle in der Spalte "Name" auf "Hinzufügen". Eine neue Variable wird angelegt.
3. Wählen Sie im Inspektorfenster die Kategorie "Eigenschaften > Eigenschaften > Allgemein" und geben Sie bei Bedarf im Feld "Name" einen eigenen Variablennamen ein. Der Variablenname muss geräteweit eindeutig sein.
4. Geben Sie bei Bedarf im Feld "Anzeigename" einen Namen ein, der in Runtime angezeigt werden soll. Der Anzeigename ist sprachabhängig und kann für die gewünschten Runtime-Sprachen übersetzt werden. Der Anzeigename ist für Basic Panels, Panels und Runtime Advanced verfügbar.
5. Wählen Sie im Feld "Verbindung" die Verbindung zur gewünschten Steuerung aus. Falls Ihnen die gewünschte Verbindung nicht angezeigt wird, müssen Sie zunächst die Verbindung zur Steuerung erstellen. Die Verbindung zu einer SIMATIC S7-Steuerung erstellen Sie im Editor "Geräte & Netze". Die Verbindung zu Fremdsteuerungen erstellen Sie im Editor "Verbindungen".  
Wenn die Steuerung im Projekt enthalten ist und integrierte Verbindungen unterstützt, dann können Sie sich die Verbindung auch automatisch erstellen lassen. Wählen Sie dazu bei der Projektierung der HMI-Variable einfach eine bestehende PLC-Variable aus, mit der Sie die HMI-Variable verbinden wollen. Die integrierte Verbindung wird dann vom System automatisch erstellt.
6. Wenn Sie mit einer integrierten Verbindung arbeiten, klicken Sie im Feld "PLC-Variable" auf die Schaltfläche  und wählen Sie in der Objektliste eine bereits angelegte PLC-Variable aus. Bestätigen Sie die Auswahl über die Schaltfläche .



7. Wenn Sie mit einer nicht integrierten Verbindung arbeiten, geben Sie im Feld "Adresse" die Adresse aus der Steuerung ein. Das Feld "PLC-Variable" bleibt leer.

8. Projektieren Sie die weiteren Eigenschaften der Variable im Inspektorfenster.

Alternativ konfigurieren Sie die Variableneigenschaften direkt in der Variablen-tabelle". Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spalten-titel.

Neue Variablen erstellen Sie alternativ auch direkt an der Verwendungsstelle, z. B. an einem EA-Feld. Hierzu klicken Sie in der Objektliste auf die Schaltfläche . Sie konfigurieren dann die neue Variable im Inspektorfenster.

## Ergebnis

Eine externe Variable ist angelegt und mit einer PLC-Variable oder einer Adresse in der Steuerung verbunden.

## Alternative Vorgehensweise

Externe HMI-Variablen können Sie auch über ein Drag&Drop von Datenbaustein-Elementen oder globalen PLC-Variablen in eine HMI-Variablen-Tabelle erzeugen.

## Siehe auch

Interne Variable anlegen (Seite 3169)

Mehrere Variablen anlegen (Seite 3171)

Variable bearbeiten (Seite 3172)

Grenzwerte einer Variable (Seite 3178)

Grundlagen zu Variablen (Seite 3160)

Adressierung externer Variablen (Seite 3164)

## Interne Variable anlegen

### Einleitung

Bei internen Variablen müssen mindestens der Name und der Datentyp definiert sein. Statt einer Verbindung zu einer Steuerung wird der Eintrag "interne Variable" gewählt.

Zu Dokumentationszwecken ist es sinnvoll, zu jeder Variable einen Kommentar einzugeben.

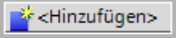
### Voraussetzung

Das Projekt ist geöffnet.

## Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner "HMI Variablen" und doppelklicken Sie den Eintrag "Standard-Variablen-Tabelle". Die Variablen-Tabelle wird geöffnet.  
Alternativ legen Sie eine neue Variablen-Tabelle an und öffnen diese anschließend.
2. Doppelklicken Sie in der Variablen-Tabelle in der Spalte "Name" auf "Hinzufügen". Eine neue Variable wird angelegt.
3. Wenn das Inspektorfenster nicht geöffnet ist, wählen Sie im Menü "Ansicht" die Option "Inspektorfenster".
4. Wählen Sie im Inspektorfenster die Kategorie "Eigenschaften > Eigenschaften > Allgemein" und geben Sie bei Bedarf im Feld "Name" einen eigenen Variablennamen ein. Der Variablenname muss projektweit eindeutig sein.
5. Geben Sie bei Bedarf im Feld "Anzeigename" einen Namen ein, der in Runtime angezeigt werden soll. Der Anzeigename ist sprachabhängig und kann für die gewünschten Runtime-Sprachen übersetzt werden. Der Anzeigename ist für Basic Panels, Panels und Runtime Advanced verfügbar.
6. Wählen Sie im Feld "Verbindung" als Verbindung den Eintrag "interne Variable".
7. Wählen Sie im Feld "Datentyp" den gewünschten Datentyp aus.
8. Abhängig vom gewählten Datentyp müssen Sie im Feld "Länge" die Anzahl der Zeichen festlegen, die maximal in der Variable abgelegt werden. Bei numerischen Variablen ist die Länge automatisch durch den Datentyp definiert.
9. Optional geben Sie einen Kommentar zur Verwendung der Variable ein. Klicken Sie dazu im Inspektorfenster auf "Eigenschaften > Eigenschaften > Kommentar" und geben Sie einen Text ein.

Alternativ konfigurieren Sie die Variableneigenschaften direkt in der Variablen-Tabelle. Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltenüberschriften.

Neue Variablen erstellen Sie alternativ auch direkt an der Verwendungsstelle, z. B. an einem EA-Feld. Hierzu klicken Sie in der Objektliste auf die Schaltfläche . Sie konfigurieren dann die neue Variable in dem sich öffnenden Eigenschaftsfenster.

## Ergebnis

Eine interne Variable wird angelegt, die Sie in Ihrem Projekt einsetzen können.

In weiteren Schritten konfigurieren Sie die Variable, indem Sie z. B. Startwert und Grenzwerte festlegen.

## Siehe auch

Externe Variable anlegen (Seite 3167)



## Mehrere Variablen anlegen

### Einleitung

Im einer Variablen-tabelle legen Sie eine größere Anzahl gleicher Variablen an, indem Sie die Tabellenzeilen unterhalb einer Variable automatisch ausfüllen lassen.

Beim automatischen Ausfüllen werden die Variablennamen automatisch hoch gezählt.

Durch automatisches Ausfüllen können Sie auch einzelne Eigenschaften einer Variable auf die darunter liegenden Tabellenzellen und damit in die entsprechenden Variablen übertragen.

Wenn Sie das automatische Ausfüllen auf bereits gefüllte Zellen einer Variablen-tabelle anwenden, werden Sie gefragt, ob Sie die Zellen überschreiben oder neue Variablen einfügen wollen.

Wenn Sie bereits konfigurierte Variablen nicht überschreiben wollen, aktivieren Sie den Einfügemodus. Den Einfügemodus aktivieren Sie, indem Sie während des Einfügens die Taste <Strg> gedrückt halten. Bereits bestehende Einträge in der Variablen-tabelle werden bei aktiviertem Einfügemodus nach unten verschoben.

### Voraussetzung

- Das Projekt ist geöffnet.
- Eine Variablen-tabelle ist geöffnet.
- Die Variable, die als Vorlage für weitere Variablen dienen soll, ist konfiguriert.

### Vorgehen

1. Wenn Sie neue Variablen erzeugen wollen, markieren Sie in der Spalte "Name" die Variable, die als Vorlage für die neuen Variablen dienen soll.  
Wenn Sie eine Eigenschaft einer Variable auf darunter liegende Variablen übertragen wollen, markieren Sie die Zelle, die diese Eigenschaft enthält.

|  | Name ▲       | Verbindung         |
|--|--------------|--------------------|
|  | Motor        | <Interne Variable> |
|  | <Hinzufügen> |                    |

2. Drücken Sie die Maustaste über die Zellen nach unten, die Sie markieren wollen. In der unteren rechten Ecke des Bereichs erscheint ein Cursor. Drücken Sie den Mauszeiger über dem Quadrat, der Cursor markiert, um den Bereich zu erweitern.  
3. Drücken Sie die Maustaste über die markierten Zellen, um den Bereich zu erweitern. In den markierten Zellen werden automatisch ausgefüllt. In den darunter liegenden Bereichen werden neue Variablen angelegt.

|  | Name ▲  | Verbindung         |
|--|---------|--------------------|
|  | Motor   | <Interne Variable> |
|  | Motor_1 | <Interne Variable> |
|  | Motor_2 | <Interne Variable> |
|  | Motor_3 | <Interne Variable> |
|  | Motor_4 | <Interne Variable> |
|  | Motor_5 | <Interne Variable> |

### Ergebnis

Abhängig davon, welche Zelle markiert war, werden einzelne Eigenschaften automatisch ausgefüllt oder neue Variablen angelegt.

## Siehe auch

Externe Variable anlegen (Seite 3167)

### 10.2.2.2 Bearbeiten von Variablen

#### Variable bearbeiten

##### Einleitung

Eine Variable können Sie jederzeit umbenennen, kopieren oder löschen.

Wenn Sie eine Variable umbenennen, muss der neue Name geräteweit eindeutig sein.

Wenn Sie eine Variable mit dem Befehl "Kopieren" in die Zwischenablage kopieren, werden die mit der Variablen Referenzen ebenfalls kopiert.

Wenn Sie eine Variable mit dem Befehl "Einfügen" in ein anderes Gerät einfügen, dann wird die Variable ohne die verbundenen Referenzen eingefügt. Von einer Referenz wird lediglich der Objektname eingefügt. Wenn im Zielsystem eine gleichnamige Referenz mit gültigen Eigenschaften existiert, dann wird die existierende Referenz mit der kopierten Variable verbunden.

Wenn Sie eine Variable kopieren, werden die mit der Variablen verknüpfte Meldungen ebenfalls kopiert. Wenn Sie die kopierte Variable in ein anderes Gerät einfügen, dann wird die Variable zusammen mit den verknüpften Meldungen eingefügt.

##### Voraussetzung

- Die Variable, die Sie umbenennen, kopieren oder löschen wollen, ist angelegt.
- Die Variablen-tabelle ist geöffnet.

##### Variable umbenennen

1. Markieren Sie das Feld "Name" der Variable in der Variablen-tabelle.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie einen neuen Namen ein.  
Die Variable wird unter dem neuen Namen angezeigt.

##### Variable kopieren

1. Markieren Sie eine oder mehrere Variablen in der Variablen-tabelle oder im Detailfenster.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".

3. Klicken Sie auf die Stelle, an der Sie die Variable einfügen wollen. Klicken Sie z. B. auf eine andere Variablen-tabelle desselben Geräts oder auf eine Variablen-tabelle in einem zweiten Gerät.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen" oder "Erweitertes einfügen". Die Variable wird wie oben beschrieben eingefügt.

### Variable löschen

1. Markieren Sie eine oder mehrere Variablen in der Variablen-tabelle.
2. Wählen Sie im Menü "Werkzeuge" den Befehl "Querverweise". Überprüfen Sie im Editor "Querverweise", wo die Variablen verwendet werden. Sie sehen so, welchen Einfluss das Löschen der Variablen auf Ihr Projekt hat.
3. Wählen Sie im Kontextmenü der Variablen den Befehl "Löschen". Alle markierten Variablen werden gelöscht.

### Export und Import von Variablen

WinCC bietet Ihnen die Möglichkeit, Variablen zu exportieren und zu importieren. Mit dem Export und Import haben Sie die Möglichkeit, Variablen aus einem Projekt zu exportieren und in ein anderes Projekt zu importieren. Außerdem haben Sie die Möglichkeit, größere Mengen von Variablen außerhalb von WinCC anzulegen, zu bearbeiten und anschließend in beliebige WinCC-Projekte zu importieren. Nähere Hinweise hierzu finden Sie unter Variablen importieren und exportieren (Seite 3865).

### Siehe auch

- Ändern der Variablenkonfiguration (Seite 3173)
- Mehrere Variablen gleichzeitig konfigurieren (Seite 3174)
- Mehrere Variablen gleichzeitig in einem Bild verwenden (Seite 3175)
- Variable neu verbinden (Seite 3177)
- Externe Variable anlegen (Seite 3167)
- Variablen importieren und exportieren (Seite 3865)

### Ändern der Variablenkonfiguration

#### Einleitung

Variablen können Sie jederzeit an geänderte Anforderungen im Projekt anpassen.

#### Variablenkonfiguration ändern

Wenn Sie die Konfiguration einer Variablen ändern wollen, öffnen Sie die Variablen-tabelle, in der die Variable enthalten ist. Alternativ öffnen Sie die Variablen-tabelle "Alle Variablen anzeigen".

Sie können in den Variablentabellen z. B. die Eigenschaften mehrerer Variablen vergleichen und aufeinander abstimmen oder die Variablen nach ihren Eigenschaften sortieren.

Ändern Sie die Eigenschaften entweder direkt in der Tabelle, oder im Inspektorfenster.

Wenn Sie eine Variableneigenschaft ändern und durch die Änderung Widersprüche zu anderen Eigenschaften auftreten, werden Sie durch eine farbige Hinterlegung darauf aufmerksam gemacht. Dies tritt z. B. ein, wenn Sie die Variable mit einer anderen Steuerung verbinden, die den eingestellten Datentyp nicht unterstützt.

## Siehe auch

Variable bearbeiten (Seite 3172)

## Mehrere Variablen gleichzeitig konfigurieren

### Einleitung

In WinCC weisen Sie mehreren Variablen in einem Arbeitsgang übereinstimmende Eigenschaften zu. Dies ermöglicht Ihnen, effizient zu projektieren.

### Voraussetzung

- Die Variablen, die Sie konfigurieren wollen, sind angelegt.
- Die Variablentabelle ist geöffnet.
- Das Inspektorfenster ist geöffnet.

### Vorgehensweise

1. Wählen Sie in der Variablentabelle alle Variablen aus, die Sie gemeinsam konfigurieren wollen.  
Wenn die Variablen in der ausgewählten Eigenschaft übereinstimmen, wird die Einstellung für diese Eigenschaft im Inspektorfenster angezeigt. Wenn die Eigenschaften nicht übereinstimmen, bleibt das zugehörige Feld leer.
2. Legen Sie die gemeinsame Eigenschaft im Inspektorfenster oder direkt in der Variablentabelle fest.

Wenn Sie an mehreren Variablen gemeinsam eine Eigenschaft ändern, dann wird nur diese eine Eigenschaft geändert. Die weiteren Eigenschaften der Variablen bleiben unverändert.

### Ergebnis

Alle markierten Variablen werden neu konfiguriert.

Wenn Sie anschließend Variableneigenschaften konfigurieren wollen, in denen sich die Variablen unterscheiden, heben Sie dazu die Mehrfachauswahl auf.

## Siehe auch

Variable bearbeiten (Seite 3172)

## Mehrere Variablen gleichzeitig in einem Bild verwenden

### Einleitung

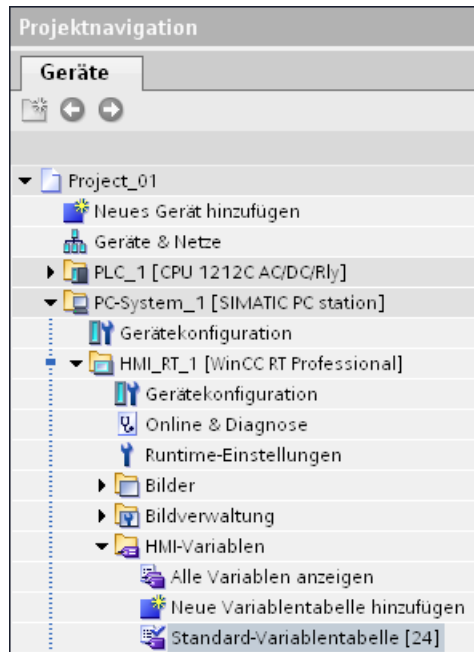
In WinCC können Sie in einem Arbeitsgang mehrere E/A-Felder in einem Bild erstellen, die mit Variablen verbunden sind. Dies ermöglicht Ihnen, effizient zu projektieren.

### Voraussetzung

- Mehrere Variablen sind angelegt.
- Ein Bild ist geöffnet.

### Vorgehen

1. Markieren Sie in der Projektnavigation unter "HMI-Variablen" die gewünschte Variablen-tabelle.



2. Wählen Sie das Detailfenster im unteren Bereich der Projektnavigation. Im Detailfenster werden die in der gewählten Variablengruppe vorhandenen Variablen angezeigt.

| ▼ Detailansicht |       |          |
|-----------------|-------|----------|
|                 | Name  | Datentyp |
|                 | Tag_1 | Bool     |
|                 | Tag_2 | Bool     |
|                 | Tag_3 | Bool     |
|                 | Tag_4 | Bool     |
|                 | Tag_5 | Bool     |
|                 | Tag_6 | Bool     |

3. Markieren Sie im Detailfenster die Variablen.
4. Ziehen Sie die Variablen per Drag&Drop in das Bild. Für jede Variable wird ein E/A-Feld erstellt, das mit der jeweiligen Variable verbunden ist.

---

#### Hinweis

Wenn Sie eine PLC-Variable mit Drag&Drop aus dem Detailfenster in den Arbeitsbereich ziehen, wird zusätzlich im Editor "Geräte & Netze" ein Netzwerk und eine Verbindung angelegt.

---

## Siehe auch

Variable bearbeiten (Seite 3172)

## Variable neu verbinden

### Einleitung

WinCC bietet Ihnen die Möglichkeit, Variablen automatisch mit Adressen in der Steuerung zu verbinden. Diese Vorgehensweise bietet sich an, wenn z. B. Änderungen an der Verbindung zwischen Bediengerät und Steuerung durchgeführt wurden und dabei die Variablenanbindungen verloren gingen. Die Funktion ist auch verwendbar, wenn Sie Steuerungsprogramm und HMI-Projekt separat projiziert haben.

Zur Durchführung steht Ihnen der Kontextmenübefehl "PLC-Variable neu verbinden" zur Verfügung.

Der Menübefehl ist unter folgenden Voraussetzungen verfügbar:

- Eine integrierte Verbindung zur Steuerung ist vorhanden.
- In der HMI-Variable ist die absolute Adresse aus der Steuerung eingetragen.
- Die HMI-Variable ist mit dem richtigen Datentyp projiziert.

An einer Variable mit symbolischer Adressierung ist der Menübefehl nicht verfügbar.

Wenn Sie mehrere Variablen selektieren, ist der Menübefehl verfügbar, wenn mindestens eine der selektierten Variablen die oben genannten Voraussetzungen erfüllt. Es werden immer nur die Variablen verbunden, die die Voraussetzungen erfüllen.

### Voraussetzung

- Eine HMI-Variable ist angelegt.
- Die Variablen-tabelle ist geöffnet.
- Eine PLC-Variable mit der absoluten Adresse aus der Steuerung ist vorhanden.

### Vorgehensweise

Um Variablen neu zu verbinden, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Variablen-tabelle die Zeile mit der Variable.
2. Öffnen Sie das Kontextmenü und wählen Sie den Menübefehl "PLC-Variable neu verbinden".

Das System sucht nach einer PLC-Variable, deren absolute Adresse und Datentyp mit den Einstellungen an der HMI-Variable übereinstimmen. Wenn eine passende PLC-Variable gefunden wird, dann wird die Variablen-Verbindung sofort etabliert.

### Ergebnis

Die PLC-Variable ist mit der HMI-Variable verbunden.

## Siehe auch

Variable bearbeiten (Seite 3172)

### 10.2.2.3 Konfigurieren von Variablen

#### Grenzwerte einer Variable

##### Einleitung

Für numerische Variablen können Sie den Wertebereich durch Grenzwerte einschränken.

##### Prinzip

Für numerische Variablen legen Sie einen Wertebereich fest, der durch einen oberen und einen unteren Grenzwert definiert wird.

Wenn der Prozesswert den Wertebereich überschreitet oder unterschreitet triggern Sie eine Funktionsliste. Wenn der Bediener einen Wert für die Variable eingibt, der außerhalb des projektierten Wertebereichs liegt, wird die Eingabe abgewiesen. Der Wert wird nicht übernommen.

---

##### Hinweis

Den Text der Analogmeldungen für Grenzwertverletzungen geben Sie im Editor "Analogmeldungen" ein.

---

#### Anwendungsbeispiel

Verwenden Sie Grenzwerte, z. B. um den Bediener rechtzeitig zu warnen, wenn der Wert einer Variable in einen kritischen Bereich eintritt.



## Siehe auch

- Variable linear skalieren (Seite 3184)
- Startwert einer Variable festlegen (Seite 3181)
- Erfassungszyklus einer Variable festlegen (Seite 3188)
- Grenzwerte einer Variable festlegen (Seite 3179)
- Startwert einer Variable (Seite 3180)
- Aktualisieren des Variablenwerts in Runtime (Seite 3182)
- Lineares Skalieren einer Variable (Seite 3183)
- Variable mit einer anderen Steuerung verbinden (Seite 3185)
- Indirekte Adressierung von Variablen (Seite 3185)
- Variable indirekt adressieren (Seite 3186)
- Variable zum Auslösen von Funktionen nutzen (Seite 3187)
- Adress-Multiplexen (Seite 3189)
- Adress-Multiplexen projektieren mit absoluter Adressierung (Seite 3191)
- Adress-Multiplexen projektieren mit symbolischer Adressierung (Seite 3193)
- Externe Variable anlegen (Seite 3167)

## Grenzwerte einer Variable festlegen

### Einleitung

Für numerische Variablen definieren Sie einen Wertebereich, indem Sie einen unteren und einen oberen Grenzwert definieren.



An das Unterschreiten oder Überschreiten des Wertebereichs können Sie zusätzlich das Abarbeiten einer Funktionsliste projektieren.

### Voraussetzung

- Die Variable, für die Sie Grenzbereiche festlegen wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften für diese Variable ist geöffnet.

## Vorgehensweise

Um Grenzwerte einer Variablen festzulegen gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Grenzwerte". Wenn Sie für einen der Grenzwerte einen konstanten Wert festlegen wollen, wählen Sie über die Schaltfläche  den Eintrag "Konstante" aus. Geben Sie im entsprechenden Feld eine Zahl ein.  
Wenn Sie für einen der Grenzwerte den Wert einer Variable verwenden wollen, wählen Sie über die Schaltfläche  den Eintrag "HMI Variable" aus. Mithilfe der Objektliste legen Sie die Variable für den Grenzwert fest.
2. Wiederholen Sie Schritt 1 sinngemäß, um einen weiteren Grenzwert für die Variable festzulegen.

## Alternative Vorgehensweise

Alternativ konfigurieren Sie den oberen und den unteren Grenzwert direkt in der Variablen-tabelle. Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

## Funktionsliste projektieren

Für das überschreiten oder unterschreiten des Wertebereichs projektieren Sie eine Funktionsliste wie folgt:

1. Wenn Sie bei Unterschreiten des Wertebereichs eine Funktionsliste starten wollen, klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Minimum unterschritten". Legen Sie dort eine Funktionsliste an.
2. Wenn Sie bei Überschreiten des Wertebereichs eine Funktionsliste starten wollen, klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Maximum überschritten". Legen Sie dort eine Funktionsliste an.

## Ergebnis

Sie haben für die ausgewählte Variable einen Wertebereich festgelegt, der durch einen unteren und einen oberen Grenzwert definiert ist. Bei Unterschreiten oder Überschreiten des Wertebereichs wird eine Funktionsliste abgearbeitet.

## Siehe auch

Grenzwerte einer Variable (Seite 3178)

## Startwert einer Variable

## Wert einer Variablen zu Runtime-Start

Für numerische Variablen und Variablen für Datum-/Uhrzeit-Werte können Sie einen Startwert projektieren. Beim Start von Runtime wird die Variable mit diesem Wert vorbelegt. Auf diese Weise schaffen Sie einen definierten Zustand der Variablen zu Runtime-Start.

Bei externen Variablen wird der Startwert so lange am Bediengerät angezeigt, bis er von der Steuerung oder durch eine Eingabe überschrieben wird.

Wenn kein Startwert projiziert wurde, enthält die Variable beim Start von Runtime den Wert "0".

In WinCC Runtime Professional können Sie an einer Variablen mit dem Datentyp "String" an Stelle des Startwerts einen Variablenwert eingeben. Der Variablenwert wird im Editor "Projekttexte" abgelegt und ist mehrsprachig. Nach der Übersetzung des Textes wird er in Runtime als sprachabhängiger Startwert angezeigt.

## Anwendungsbeispiel

Ein E/A-Feld können Sie mit einem Defaultwert vorbelegen. Legen Sie den gewünschten Defaultwert als Startwert für die Variable fest, die mit dem E/A-Feld verknüpft ist.

## Siehe auch

Startwert einer Variable festlegen (Seite 3181)

Grenzwerte einer Variable (Seite 3178)

## Startwert einer Variable festlegen

### Einleitung

In WinCC können Sie für eine numerische Variable und eine Variable für Datum-/Uhrzeit-Werte einen Startwert projektieren, den diese bei Runtime-Start annimmt.

### Voraussetzung

- Die Variable, für die Sie einen Startwert festlegen wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften der Variable ist geöffnet.

### Vorgehensweise

Um einen Startwert zu projektieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Werte".
2. Geben Sie den gewünschten "Startwert" ein.

### Alternative Vorgehensweise

Alternativ konfigurieren Sie den Startwert direkt in der Variablentabelle. Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spaltentitel.

### Ergebnis

Der gewählte Startwert für die Variable wird in das Projekt übernommen.

## Siehe auch

Grenzwerte einer Variable (Seite 3178)

Startwert einer Variable (Seite 3180)

## Aktualisieren des Variablenwerts in Runtime

### Einleitung

Variablen enthalten Prozesswerte, die sich im Verlauf von Runtime ändern. Bei internen und externen Variablen werden Wertänderungen auf unterschiedliche Weise ausgelöst.

### Prinzip

Beim Start von Runtime ist der Wert einer Variable gleich ihrem Startwert. In Runtime kommt es zu Wertänderungen der Variablen.

In Runtime haben Sie folgende Möglichkeiten, den Wert einer Variable zu ändern:

- Durch Wertänderung einer externen Variablen in der Steuerung.
- Durch Eingabe, z. B. in einem E/A-Feld.
- Durch Ausführen einer Systemfunktion zur Wertänderung, z. B. der Systemfunktion "SetzeWert".

## Aktualisierung des Werts von externen Variablen

Der Wert einer externen Variable wird folgendermaßen aktualisiert:

- **Zyklisch im Betrieb**  
Wenn Sie die Erfassungsart "Zyklisch im Betrieb" wählen, wird die Variable in Runtime aktualisiert, so lange sie in einem Bild angezeigt wird. Der Erfassungszyklus bestimmt, in welchem Zeitintervall der Wert der Variable am Bediengerät aktualisiert wird. Die zyklische Erfassung beruht auf der gewählten Zykluszeit.
- **Zyklisch fortlaufend**  
Wenn Sie die Erfassungsart "Zyklisch fortlaufend" wählen, wird die Variable in Runtime ständig aktualisiert, auch wenn sie sich nicht im aufgeschlagenen Bild befindet. Die Einstellung wird z. B. für Variablen aktiviert, an deren Wertänderung eine Funktionsliste projiziert ist.  
Verwenden Sie "Zyklisch fortlaufend" nur für Variablen, die tatsächlich immer aktualisiert werden müssen. Häufige Lesevorgänge führen zu einer höheren Kommunikationslast.
- **Auf Anforderung**  
Wenn Sie die Erfassungsart "Auf Anforderung" wählen, wird die Variable nicht zyklisch aktualisiert. Die Aktualisierung erfolgt nur auf Anforderung, z. B. durch die Systemfunktion "Aktualisiere Variable".

## Siehe auch

Grenzwerte einer Variable (Seite 3178)

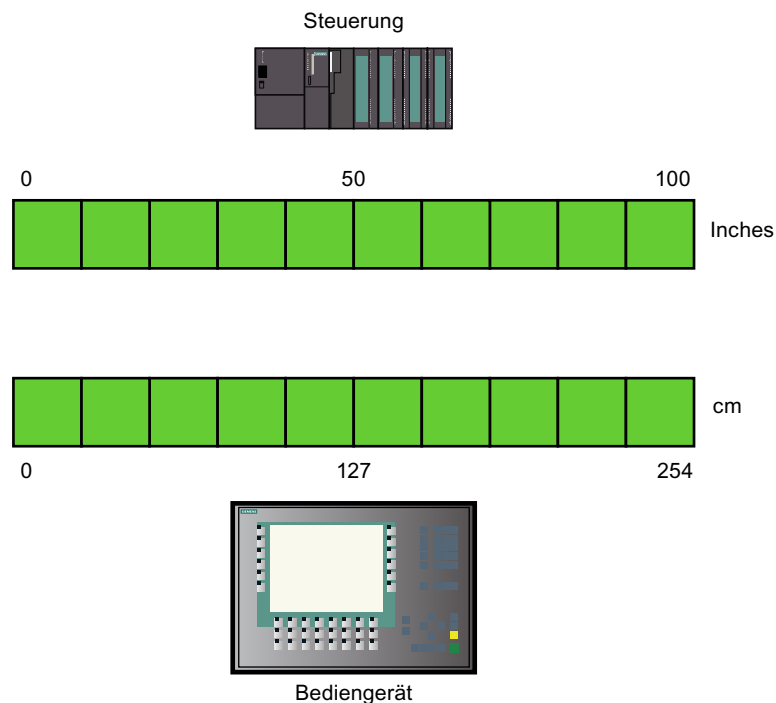
## Lineares Skalieren einer Variable

### Einleitung

Numerische Datentypen können Sie linear skalieren. Die in der Steuerung vorhandenen Prozesswerte einer externen Variable lassen sich auf einen bestimmten Wertebereich im Projekt abbilden.

### Prinzip

Für die lineare Skalierung einer Variable werden ein Wertebereich am Bediengerät und ein Wertebereich an der Steuerung angegeben. Die Wertebereiche werden linear aufeinander abgebildet.



Sobald ein Prozesswert vom Bediengerät in eine externe Variable geschrieben wird, wird er automatisch auf den Wertebereich der Steuerung abgebildet. Sobald ein Prozesswert vom Bediengerät aus der externen Variablen gelesen wird, wird eine entsprechende Umkehrtransformation vorgenommen.

---

#### Hinweis

Für das automatische Umrechnen von Prozesswerten können Sie auch die Systemfunktionen "LineareSkalierung" und "InvertiereLineareSkalierung" einsetzen.

---

## Anwendungsbeispiel

Der Benutzer gibt Längenmaße in cm ein, die Steuerung erwartet jedoch Inch. Die eingegebenen Werte werden automatisch umgerechnet, bevor sie an die Steuerung weitergeleitet werden. Mit der linearen Skalierung passen Sie den Wertebereich der Steuerung [0 ... 100] auf den Wertebereich [0 ... 254] im Bediengerät an.

## Siehe auch

Grenzwerte einer Variable (Seite 3178)

## Variable linear skalieren

### Einleitung

Für die lineare Skalierung einer Variable werden ein Wertebereich am Bediengerät und ein Wertebereich an der Steuerung angegeben. Die Wertebereiche werden linear aufeinander abgebildet.

### Voraussetzung

- Die externe Variable, die Sie linear skalieren wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften für diese Variable ist geöffnet.

### Vorgehensweise

Um eine Variable linear zu skalieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Lineare Skalierung".
2. Klicken Sie auf "Aktivieren", damit die lineare Skalierung eingeschaltet wird.  
Mit dieser Option schalten Sie die lineare Skalierung zeitweilig aus, z. B. zu Testzwecken.  
Bereits vorgenommene Einstellungen für die lineare Skalierung bleiben dabei erhalten.
3. Geben Sie im Bereich "Steuerung" den Endwert und den Anfangswert für den Wertebereich ein, der für die Prozesswerte in der Steuerung gilt.
4. Geben Sie im Bereich "Bediengerät" den Endwert und den Anfangswert für den Wertebereich ein, der für die Prozesswerte am Bediengerät verwendet wird.

### Ergebnis

In Runtime werden die Prozesswerte automatisch von einem Wertebereich auf den anderen abgebildet.

---

#### Hinweis

Für das automatische Umrechnen von Prozesswerten stehen Ihnen auch die Systemfunktionen "LineareSkalierung" und "InvertiereLineareSkalierung" zur Verfügung.

---

## Siehe auch

Grenzwerte einer Variable (Seite 3178)

## Variable mit einer anderen Steuerung verbinden

### Einleitung

Sie können in WinCC jederzeit die Steuerungsanbindung einer Variable ändern. Dies ist z. B. nötig, wenn Sie die Konfiguration Ihrer Anlage ändern.

Abhängig von der gewählten Steuerung müssen Sie anschließend die Konfiguration der Variable anpassen. Anhand der farbigen Hinterlegung sehen Sie, welche Variableneigenschaften Sie ändern müssen.

### Voraussetzung

- Die externe Variable, deren Verbindung Sie ändern wollen, ist angelegt.
- Die Verbindungen zu den Steuerungen sind angelegt.
- Das Eigenschaftsfenster für die Variable ist geöffnet.

### Vorgehensweise

Um die Steuerungsanbindung einer Variable zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wählen Sie im Feld "Verbindung" die neue Verbindung aus.  
Die Variableneigenschaften, die Sie anpassen müssen, werden in der Variablen-tabelle und im Inspektorfenster farbig hinterlegt.
3. Passen Sie alle farbig hinterlegten Eigenschaften der Variable an die Erfordernisse der neuen Steuerung an.

### Ergebnis

Die externe Variable wird mit der neuen Steuerung verbunden.

## Siehe auch

Grenzwerte einer Variable (Seite 3178)

## Indirekte Adressierung von Variablen

### Prinzip

Beim Multiplexen, einer Art indirekte Adressierung, wird die genutzte Variable erst zur Laufzeit ermittelt. Für die Multiplexvariable wird eine Liste von Variablen festgelegt. Aus der Liste von

Variablen wird die relevante Variable in Runtime ausgewählt. Die Auswahl der Variable hängt vom Wert der Indexvariablen ab.

In Runtime wird zunächst der Wert der Indexvariable ausgelesen. Dann wird auf die Variable zugegriffen, die an der entsprechenden Stelle der Variablenliste steht.

### Anwendungsbeispiel

Mit der indirekten Adressierung können Sie folgendes Szenario projektieren:

Der Bediener wählt aus einer Auswahlliste eine von mehreren Maschinen aus. Abhängig von der Auswahl des Bedieners wird ein Prozesswert der Maschine in einem Ausgabefeld angezeigt.

Um ein solches Szenario zu projektieren, projektieren Sie an ein symbolisches E/A-Feld die Indexvariable. An ein E/A-Feld projektieren Sie die Multiplexvariable. Die Variablenliste der Multiplexvariable projektieren Sie parallel zum Aufbau der Auswahlliste.

Wenn der Bediener eine andere Maschine auswählt, ändert sich dadurch der Wert der Indexvariable. Das Auswahlfeld zeigt dann den Inhalt der Variablen an, die in der Variablenliste der Multiplexvariable zum neuen Indexwert gehört.

### Siehe auch

Grenzwerte einer Variable (Seite 3178)

### Variable indirekt adressieren

#### Einleitung

Bei der indirekten Adressierung wird die genutzte Variable erst zur Laufzeit ermittelt. Anstelle einer einzelnen Variable wird eine Liste von Variablen festgelegt. Die Listeneinträge bestehen aus einem Indexwert und dem Namen der Variablen, die genutzt werden soll. Über eine Indexvariable steuern Sie, auf welchen Eintrag der Variablenliste zugegriffen wird.

#### Voraussetzung

- Die Variable, die Sie für die indirekte Adressierung verwenden wollen, ist angelegt.
- Die Indexvariable ist angelegt.
- Die Variablen, die in der Variablenliste enthalten sein sollen, sind angelegt.
- Das Inspektorfenster mit den Eigenschaften für Variablen ist geöffnet.



## Vorgehensweise

Um Variablen indirekt zu adressieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Multiplexen".
2. Aktivieren Sie die Option "Multiplexen", damit die indirekte Adressierung eingeschaltet wird. Mit dieser Option schalten Sie die indirekte Adressierung zeitweilig aus, z. B. zu Testzwecken. Bereits vorgenommene Einstellungen für die indirekte Adressierung bleiben erhalten.
3. Wählen Sie im Feld "Indexvariable" die vorgesehene Indexvariable aus oder definieren Sie eine neue Variable mithilfe der Objektliste.
4. Klicken Sie in der Variablenliste auf den ersten Eintrag der Spalte "Variablen".
5. Wählen Sie eine Variable als Listeneintrag aus oder erstellen Sie eine neue Variable mithilfe der Objektliste.  
Der Eintrag in der Spalte "Index" wird automatisch erzeugt.
6. Wiederholen Sie Schritt 5 für alle Variablen, die Sie in die Variablenliste aufnehmen wollen.
7. Bei Bedarf ändern Sie die Reihenfolge der Listeneinträge mit Drag&Drop.

## Ergebnis

In Runtime wird dynamisch auf die Variable aus der Variablenliste zugegriffen, deren Index gleich dem Wert der Indexvariablen ist.

## Siehe auch

Grenzwerte einer Variable (Seite 3178)

## Variable zum Auslösen von Funktionen nutzen

### Einleitung

Variablenwerte können Sie als auslösendes Ereignis für eine Aktion in Runtime nutzen. Um eine Aktion in Runtime zu starten, projektieren Sie eine Funktionsliste an eine Variable. In die Funktionsliste nehmen Sie eine oder mehrere Systemfunktionen auf. Die Funktionsliste wird beim Eintritt des projektierten Ereignisses abgearbeitet.

Folgende Ereignisse sind an einer Variablen verfügbar:

- Wertänderung der Variablen  
Die Funktionsliste wird bei jeder Änderung des Variablenwerts abgearbeitet.  
Wenn die Variable Arrays enthält, dann wird die Funktionsliste bei jeder Änderung eines Arrayelements abgearbeitet.
- Überschreiten des oberen Grenzwerts der Variablen  
Die Funktionsliste wird bei Überschreiten des oberen Grenzwerts abgearbeitet.
- Unterschreiten des unteren Grenzwerts der Variablen  
Die Funktionsliste wird bei Unterschreiten des unteren Grenzwerts abgearbeitet.

### Voraussetzung

- Die Variable, deren Wert Sie als Ereignis nutzen wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften für diese Variable ist geöffnet.

### Vorgehensweise

Um eine Funktionsliste an eine Variable zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Inspektorfenster unter "Eigenschaften > Ereignisse" das Ereignis, für das Sie eine Funktionsliste erstellen wollen.  
Die zum selektierten Ereignis gehörende Funktionsliste wird angezeigt.
2. Klicken Sie auf "<Funktion hinzufügen>". In der zweiten Tabellenspalte wird eine Auswahl Schaltfläche angezeigt.
3. Klicken Sie auf die Auswahl Schaltfläche und wählen Sie eine Systemfunktion aus.
4. Legen Sie die Parameterwerte fest.

### Ergebnis

Wenn das projizierte Ereignis in Runtime eintritt, wird die Funktionsliste abgearbeitet.

### Siehe auch

Grenzwerte einer Variable (Seite 3178)

## Erfassungszyklus einer Variable festlegen

### Einleitung

Der Wert einer externen Variable kann in Runtime durch die Steuerung geändert werden, mit der die Variable verknüpft ist. Damit das Bediengerät von einer Wertänderung durch die Steuerung erfährt, muss der Variablenwert am Bediengerät aktualisiert werden. Die Aktualisierung erfolgt in regelmäßigen Abständen, solange die Variable im Prozessbild angezeigt wird oder archiviert wird. Eine regelmäßige Aktualisierung wird durch einen Erfassungszyklus gesteuert. Die Aktualisierung kann aber auch fortlaufend erfolgen.

### Voraussetzung

- Die Variable, für die Sie einen Erfassungszyklus festlegen wollen, ist angelegt.
- Das Inspektorfenster mit den Eigenschaften für die Variable ist geöffnet.

## Vorgehen

Um einen Erfassungszyklus für eine Variable zu projektieren, gehen sie folgendermaßen vor:

1. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
2. Wenn Sie die Variable regelmäßig aktualisieren wollen, solange sie im Bild angezeigt oder archiviert wird, wählen Sie die Erfassungsart "Zyklisch im Betrieb".  
Oder:  
Wenn Sie die Variable regelmäßig aktualisieren wollen, auch wenn sie nicht im Bild angezeigt oder archiviert wird, wählen Sie die Erfassungsart "Zyklisch fortlaufend".  
Die Einstellung "Zyklisch fortlaufend" wird z. B. für eine Variable aktiviert, an deren Wertänderung eine Funktionsliste projiziert ist und die in keinem Bild direkt sichtbar ist.
3. Wählen Sie im Feld "Erfassungszyklus" die gewünschte Zykluszeit aus oder definieren Sie einen eigenen Erfassungszyklus mithilfe der Objektliste.

Alternativ konfigurieren Sie den Erfassungszyklus direkt im Arbeitsbereich der Variablen-tabelle. Spalten, die nicht angezeigt werden, aktivieren Sie über das Kontextmenü der Spalten-titel.

---

### Hinweis

Verwenden Sie die Erfassungsart "Zyklisch fortlaufend" nur für Variablen, die tatsächlich fortlaufend aktualisiert werden müssen. Häufige Lesevorgänge erzeugen eine höhere Kommunikationslast.

---

## Ergebnis

Die projizierte Variable wird in Runtime mit dem gewählten Erfassungszyklus aktualisiert.

## Siehe auch

Grenzwerte einer Variable (Seite 3178)

## Adress-Multiplexen

### Einleitung

Beim Adress-Multiplexen können Sie mit einer einzigen Variablen eine Vielzahl an Speicherplätzen im Adressbereich der Steuerung ansprechen. Sie können auf die Adressen schreibend und lesend zugreifen, ohne für jede einzelne Adresse eine Variable zu definieren.

### Multiplexen mit absoluter Adressierung

Beim Multiplexen mit absoluten Adressen projektieren Sie Variablen als Platzhalter für die anzusprechende Adresse in der Steuerung.

Wenn Sie z. B. auf eine Adresse des Formats "%DBx.DBWy" zugreifen wollen, sieht der Ausdruck für das Multiplexen folgendermaßen aus:

```
"%DB[HMITag1].DBW[HMITag2]"
```

Die Variable "HMITag1" versorgen Sie in Runtime mit dem gewünschten Wert für den Datenbaustein, den Sie adressieren wollen.

Die Variable "HMITag2" versorgen Sie in Runtime mit der gewünschten Adresse aus dem Datenbaustein.

Die Versorgung der Variablen geschieht z. B. mithilfe von Werten aus der Steuerung oder über ein Skript.

Das Multiplexen mit absoluten Adressen wird für folgende Steuerungen und Kommunikationstreiber unterstützt.

- SIMATIC S7 300/400
- SIMATIC S7 1200

Das Multiplexen mit absoluten Adressen steht für Datenbausteine mit optimiertem Zugriff nicht zur Verfügung.

### **Multiplexen mit symbolischer Adressierung**

Beim Multiplexen mit symbolischer Adressierung greifen Sie mithilfe einer Multiplexvariable und einer Indexvariable auf ein Arrayelement einer Arrayvariable in einem Datenbaustein der verbundenen Steuerung zu. Die Multiplexvariable enthält die symbolische Adresse des Datenbausteins, auf den Sie zugreifen wollen. Die symbolische Adresse enthält außerdem die Indexvariable, über die Sie auf den Index der Arrayvariable zugreifen.

Wenn Sie z. B. auf die Arrayvariable "Arraytag\_1" im Datenbaustein "Datablock\_1" zugreifen wollen, sieht der Ausdruck für die symbolische Adressierung folgendermaßen aus:

```
"Datablock_1.Arraytag_1["HMITag_1"]
```

Mit der HMI-Variablen "HMITag\_1" steuern Sie den Zugriff auf den Index der Arrayelemente. Die Variable versorgen Sie in Runtime mit dem Index des Arrayelements, auf das Sie jeweils zugreifen wollen.

Das Multiplexen mit symbolischer Adressierung ist nur verfügbar, wenn folgende Komponenten die symbolische Adressierung unterstützen:

- das HMI-Bediengerät
- die Steuerung
- der Kommunikationstreiber

Die symbolische Adressierung wird derzeit nur vom Kommunikationstreiber SIMATIC S7 1200 unterstützt.

### **Siehe auch**

Grenzwerte einer Variable (Seite 3178)

## Adress-Multiplexen projektieren mit absoluter Adressierung

### Einleitung

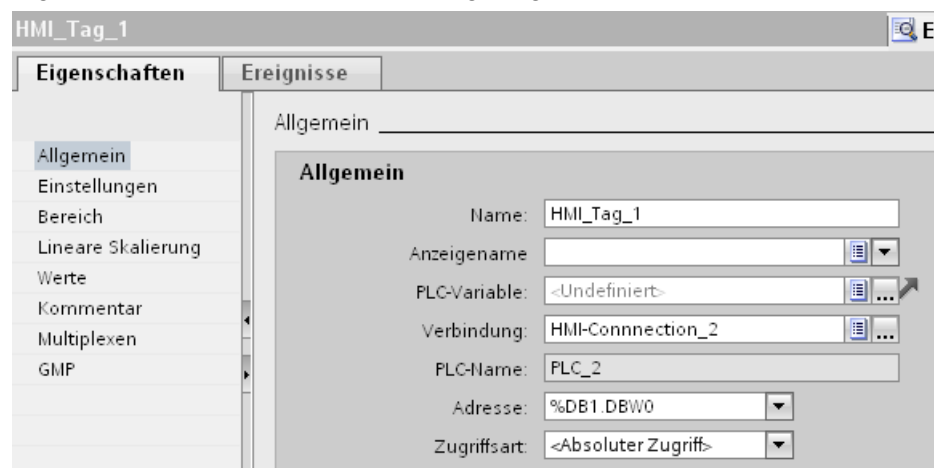
Beim Adress-Multiplexen greifen Sie mithilfe von wenigen Variablen effizient auf unterschiedliche Adressen in der Steuerung zu. An Stelle der absoluten Adresse in der Steuerung verwenden Sie Variablen, um die Adresse in Runtime verändern zu können.

### Voraussetzung

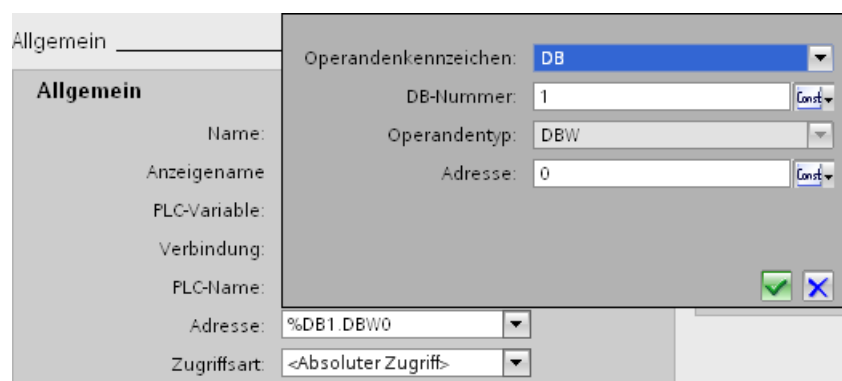
- Die Variable für das Adress-Multiplexen ist angelegt und mit der Steuerung verbunden.
- Das Eigenschaftsfenster für diese Variable ist geöffnet.

### Vorgehensweise

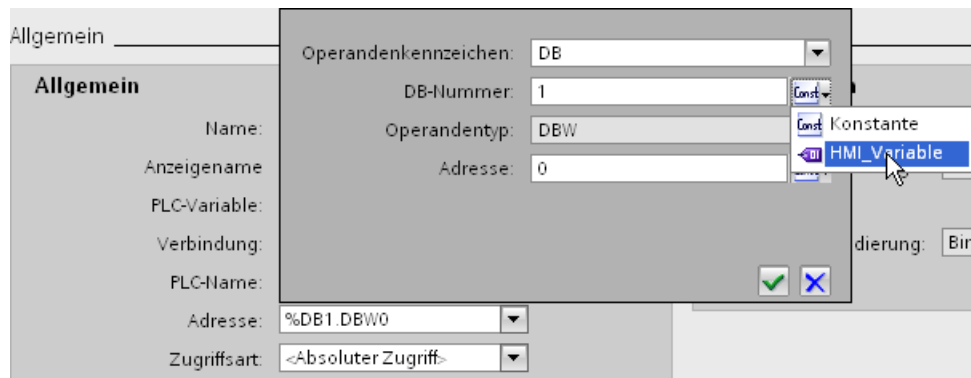
1. Selektieren Sie in der Variablen-tabelle die Variable für das Adress-Multiplexen und wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein". Die allgemeinen Eigenschaften der Variable werden angezeigt.



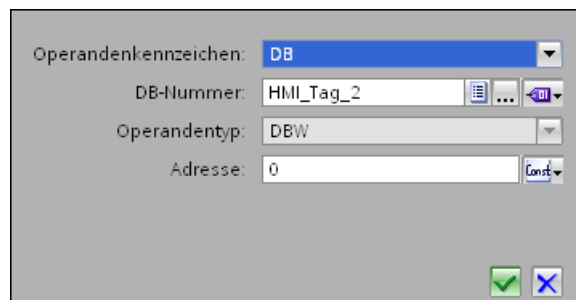
2. Wählen Sie für dieses Beispiel den Datentyp "Int".
3. Wählen Sie die Zugriffsart "Absolute Adressierung".
4. Klicken Sie auf die Auswahl-schaltfläche im Feld "Adresse". Der Adressdialog wird geöffnet.



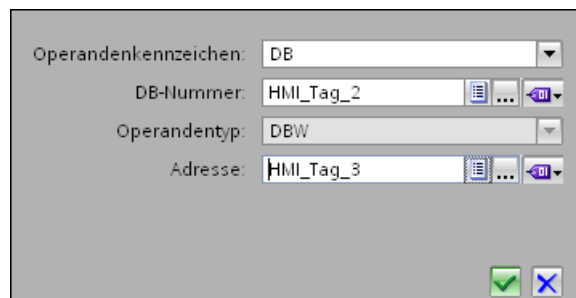
5. Klicken Sie auf die Auswahl Schaltfläche im Feld "DB-Nummer" und wählen Sie den Eintrag "HMI-Variable".



6. Klicken Sie im Feld "DB-Nummer" auf die Schaltfläche ... und wählen Sie in der Objektliste eine Variable für die DB-Nummer aus. Alternativ legen Sie mithilfe der Objektliste eine neue Variable an. Übernehmen Sie die Variable durch Klicken auf die Schaltfläche ✓.



7. Wiederholen Sie die Schritte 3 und 4 für das Feld "Adresse" und projizieren Sie eine weitere Variable für den Aufruf des Adressbereichs im Datenbaustein.



Die Auswahlmöglichkeiten im Adress-Dialog sind abhängig vom gewählten Datentyp der Multiplex-Variablen. Der Adressdialog bietet nur Adresseinstellungen an, die mit dem gewählten Datentyp projektierbar sind.

## Ergebnis

In Runtime wird mit der Multiplex-Variablen auf den Speicherbereich zugegriffen, dessen Adresse die Variable aktuell enthält. Mit der Variable im Feld DB-Nummer steuern Sie den

Zugriff auf den Datenbaustein. Mit der Variable im Feld "Adresse" steuern Sie den Zugriff auf die Adresse im gewählten Datenbaustein.

---

#### **Hinweis**

Der Wert im Speicherbereich wird erst mit dem nächsten Aktualisierungszyklus der angesprochenen Variablen gelesen.

Wenn Sie z. B. in einem Skript eine Multiplex-Variable verwenden, greifen Sie nicht unmittelbar nach Änderung des Speicherbereichs auf dessen Inhalt zu.

---

#### **Siehe auch**

Grenzwerte einer Variable (Seite 3178)

### **Adress-Multiplexen projektieren mit symbolischer Adressierung**

#### **Einleitung**

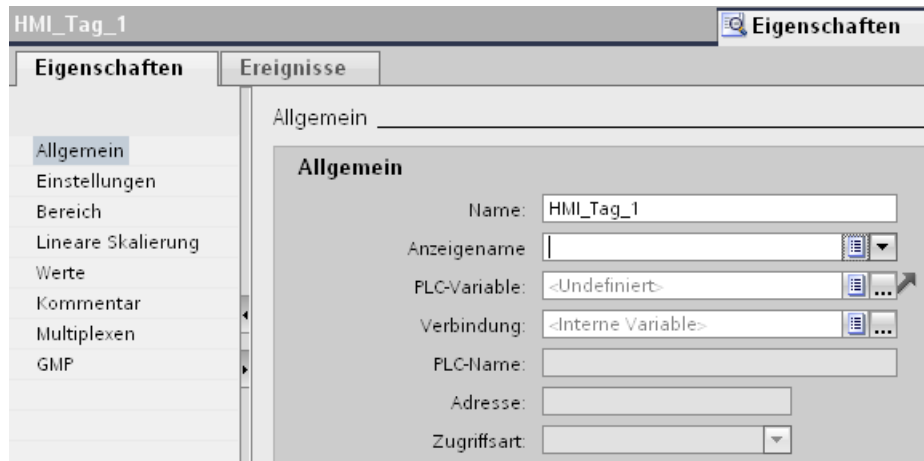
Beim Adress-Multiplexen greifen Sie mithilfe von wenigen Variablen effizient auf unterschiedliche Adressen in der Steuerung zu. An Stelle der symbolischen Adresse in der Steuerung verwenden Sie Variablen, um die Adresse in Runtime verändern zu können.

#### **Voraussetzung**

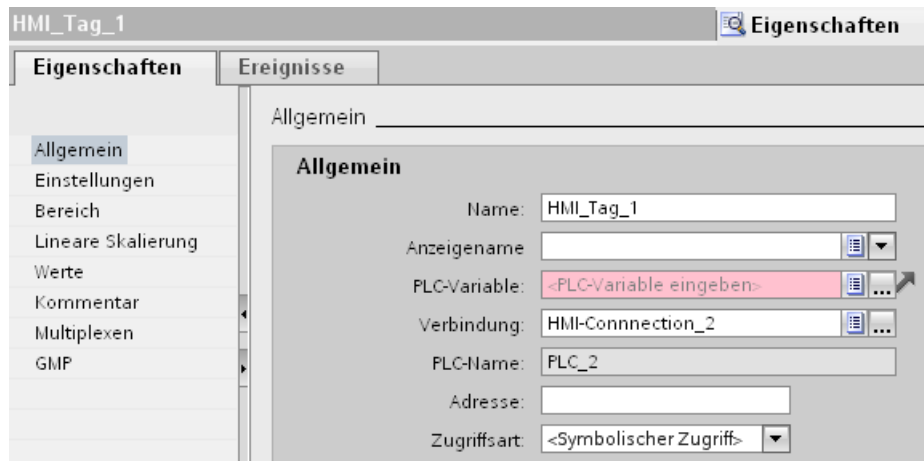
- Die Variable für das Adress-Multiplexen ist angelegt.
- Das Eigenschaftsfenster für diese Variable ist geöffnet.
- In der verbundenen Steuerung ist ein Datenbaustein mit einer Array-Variable angelegt.
- Der Datenbaustein wurde übersetzt.

### Vorgehensweise

1. Selektieren Sie in der Variablen-tabelle die Variable für das Adress-Multiplexen und wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein". Die allgemeinen Eigenschaften der Variable werden angezeigt.



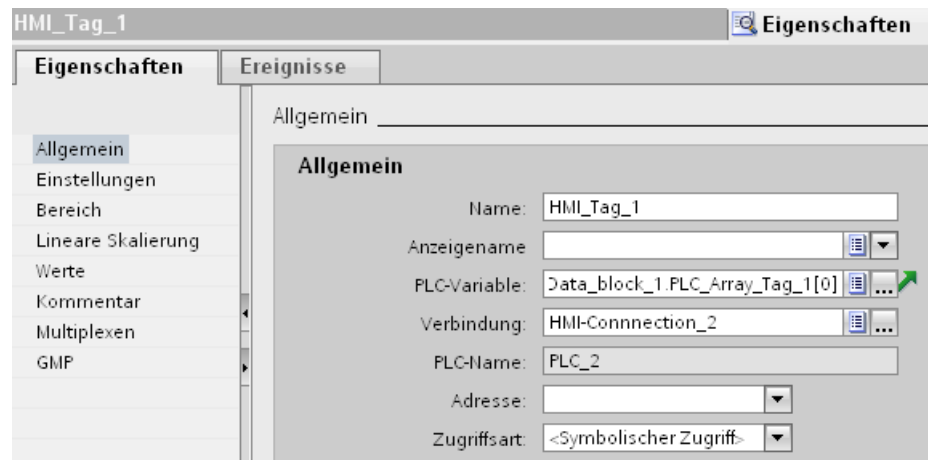
2. Wählen Sie über das Feld "Verbindung" die Verbindung zur Steuerung aus.



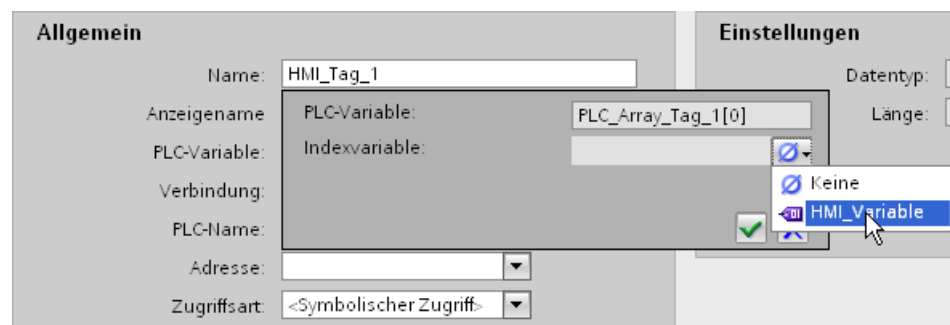
3. Wählen Sie die Zugriffsart "Symbolische Adressierung".



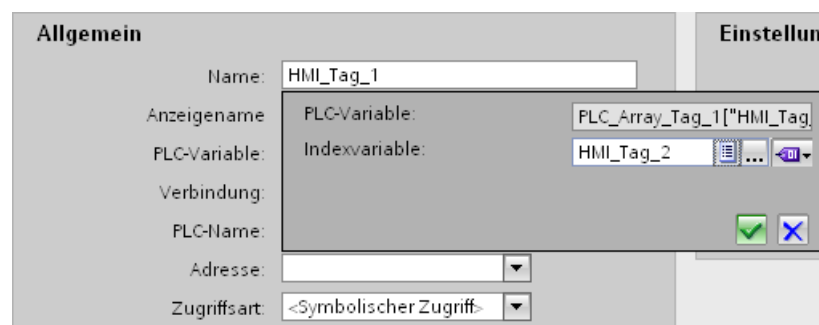
4. Navigieren Sie über das Feld "PLC-Variable" zu dem Datenbaustein der Steuerung und wählen Sie ein Array-Element der Arrayvariable aus.



5. Klicken Sie im Feld "Adresse" auf die Auswahlschaltfläche. Der Adressdialog wird geöffnet.
6. Klicken Sie im Feld "Indexvariable" auf die Auswahlschaltfläche und wählen Sie den Eintrag "HMI-Variable".



7. Klicken Sie im Feld "Indexvariable" auf die Schaltfläche ... und wählen Sie in der Objektliste eine Variable für den Array-Index aus. Alternativ legen Sie mithilfe der Objektliste eine neue Variable an. Übernehmen Sie die Variable durch Klicken auf die Schaltfläche ✓.



## Ergebnis

In Runtime wird auf das Array-Element zugegriffen, dessen Indexwert in der Indexvariable enthalten ist.

**Siehe auch**

Grenzwerte einer Variable (Seite 3178)

### 10.2.3 Arbeiten mit Arrays

#### 10.2.3.1 Grundlagen zu Arrays

**Definition**

Mit Arrays werden Daten eines einheitlichen Datentyps hintereinander angeordnet und im Adressbereich fortlaufend adressiert, sodass Sie über einen Index auf die Daten zugreifen können. Das Array verwendet zur Adressierung seiner einzelnen Arrayelemente einen ganzzahligen Index. Die Eigenschaften eines jeden Arrayelements sind gleich und werden an der Arrayvariable in einem Datenbaustein des Steuerungsprogramms projiziert.

| Default tag table |                   |                     |                        |
|-------------------|-------------------|---------------------|------------------------|
| Name ▲            | Variablen-tabelle | Datentyp            | Verbindung             |
| ▼ HMI_Tag_1       | Default tag table | Array [0..4] of Int | <Interne Variable> ... |
| ■ [0]             | Default tag table | Int                 | <Interne Variable>     |
| ■ [1]             | Default tag table | Int                 | <Interne Variable>     |
| ■ [2]             | Default tag table | Int                 | <Interne Variable>     |
| ■ [3]             | Default tag table | Int                 | <Interne Variable>     |
| ■ [4]             | Default tag table | Int                 | <Interne Variable>     |

**Vorteile**

Mit nur einer Arrayvariablen projizieren Sie auf ein Mal viele Arrayelemente mit den gleichen Eigenschaften. Jedes einzelne Arrayelement verwenden Sie dann in der Projektierung wie jede andere Variable auch.

**Einschränkungen**

Für die Verwendung von Arrays gibt es folgende Einschränkungen:

- Nicht alle Bediengeräte unterstützen Arrayvariablen.
- Ein Array darf lediglich eine Dimension enthalten.
- Der untere Index eines Arrays muss bei "0" beginnen.

## Anwendungsbeispiele


Arrayvariablen verwenden Sie in folgenden Fällen:

- Um Prozesswerte in Profilkurven zusammenzufassen: Z. B. bilden Sie Prozesswerte in Kurven ab, die zu verschiedenen Zeitpunkten erfasst wurden.
- Um gezielt auf die in Kurven zusammengefassten Prozesswerte zuzugreifen: Z. B. geben Sie alle erfassten Werte der Profilkurve aus, in dem Sie die Indexvariable schrittweise erhöhen.
- Um Bitmeldungen mit einer fortlaufenden Bitnummer zu projektieren.
- Um Maschinendatensätze als Ganzes in einer Rezeptur abzulegen.

## Lizenzregelung für Runtime


Bei WinCC Runtime wird eine Arrayvariable unabhängig von der Zahl der Arrayelemente als 1 PowerTag gezählt.

## Besonderheiten

|  |
|--|
|  <b>WARNUNG</b>  |
| <b>Erhöhte Systemauslastung und Performanceeinbußen</b>  |
| Bei einem Lesezugriff oder Schreibzugriff auf ein einzelnes Arrayelement werden immer alle Arrayelemente der Arrayvariablen gelesen oder geschrieben. Daher dauert bei großen Arrays die Datenübertragung von und zur Steuerung generell länger als bei einem elementaren Datentyp. Dies kann die Kommunikation überlasten und dadurch stören. |

### Beispiel:

- Eine Arrayvariable mit 100 Arrayelementen vom Datentyp "Real" ist projektiert.
- Wenn sich ein 4 Byte großes Arrayelement ändert, werden 100 x 4 Bytes in die Steuerung geschrieben.

|  |
|--|
|  <b>VORSICHT</b>  |
| <b>Dateninkonsistenz bei Arrayvariablen</b>  |
| Wenn in einer Arrayvariablen der Wert eines einzelnen Elements geändert werden soll, wird das ganze Array gelesen, geändert und als vollständiges Array wieder zurückgeschrieben. Änderungen an anderen Arrayelementen, die zwischenzeitlich in der Steuerung vorgenommen werden, werden beim Zurückschreiben überschrieben. |
| Achten Sie darauf, dass nicht zur selben Zeit unterschiedliche Stellen z. B. Bediengerät und Steuerung Werte in dieselbe Arrayvariable schreiben. Um die Arrayvariable mit der Steuerung zu synchronisieren, benutzen Sie z. B. die synchrone Übertragung von Rezepturdatensätzen.   |

## Siehe auch

- Arrayvariable anlegen (Seite 3198)
- Beispiele zu Arrays (Seite 3199)
- Grundlagen zu Variablen (Seite 3160)

### 10.2.3.2 Arrayvariable anlegen

#### Einleitung

Um eine große Zahl von Variablen gleichen Datentyps zu projektieren, legen Sie eine Arrayvariable an. Die einzelnen Arrayelemente werden in einem fortlaufenden Adressbereich gespeichert.

Eine Arrayvariable können Sie als interne Variable oder als externe Variable anlegen.



Wenn Sie eine Arrayvariable als externe Variable anlegen wollen, dann projektieren Sie zuerst eine Arrayvariable in einem Datenbaustein der verbundenen Steuerung. Anschließend verbinden Sie die Arrayvariable mit einer HMI-Variable.

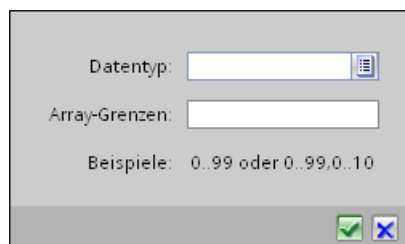
#### Voraussetzung

- Die HMI-Variablen-tabelle ist geöffnet.

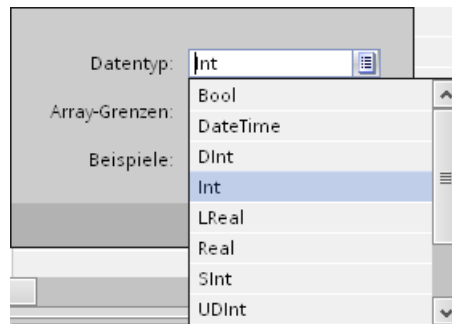
#### Vorgehensweise

Um eine Arrayvariable anzulegen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der HMI-Variablen-tabelle in der Spalte "Name" auf <Hinzufügen>. Eine neue HMI-Variable wird angelegt.
2. Klicken Sie in der Spalte Datentyp auf  und wählen Sie den Datentyp "Array".
3. Klicken Sie in der Spalte Datentyp auf . Der Dialog zur Projektierung des Arrays wird geöffnet.



4. Wählen Sie im Feld "Datentyp" den gewünschten Datentyp für die Arrayvariable.



5. Legen Sie im Feld "Array-Grenzen" die Anzahl der Arrayelemente fest. Die untere Grenze muss mit "0" beginnen.
6. Klicken Sie auf . Die Einstellungen für das Array werden übernommen.
7. Speichern Sie das Projekt.

## Ergebnis

Eine Arrayvariable ist angelegt. Die Eigenschaften der Arrayelemente werden von der übergeordneten Arrayvariablen übernommen.

## Siehe auch

Grundlagen zu Arrays (Seite 3196)

### 10.2.3.3 Beispiele zu Arrays

#### Einleitung

Arrayvariablen fassen viele Variablen, z. B. 100 Arrayelemente zusammen. Arrayvariablen verwenden Sie als vollständige Arrays an folgenden Stellen:

- Im Editor "Meldungen"
- Im Editor "Rezepturen"
- Beim Adress-Multiplexen
- In der Kurvenanzeige

Einzelne Arrayelemente verwenden Sie überall in der Projektierung, wie HMI-Variablen.

#### Beispiele

Wenn Sie viele Variablen vom selben Datentyp haben, projektieren Sie nur eine Arrayvariable mit der entsprechenden Anzahl Arrayelemente.

- Greifen Sie auf die einzelnen Arrayelemente indirekt zu, z. B. über eine Multiplex-Indexvariable.
- Über die Indexvariable bedienen und beobachten Sie die Arrayelemente.

## Siehe auch

Grundlagen zu Arrays (Seite 3196)

## 10.2.4 Arbeiten mit Zyklen

### 10.2.4.1 Grundlagen zu Zyklen

#### Einleitung

Zyklen werden verwendet, um regelmäßig wiederkehrende Aktionen in Runtime zu steuern. Klassische Anwendungen sind der Erfassungszyklus und der Aktualisierungszyklus.

#### Prinzip

In Runtime werden regelmäßig wiederkehrende Aktionen durch Zyklen gesteuert. Typische Anwendungen für Zyklen:

- Erfassung von externen Variablen  
Der Erfassungszyklus bestimmt, wann der Prozesswert einer externen Variablen vom Bediengerät aus der Steuerung ausgelesen wird. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs.  
Wählen Sie den Erfassungszyklus nicht zu klein, da sonst die Kommunikationslast des Prozesses stark ansteigt.

Der kleinste Wert für einen Zyklus hängt vom Bediengerät ab, für das Sie projektieren. Für die meisten Bediengeräte beträgt dieser Wert 100 ms. Die Werte aller weiteren Zyklen sind immer ein ganzzahliges Vielfaches des kleinsten Werts.

#### Anwendungsbeispiel

Zyklen verwenden Sie z. B. für folgende Aufgaben:

- Um eine Variable regelmäßig zu aktualisieren.
- Um auf Wartungsintervalle aufmerksam zu machen.

## Siehe auch

Grundlagen zu Variablen (Seite 3160)

## 10.2.5 Variablen darstellen

### 10.2.5.1 Ausgeben von Variablenwerten in Bildern

#### Einleitung

In Runtime können Sie Variablenwerte in den Bildern am Bediengerät in Form einer Kurve ausgeben. Eine Kurve ist die grafische Darstellung der Werte, die eine Variable in Runtime annimmt. Für die Darstellung verwenden Sie das Grafikobjekt "Kurvenanzeige". Prozesswerte für die Darstellung als Kurve werden aus dem laufenden Prozess von der Steuerung geladen.

Die darzustellenden Werte werden einzeln in einem festen, einstellbaren Zyklus ermittelt. Zyklisch getriggerte Kurven eignen sich für die Darstellung kontinuierlicher Verläufe, z. B. die Darstellung der Betriebstemperatur eines Motors.

#### Dargestellte Werte

Damit Variablenwerte am Bediengerät angezeigt werden, projizieren Sie in einem Bild eine Kurvenanzeige. Bei der Projektierung der Kurvenanzeige geben Sie an, welche Variablenwerte dargestellt werden.

Die Aktualisierung der Kurvenanzeige steuern Sie durch die Definition der Zykluszeit.

### 10.2.5.2 Kurvenanzeige für Werte aus der Steuerung projizieren

#### Einleitung

Mit einer Kurvenanzeige stellen Sie Werte grafisch dar, die eine Variable im Prozess annimmt.

#### Voraussetzung

- Ein Bild ist geöffnet.
- Das Inspektorfenster mit den Eigenschaften für die Kurvenanzeige ist geöffnet.

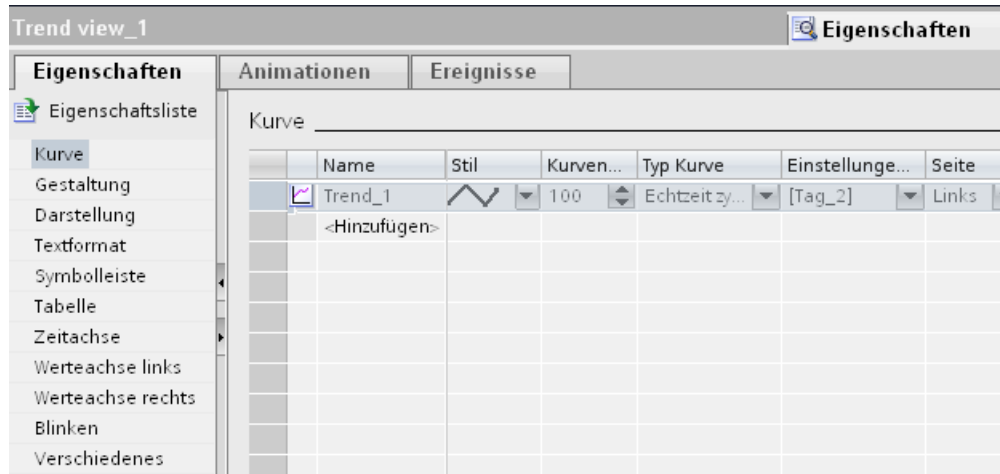
## Vorgehen

Um eine Kurvenanzeige zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus dem Werkzeugfenster aus der Gruppe "Control" das Objekt "Kurvenanzeige" in das Bild ein.



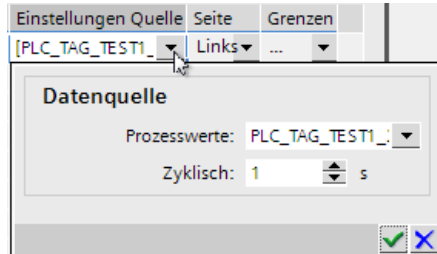
2. Selektieren Sie im Inspektorfenster in der Gruppe "Eigenschaften" die Kategorie "Kurve" und doppelklicken Sie in der Spalte "Name" den Eintrag "<Hinzufügen>".



3. Vergeben Sie in der Spalte "Name" einen Namen für die Kurve.
4. Öffnen Sie in der Spalte "Stil" über die Auswahlfläche den Dialog "Stil" und wählen Sie die Darstellung der Linie aus.
5. Wählen Sie in der Spalte "Kurvenwerte" die Anzahl der Kurvenwerte.



- Öffnen Sie in der Spalte "Einstellungen" über die Auswahlschaltfläche den Dialog "Datenquelle" und wählen Sie die Variable für die Werterversorgung der Kurve aus. Geben Sie einen Zyklus für das Lesen der Variablen aus der Steuerung an.



- Nehmen Sie weitere Einstellungen in den Dialogen des Inspektorfensters vor. Aktivieren Sie z. B. in der Kategorie "Tabelle" die Option "Tabelle anzeigen", um unter der Kurvenanzeige eine Wertetabelle anzuzeigen.

---

#### Hinweis

Wenn Sie die Taste <STRG> gedrückt halten und einen Doppelklick auf die Kurvenanzeige machen, wird die Kurvenanzeige aktiv geschaltet. Im aktiven Modus können Sie im Tabellenkopf die Spaltenbreite und die Position der Spalten der Wertetabelle einstellen. Zur Aktivierung der Kurvenanzeige muss der Zoomfaktor auf 100 % eingestellt sein.

---

## Ergebnis

In Runtime werden in der projektierten Kurvenanzeige die Werte der gewählten Variablen angezeigt.

## 10.3 Mit Meldungen arbeiten

### 10.3.1 Grundlagen

#### 10.3.1.1 Das Meldesystem in WinCC

## Einleitung

Das Meldesystem ermöglicht am Bediengerät die Anzeige von Betriebszuständen und Störungen, die in einer Anlage anstehen oder auftreten.

Eine Meldung hat z. B. folgende Inhalte:

| N r. | Uhrzeit       | Datum       | Meldetext                              | Zustand              | Meldeklasse           |
|------|---------------|-------------|--|----------------------|-----------------------|
| 5    | 12:50:24 :590 | 24.02. 2007 | Kesseldruck liegt über der Obergrenze. | Gekommen<br>Gegangen | Warnung:<br>Farbe Rot |

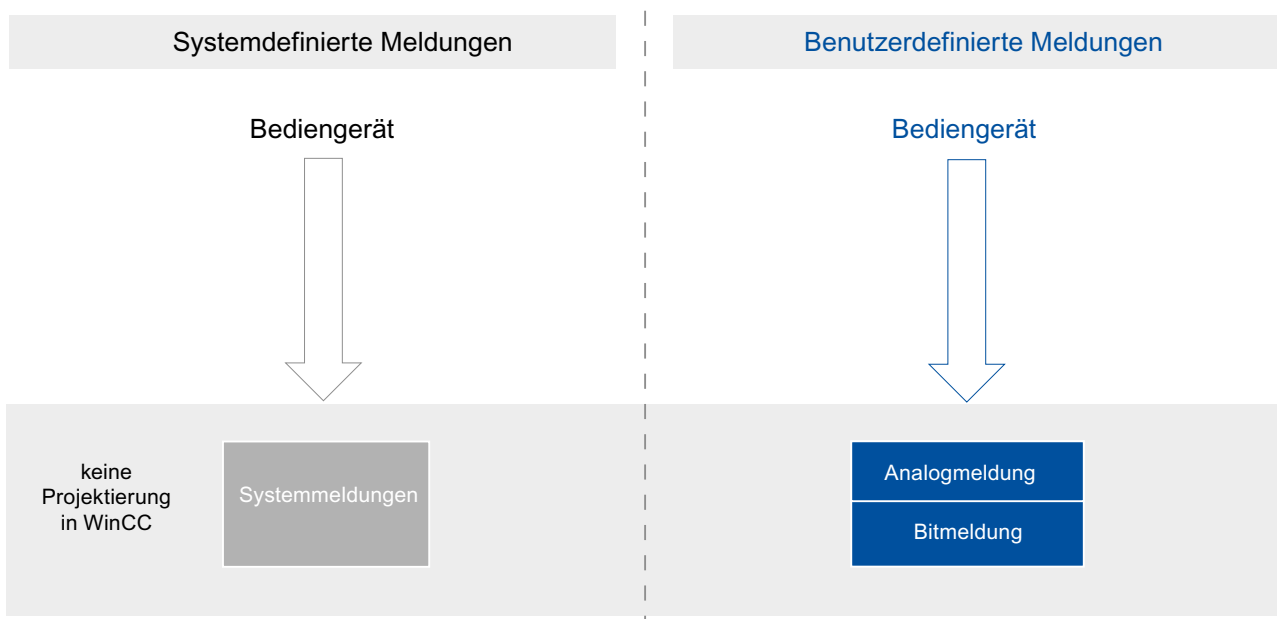
### Das Meldesystem in WinCC

Das Meldesystem verarbeitet verschiedene Meldeverfahren. Die Meldeverfahren teilen sich auf in systemdefinierte Meldungen und benutzerdefinierte Meldungen:

- Benutzerdefinierte Meldungen dienen zur Überwachung des Anlagenprozesses.
- Systemdefinierte Meldungen dienen zur Überwachung des Bediengeräts.

Die erfassten Meldeereignisse werden am Bediengerät angezeigt. Gezielte Zugriffe auf die Meldungen und ergänzende Informationen zu einzelnen Meldungen gewährleisten eine zügige Störungslokalisierung und Störungsbehebung. Stillstandszeiten werden verringert oder ganz vermieden.

Das folgende Bild zeigt den Aufbau des Meldesystems:



### 10.3.1.2 Meldeverfahren

#### Übersicht Meldeverfahren

##### Einleitung

Die Meldeverfahren dienen verschiedenen Zwecken zur Überwachung der Anlage. Die Meldungen der einzelnen Meldeverfahren werden auf unterschiedliche Arten projiziert und ausgelöst.

Sie projizieren Meldungen nach den einzelnen Meldeverfahren in der jeweiligen Registerkarte im Editor "HMI-Meldungen".

##### Meldeverfahren in WinCC

WinCC unterstützt folgende Meldeverfahren:

##### Benutzerdefinierte Meldungen

- **Analogmeldungen**
  - Analogmeldungen dienen zur Grenzwertüberwachung.
- **Bitmeldungen**
  - Bitmeldungen dienen zur Zustandsüberwachung.

##### Systemdefinierte Meldungen

- **Systemmeldungen**
  - Systemmeldungen gehören zum jeweiligen Bediengerät und werden in das Projekt importiert.
  - Systemmeldungen dienen zur Überwachung des Bediengeräts.

##### Systemdefinierte Meldungen

##### Systemmeldungen

##### Beispiele für Meldungen

- "Online-Verbindung zur Steuerung wurde erfolgreich aufgebaut."

##### Beschreibung

Eine Systemmeldung zeigt den Status des Systems an und Kommunikationsfehler zwischen Bediengerät und System.

Unter "Runtime-Einstellungen > Meldungen" legen Sie fest, wie lange eine Systemmeldung am Bediengerät angezeigt wird.

## Support

Sie finden in der Referenz eine Liste der möglichen Systemmeldungen mit Angabe der Ursache und Möglichkeiten zur Abhilfe. Wenn Sie sich wegen einer Systemmeldung des Bediengeräts an den Online Support wenden, benötigen Sie die Meldenummer und die vorhandenen Variablen der Systemmeldung.

## Benutzerdefinierte Meldungen

### Analogmeldungen

#### Beschreibung

Analogmeldungen zeigen Grenzwertverletzungen im laufenden Prozess an.

#### Beispiel

Die Drehzahl des Mixers in einer Fruchtsaftmischanlage darf nicht zu hoch und nicht zu niedrig sein. Um die Drehzahl des Mixers zu überwachen, projektieren Sie die entsprechenden Analogmeldungen. Wenn der Grenzwert für die Drehzahl des Mixers unter- oder überschritten wird, dann wird auf dem Bediengerät eine Meldung ausgegeben, die z. B. folgenden Meldetext enthält: "Drehzahl des Mixers zu niedrig".

### Bitmeldungen

#### Beschreibung

Bitmeldungen zeigen einen Zustand im laufenden Prozess an.

#### Beispiel

Eine Fruchtsaftmischanlage hat mehrere Tanks mit den Zutaten. Für das gewünschte Mischungsverhältnis von Wasser, Fruchtkonzentrat, Zucker und Aroma müssen sich die Ventile in den Zuleitungen im richtigen Moment schließen und öffnen. Dieser Vorgang soll überwacht werden.

Dazu projektieren Sie für alle Zustände der Ventile eine entsprechende Bitmeldung. Wenn sich ein Ventil der vier Tanks öffnet oder schließt, wird dann eine Meldung angezeigt, z. B. "Ventil Wasser geschlossen".

Der Benutzer überwacht so, ob die Anlage korrekt produziert.

### 10.3.1.3 Meldezustände

#### Einleitung

Eine Meldung nimmt in Runtime verschiedene Meldezustände an. Anhand der Meldezustände analysiert und protokolliert der Benutzer den Verlauf des Prozesses.

#### Beschreibung

Jede Meldung befindet sich in einem Meldezustand. Die Meldezustände setzen sich aus folgenden Ereignissen zusammen:

- **Gekommen**  
Die Bedingung zum Auslösen einer Meldung trifft zu. Die Meldung wird angezeigt, z. B. "Kesseldruck zu hoch".
- **Gegangen**  
Die Bedingung zum Auslösen einer Meldung trifft nicht mehr zu. Die Meldung wird nicht mehr angezeigt, weil der Kessel entlüftet wurde.
- **Quittieren**  
Der Benutzer quittiert die Meldung.

#### Meldungen ohne Quittierpflicht

Die folgende Tabelle zeigt die Meldezustände für Meldungen ohne Quittierpflicht:

| Zustand  | Beschreibung                                      |
|----------|---|
| Gekommen | Die Bedingung einer Meldung trifft zu.            |
| Gegangen | Die Bedingung einer Meldung trifft nicht mehr zu. |

#### Meldungen mit Quittierpflicht

Die folgende Tabelle zeigt die Meldezustände für Meldungen mit Quittierpflicht:

| Zustand                      | Beschreibung   |
|------------------------------|--|
| Gekommen                     | Die Bedingung einer Meldung trifft zu  |
| Gegangen,<br>nicht quittiert | Die Bedingung einer Meldung trifft nicht mehr zu. Der Benutzer hat die Meldung nicht quittiert.                          |
| Gegangen,<br>dann quittiert  | Die Bedingung einer Meldung trifft nicht mehr zu. Der Benutzer hat die Meldung nach diesem Zeitpunkt quittiert.          |
| Gekommen,<br>quittiert       | Die Bedingung einer Meldung trifft zu. Der Benutzer hat die Meldung quittiert.   |
| Gegangen,<br>zuvor quittiert | Die Bedingung einer Meldung trifft nicht mehr zu. Der Benutzer hat die Meldung quittiert, als die Bedingung noch zutraf. |

Jedes Eintreten dieser Zustände wird bei Bedarf am Bediengerät angezeigt.

---

**Hinweis**

Der Anzeigetext für die Zustände einer Meldung ist sprach- und projektierungsabhängig.

---

### 10.3.1.4 Meldeklassen

## Grundlagen zu Meldeklassen

### Einleitung

In einer Anlage treten viele Meldungen von unterschiedlicher Wichtigkeit auf. Um dem Benutzer klar ersichtlich zu machen, welche Meldungen am wichtigsten sind, ordnen Sie die Meldungen Ihres Projekts in Meldeklassen ein.

### Beschreibung

Die Meldeklasse legt die Darstellung einer Meldung fest. Die Meldeklasse legt fest, ob und wie der Benutzer die Meldungen dieser Meldeklasse quittieren muss.

In WinCC wird eine neue Meldeklasse als quittierpflichtige Meldeklasse angelegt.

---

**Hinweis**

Die Auswahl der Darstellungsarten für Meldeklassen ist abhängig von den Möglichkeiten Ihres Bediengeräts.

---

## Beispiele zur Anwendung von Meldeklassen

- Die Meldung "Drehzahl Lüfter 1 im oberen Toleranzbereich" hat die Meldeklasse "Warnings". Die Meldung wird mit weißem Hintergrund dargestellt. Die Meldung ist nicht quittierpflichtig.
- Die Meldung "Drehzahl Lüfter 2 hat oberen Warnbereich überschritten" ist der Meldeklasse "Errors" zugeordnet. Die Meldung wird mit rotem Hintergrund dargestellt und blinkt in Runtime mit einer hohen Frequenz. Die Meldung wird so lange angezeigt, bis der Benutzer die Meldung quittiert hat.

## Meldeklassen verwenden

Um das Quittierkonzept und die Darstellung der Meldungen für Ihr Projekt festzulegen, nutzen Sie folgende Meldeklassen:

- **Vordefinierte Meldeklassen**  
Vordefinierte Meldeklassen können Sie nicht löschen und nur eingeschränkt bearbeiten. Für jedes Bediengerät sind vordefinierte Meldeklassen unter "HMI-Meldungen > Meldeklassen" angelegt.
- **Benutzerdefinierte Meldeklassen**  
Sie legen Meldeklassen unter "HMI-Meldungen > Meldeklassen" neu an und projektieren die gewünschte Darstellung von Meldungen und ein Quittierkonzept für die Meldungen dieser Meldekategorie. Die mögliche Anzahl benutzerdefinierter Meldeklassen ist abhängig von der in Ihrem Projekt verwendeten Runtime.

## Siehe auch

Meldeklassen anlegen (Seite 3215)

## Vordefinierte Meldeklassen

## Vordefinierte Meldeklassen

Die folgenden Meldeklassen sind in WinCC für jedes Bediengerät bereits angelegt:

## Meldeklassen für benutzerdefinierte Meldungen

- **"Warnings"**  
Die Meldekategorie "Warnings" ist dazu vorgesehen, reguläre Zustände und Verläufe im Prozess anzuzeigen. Meldungen dieser Meldekategorie quittiert der Benutzer nicht.
- **"Errors"**  
Die Meldekategorie "Errors" ist dazu vorgesehen, kritische oder gefährliche Zustände oder Grenzwertverletzungen im Prozess anzuzeigen. Meldungen dieser Meldekategorie muss der Benutzer quittieren.

## Meldekategorie für systemdefinierte Meldungen

- **"System"**  
Die Meldekategorie "System" enthält Meldungen, die Zustände des Bediengeräts und der Steuerungen anzeigen.

## Siehe auch

Meldeklassen anlegen (Seite 3215)

### 10.3.1.5 Quittierung

#### Quittieren von Meldungen

##### Einleitung

Um sicherzustellen, dass eine Meldung vom Benutzer einer Anlage registriert wurde, projektieren Sie diese Meldung so, dass sie so lange angezeigt wird, bis sie der Benutzer quittiert hat. Meldungen, die kritische oder gefährliche Zustände im Prozess anzeigen, müssen quittierpflichtig sein.

##### Beschreibung

Durch die Quittierung einer Meldung ändert sich der Meldezustand einer Meldung von "Gekommen" auf "Quittiert". Mit der Quittierung einer Meldung bestätigt der Benutzer die Bearbeitung des Zustandes, der die Meldung ausgelöst hat.

##### Quittierung einer Meldung triggern

Die Quittierung einer Meldung triggert der Bediener in Runtime auf verschiedene Arten:

- Quittieren durch den berechtigten Benutzer am Bediengerät
- Automatische Quittierung durch das System ohne Bedienaktion, z. B. über
  - Variablen
  - Steuerung
  - Systemfunktionen in Funktionslisten

##### Zusammengehörige Meldungen gemeinsam quittieren

Um das Meldesystem in Runtime übersichtlich zu gestalten und einfach bedienbar zu machen, projektieren Sie bei Bedarf eine Meldegruppe. Der Benutzer quittiert mit einer Bedienaktion alle Meldungen, die der Meldegruppe angehören.

##### Quittieren durch die Steuerung

Bitmeldungen werden bei Bedarf automatisch von der Steuerung quittiert. Die Quittierung wird von einem Bit der "Quittiervariable Steuerung" ausgelöst. Bit und Variable legen Sie bei der Projektierung fest.



## Quittieren einer Meldung am Bediengerät

Der Benutzer quittiert in Runtime eine Meldung abhängig von der Projektierung auf folgende Arten:

- Über die Quittiertaste <ACK> auf dem Bediengerät
- Über die Schaltfläche der Meldeanzeige
- Über projektierte Funktionstasten oder Schaltflächen in Bildern

---

### Hinweis

#### Quittiertaste <ACK> am Bediengerät

Um kritische Meldungen nur von berechtigten Benutzern bearbeitet zu lassen, schützen Sie die Taste "ACK" der Bediengeräte und die Bedien- und Anzeigeobjekte der Meldungen. Verwenden Sie dazu einer entsprechende Bedienberechtigung.

---

### Hinweis

#### Bediengeräteabhängigkeit

Die Quittiertaste <ACK> steht nicht auf allen Bediengeräten zur Verfügung.

---

## Quittierkonzept

### Übersicht

Das Quittierkonzept legen Sie für eine Meldeklasse fest. Meldungen, die dieser Meldeklasse zugeordnet sind, werden nach diesem Quittierkonzept quittiert. In WinCC gibt es folgende Quittierkonzepte:

- Meldung ohne Quittierung  
Diese Meldung kommt und geht, ohne dass sie eine Quittierung erfordert. Das System zeigt keine Reaktion.
- Meldung mit Einfachquittierung  
Diese Meldung erfordert eine Quittierung, sobald das Ereignis, das die Meldung auslöst, eingetreten ist. Die Meldung steht an bis sie quittiert ist.

### 10.3.1.6 Meldegruppen

#### Einleitung

In einer Anlage treten viele Meldungen aus unterschiedlichen Bereichen und Prozessen auf. In Meldegruppen fassen Sie zusammengehörige Meldungen zusammen.

#### Meldegruppen

Mithilfe der Meldegruppen beobachten Sie Teile der Anlage und quittieren die dazugehörigen Meldungen bei Bedarf gemeinsam.

Meldegruppen enthalten bei Bedarf Meldungen verschiedener Meldeklassen. Sie ordnen nur quittierpflichtige Meldungen einer Meldegruppe zu.

### Meldegruppen verwenden

Die Zusammenstellung von Meldegruppen ist z. B. für folgende Meldungen sinnvoll:

- Meldungen, welche die gleiche Störung als Ursache haben.
- Gleichartige Meldungen
- Meldungen aus einer Maschineneinheit, z. B. "Störung Antrieb XY"
- Meldungen aus einem zusammengehörigen Teil des Prozesses, z. B. "Störung Kühlwasserversorgung"

### Darstellung in Runtime

In Runtime wird in der Spalte "Meldegruppe" die Nummer der Meldegruppe angezeigt, zu der die Meldung gehört.

#### 10.3.1.7 Meldenummer

### Meldenummernvergabe

Das System vergibt innerhalb eines Meldeverfahrens eindeutige Meldenummern.

---

#### Hinweis

Wenn Sie Meldenummern anpassen, achten Sie auf die Eindeutigkeit der Meldenummer innerhalb eines Meldeverfahrens.

---

## 10.3.2 Arbeiten mit Meldungen

### 10.3.2.1 Bestandteile und Eigenschaften von Meldungen

#### Übersicht

In WinCC projektieren Sie die Bestandteile von Meldungen. Die grundlegenden Bestandteile von Meldungen zeigt die folgende Tabelle:

| Meldeklasse | Meldenummer | Uhrzeit  | Datum      | Meldezustand | Meldetext                           | Meldegruppe | Tooltip               | Triggervariable | Grenzwert |
|-------------|-------------|----------|------------|--------------|-------------------------------------|-------------|-----------------------|-----------------|-----------|
| Warning     | 1           | 11:09:14 | 06.08.2007 | KG           | Maximale Drehzahl erreicht.         | 2           | Diese Meldung ist.... | speed_1         | 27        |
| System      | 110001      | 11:25:58 | 06.08.2007 | K            | Wechsel in die Betriebsart "Online" | 0           | Diese Meldung ist.... | PLC-Variable_1  | –         |

#### Meldeklasse

Meldeklassen sind z. B. "Warnings" oder "Errors". Die Meldeklasse legt für eine Meldung Folgendes fest:

- Quittierkonzept
- Darstellung in Runtime (z. B. Farbe)

#### Meldenummer

Eine Meldung wird über eine eindeutige Meldenummer identifiziert. Die Meldenummer wird vom System vergeben. Die Meldenummer ändern Sie bei Bedarf, um z. B. Meldungen, die in Ihrem Projekt zusammengehören, mit einer fortlaufenden Meldenummer zu kennzeichnen.

#### Uhrzeit und Datum

Jede Meldung zeigt in einem Zeitstempel die Uhrzeit und Datum, zu der die Meldung ausgelöst wurde.

#### Meldezustand

Eine Meldung hat die Ereignisse "Gekommen", "Gegangen", "Quittieren". Für jedes Ereignis wird eine neue Meldung ausgegeben mit dem aktuellen Zustand der Meldung.

#### Meldetext

Der Meldetext beschreibt die Ursache der Meldung.

Der Meldetext kann Ausgabefelder für aktuelle Werte enthalten. Welche Werte Sie einfügen können, ist abhängig von der verwendeten Runtime. Der Wert wird zu dem Zeitpunkt festgehalten, zu dem sich der Meldezustand ändert.

### **Meldegruppe**

Die Meldegruppe fasst einzelne Meldungen zusammen.

### **Tooltip**

Sie projektieren bei Bedarf zu jeder Meldung einen eigenen Tooltip, den der Benutzer in Runtime anzeigen lassen kann.

### **Triggervariable**

Jeder Meldung ist eine Variable als Auslöser zugewiesen. Die Meldung wird ausgegeben, wenn diese Triggervariable die festgelegte Bedingung erfüllt, z. B. wenn sie den Zustand ändert oder einen Grenzwert überschreitet.

### **Grenzwert**

Analogmeldungen zeigen Grenzwertverletzungen an. Je nach Konfiguration gibt WinCC die Analogmeldung aus sobald die Triggervariable den konfigurierten Grenzwert z. B. überschreitet oder unterschreitet.

## **10.3.2.2 Projektieren von Meldungen**

### **Übersicht über das Projektieren von Meldungen**

#### **Arbeitsschritte zur Projektierung von Meldungen**

Sie projektieren Meldungen in WinCC in folgenden Arbeitsschritten:

1. Meldeklassen bearbeiten und erstellen  
Sie legen über die Meldeklasse die Darstellung einer Meldung in Runtime und das Quittierkonzept einer Meldung fest.
2. Im Editor "HMI-Variablen" Variablen anlegen
  - Sie projektieren die Variablen für Ihr Projekt.
  - Sie legen für die Variablen Bereichswerte an.

3. Im Editor "HMI-Meldungen" Meldungen anlegen
  - Sie legen benutzerdefinierte Meldungen an und ordnen Ihnen die zu überwachende Variable, Meldeklassen, Meldegruppen und weitere Eigenschaften zu.
  - Bei Bedarf ordnen Sie den Ereignissen von Meldungen Systemfunktionen oder Skripte zu.
4. Ausgabe von Meldungen projektieren  
Um die projektieren Meldungen auszugeben, projektieren Sie im Editor "Bilder" eine Meldeanzeige oder ein Meldefenster.

### Weitergehende Projektierungsschritte

Abhängig von den Anforderungen Ihres Projekts sind weitere Arbeitsschritte zum Projektieren von Meldungen erforderlich:

1. Meldegruppen anlegen  
In Meldegruppen ordnen Sie die Meldungen Ihres Projekts nach Zusammengehörigkeit, z. B. nach Fehlerursache (z. B. Stromausfall) oder Fehlerquelle (z. B. Motor 1).
2. Loop-In-Alarm projektieren  
Um nach dem Eingang einer Meldung zu dem Bild zu wechseln, das Informationen zu dem aufgetretenen Fehler enthält, wird ein Loop-In-Alarm projektiert.

### Meldeklassen anlegen

#### Einleitung

Meldeklassen legen Sie im Editor "HMI-Meldungen" auf der Registerkarte "Meldeklassen" an. Standardmäßig sind für jedes Projekt einige Meldeklassen bereits angelegt. Nach Bedarf legen Sie weitere benutzerdefinierte Meldeklassen an. Insgesamt legen Sie maximal 32 Meldeklassen an.

#### Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

| Meldeklassen |             |                  |                            |               |                |            |
|--------------|-------------|------------------|----------------------------|---------------|----------------|------------|
|              | Anzeigename | Name ▲           | Quittierkonzept            | Archiv        | E-Mail-Adresse | Hintergr.. |
| Um           | S7          | Diagnosis events | Meldung ohne Quittieru...  | <Kein Archiv> |                | 255...     |
|              | !           | Errors           | Meldung mit Einfachquit... | <Kein Archiv> |                | 255...     |
| 1.           | \$          | System           | Meldung ohne Quittieru...  | <Kein Archiv> |                | 255...     |
|              |             | Warnings         | Meldung ohne Quittieru...  | <Kein Archiv> |                | ...        |
|              |             | <Hinzufügen>     |                            |               |                |            |

**Vorgehen**

1. Doppelklicken Sie in der Tabelle auf "<Hinzufügen>".  
 Eine neue Meldeklasse wird angelegt. Jeder neuen Meldeklasse wird automatisch eine feste ID zugewiesen.  
 Die Eigenschaften der neuen Meldeklasse werden im Inspektorfenster angezeigt.
2. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldeklasse
  - Geben Sie einen "Namen" und den "Anzeigenamen" ein.
  - Abhängig vom Bediengerät aktivieren Sie bei Bedarf auch die Archivierung oder den automatischen Versand von E-Mails.
3. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung" das Quittierkonzept der Meldeklasse fest.
4. Ändern Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Zustand" den voreingestellten Text.  
 Mit diesem Text wird in Runtime der Zustand einer Meldung angezeigt.
5. Ändern Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Farben" die voreingestellten Farben. Abhängig vom Bediengerät ändern Sie bei Bedarf auch das Blinkverhalten.

Mit diesen Einstellungen legen Sie fest, wie die Meldungen dieser Meldeklasse in Runtime dargestellt werden.

**Hinweis**

Um Meldeklassen in Runtime farbig darzustellen, muss die Option "Farben Meldeklassen verwenden" aktiviert sein. Aktivieren Sie dazu in der Projektnavigation "Runtime-Einstellungen > Meldungen > Allgemein > Farben Meldeklassen verwenden". In einem neuen Projekt in WinCC ist diese Option aktiviert.

**Meldegruppen projektieren**

**Einleitung**

Im Editor "HMI-Meldungen" in der Registerkarte "Meldegruppen" legen Sie Meldegruppen an. Die Meldegruppe ist eine Zusammenfassung einzelner Meldungen. Sie ordnen Meldungen in einer Meldegruppe nach Zusammengehörigkeit, z. B. Fehlerursache oder Fehlerquelle. Wenn Sie in Runtime eine Meldung dieser Meldegruppe quittieren, werden alle Meldungen der Meldegruppe automatisch auch quittiert.

## Voraussetzung

- Ein Projekt ist angelegt.



## Anlegen einer neuen Meldegruppe

1.

| Meldegruppen  |    |
|---------------|----|
| Name ▲        | ID |
| Meldegruppe_1 | 17 |
| <Hinzufügen>  |    |
|               |    |
|               |    |
|               |    |
|               |    |
|               |    |

2. Doppelklicken Sie in der Tabelle im Arbeitsbereich in der ersten freien Zeile auf "<Hinzufügen>".  
Eine neue Meldegruppe wird angelegt.
3. Überschreiben Sie bei Bedarf den vorgeschlagenen "Namen".

## Ergebnis

Eine Meldegruppe ist angelegt. Um Meldungen in Runtime gemeinsam zu quittieren, ordnen Sie zusammengehörige, quittierpflichtige Meldungen einer Meldegruppe zu.

## Bitmeldungen projektieren

### Einleitung

Bitmeldungen zeigen Zustandsänderungen in einer Anlage an und werden von der Steuerung getriggert. Sie zeigen z. B. an, ob ein Ventil geöffnet oder geschlossen ist.

Im Folgenden ist die Projektierung im Editor "HMI-Meldungen" beschrieben. Alternativ projektieren Sie Bitmeldungen im Editor "HMI-Variablen".

## Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Die benötigten Meldeklassen und Meldegruppen sind angelegt.

## Vorgehen

Um eine Bitmeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Registerkarte "Bitmeldungen".
2. Um eine neue Bitmeldung anzulegen, doppelklicken Sie im Arbeitsbereich auf "<Hinzufügen>".  
Eine neue Bitmeldung wird angelegt.

3. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldung:
  - Geben Sie als Ereignistext den gewünschten Meldetext ein. Nutzen Sie die Funktionen des Kontextmenüs, um den Text zeichenweise zu formatieren und um Ausgabefelder für HMI-Variablen oder Texte aus Textlisten einzufügen.
  - Ändern Sie bei Bedarf die Meldenummer der Meldung.
  - Wählen Sie die Meldeklasse und bei Bedarf die Meldegruppe.
4. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger" die Variable und das Bit, durch das die Meldung getriggert wird. Beachten Sie dabei Folgendes:
  - Wenn Sie eine HMI Variable wählen, verwenden Sie die Datentypen "Int" oder "UInt".
  - Wenn Sie eine PLC Variable wählen, verwenden Sie die Datentypen "Int" oder "Word".
  - Verwenden Sie ein Bit einer Triggervariablen nur für eine Meldung.
  - Verwenden Sie eine Triggervariable ausschließlich als Triggervariable.
  - Wenn Sie die Meldung über die Steuerung quittieren lassen, verwenden Sie diese Variable auch als PLC Quittivariable.

---

**Hinweis**

Beachten Sie bei der Angabe des Bits die Zählweise der Bits in der verwendeten Steuerung. Weitere Hinweise dazu finden Sie in der Online-Hilfe zu der jeweiligen Steuerung im Abschnitt "Kommunikation".

---

**Hinweis**

Wenn in der Auswahl das gewünschte Objekt noch nicht existiert, legen Sie es direkt in der Objektliste an und ändern dessen Eigenschaften später.

---

### Zustandsabhängige Meldetexte

Um abhängig vom Zustand der Meldung einen anderen Text anzuzeigen, binden Sie in den Meldetext eine Textliste ein. Die Textliste steuern Sie über eine Variable.



## Weitere Einstellungen für Bitmeldungen

### Tooltip erstellen

Um die Meldung mit einem Tooltip auszustatten, gehen Sie folgendermaßen vor:

- Geben Sie unter "Eigenschaften > Eigenschaften > Tooltip" den gewünschten Text ein.

### Ereignisgesteuerte Aufgaben projektieren

Um ereignisgesteuerte Aufgaben wie einen Loop-In-Alarm zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Bitmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse" und projektieren Sie für das gewünschte Ereignis eine Funktionsliste.

### Siehe auch

Loop-In-Alarm projektieren (Seite 3223)

## Analogmeldungen projektieren

### Einleitung

Analogmeldungen zeigen Grenzwertverletzungen an. Wenn z. B. die Drehzahl eines Motors unter einen bestimmten Wert sinkt, wird eine Analogmeldung getriggert.

### Voraussetzung

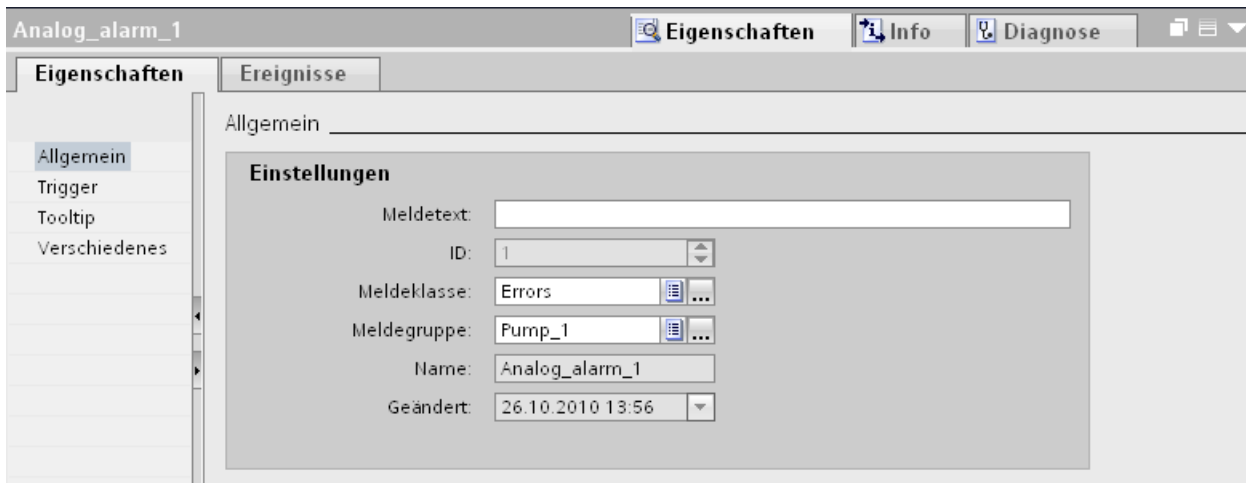
- Der Editor "HMI-Meldungen" ist geöffnet.
- Das Inspektorfenster ist geöffnet.
- Die benötigten Meldeklassen und Meldegruppen sind angelegt.

### Vorgehen

Um eine Analogmeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Analogmeldungen".
2. Um eine neue Analogmeldung anzulegen, doppelklicken Sie in der Tabelle auf "<Hinzufügen>".  
Eine neue Analogmeldung wird angelegt.

3. Konfigurieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldung:
  - Geben Sie als Ereignistext den gewünschten Meldetext ein.  
Über das Kontextmenü formatieren Sie bei Bedarf den Text zeichenweise.  
Über das Kontextmenü fügen Sie bei Bedarf Ausgabefelder für HMI-Variablen oder Texte aus Textlisten ein.
  - Ändern Sie bei Bedarf die Meldenummer der Meldung.
  - Wählen Sie die Meldeklasse und bei Bedarf die Meldegruppe.



4. Konfigurieren Sie unter "Eigenschaften > Eigenschaften > Trigger > Einstellungen" die Variable, welche die Meldung auslöst.  
Verwenden Sie eine Triggervariable ausschließlich als Triggervariable.

### Grenzwerte für Analogmeldung projektieren

1. Klicken Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger > Grenze > Wert" auf die Schaltfläche .
  - Um eine Konstante als Grenzwert zu verwenden, wählen Sie "Konstante".  
Geben Sie den gewünschten Grenzwert ein.
  - Um eine Variable als Grenzwert zu verwenden, wählen Sie "HMI\_Variable".  
Die Schaltfläche wird eingeblendet. Wählen Sie damit die Variable.

---

#### Hinweis

Wenn in der Auswahl die gewünschte Variable noch nicht existiert, legen Sie diese in der Objektliste an und ändern Sie die Eigenschaften später.

---

2. Wählen Sie den Modus:
  - "Überschreitung": Die Meldung wird bei Überschreitung des Grenzwerts getriggert.
  - "Unterschreitung": Die Meldung wird bei Unterschreitung des Grenzwerts getriggert.

## Optionale Einstellungen für Analogmeldungen

### Verzögerungszeit einstellen

Um die Verzögerungszeit einzustellen, gehen Sie folgendermaßen vor:

- Geben Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Trigger > Einstellungen > Verzögerung" eine Zeitspanne ein.  
Die Meldung wird nur dann ausgelöst, wenn die Triggerbedingung nach Ablauf der Verzögerungszeit noch ansteht.

### Totzone einstellen

---

#### Hinweis

Wenn ein Prozesswert um den Grenzwert pendelt, wird für diese Störung die zugehörige Meldung mehrmals getriggert. Um dies zu verhindern, projektieren Sie eine Totzone oder eine Verzögerungszeit.

---

Um die Totzone einzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie unter "Eigenschaften > Eigenschaften > Trigger > Totzone > Modus" bei welcher Änderung des Meldezustandes die Totzone berücksichtigt wird.
2. Geben Sie unter "Wert" einen konstanten Wert ein.
3. Um festzulegen, dass der Wert der Totzone als Prozentangabe auf den Grenzwert bezogen wird, aktivieren Sie die Option "in %".

### Tooltip erstellen

Um die Meldung mit einem Tooltip auszustatten, gehen Sie folgendermaßen vor:

- Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Tooltip" und geben Sie den gewünschten Text ein.

### Ereignisgesteuerte Aufgaben projektieren

Um ereignisgesteuerte Aufgaben wie einen Loop-In-Alarm zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Analogmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse" und projektieren Sie für das gewünschte Ereignis eine Funktionsliste.

### Siehe auch

Loop-In-Alarm projektieren (Seite 3223)

## Ausgabefeld in Meldetext einfügen

### Einleitung


In WinCC fügen Sie in den Meldetext Ausgabefelder ein, die den Inhalt von Variablen anzeigen.

### Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die Meldung ist markiert.

### Wert einer Variablen im Meldetext ausgeben

Um ein Ausgabefeld für den Wert einer Variablen in den Meldetext einzufügen, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Cursor im Ereignistext an der gewünschten Stelle.
2. Wählen Sie im Kontextmenü "Variablenausgabefeld einfügen".
3. Öffnen Sie unter "Variable" die Objektliste und wählen Sie die gewünschte Variable aus. Bei Bedarf können Sie die Variable in der Objektliste anlegen.
4. Legen Sie unter "Format" die Länge des Ausgabefeldes und das Format fest, in dem der Wert der Variable im Meldetext ausgegeben wird. Projektieren Sie das Ausgabefeld in ausreichender Größe. Andernfalls wird der Inhalt der Variablen in der Meldung nicht vollständig ausgegeben.
5. Um die Eingaben zu speichern, klicken Sie auf das Symbol .

WinCC fügt in den Meldetext einen Platzhalter für das Ausgabefeld ein: "<tag: n, [Variablenname]>" wobei n = Textlänge in Zeichen.

### Eigenschaften eines Ausgabefelds ändern

Um die Eigenschaften eines Ausgabefelds zu ändern, gehen Sie folgendermaßen vor:

- Doppelklicken Sie im Meldetext das Ausgabefeld und ändern Sie die Einstellungen wie gewünscht.

### Ausgabefeld aus dem Meldetext löschen

Um ein Ausgabefeld aus dem Meldetext zu löschen, gehen Sie folgendermaßen vor:

- Markieren Sie im Meldetext das Ausgabefeld und wählen Sie im Kontextmenü "Löschen".

---

#### Hinweis

Die Reihenfolge der Ausgabefelder für Variablen im Meldetext ist sprachabhängig. Die Änderung der Variable eines Ausgabefeldes in einer Sprache führt dazu, dass in allen anderen Sprachen das geänderte Ausgabefeld am Ende des Meldetextes steht.

---

## Meldetext formatieren

### Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Eine Meldung ist angelegt.

### Vorgehen

Um einen Meldetext zu formatieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldung, die Sie bearbeiten wollen.
2. Markieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein > Meldetext" die Zeichen, die Sie formatieren wollen.
3. Wählen Sie im Kontextmenü die gewünschte Formatierung, z. B. "Unterstrichen" oder "Großbuchstaben".

### Ergebnis

Die ausgewählten Zeichen werden in Runtime in der gewählten Formatierung dargestellt.

## Formatierungen entfernen

Um alle Formatierungen eines Texts zu entfernen, gehen Sie folgendermaßen vor:

1. Markieren Sie im Inspektorfenster im Meldetext die Zeichen, deren Formatierung Sie entfernen wollen.
2. Wählen Sie im Kontextmenü "Formatierungszeichen löschen".

### Ergebnis

Die ausgewählten Zeichen werden in Runtime unformatiert dargestellt.

## Loop-In-Alarm projektieren

### Einleitung

Um nach dem Eingang einer Meldung zu dem Bild zu wechseln, das Informationen zu dem aufgetretenen Fehler enthält, wird ein Loop-In-Alarm projiziert.

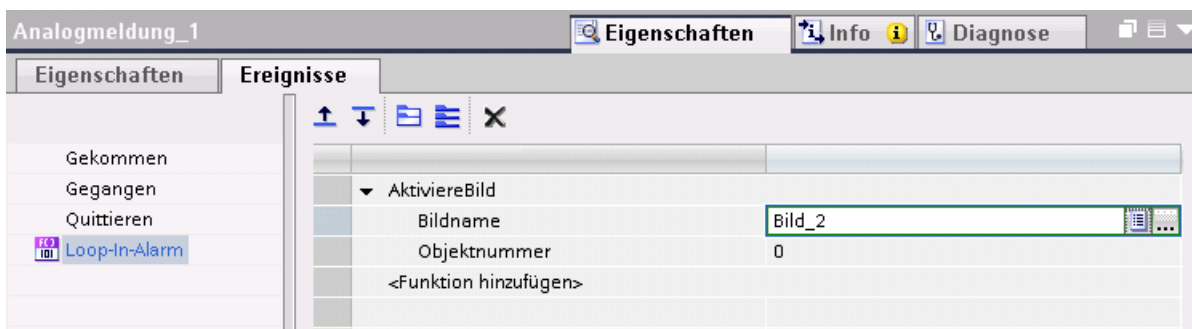
### Voraussetzung

- Das Bild, das der Loop-In-Alarm aufruft, ist angelegt.
- Der Editor "HMI-Meldungen" ist geöffnet.

## Vorgehen

Um für eine Meldung einen Loop-In-Alarm zu projektieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte, welche die Meldung enthält, für die Sie den Loop-In-Alarm projektieren wollen.
2. Markieren Sie die Meldung.
3. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Loop-In-Alarm".
4. Wählen Sie die Systemfunktion "AktiviereBild".
5. Wählen Sie als Parameter das Bild, das durch den Loop-In-Alarm aufgerufen wird.



## Hinweis

Um den Loop-In-Alarm für eine Meldeanzeige der Ausprägung "Meldezeile" zu projektieren, verwenden Sie eine Schaltfläche mit den folgenden Systemfunktionen:

- "BearbeiteMeldung" für Bediengeräte mit Tasten
- "MeldeanzeigeBearbeiteMeldung" für Bediengeräte ohne Tasten

Diese Systemfunktionen lösen das Ereignis "Loop-In-Alarm" aus. Die Meldezeile hat keine Schaltflächen.

## Ergebnis

Wenn Sie in Runtime auf die Schaltfläche "Loop-In-Alarm" der Meldeanzeige klicken, wird ein Bild mit Information zur markierten Meldung geöffnet.

## Siehe auch

Analogmeldungen projektieren (Seite 3219)

Bitmeldungen projektieren (Seite 3217)

## Meldungen im Editor "HMI-Variablen"

### Bitmeldungen projektieren im Editor "HMI-Variablen"

#### Einleitung

In WinCC können Sie auch im Editor "HMI-Variablen" Bit- und Analogmeldungen zusammen mit der Triggervariablen anlegen und bearbeiten.

---

#### Hinweis

Wenn Sie im Editor "HMI-Variablen" löschen, verschieben oder kopieren, wirken sich diese Änderungen auch im Editor "HMI-Meldungen" aus.

---

#### Voraussetzung

Der Editor "HMI-Variablen" ist geöffnet.

#### Vorgehen

Um eine Bitmeldung zu projektieren, gehen Sie folgendermaßen vor:

1. Um eine Variable anzulegen, klicken Sie im oberen Teil des Arbeitsbereichs in der Tabelle auf "<Hinzufügen>".  
Eine neue Variable wird angelegt.
2. Projektieren Sie nach Bedarf eine interne oder externe Variable.
  - Wenn Sie eine HMI Variable wählen, verwenden Sie die Datentypen "Int" oder "UInt".
  - Wenn Sie eine PLC Variable wählen, verwenden Sie die Datentypen "Int" oder "Word"
3. Markieren Sie die Variable im oberen Teil des Arbeitsbereichs.
4. Klicken Sie im unteren Teil des Arbeitsbereichs auf der Registerkarte "Bitmeldungen" in der Tabelle auf "<Hinzufügen>".  
Eine neue Bitmeldung wird für die Variable angelegt. Wenn Sie den falschen Datentyp gewählt haben, wird die Variable in der Bitmeldung markiert.

5. Projektieren Sie im Inspektorfenster die Bitmeldung:
  - Geben Sie unter "Eigenschaften >Eigenschaften > Allgemein > Meldetext" den gewünschten Meldetext ein.  
Fügen Sie bei Bedarf Ausgabefelder in den Meldetext ein.
  - Wählen Sie eine Meldeklasse.
  - Wählen Sie unter "Eigenschaften > Eigenschaften > Trigger" das Triggerbit der Variablen, das die Bitmeldung auslöst.
6. Legen Sie bei Bedarf weitere Bitmeldungen zur Überwachung der Variablen an.

---

#### Hinweis

Eine Variable wird nur mit einem Meldeverfahren überwacht. Legen Sie also für eine Variable entweder Analogmeldungen **oder** Bitmeldungen an.

---

## Ergebnis

Die projektierten Bitmeldungen sind im Editor "HMI-Variablen" angelegt und werden in den Editoren "HMI-Meldungen" und "HMI-Variablen" angezeigt.

## Analogmeldungen projektieren im Editor "HMI-Variablen"

### Einleitung

In WinCC legen Sie im Editor "HMI-Variablen" Bit- und Analogmeldungen zusammen mit der Triggervariablen an. Darüber hinaus bearbeiten Sie die Meldungen wie im Editor "HMI-Meldungen". Sie legen für eine Variable zwei Bereichswerte an, die Sie mit Analogmeldungen überwachen.

### Voraussetzung

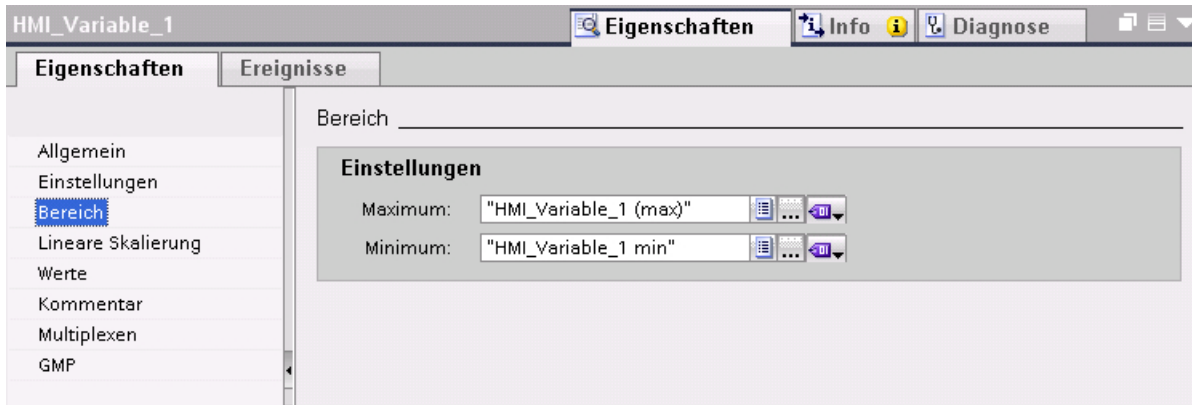
Der Editor "HMI-Variablen" ist geöffnet.

### Vorgehen

Um eine Analogmeldung im Editor "HMI-Variablen" zu projektieren, gehen Sie folgendermaßen vor:

1. Um eine Variable anzulegen, klicken Sie im oberen Teil des Arbeitsbereichs in der Tabelle auf "<Hinzufügen>".  
Eine neue Variable wird angelegt.
2. Projektieren Sie nach Bedarf eine interne oder externe Variable.
3. Projektieren Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Bereich" die Bereichswerte der Variablen:
  - Wählen Sie für die gewünschten Bereichswerte, ob eine "Konstante" oder "HMI\_Variable" als Grenzwert verwendet wird. Wenn Sie "HMI\_Variable" wählen, wird die Objektliste geöffnet. Wählen Sie die gewünschte Variable.





1. Klicken Sie im unteren Teil des Arbeitsbereichs auf die Registerkarte "Analogmeldungen". Legen Sie für beide Bereichswerte eine Analogmeldung an.
2. Markieren Sie eine Analogmeldung und konfigurieren Sie diese im Inspektorfenster:
  - Geben Sie unter "Eigenschaften > Eigenschaften > Allgemein > Meldetext" den gewünschten Meldetext ein.
  - Fügen Sie bei Bedarf Ausgabefelder in den Meldetext ein.
  - Ändern Sie bei Bedarf die voreingestellte Meldeklasse.
3. Projektieren Sie für Analogmeldungen weiter wie im Editor "HMI-Meldungen".
4. Fahren Sie fort, bis alle Analogmeldungen konfiguriert sind.

---

#### Hinweis

Eine Variable wird nur mit einem Meldeverfahren überwacht. Legen Sie also für eine Variable entweder Analogmeldungen **oder** Bitmeldungen an.

---

## Ergebnis

Die projektierten Analogmeldungen sind im Editor "HMI-Variablen" angelegt und werden in den Editoren "HMI-Meldungen" und "HMI-Variablen" angezeigt.

### 10.3.2.3 Projektieren der Ausgabe von Meldungen

#### Übersicht zum Projektieren der Ausgabe von Meldungen

#### Arbeitsschritte zum Projektieren der Ausgabe von Meldungen

Sie projektieren die Ausgabe von Meldungen in WinCC in folgenden Arbeitsschritten:

1. Meldeanzeige erstellen  
Um Meldungen in Runtime anzuzeigen, verwenden Sie die Anzeige- und Bedienobjekte im Editor "Bilder".
2. Quittierung projektieren  
Im Editor "Bilder" richten Sie ein, mit welcher Bedienaktion die Quittierung getriggert wird.

#### Weitergehende Projektierungsschritte

Abhängig von den Anforderungen Ihres Projekts sind weitere Arbeitsschritte zum Projektieren der Meldeanzeige erforderlich:

1. Berechtigungen einrichten  
Um sicherzustellen, dass nur berechtigte Benutzer die Meldungen bearbeiten, vergeben Sie Berechtigungen für die Meldeanzeige und die Funktionstasten des Bediengeräts.
2. Filtern der Meldeanzeige projektieren  
Sie projektieren im Editor "Bilder" das Filtern der Meldungen in Runtime. Sie projektieren bei Bedarf auch Meldeanzeigen, die nur ausgewählte Meldungen anzeigen.
3. Bedienmeldungen projektieren  
Projektieren Sie im Editor "Bilder" an die Bedienelemente des Anzeige- und Bediengeräts Bedienmeldungen. Bei einer Bedienaktion wird eine vorkonfigurierte Bedienmeldung ausgegeben. Eine Bedienaktion ist z. B. die Quittierung einer Meldung.

#### Anzeige der Meldungen

#### Möglichkeiten zur Anzeige von Meldungen am Bediengerät

Um Meldungen am Bediengerät anzuzeigen, bietet WinCC folgende Möglichkeiten:

- Meldeanzeige  
Die Meldeanzeige projektieren Sie in einem Bild. Abhängig von ihrer projizierten Größe zeigt sie mehrere Meldungen gleichzeitig an. Sie projektieren mehrere Meldeanzeigen mit unterschiedlichen Inhalten.
- Meldefenster  
Das Meldefenster projektieren Sie im Editor "Globales Bild". Abhängig von seiner projizierten Größe zeigt das Meldefenster mehrere Meldungen gleichzeitig an. Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört. Je nach Projektierung wird es erst wieder geschlossen, wenn die Meldung quittiert wurde.

## Zusätzliche Signale

- **Meldeindikator**  
Der Meldeindikator ist ein projektierbares grafisches Symbol. Wenn eine Meldung kommt, wird der Meldeindikator am Bediengerät angezeigt. Sie projektieren den Meldeindikator im Editor "Globales Bild".  
Der Meldeindikator hat zwei Zustände:
  - Blinkend: Mindestens eine quittierpflichtige Meldung steht an.
  - Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen.  
Abhängig vom Bediengerät zeigt der Meldeindikator auch die Anzahl anstehender Meldungen an.
- **Systemfunktionen**  
Sie projektieren an ein Ereignis einer Meldung eine Funktionsliste, die beim Eintreten des Ereignisses in Runtime abgearbeitet wird.  
Um die Meldeanzeige oder das Meldefenster nicht über die Symbolleiste zu bedienen, verwenden Sie in WinCC Systemfunktionen für Meldungen.

## Anzeige der vordefinierten Meldeklassen in Runtime

Die folgende Tabelle zeigt die Symbole, mit der die vordefinierten Meldeklassen in der Meldeanzeige dargestellt werden:

| Meldeklasse | Angezeigtes Symbol |
|-------------|--------------------|
| "Errors"    | !                  |
| "System"    | \$                 |
| "Warnings"  | <kein Symbol>      |

## Meldeanzeige projektieren

### Einleitung

Aktuelle Meldungen werden in Runtime in einer Meldeanzeige oder einem Meldefenster angezeigt.

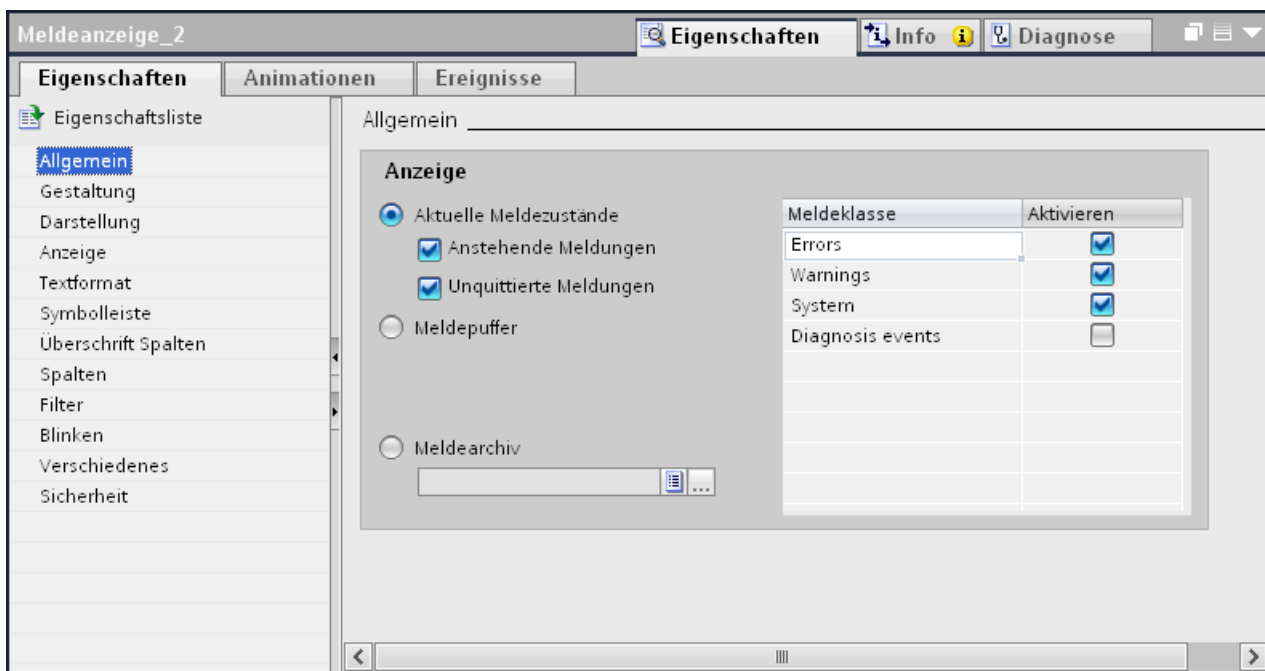
### Voraussetzung

- Im Editor "Bilder" ist ein Bild geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.

### Meldungen für die Meldeanzeige projektieren

Um festzulegen, welche Meldungen in der Meldeanzeige angezeigt werden, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Task Card "Werkzeuge" das Objekt "Meldeanzeige" in das Bild ein.
2. Markieren Sie die Meldeanzeige.
  - Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Anzeige > Aktuelle Meldezustände".  
Legen Sie bei Bedarf fest, ob quittierpflichtige und/oder Meldungen ohne Quittierpflicht angezeigt werden.
  - Um alle Meldungen anzuzeigen, die sich im Meldepuffer befinden, aktivieren Sie "Meldepuffer".



3. Aktivieren Sie in der Tabelle die Meldeklassen, die in der Meldeanzeige angezeigt werden.

## Darstellung der Meldeanzeige projektieren

Um festzulegen, wie die Meldungen in der Meldeanzeige dargestellt werden, gehen Sie folgendermaßen vor.

1. Legen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Darstellung > Einstellungen > Zeilen pro Meldung" die Anzahl der Zeilen fest, mit der jede Meldung angezeigt wird.
2. Wählen Sie unter "Eigenschaften > Eigenschaften > Anzeige" die Bedienelemente, die am Bediengerät zur Verfügung stehen.
3. Konfigurieren Sie unter "Eigenschaften > Eigenschaften > Spalten" die Spalten:
  - Wählen Sie unter "Sichtbare Spalten" die Spalten, die in der Meldeanzeige ausgegeben werden.
  - Legen Sie unter "Eigenschaften Spalten" die Eigenschaften der Spalten fest.
  - Wählen Sie unter "Sortieren" die Sortierreihenfolge der Meldungen.



## Ergebnis

In Runtime werden in der Meldeanzeige Meldungen verschiedener Meldeklassen ausgegeben.

## Siehe auch

Meldeanzeige (Seite 3142)

## Meldefenster projektieren

## Einleitung

Das Meldefenster zeigt aktuelle Meldungen an. Das Meldefenster wird im Editor "Globales Bild" projiziert. Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung

wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört. Je nach Projektierung wird es erst wieder geschlossen, wenn die Meldung quittiert wurde.. Auch wenn Meldungen anstehen und angezeigt werden, bleibt das Bediengerät bedienbar. Ein Meldefenster wird dargestellt und projektiert wie eine Meldeanzeige.

Um Meldefenster bei der Projektierung gezielt auszublenden, legen Sie diese in einer eigenen Ebene an.

### Voraussetzung

- Der Editor "Globales Bild" ist geöffnet.
- Die Task Card "Werkzeuge" wird angezeigt.
- Das Inspektorfenster ist geöffnet.

### Vorgehen

Um ein Meldefenster zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Task Card "Werkzeuge" das Objekt "Meldefenster" in das Globale Bild ein.
2. Projektieren Sie das Meldefenster wie eine Meldeanzeige.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Modus > Fenster" wie sich das Meldefenster in Runtime verhält und bedient wird.
  - Wenn das Meldefenster in Runtime bei einem Bildwechsel den Fokus behalten soll, aktivieren Sie "Gebunden".  
Diese Option ist wichtig, da das Hin- und Herschalten zwischen Bild und verschiedenen Fenstern mit <Strg+Tab> nicht unterstützt wird.

### Ergebnis

In Runtime werden die Meldungen der gewählten Meldeklasse im Meldefenster angezeigt.

### Siehe auch

Meldefenster (Seite 3144)

### Meldeindikator projektieren

#### Einleitung

Der Meldeindikator zeigt mit einem Warndreieck an, dass Meldungen anstehen oder quittiert werden müssen. Wenn eine Meldung einer mit Meldeindikator projektierten Meldeklasse kommt, wird der Meldeindikator eingeblendet.

Der Meldeindikator hat zwei Zustände:

- Blinkend: Mindestens eine quittierpflichtige Meldung steht an.
- Statisch: Mindestens eine der quittierten Meldungen ist noch nicht gegangen.

Sie legen bei der Projektierung bei Bedarf fest, dass in Runtime das Meldefenster geöffnet wird, wenn der Benutzer den Meldeindikator bedient.

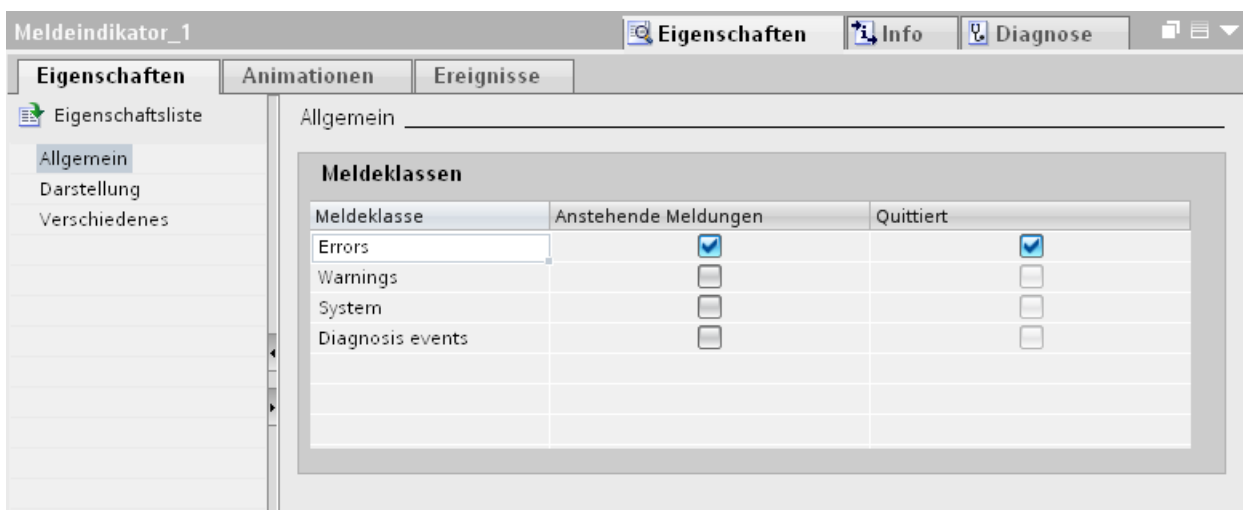
## Voraussetzung

- Der Editor "Globales Bild" ist geöffnet.
- Die Task Card "Werkzeuge" ist geöffnet.
- Das Inspektorfenster ist geöffnet.

## Vorgehen

Um den Meldeindikator zu projektieren, gehen Sie folgendermaßen vor:

1. Fügen Sie aus der Task Card "Werkzeuge" das Objekt "Meldeindikator" in den Arbeitsbereich ein.
2. Markieren Sie den Meldeindikator.
3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldeklassen, die durch den Meldeindikator angezeigt werden. Geben Sie an, ob anstehende und/oder quittierte Meldungen mit Meldeindikator angezeigt werden.



4. Projektieren Sie unter "Eigenschaften > Ereignis" die Systemfunktion "ZeigeMeldefenster" an ein Ereignis des Meldeindikators.

### Hinweis

Wenn Sie im Bild oder der Vorlage ein Permanentfenster projiziert haben, positionieren Sie Meldefenster und Meldeindikator nicht im Bereich des Permanentfensters. Das Meldefenster und der Meldeindikator werden sonst in Runtime nicht angezeigt. Das Permanentfenster sehen Sie im Editor "Globales Bild" jedoch nicht.

## Ergebnis

Wenn in Runtime Meldungen der ausgewählten Meldeklasse anstehen oder quittiert werden müssen, wird der Meldeindikator angezeigt. Wenn der Benutzer den Meldeindikator bedient, wird das Meldefenster geöffnet.

## Siehe auch

Meldeindikator (Seite 3147)

### 10.3.2.4 Quittieren von Meldungen

#### Quittierpflicht einer Meldung über Meldeklasse projektieren

## Einleitung

Um eine Meldung mit Quittierpflicht zu projektieren, ordnen Sie diese einer Meldeklasse mit dem Quittierkonzept "Meldung mit Einfachquittierung" zu.

## Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die gewünschte Meldeklasse ist angelegt.
- Die gewünschte Meldung ist angelegt.

#### Quittierkonzept für eine Meldeklasse wählen

Das Quittierkonzept vordefinierter Meldeklassen ist festgelegt. Sie wählen das Quittierkonzept nur bei benutzerdefinierten Meldeklassen. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Meldeklasse" und markieren Sie die gewünschte Meldeklasse.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung" das gewünschte Quittierkonzept.

#### Meldungen einer quittierpflichtigen Meldeklasse zuordnen

Um einer Meldung eine Meldeklasse mit Quittierpflicht zuzuordnen, gehen Sie folgendermaßen vor.

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte des gewünschten Meldeverfahrens und markieren Sie die gewünschte Meldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Allgemein" die Meldeklasse der Meldung aus.



## Ergebnis

Die Meldung wird in Runtime erst ausgeblendet, wenn sie der Bediener quittiert hat.

## Trigger zum Quittieren von Meldungen projektieren

### Einleitung

Grundsätzlich legen sie die Quittierpflicht einer Meldung über die Meldeklasse fest. Dann quittiert der Bediener die Meldung über die Funktionstaste "ACK" des Bediengeräts oder die Schaltfläche "Quittieren" der Meldeanzeige.

Außerdem gibt es folgende Möglichkeiten die Quittierung zu triggern:

- Schaltfläche zum Quittieren einer Meldung projektieren
- Bitmeldung durch die Steuerung quittieren

### Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die gewünschte Meldeklasse ist angelegt.
- Die gewünschte Meldung ist angelegt.
- Im Editor "Bilder" sind eine Meldeanzeige und eine Schaltfläche angelegt.

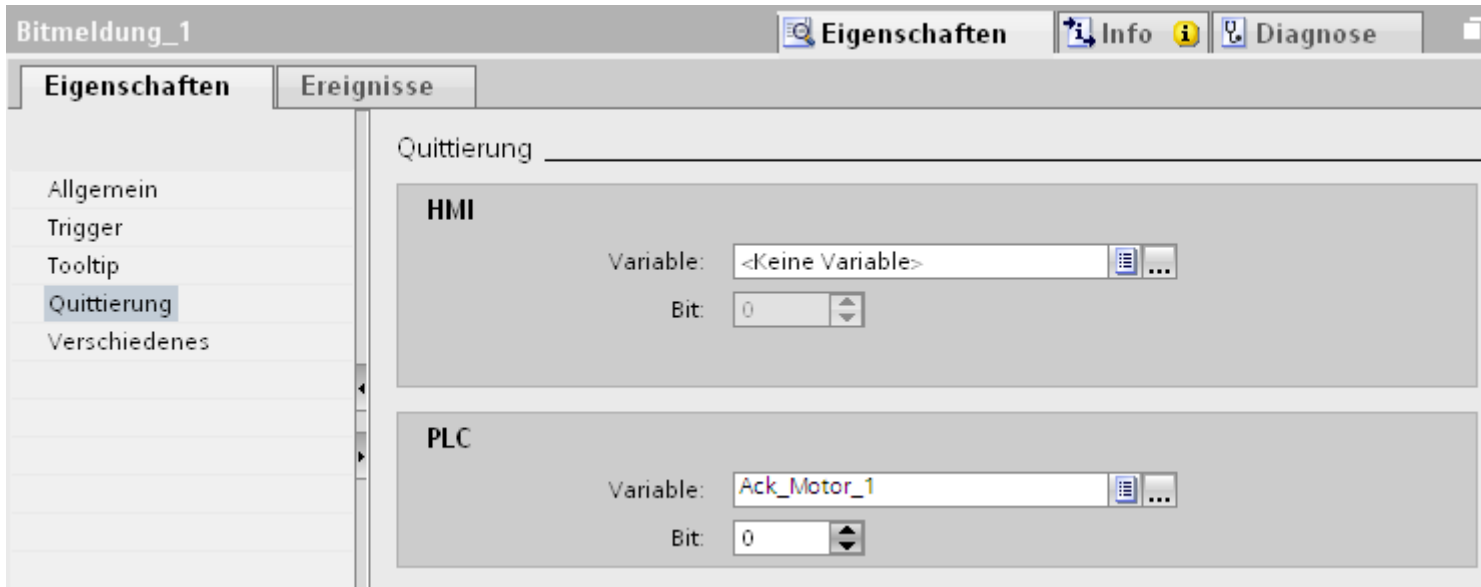
## Schaltfläche zum Quittieren einer Meldung projektieren

Um eine Schaltfläche zum Quittieren einer Meldung zu projektieren, gehen Sie folgendermaßen vor:

1. Markieren Sie im Editor "Bilder" die Schaltfläche.
2. Projektieren Sie im Inspektorfenster unter "Eigenschaften > Ereignisse" die Systemfunktion "MeldeanzeigeQuittiereMeldung" an das Ereignis "Klicken".
3. Wählen Sie als Parameter die Meldeanzeige.

### Bitmeldung durch die Steuerung quittieren

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Bitmeldung" und markieren Sie die gewünschte Bitmeldung.
2. Wählen Sie im Inspektorfenster unter "Eigenschaften > Eigenschaften > Quittierung > PLC" die Variable und das Bit, durch das die Meldung von der Steuerung quittiert wird.



### Meldungsquittierung an die Steuerung senden

#### Voraussetzung

- Der Editor "HMI-Meldungen" ist geöffnet.
- Die gewünschte Meldung ist angelegt und einer Meldeklasse mit Quittierpflicht zugeordnet.

---

#### Hinweis

Die Quittierung von Analogmeldungen kann nicht an die Steuerung gesendet werden.

---

## Meldungsquittierung an die Steuerung senden

Um zu projektieren, dass die Quittierung einer Meldung an die Steuerung gesendet wird, gehen Sie folgendermaßen vor:

1. Klicken Sie im Editor "HMI-Meldungen" auf die Registerkarte "Bitmeldung" und markieren Sie die gewünschte Bitmeldung.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Quittierung".
3. Wählen Sie unter "HMI" die Variable und das Bit, das durch die Meldungsquittierung gesetzt wird.

---

### Hinweis

Auf den Speicherbereich der Quittiervariablen darf von Bediengerät und Steuerung nur lesend zugegriffen werden.

---

## Ergebnis

Wenn der Bediener in Runtime die Meldung quittiert, wird der Bedienvorgang an die Steuerung weitergeleitet.

## 10.3.3 Meldungen in Runtime bedienen

### 10.3.3.1 Meldungen in Runtime

## Meldungen

Meldungen zeigen am Bediengerät Ereignisse und Zustände an, die in der Anlage bzw. im Prozess oder am Bediengerät auftreten. Ein Zustand wird bei dessen Eintritt gemeldet.

Für eine Meldung können folgende Meldeereignisse auftreten:

- Gekommen
- Gegangen
- Quittieren
- Loop-In-Alarm

Der Projektteur legt fest, welche Meldungen vom Benutzer quittiert werden müssen.

Eine Meldung kann folgende Informationen enthalten:

- Datum
- Uhrzeit
- Meldetext
- Störort
- Zustand

- Meldeklasse
- Meldenummer
- Meldegruppe

## Meldeklassen

Meldungen sind verschiedenen Meldeklassen zugeordnet.

- "Warnings"  
Meldungen dieser Klasse zeigen normalerweise Zustände in der Anlage an, z. B. "Motor eingeschaltet". Meldungen dieser Meldeklasse müssen nicht quittiert werden.
- "Errors"  
Meldungen dieser Klasse müssen immer quittiert werden. Störmeldungen zeigen normalerweise kritische Störungen in der Anlage an, z. B. "Motortemperatur zu hoch".
- "System"  
Systemmeldungen zeigen Zustände oder Ereignisse des Bediengeräts selbst an. Systemmeldungen informieren z. B. über Fehlbedienungen oder Störungen in der Kommunikation.
- Benutzerdefinierte Meldeklassen  
Die Eigenschaften dieser Meldeklasse werden bei der Projektierung festgelegt.

## Meldepuffer

Die Meldeereignisse werden in einem internen Puffer gespeichert. Die Größe dieses Meldepuffers hängt vom Typ des Bediengeräts ab.

## Meldeanzeige

In der Meldeanzeige werden ausgewählte Meldungen oder Meldeereignisse aus dem Meldepuffer angezeigt. Über die Projektierung wird festgelegt, ob die auftretenden Meldungen quittiert werden müssen.

## Meldefenster

Das Meldefenster ist keinem Bild zugeordnet. Abhängig von der Projektierung wird das Meldefenster geöffnet, wenn eine Meldung ansteht, die einer bestimmten Meldeklasse angehört.

Die Sortierung der angezeigten Meldungen ist projektierbar. Wahlweise kann die aktuelle oder die älteste Meldung zuerst angezeigt werden. Im Meldefenster kann weiterhin der genaue Störort mit Datum/Uhrzeit des Meldeereignisses angezeigt werden. Über die Projektierung kann die Anzeige so gefiltert werden, dass nur Meldungen angezeigt werden, die in Ihrem Meldetext eine bestimmte Zeichenkette enthalten.

## Meldeindikator

Der Meldeindikator ist ein grafisches Symbol, das am Bildschirm angezeigt wird, wenn eine Meldung der festgelegten Meldeklasse kommt.

Der Meldeindikator kann zwei Zustände haben:

- Blinkend: Mindestens eine unquitierte Meldung steht an.
- Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen. Die angezeigte Zahl bedeutet die Anzahl noch anstehender Meldungen.

### 10.3.3.2 Einfache Meldeanzeige, Einfaches Meldefenster in Runtime

#### Verwendung

In der einfachen Meldeanzeige werden ausgewählte Meldungen oder Meldeereignisse aus dem Meldepuffer angezeigt. Aussehen und Bedienung des einfachen Meldefensters entsprechen dem der einfachen Meldeanzeige.

#### Hinweis

Im Engineering System dynamisieren Sie im Inspektorfenster im Register "Animationen", z. B. die Sichtbarkeit eines Objekts. Das Objekt "Einfache Meldeanzeige" unterstützt in Runtime keine Animation. Wenn Sie eine Animation projiziert haben und z. B. eine Konsistenzprüfung des Projekts ausführen, dann wird im Ausgabefenster eine Fehlermeldung ausgegeben.



#### Darstellung

Abhängig von der Projektierung werden in der Meldeanzeige verschiedene Spalten mit Informationen zu einer Meldung oder einem Meldeereignis angezeigt.

Zur Unterscheidung der verschiedenen Meldeklassen enthält die erste Spalte in der Meldeanzeige ein Symbol:

| Symbol                     | Meldekategorie                     |
|----------------------------|------------------------------------|
| !                          | "Errors"                           |
| leer                       | "Warnings"                         |
| abhängig von Projektierung | Benutzerdefinierte Meldekategorien |
| \$                         | "System"                           |

### Bedienung

Abhängig von der Projektierung bedienen Sie die Meldeanzeige wie folgt:

- Meldungen quittieren
- Meldungen bearbeiten

### Bedienelemente

Die Schaltflächen haben folgende Funktionen:

| Schaltfläche | Funktion  |
|--------------|---|
|              | Meldung quittieren  |
|              | Loop-In-Alarm<br>Wechselt zu dem Bild, das Informationen zu dem aufgetretenen Fehler enthält  |
|              | Tooltip für eine Meldung anzeigen   |
|              | Anzeige des kompletten Meldetexts zur ausgewählten Meldung in einem separaten Fenster, dem Meldetextfenster<br>Im Meldetextfenster zeigen Sie Meldetexte an, die mehr Platz erfordern, als in der Meldeanzeige vorhanden ist. Schließen Sie das Meldetextfenster mit der Schaltfläche |
|              | Blättert um eine Meldung nach oben.   |
|              | Blättert in der Meldeanzeige seitenweise nach oben.   |
|              | Blättert in der Meldeanzeige seitenweise nach unten.  |
|              | Blättert um eine Meldung nach unten.  |

### Darstellung der Bedienelemente

Die Darstellung der Schaltflächen zur Bedienung der einfachen Meldeanzeige ist abhängig von der projektierten Größe. Überprüfen Sie daher auf dem Bediengerät, ob alle benötigten Schaltflächen zur Verfügung stehen.

### 10.3.3.3 Meldeindikator in Runtime

#### Verwendung

Wenn Meldungen der festgelegten Meldeklasse anstehen oder quittiert werden müssen, wird der Meldeindikator angezeigt.



#### Darstellung

Der Meldeindikator kann zwei Zustände haben:

- Blinkend: Mindestens eine unquittierte Meldung steht an.
- Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen. Die angezeigte Zahl bedeutet die Anzahl noch anstehender Meldungen.

#### Bedienung

Abhängig von der Projektierung wird bei Bedienung des Meldeindikators ein Meldefenster geöffnet. Der Meldeindikator ist nur per Touch-Screen bedienbar.

### 10.3.3.4 Meldungen quittieren

#### Einleitung

Meldungen quittieren Sie in Runtime abhängig von der Projektierung Ihres Projekts. Sie quittieren Meldungen folgendermaßen:

- Mit den Schaltflächen der Anzeige- und Bedienobjekte
- Mit der Taste "ACK" an Ihrem Bediengerät
- Mit individuell projektierte Funktionstasten oder Schaltflächen

Wenn eine Bedienberechtigung für die Bedienelemente projektiert ist, dann sind die Meldungen nur von berechtigten Benutzern quittierbar.

Um Meldungen in Runtime automatisch zu quittieren, verwenden Sie Systemfunktionen und die Option "Quittieren durch die Steuerung".

### Varianten der Quittierung

Sie quittieren in Runtime einzelne Meldungen oder mehrere Meldungen gemeinsam. Sie unterscheiden dabei folgendermaßen:


- Einzelquittierung  
Quittieren einer Meldung über eine Schaltfläche oder Funktionstaste.
- Meldegruppen quittieren  
Quittierung aller Meldungen einer Meldegruppe über eine Schaltfläche oder Funktionstaste.

### Voraussetzung

- Eine Meldung wird am Bediengerät angezeigt.

### Vorgehen

Um eine Meldung zu quittieren, gehen Sie folgendermaßen vor:

1. Markieren Sie die Meldung.
2. Klicken Sie auf die Schaltfläche .

### Ergebnis

Die Meldung erhält den Status "quittiert". Wenn die Bedingung zum Auslösen einer Meldung nicht mehr zutrifft, erhält die Meldung zusätzlich den Zustand "gegangen" und wird am Bediengerät nicht mehr angezeigt.

## 10.3.4 Referenz

### 10.3.4.1 Systemfunktionen für Meldungen

#### Systemfunktionen

Systemfunktionen sind vordefinierte Funktionen, mit denen Sie auch ohne Programmierkenntnisse viele Aufgaben in Runtime realisieren. Systemfunktionen verwenden Sie in einer Funktionsliste.

Die Tabelle zeigt alle Systemfunktionen zum Darstellen und Bearbeiten von Meldungen.

| Systemfunktion     | Auswirkung  |
|--------------------|---|
| BearbeiteMeldung   | Löst das Ereignis "Loop-In-Alarm" für alle gewählten Meldungen aus. |
| LoescheMeldepuffer | Löscht Meldungen aus dem Meldepuffer am Bediengerät.                |



| Systemfunktion               | Auswirkung   |
|------------------------------|--|
| LoescheMeldepufferProTool    | Funktion wie "LoescheMeldepuffer". Diese Systemfunktion ist aus Kompatibilitätsgründen enthalten und verwendet die ProTool-Nummerierung. |
| MeldeanzeigeBearbeiteMeldung | Löst das Ereignis "Loop-In-Alarm" für alle Meldungen aus, die in der angegebenen Meldeanzeige markiert sind.                             |
| MeldeanzeigeQuittiereMeldung | Quittiert die Meldungen, die in der angegebenen Meldeanzeige markiert sind.  |
| MeldeanzeigeZeigeHilfetext   | Zeigt den projektierten Tooltip der Meldung an, welcher in der angegebenen Meldeanzeige markiert ist.                                    |
| QuittiereMeldung             | Quittiert alle markierten Meldungen.   |
| ZeigeMeldefenster            | Blendet das Meldefenster am Bediengerät ein oder aus.  |

### 10.3.4.2 Systemmeldungen

#### Grundlagen zu Systemmeldungen

#### Systemmeldungen

Systemmeldungen geben am Bediengerät Auskunft über interne Zustände des Bediengeräts und der Steuerung.

Nachfolgend erhalten Sie eine Übersicht, wann eine Systemmeldung auftritt und wie Sie die Fehlerursache beheben.

---

#### Hinweis

##### Bediengeräteabhängigkeit

Für die einzelnen Bediengeräte gilt je nach Funktionsumfang eine Teilmenge der beschriebenen Systemmeldungen.

---



---

#### Hinweis

Systemmeldungen werden in einer Meldeanzeige ausgegeben. Systemmeldungen werden in der Sprache ausgegeben, die aktuell an Ihrem Bediengerät eingestellt ist.

---

#### Parameter der Systemmeldungen

Systemmeldungen können verschlüsselte Parameter enthalten. Die Parameter sind zur Verfolgung eines Fehlers relevant, da sie Hinweise auf den Quellcode der Runtime-Software geben. Die Parameter werden nach dem Text "Fehlercode:" ausgegeben.

## 30000 - Meldungen Fehler bei Verwendung von Systemfunktionen

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können. Die Systemmeldungen sind in verschiedene Bereiche aufgeteilt.

Tabelle 10-2 30000 - Meldungen Fehler bei Verwendung von Systemfunktionen

| Nummer | Wirkung/Ursachen   | Abhilfe   |
|--------|--|---|
| 30010  | Die Variable konnte das Ergebnis der Systemfunktion nicht aufnehmen, z. B. bei Wertebereichsüberschreitung.                            | Überprüfen Sie die Variablentypen der Parameter der Systemfunktion.   |
| 30011  | Eine Systemfunktion konnte nicht ausgeführt werden, da im Parameter der Systemfunktion ein unzulässiger Wert oder Typ übergeben wurde. | Überprüfen Sie den Parameterwert und Variablentyp des unzulässigen Parameters. Falls als Parameter eine Variable verwendet wird, überprüfen Sie deren Wert. |
| 30012  | Eine Systemfunktion konnte nicht ausgeführt werden, da im Parameter der Systemfunktion ein unzulässiger Wert oder Typ übergeben wurde. | Überprüfen Sie den Parameterwert und Variablentyp des unzulässigen Parameters. Falls als Parameter eine Variable verwendet wird, überprüfen Sie deren Wert. |

## 40000 - Meldungen Lineare Skalierung

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können. Die Systemmeldungen sind in verschiedene Bereiche aufgeteilt.

Tabelle 10-3 40000 - Meldungen Lineare Skalierung

| Nummer | Wirkung/Ursachen  | Abhilfe   |
|--------|---|---|
| 40010  | Die Systemfunktion konnte nicht ausgeführt werden, da die Parameter nicht auf einen gemeinsamen Variablentyp konvertiert werden können. | Überprüfen Sie die Parametertypen in der Projektierung. |
| 40011  | Die Systemfunktion konnte nicht ausgeführt werden, da die Parameter nicht auf einen gemeinsamen Variablentyp konvertiert werden können. | Überprüfen Sie die Parametertypen in der Projektierung. |

## 50000 - Meldungen Datenserver

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können. Die Systemmeldungen sind in verschiedene Bereiche aufgeteilt.

Tabelle 10-4 50000 - Meldungen Datenserver

| Nummer | Wirkung/Ursachen   | Abhilfe |
|--------|--|---------|
| 50000  | Das Bediengerät erhält schneller Daten als es bearbeiten kann. Es werden daher solange keine neuen Daten angenommen, bis die vorhandenen Daten bearbeitet wurden. Danach wird der Datenaustausch wieder aufgenommen. | --      |
| 50001  | Der Datenaustausch wurde wieder aufgenommen.   | --      |

## 70000 - Meldungen Win32 Funktionen

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können.

Tabelle 10-5 70000 - Meldungen Win32 Funktionen

| Nummer | Wirkung/Ursachen   | Abhilfe   |
|--------|--|---|
| 70010  | Das Programm konnte nicht gestartet werden, da es im angegebenen Pfad nicht gefunden wurde oder weil nicht genügend freier Speicherplatz vorhanden ist.  | Überprüfen Sie, ob das Programm im angegebenen Pfad oder Suchpfad existiert oder schließen Sie andere Programme.  |
| 70011  | Die Systemzeit konnte nicht geändert werden. Die Fehlermeldung erscheint nur im Zusammenhang mit Bereichszeiger "Datum/Uhrzeit Steuerung". Mögliche Ursachen: <ul style="list-style-type: none"> <li>Im Steuerungsauftrag wurde eine unzulässige Zeit übergeben.</li> <li>Der Windows-Benutzer hat kein Benutzerrecht, um die Systemzeit zu ändern.</li> </ul> Wird in der Systemmeldung als erster Parameter der Wert 13 angezeigt, so kennzeichnet der zweite Parameter das Byte, das den falschen Wert hat. | Überprüfen Sie die Zeit, die gesetzt werden soll. Unter Windows NT/XP: Der Benutzer, der WinCC Runtime ausführt, muss das Recht erhalten, die Systemzeit des Betriebssystems zu ändern. |
| 70012  | Bei der Ausführung der Systemfunktion "StoppeRuntime" mit der Option "Runtime und Betriebssystem" trat ein Fehler auf. Windows und WinCC Runtime werden nicht beendet. Eine mögliche Ursache ist, dass sich andere Programme nicht beenden lassen.   | Beenden Sie alle zur Zeit laufenden Programme. Beenden Sie dann Windows.  |
| 70013  | Die Systemzeit konnte nicht geändert werden, da der eingegebene Wert nicht zulässig ist. Evtl. wurden falsche Trennzeichen verwendet.  | Überprüfen Sie die Zeit, die gesetzt werden soll.   |
| 70014  | Die Systemzeit konnte nicht geändert werden. Mögliche Ursachen: <ul style="list-style-type: none"> <li>Es wurde eine unzulässige Zeit übergeben.</li> <li>Der Windows-Benutzer hat kein Benutzerrecht um die Systemzeit zu ändern.</li> </ul> Windows lehnt das Setzen ab.   | Überprüfen Sie die Zeit, die gesetzt werden soll. Unter Windows NT/XP: Der Benutzer, der WinCC Runtime ausführt, muss das Recht erhalten, die Systemzeit des Betriebssystems zu ändern. |

| Nummer | Wirkung/Ursachen   | Abhilfe   |
|--------|--|---|
| 70015  | Die Systemzeit konnte nicht gelesen werden, da Windows das Lesen ablehnt.  | --  |
| 70016  | Es wurde versucht, durch eine Systemfunktion oder einen Auftrag ein Bild anzuwählen. Dies ist nicht möglich, da die projektierte Bildnummer nicht existiert.<br>Oder: ein Bild konnte wegen unzureichendem Systemspeicher nicht aufgebaut werden.<br>Oder: Das Bild ist gesperrt.<br>Oder: Bildaufruf wird nicht korrekt ausgeführt. | Vergleichen Sie die Bildnummer in der Systemfunktion oder im Auftrag mit den projektierten Bildnummern. Weisen Sie gegebenenfalls die Nummer einem Bild zu. Kontrollieren Sie die Angaben zum Bildaufruf und ob das Bild für bestimmte Benutzer gesperrt ist. |
| 70017  | Datum/Uhrzeit wird nicht aus dem Bereichszeiger gelesen, da die eingestellte Adresse in der Steuerung nicht vorhanden oder nicht eingerichtet ist.   | Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung ein.   |
| 70018  | Rückmeldung für erfolgreichen Import der Kennwortliste.  | --  |
| 70019  | Rückmeldung für erfolgreichen Export der Kennwortliste.  | --  |
| 70020  | Rückmeldung für Aktivierung der Meldeprotokollierung.  | --  |
| 70021  | Rückmeldung für Deaktivierung der Meldeprotokollierung.  | --  |
| 70022  | Rückmeldung für den Start der Aktion Kennwortliste importieren.  | --  |
| 70023  | Rückmeldung für den Start der Aktion Kennwortliste exportieren.  | --  |
| 70024  | Der Wertebereich der Variablen wurde bei Ausführung der Systemfunktion überschritten. Die Berechnung der Systemfunktion wird nicht durchgeführt.   | Überprüfen Sie die gewünschte Berechnung und korrigieren Sie diese gegebenenfalls.  |
| 70025  | Der Wertebereich der Variablen wurde bei Ausführung der Systemfunktion überschritten. Die Berechnung der Systemfunktion wird nicht durchgeführt.   | Überprüfen Sie die gewünschte Berechnung und korrigieren Sie diese gegebenenfalls.  |
| 70026  | Es sind keine weiteren Bilder in dem internen Bilderspeicher gespeichert.<br>Es ist keine weitere Bildanwahl möglich.  | --  |
| 70027  | Das Sichern des RAM-Dateisystems wurde gestartet.  | --  |
| 70028  | Das Sichern des RAM-Dateisystems wurde erfolgreich beendet.<br>Die Dateien aus dem RAM werden ausfallsicher in den Flash-Speicher kopiert. Bei Neuanlauf werden diese gesicherten Dateien wieder in das RAM-Dateisystem zurückkopiert.   | --  |
| 70029  | Das Sichern des RAM-Dateisystems ist fehlgeschlagen.<br>Das RAM-Dateisystem wurde nicht gesichert.   | Überprüfen Sie die Einstellungen im Dialog "Control Panel > OP" und sichern Sie das RAM-Dateisystem über die Schaltfläche "Save Files" in der Registerkarte "Persistent Storage".   |

| Nummer | Wirkung/Ursachen   | Abhilfe   |
|--------|--|---|
| 70030  | Die projektierten Parameter der Systemfunktion sind fehlerhaft.<br>Die Verbindung zur neuen Steuerung wurde nicht aufgebaut.   | Vergleichen Sie die projektierten Parameter der Systemfunktion mit den projektierten Parametern der Steuerungen und korrigieren Sie diese gegebenenfalls.   |
| 70031  | Die projektierte Steuerung in der Systemfunktion ist keine S7-Steuerung.<br>Die Verbindung zur neuen Steuerung wurde nicht aufgebaut.  | Vergleichen Sie den projektierten Parameter S7-Steuerungsname der Systemfunktion mit den projektierten Parametern der Steuerungen und korrigieren Sie diesen gegebenenfalls.  |
| 70032  | In dem angewählten Bild ist das projektierte Objekt mit dieser Nummer in der Tab-Reihenfolge nicht vorhanden.<br>Der Bildwechsel wird ausgeführt, der Fokus wird jedoch auf das erste Objekt gesetzt.  | Überprüfen Sie die Nummer in der Tab-Reihenfolge und korrigieren Sie diese gegebenenfalls.  |
| 70033  | Eine E-Mail konnte nicht versandt werden, weil keine TCP/IP-Verbindung zum SMTP-Server mehr besteht.<br>Die Systemmeldung wird nur beim ersten Fehlversuch generiert. Alle folgenden erfolglosen Versuche, eine E-Mail zu versenden, erzeugen keine Systemmeldung mehr. Die Meldung wird erst wieder erzeugt, wenn inzwischen eine E-Mail erfolgreich versandt werden konnte.<br>Die zentrale E-Mail-Komponente in WinCC Runtime versucht in regelmäßigen Abständen ( 1 min.), die Verbindung zum SMTP-Server aufzubauen und die verbleibenden E-Mails zu versenden. | Überprüfen Sie die Netzwerkverbindung zum SMTP-Server und stellen Sie diese gegebenenfalls wieder her.  |
| 70034  | Nach einem Verbindungsabbruch konnte die TCP/IP Verbindung zum SMTP-Server wieder hergestellt werden.<br>Die in der Warteschlange verbliebenen E-Mails werden versandt.  | --  |
| 70036  | Es wurde kein SMTP-Server für den Versand von E-Mails konfiguriert. Eine Verbindung zu einem SMTP-Server kann daher nicht hergestellt werden und es können keine E-Mails gesendet werden.<br>Die Systemmeldung wird durch WinCC Runtime beim ersten Versuch eine E-Mail zu senden, generiert.  | Konfigurieren Sie einen SMTP-Server:<br>In WinCC Engineering System über "Geräteeinstellungen > Geräteeinstellungen"<br>Im Betriebssystem Windows CE über "Control Panel > Internet Settings > Email > SMTP Server" |
| 70037  | Eine E-Mail konnte aus unbekanntem Gründen nicht verschickt werden.<br>Der Inhalt der E-Mail wird verworfen.   | Überprüfen Sie die E-Mail-Parameter (Empfänger etc).  |
| 70038  | Der SMTP-Server hat die Weiterleitung bzw. den Versand der E-Mail abgelehnt, weil die Domäne des Empfängers dem Server nicht bekannt ist oder weil der SMTP-Server eine Authentifizierung benötigt.<br>Inhalt der E-Mail wird verworfen.   | Domäne der Empfänger Adresse überprüfen oder Authentifizierung am SMTP-Server deaktivieren falls möglich. Eine SMTP-Authentifizierung wird gegenwärtig von WinCC Runtime nicht behandelt.                           |
| 70039  | Die Syntax der E-Mail Adresse ist falsch oder enthält ungültige Zeichen.<br>E-Mail Inhalt wird verworfen.  | E-Mail Adresse des Empfängers Adresse überprüfen.   |
| 70040  | Die Syntax der E-Mail Adresse ist falsch oder enthält ungültige Zeichen.   | --  |

| Nummer | Wirkung/Ursachen  | Abhilfe   |
|--------|---|---|
| 70041  | Der Import der Benutzerverwaltung wurde wegen eines Fehlers abgebrochen.<br>Es wurde kein Import durchgeführt.                                      | Überprüfen Sie ihre Benutzerverwaltung oder laden Sie Ihre Benutzerverwaltung erneut auf das Panel. |
| 70042  | Der Wertebereich der Variablen wurde bei Ausführung der Systemfunktion überschritten.<br>Die Berechnung der Systemfunktion wird nicht durchgeführt. | Überprüfen Sie die gewünschte Berechnung und korrigieren Sie diese gegebenenfalls.                  |
| 70043  | Der Wertebereich der Variablen wurde bei Ausführung der Systemfunktion überschritten.<br>Die Berechnung der Systemfunktion wird nicht durchgeführt. | Überprüfen Sie die gewünschte Berechnung und korrigieren Sie diese gegebenenfalls.                  |
| 70044  | Beim Senden der E-Mails ist ein Fehler aufgetreten.<br>Die E-Mails wurden nicht gesendet.   | Überprüfen Sie die SMTP-Einstellungen und beachten Sie die Fehlermeldung in der Systemmeldung.      |
| 70045  | Eine Datei, die für Verschlüsselung der E-Mail notwendig ist, konnte nicht geladen werden.  | Aktualisieren Sie das Betriebssystem und die Runtime.   |
| 70046  | Der Server unterstützt die Verschlüsselung nicht.   | Wählen Sie einen SMTP-Server, der eine Verschlüsselung unterstützt.                                 |
| 70047  | Möglicherweise passen die SSL-Versionen von Bediengerät und SMTP-Server nicht zusammen.   | Kontaktieren Sie Ihren Netzwerkadministrator oder den Betreiber des SMTP-Servers.                   |

## 110000 - Meldungen Offline Funktionen

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können.

Tabelle 10-6 110000 - Meldungen Offline Funktionen

| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 110000 | Es wurde ein Wechsel der Betriebsart durchgeführt.<br>Die Betriebsart ist jetzt "Offline".   | --   |
| 110001 | Es wurde ein Wechsel der Betriebsart durchgeführt.<br>Die Betriebsart ist jetzt "Online".  | --   |
| 110002 | Die Betriebsart wurde nicht gewechselt.  | Überprüfen Sie die Verbindung zu den Steuerungen.<br>Überprüfen Sie, ob der Adressbereich für den Bereichszeiger 88"Koordinierung" in der Steuerung vorhanden ist. |
| 110003 | Die Betriebsart der angegebenen Steuerung wurde durch die Systemfunktion "SetzeVerbindungsmodus" gewechselt.<br>Die Betriebsart ist jetzt "Offline". | --   |
| 110004 | Die Betriebsart der angegebenen Steuerung wurde durch die Systemfunktion "SetzeVerbindungsmodus" gewechselt.<br>Die Betriebsart ist jetzt "Online".  | --   |

| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 110005 | Es wurde versucht, über die Systemfunktion "SetzeVerbindungsmodus" die angegebene Steuerung in die Betriebsart "Online" zu schalten, obwohl sich das Gesamtsystem in der Betriebsart "Offline" befindet. Diese Umschaltung ist nicht zulässig. Die Betriebsart der Steuerung bleibt "Offline". | Schalten Sie das Gesamtsystem in die Betriebsart "Online" und führen Sie dann die Systemfunktion erneut aus.   |
| 110006 | Der Inhalt des Bereichszeigers "Projektkennung" stimmt nicht mit der in WinCC projektierten Projektkennung überein. Deshalb wird WinCC Runtime beendet.  | Überprüfen Sie: <ul style="list-style-type: none"> <li>die in der Steuerung eingetragene Projektkennung.</li> <li>die in WinCC eingetragene Projektkennung.</li> </ul> |

## 120000 - Meldungen Kurvendarstellung

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können.

Tabelle 10-7 120000 - Meldungen Kurvendarstellung

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 120000 | Die Kurve wird nicht dargestellt, da eine falsche Achse zur Kurve bzw. eine falsche Kurve projektiert wurde.        | Ändern Sie die Projektierung.  |
| 120001 | Die Kurve wird nicht dargestellt, da eine falsche Achse zur Kurve bzw. eine falsche Kurve projektiert wurde.        | Ändern Sie die Projektierung.  |
| 120002 | Die Kurve wird nicht dargestellt, da die zugeordnete Variable auf eine ungültige Adresse in der Steuerung zugreift. | Überprüfen Sie, ob der Datenbereich für die Variable in der Steuerung existiert, ob die projektierte Adresse richtig ist, oder ob der Wertebereich der Variablen stimmt. |

## 140000 - Meldungen Verbindung: Verbindung + Gerät

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können.

Tabelle 10-8 140000 - Meldungen Verbindung: Verbindung + Gerät

| Nummer | Wirkung/Ursachen   | Abhilfe |
|--------|--|---------|
| 140000 | Online-Verbindung zur Steuerung wurde erfolgreich aufgebaut. | --      |
| 140001 | Online-Verbindung zur Steuerung wurde abgebaut.              | --      |

| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 140003 | Es erfolgt keine Variablenaktualisierung bzw. Schreiben.   | Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist.<br>Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" die eingestellten Parameter.<br>Führen Sie einen Neuanlauf durch.   |
| 140004 | Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da der Zugangspunkt oder die Baugruppenparametrierung fehlerhaft ist.   | Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist.<br>Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" den Zugangspunkt bzw. die Baugruppenparametrierung (MPI, PPI, PROFIBUS).<br>Führen Sie einen Neuanlauf durch.               |
| 140005 | Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Adresse des Bediengerätes fehlerhaft ist (eventuell zu groß).  | Verwenden Sie eine andere Adresse für das Bediengerät.<br>Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist.<br>Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" die eingestellten Parameter.<br>Führen Sie einen Neuanlauf durch. |
| 140006 | Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Baudrate fehlerhaft ist.   | Wählen Sie in WinCC eine andere Baudrate (abhängig von Baugruppe, Profil, Kommunikationspartner etc.).   |
| 140007 | Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da das Busprofil fehlerhaft ist (s. %1).<br>Folgende Parameter können nicht in die Registrierungsdatenbank eingetragen werden:<br>1: Tslot<br>2: Tqui<br>3: Tset<br>4: MinTsdr<br>5: MaxTsdr<br>6: Trdy<br>7: Tid1<br>8: Tid2<br>9: Gap Faktor<br>10: Retry Limit   | Überprüfen Sie das benutzerdefinierte Busprofil.<br>Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist.<br>Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" die eingestellten Parameter.<br>Führen Sie einen Neuanlauf durch.       |
| 140008 | Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Projektierungsdaten fehlerhaft sind: Folgende Parameter können nicht in die Registrierungsdatenbank eingetragen werden:<br>0: allgemeiner Fehler<br>1: falsche Version<br>2: Profil kann nicht in die Registrierungsdatenbank eingetragen werden.<br>3: Subnet-Type kann nicht in die Registrierungsdatenbank eingetragen werden.<br>4: Target Rotation Time kann nicht in die Registrierungsdatenbank eingetragen werden.<br>5: Höchste Adresse (HSA) fehlerhaft. | Kontrollieren Sie die Verbindung und ob die Steuerung eingeschaltet ist.<br>Überprüfen Sie in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" die eingestellten Parameter.<br>Führen Sie einen Neuanlauf durch.   |
| 140009 | Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Baugruppe für die S7-Kommunikation nicht gefunden wurde.   | Installieren Sie die Baugruppe in der Systemsteuerung mit "PG/PC-Schnittstelle einstellen" neu.  |



| Nummer | Wirkung/Ursachen  | Abhilfe   |
|--------|---|---|
| 140010 | Es wurde kein S7-Kommunikationspartner gefunden, da die Steuerung ausgeschaltet ist.<br>DP/T:<br>In der Systemsteuerung ist unter "PG/PC-Schnittstelle einstellen" die Option "PG/PC ist einziger Master am Bus" nicht eingestellt.   | Schalten Sie die Steuerung ein.<br>DP/T:<br>Befindet sich nur ein Master am Netz, aktivieren Sie unter "PG/PC-Schnittstelle einstellen" die Option "PG/PC ist einziger Master am Bus".<br>Befinden sich mehr als ein Master am Netz, schalten Sie diese Master ein. Ändern Sie dabei keine Einstellungen, sonst kommt es zu Busstörungen. |
| 140011 | Es erfolgt keine Variablenaktualisierung bzw. Schreiben, da die Kommunikation unterbrochen ist.   | Kontrollieren Sie die Verbindung und ob der Kommunikationspartner eingeschaltet ist.  |
| 140012 | Es liegt ein Initialisierungsproblem vor (z. B. wenn WinCC Runtime im Task-Manager beendet wurde).<br>Oder:<br>ein weiteres Programm (z. B. STEP 7) ist bereits mit anderen Busparametern aktiv und die Treiber können mit den neuen Busparametern (z. B. Baudrate) nicht gestartet werden. | Starten Sie das Bediengerät neu.<br>Oder:<br>Starten Sie zuerst WinCC Runtime und danach weitere Programme.   |
| 140013 | Das MPI-Kabel steckt nicht und damit fehlt die Stromversorgung.   | Überprüfen Sie die Verbindungen.  |
| 140014 | Projektierte Adresse am Bus bereits belegt.   | Ändern Sie in der Projektierung unter Steuerung die Adresse des Bediengerätes.  |
| 140015 | Falsche Baudrate<br>Oder:<br>Falsche Busparameter (z. B. HSA)<br>Oder:<br>OP-Adresse > HSA oder: falscher Interruptvektor (Interrupt kommt nicht zum Treiber durch)   | Korrigieren Sie die falschen Parameter.   |
| 140016 | Eingestellter Interrupt wird nicht von der Hardware unterstützt.  | Ändern Sie die Interruptnummer.   |
| 140017 | Eingestellter Interrupt wird von einem anderen Treiber genutzt.   | Ändern Sie die Interruptnummer.   |
| 140018 | Der Konsistenzcheck wurde durch SIMOTION Scout deaktiviert. Es erscheint nur ein entsprechender Hinweis.  | Aktivieren Sie den Konsistenzcheck mit SIMOTION Scout erneut und laden Sie das Projekt erneut in die Steuerung.   |
| 140019 | SIMOTION Scout lädt ein neues Projekt in die Steuerung. Die Verbindung zur Steuerung wird abgebrochen.  | Warten Sie das Ende der Umkonfiguration ab.   |
| 140020 | Die Version in der Steuerung und die Version in der Projektierung (FWX-Datei) stimmen nicht überein. Die Verbindung zur Steuerung wird abgebrochen.   | Zur Abhilfe stehen folgende Möglichkeiten zur Verfügung:<br>Laden Sie mit SIMOTION Scout die aktuelle Version in die Steuerung.<br>Übersetzen Sie das Projekt mit WinCC ES neu, beenden Sie WinCC Runtime und starten Sie mit neuer Projektierung.  |
| 140021 | Die Verbindung zur Steuerung wurde nicht aufgebaut. Das "Zugangspasswort" für die Verbindung zur Steuerung wurde falsch projektiert.  | Überprüfen Sie im Editor "Verbindungen" im Bereich "Zugangspasswort" das eingegebene Passwort für die Verbindung zur Steuerung.<br>Vergeben Sie das korrekte Passwort.<br>Das "Passwort" für die Verbindung zur Steuerung wird in den Eigenschaften der Steuerung unter "Schutz" vergeben.  |

| Nummer | Wirkung/Ursachen  | Abhilfe   |
|--------|---|---|
| 140022 | Die Verbindung zur Steuerung wurde nicht aufgebaut.<br>Es wurde kein Zugangspasswort für die Verbindung zur Steuerung projiziert. | Geben Sie im Editor "Verbindungen" im Bereich "Zugangspasswort" das Passwort für die Verbindung zur Steuerung ein.<br>Das "Passwort" wird in den Eigenschaften der Steuerung unter "Schutz" vergeben. |
| 140025 | Die Verbindung zur Steuerung wurde nicht aufgebaut.<br>Im Display der Steuerung wurde das Zugangspasswort der Steuerung gesperrt. | Entsperren Sie das Zugangspasswort im Display der Steuerung.  |

## 180000 - Meldungen Allgemeines

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können.

Tabelle 10-9 180000 - Meldungen Allgemeines

| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 180000 | Eine Komponente/OCX erhielt Projektierungsdaten mit einer Versionskennung, die nicht unterstützt wird.   | Installieren Sie eine neuere Komponente.   |
| 180001 | Das System ist überlastet, da zu viele Aktionen gleichzeitig aktiviert wurden. Es können nicht alle Aktionen ausgeführt werden, einige werden verworfen.             | Es stehen mehrere verschiedene Abhilfemöglichkeiten zur Verfügung: <ul style="list-style-type: none"> <li>• Erzeugen Sie Meldungen langsamer (pollen).</li> <li>• Stoßen Sie Skripte und Systemfunktionen in größeren Zeitabständen an.</li> </ul> Falls die Meldung häufiger erscheint:<br>Starten Sie das Bediengerät neu. |
| 180002 | Die Bildschirmtastatur konnte nicht aktiviert werden. Mögliche Ursache:<br>Die Datei "TouchInputPC.exe" wurde durch fehlerhaft ausgeführtes Setup nicht registriert. | Installieren Sie WinCC Runtime erneut.   |

## 190000 - Meldungen Variablen

### Bedeutung der Systemmeldungen

Nachfolgend werden Ihnen alle Systemmeldungen dargestellt, die angezeigt werden können.

Tabelle 10-10 190000 - Meldungen Variablen

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 190000 | Die Variable wird evtl. nicht aktualisiert.   | --   |
| 190001 | Die Variable wird nach einem fehlerhaften Zustand wieder aktualisiert, nachdem der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb).   | --   |
| 190002 | Die Variable wird nicht aktualisiert, da die Kommunikation mit der Steuerung unterbrochen ist.  | Schalten Sie die Kommunikation über die Systemfunktion "SetOnline" ein.  |
| 190004 | Die Variable wird nicht aktualisiert, da die projektierte Adresse für diese Variable nicht vorhanden ist.   | Überprüfen Sie die Projektierung.  |
| 190005 | Die Variable wird nicht aktualisiert, da der projektierte Steuerungstyp für diese Variable nicht existiert.   | Überprüfen Sie die Projektierung.  |
| 190006 | Die Variable wird nicht aktualisiert, da eine Abbildung vom Steuerungstyp in den Datentyp der Variablen nicht möglich ist.  | Überprüfen Sie die Projektierung.  |
| 190007 | Der Variablenwert wird nicht geändert, da die Verbindung zur Steuerung unterbrochen oder die Variable offline ist.  | Schalten Sie Online bzw. stellen Sie die Verbindung zur Steuerung wieder her.  |
| 190008 | Die projektierten Grenzwerte der Variablen wurden durch eines der folgenden Ereignisse verletzt: <ul style="list-style-type: none"> <li>• Werteingabe</li> <li>• Systemfunktion</li> <li>• Skript</li> </ul>  | Beachten Sie die projektierten oder aktuellen Grenzwerte der Variablen.  |
| 190009 | Es wurde versucht, der Variablen einen Wert zuzuweisen, der außerhalb des für diesen Datentyp zulässigen Wertebereichs liegt.<br>Z. B. Werteingabe von 260 für eine Byte-Variable oder Werteingabe von -3 für eine vorzeichenlose Wort-Variable.  | Beachten Sie den Wertebereich des Datentyps der Variablen.   |
| 190010 | Die Variable wird zu oft mit Werten beschrieben (z. B. in einer Schleife von einem Skript aus). Es gehen Werte verloren, da maximal 100 Vorgänge zwischengespeichert werden.  | Zur Abhilfe stehen folgende Möglichkeiten zur Verfügung: <ul style="list-style-type: none"> <li>• Erhöhen Sie die Zeit zwischen mehrmaligem Schreiben.</li> <li>• Verwenden Sie bei einer projektierten Quittierung am Bediengerät für "HMI-Quittvariable" keine Arrayvariable, die länger als 6 Worte ist.</li> </ul> |
| 190011 | Mögliche Ursache 1:<br>Der eingegebene Wert konnte nicht in die projektierte Steuerungsvariable geschrieben werden, da der Wertebereich über- oder unterschritten wurde.<br>Die Eingabe wurde verworfen und der ursprüngliche Wert wurde wieder hergestellt.<br>Mögliche Ursache 2:<br>Die Verbindung zur Steuerung wurde unterbrochen. | Beachten Sie, dass der eingegebene Wert innerhalb des Wertebereichs der Steuerungsvariablen liegen muss.<br><br>Kontrollieren Sie die Verbindung zur Steuerung.  |

| Nummer | Wirkung/Ursachen   | Abhilfe   |
|--------|--|---|
| 190012 | Es ist nicht möglich den Wert von einem Quellformat in ein Zielformat zu wandeln, z. B.:<br>Es soll ein Wert außerhalb des gültigen steuerungsabhängigen Wertebereichs für einen Zähler geschrieben werden.<br>Einer Variablen von Typ Integer soll ein Wert vom Typ String zugewiesen werden. | Kontrollieren Sie den Wertebereich oder den Datentyp der Variablen.           |
| 190013 | Der Benutzer hat einen String eingegeben, dessen Länge größer als die der Variable ist. Der String wird automatisch auf die zulässige Länge gekürzt.   | Geben Sie nur Strings ein, welche die Länge der Variable nicht überschreiten. |

**190100 - Meldungen Bereichszeiger**

**190100 - Meldungen Bereichszeiger**

| Nummer | Wirkung/Ursachen  | Abhilfe                           |
|--------|---|-----------------------------------|
| 190100 | Der Bereichszeiger wird nicht aktualisiert, da die projektierte Adresse für diesen Bereichszeiger nicht vorhanden ist.<br>Typ:<br>1 Warnungen<br>2 Fehler<br>3 Quittierung Steuerung<br>4 Quittierung Bediengerät<br>5 LED-Abbild<br>6 Kurvenanforderung<br>7 Kurvenübertragung 1<br>8 Kurvenübertragung 2<br>Nr.:<br>ist die in WinCC ES angezeigte fortlaufende Nummer. | Überprüfen Sie die Projektierung. |
| 190101 | Der Bereichszeiger wird nicht aktualisiert, da eine Abbildung vom Steuerungstyp in den Typ des Bereichszeigers nicht möglich ist.<br>Parameter Typ und Nr.:<br>siehe Meldung 190100   | --                                |
| 190102 | Der Bereichszeiger wird nach einem fehlerhaften Zustand wieder aktualisiert, da der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb).Parameter Typ und Nr.: Siehe Meldung 190100   | --                                |

**200000 - Meldungen PLC-Koordination****200000 - Meldungen PLC-Koordination**

| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 200000 | Die Koordinierung wird nicht ausgeführt, da die projektierte Adresse in der Steuerung nicht vorhanden/eingerichtet ist.  | Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung ein.  |
| 200001 | Die Koordinierung wird nicht ausgeführt, da an die projektierte Adresse in der Steuerung nicht geschrieben werden kann.  | Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung in einem schreibbaren Bereich ein.  |
| 200002 | Die Koordinierung wird momentan nicht ausgeführt, da das Adressformat des Bereichszeigers nicht zum internen Ablageformat passt.   | Interner Fehler  |
| 200003 | Die Koordinierung wird wieder ausgeführt, da der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb).  | --   |
| 200004 | Die Koordinierung wird evtl. nicht ausgeführt.   | --   |
| 200005 | Es werden keine Daten mehr geschrieben oder gelesen. Mögliche Ursachen: <ul style="list-style-type: none"> <li>• Das Kabel ist unterbrochen.</li> <li>• Die Steuerung antwortet nicht, ist defekt, etc.</li> <li>• Das System ist überlastet.</li> </ul> | Überprüfen Sie, ob das Kabel steckt oder die Steuerung in Ordnung ist.<br>Starten Sie neu, wenn die Systemmeldung weiterhin dauerhaft ansteht. |

**210000 - Meldungen PLC-Job****210000 - Meldungen PLC-Job**

| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 210000 | Aufträge werden nicht bearbeitet, da die projektierte Adresse in der Steuerung nicht vorhanden/eingerichtet ist. | Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung ein.  |
| 210001 | Aufträge werden nicht bearbeitet, da die projektierte Adresse in der Steuerung nicht lesbar/schreibbar ist.      | Ändern Sie die Adresse oder richten Sie die Adresse in der Steuerung in einem lesbaren/schreibbaren Bereich ein. |
| 210002 | Aufträge werden nicht ausgeführt, da das Adressformat des Bereichszeigers nicht zum internen Ablageformat passt. | Interner Fehler  |
| 210003 | Das Auftragsfach wird wieder bearbeitet, da der letzte Fehlerzustand beseitigt ist (Rückkehr zum Normalbetrieb). | --   |
| 210004 | Das Auftragsfach wird evtl. nicht bearbeitet.  | --   |

| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 210005 | Es wurde ein Steuerungsauftrag mit einer unzulässigen Nummer angestoßen.   | Überprüfen Sie das Steuerungsprogramm.   |
| 210006 | Während der Ausführung des Steuerungsauftrags trat ein Fehler auf. Der Steuerungsauftrag wird deshalb nicht ausgeführt. Beachten Sie gegebenenfalls auch die nachfolgende / vorhergehende Systemmeldung. | Überprüfen Sie die Parameter des Steuerungsauftrags. Übersetzen Sie die Projektierung neu. |

**220000 - Meldungen WinCC-Kommunikationstreiber**

**220000 - Meldungen WinCC-Kommunikationstreiber**

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 220001 | Die Variable wird nicht übertragen, da der unterlagerte Kommunikationstreiber/das Bediengerät beim Schreiben den Datentyp Bool/Bit nicht unterstützt. | Ändern Sie die Projektierung.  |
| 220002 | Die Variable wird nicht übertragen, da der unterlagerte Kommunikationstreiber/das Bediengerät beim Schreiben den Datentyp Byte nicht unterstützt.     | Ändern Sie die Projektierung.  |
| 220003 | Der Kommunikationstreiber konnte nicht geladen werden. Evtl. ist der Treiber nicht installiert.   | Installieren Sie den Treiber indem Sie WinCC Runtime neu installieren. |
| 220004 | Die Kommunikation ist unterbrochen, es erfolgt keine Aktualisierung, da das Kabel nicht steckt, oder defekt ist, etc.                                 | Überprüfen Sie die Verbindung.   |
| 220005 | Die Kommunikation läuft.  | --   |
| 220006 | Die Verbindung zur angegebenen Steuerung an der angegebenen Schnittstelle ist hergestellt.  | --   |

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 220007 | Die Verbindung zur angegebenen Steuerung an der angegebenen Schnittstelle ist unterbrochen.   | Überprüfen Sie Folgendes: <ul style="list-style-type: none"> <li>• Steckt das Kabel?</li> <li>• Ist die Steuerung in Ordnung?</li> <li>• Wird die richtige Schnittstelle verwendet?</li> <li>• Ist ihre Projektierung in Ordnung (Schnittstellenparameter, Protokolleinstellungen, Steuerungsadresse)?</li> </ul> Starten Sie neu, wenn die Systemmeldung weiterhin dauerhaft ansteht. |
| 220008 | Der Kommunikationstreiber kann nicht auf die angegebene Schnittstelle zugreifen oder diese öffnen. Möglicherweise verwendet bereits ein anderes Programm diese Schnittstelle oder es wird eine nicht am Zielgerät vorhandene Schnittstelle verwendet.<br>Es findet keine Kommunikation mit der Steuerung statt. | Beenden Sie alle Programme, die auf die Schnittstelle zugreifen und starten Sie den Rechner neu.<br>Verwenden Sie eine andere, im System vorhandene, Schnittstelle.  |

### 230000 - Meldungen Bildobjekte

### 230000 - Meldungen Bildobjekte

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 230000 | Der eingegebene Wert konnte nicht übernommen werden. Die Eingabe wird verworfen und der vorherige Wert wird wieder hergestellt.<br>Mögliche Ursache: <ul style="list-style-type: none"> <li>• Der Wertebereich wurde überschritten</li> <li>• Sie haben unzulässige Zeichen eingegeben</li> <li>• Die maximal zulässige Anzahl der Benutzer wurde überschritten.</li> </ul> | Geben Sie einen sinnvollen Wert ein oder löschen Sie einen nicht mehr benötigten Benutzer. |
| 230002 | Da der angemeldete Benutzer die erforderliche Berechtigung nicht besitzt, wird die Eingabe verworfen und der vorherige Wert wird wiederhergestellt.   | Melden Sie sich als Benutzer mit ausreichender Berechtigung an.                            |
| 230003 | Der Wechsel zum angegebenen Bild wird nicht durchgeführt, da das Bild nicht vorhanden/projektiert ist. Das bisherige Bild bleibt angewählt.   | Projektieren Sie das Bild und überprüfen Sie die Anwahlfunktion.                           |
| 230005 | Der Wertebereich der Variablen im E/A-Feld wurde überschritten.<br>Der ursprüngliche Wert der Variable bleibt erhalten.   | Berücksichtigen Sie bei der Eingabe den Wertebereich der Variablen.                        |

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 230100 | Nach Navigation im Web-Browser wurde eine Meldung zurückgeliefert, die für den Benutzer interessant sein könnte.<br>Der Web-Browser läuft weiter, zeigt die neue Seite aber evtl. nicht (vollständig) an.   | Auf eine andere Seite navigieren.  |
| 230200 | Die Verbindung des HTTP-Kanals wurde unterbrochen, weil ein Fehler aufgetreten ist. Dieser Fehler wird durch eine weitere Systemmeldung näher erläutert.<br>Es werden keine Daten mehr ausgetauscht.  | Überprüfen Sie die Netzwerkverbindung.<br>Überprüfen Sie die Konfiguration des Servers.  |
| 230201 | Die Verbindung des HTTP-Kanals wurde hergestellt.<br>Es werden Daten ausgetauscht.  | --   |
| 230202 | Die WININET.DLL hat einen Fehler festgestellt. Dieser Fehler tritt zumeist auf wenn keine Verbindung zum Server möglich ist, oder der Server eine Verbindung ablehnt, weil der Client sich nicht richtig autorisiert hat.<br>Bei einer Verschlüsselten Verbindung über SSL kann auch ein nicht akzeptiertes Server-Zertifikat die Ursache sein.<br>Nähere Erläuterung liefert der Fehlertext in der Meldung.<br>Dieser Text ist immer in der Sprache der Windows-Installation, da er von Windows geliefert wird.<br>Es werden keine Prozesswerte ausgetauscht. Möglicherweise wird der Teil der Meldung nicht angezeigt, der vom Betriebssystem Windows geliefert wird, z. B. "Ein Fehler ist aufgetreten. Die WININET.DLL liefert folgenden Fehler: Nummer: 12055 Text:HTTP: <no error text available>." | Abhängig von der Ursache:<br>Wenn die Verbindung nicht hergestellt werden kann oder ein Timeout auftritt: <ul style="list-style-type: none"> <li>• Netzwerkverbindung und Netzwerk überprüfen.</li> <li>• Server-Adresse überprüfen.</li> <li>• Überprüfen, ob der WebServer auf dem Zielrechner tatsächlich läuft.</li> </ul> Bei fehlerhafter Autorisierung: <ul style="list-style-type: none"> <li>• Projektierter Benutzername und/oder Passwort stimmen nicht mit denen des Servers überein. Übereinstimmung herstellen.</li> </ul> Bei nicht akzeptierten Server-Zertifikat:<br>Zertifikat von unbekanntem CA ( ) signiert: <ul style="list-style-type: none"> <li>• Ignorieren Sie entweder diesen Punkt oder installieren Sie ein Zertifikat, das mit einem dem Clientrechner bekannten Root-Zertifikat signiert wurde.</li> </ul> Bei ungültigen Datum des Zertifikats: <ul style="list-style-type: none"> <li>• Ignorieren Sie entweder diesen Punkt oder installieren Sie ein Zertifikat mit gültigem Datum auf dem Server.</li> </ul> Bei ungültigen CN (Common Name oder Computer Name): <ul style="list-style-type: none"> <li>• Ignorieren Sie entweder diesen Punkt oder installieren Sie ein Zertifikat mit einem der Server-Adresse entsprechenden Namen.</li> </ul> |



| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 230203 | <p>Obwohl eine Verbindung zum Server möglich ist, hat der HTTP-Server die Verbindung abgelehnt.<br/>Mögliche Ursachen:</p> <ul style="list-style-type: none"> <li>• WinCC Runtime läuft nicht auf dem Server nicht läuft</li> <li>• Der HTTP-Kanal wird nicht unterstützt (503 Service unavailable).</li> </ul> <p>Andere Fehler können nur auftreten, wenn der Webserver den HTTP-Kanal nicht unterstützt.<br/>Die Sprache des Fehlertext hängt vom Webserver ab.<br/>Es werden keine Daten ausgetauscht.</p> | <p>Bei Fehler 503 Service unavailable:<br/>Überprüfen Sie, ob WinCC Runtime auf dem Server läuft und ob der HTTP-Kanal unterstützt wird.</p> |
| 230301 | <p>Ein interner Fehler ist aufgetreten. Ein englischer Text erläutert den Fehler in der Meldung noch etwas genauer. Eine mögliche Ursache kann z.B. ungenügender Speicher sein.<br/>Das OCX funktioniert nicht.</p>  | --   |
| 230302 | <p>Der Name des Remote-Servers kann nicht aufgelöst werden.<br/>Es kann keine Verbindung hergestellt werden.</p>   | <p>Überprüfen Sie die projektierte Server-Adresse.<br/>Überprüfen Sie, ob der DNS-Service des Netzwerkes aktiv ist.</p>                      |
| 230303 | <p>Der Remote-Server ist auf dem adressierten Rechner nicht aktiv.<br/>Die Server-Adresse ist falsch.<br/>Es kann keine Verbindung hergestellt werden.</p>   | <p>Überprüfen Sie die projektierte Server-Adresse.<br/>Überprüfen Sie, ob der Remote-Server des Zielrechners läuft.</p>                      |
| 230304 | <p>Der Remote-Server des adressierten Rechners ist inkompatibel zum VNCOCX.<br/>Es kann keine Verbindung hergestellt werden.</p>   | <p>Verwenden Sie einen kompatiblen Remote-Server.</p>  |
| 230305 | <p>Die Authentifizierung ist wegen falschem Kennwort gescheitert.<br/>Es kann keine Verbindung hergestellt werden.</p>   | <p>Projektieren Sie das richtige Kennwort.</p>   |
| 230306 | <p>Die Verbindung zum Remote-Server ist gestört. Dies kann bei Netzproblemen auftreten.<br/>Es kann keine Verbindung hergestellt werden.</p>   | <p>Überprüfen Sie, ob das Netzkabel steckt oder Netzprobleme vorliegen.</p>  |
| 230307 | <p>Die Verbindung wurde vom Remote-Server beendet. Mögliche Ursachen:</p> <ul style="list-style-type: none"> <li>• Der Remote-Server wurde beendet</li> <li>• Der Benutzer hat den Server angewiesen hat, alle Verbindungen zu beenden.</li> </ul> <p>Die Verbindung wird abgebrochen.</p>   | --   |
| 230308 | <p>Diese Meldung informiert Sie über den Verbindungsaufbau.<br/>Eine Verbindung wird gerade aufgebaut.</p>   | --   |

**260000 - Meldungen Passwortssystem**

**260000 - Meldungen Passwortssystem**

| Nummer | Wirkung/Ursachen  | Abhilfe   |
|--------|---|---|
| 260000 | Es wurde ein unbekannter Benutzer oder ein unbekanntes Kennwort am System eingegeben. Der aktuelle Benutzer wird vom System abgemeldet.   | Melden Sie sich als Benutzer mit gültigem Kennwort am System an.  |
| 260001 | Der angemeldete Benutzer hat nicht die ausreichende Berechtigung, um die geschützte Funktionalität auszuführen.   | Melden Sie sich als Benutzer am System an, der die notwendige Berechtigung besitzt.   |
| 260002 | Diese Meldung wird bei Auslösung der Systemfunktion "VerfolgeBenutzeraenderung" ausgegeben.   | --  |
| 260003 | Der Benutzer hat sich am System abgemeldet.   | --  |
| 260004 | Der in der Benutzeranzeige neu eingegebene Benutzername ist bereits in der Benutzerverwaltung vorhanden.  | Wählen Sie einen anderen Benutzernamen, da Benutzernamen in der Benutzerverwaltung eindeutig sein müssen.                               |
| 260005 | Eingabe wird verworfen.   | Kürzeren Benutzernamen eingeben.  |
| 260006 | Eingabe wird verworfen.   | Kürzeres bzw. längeres Kennwort eingeben.   |
| 260007 | Die eingegebene Abmeldezeit liegt außerhalb des gültigen Bereichs von 0 bis 60 Minuten. Der eingegebene Wert wird verworfen und der ursprüngliche Wert bleibt bestehen.                         | Geben Sie einen Wert zwischen 0 und 60 Minuten für die Abmeldezeit ein.   |
| 260008 | Es wurde versucht, eine mit ProTool V 6.0 erstellte Datei PTProRun.pwl, in WinCC zu lesen.<br>Das Lesen der Datei wurde wegen Inkompatibilität des Formats abgebrochen.                         | --  |
| 260009 | Sie haben versucht einen der Benutzer "Administrator" oder "PLC User" zu löschen. Diese beiden Benutzer sind fester Bestandteil der Benutzerverwaltung und dürfen nicht gelöscht werden.        | Falls es nötig ist, einen Benutzer zu löschen, z.B. wenn maximale Anzahl der Benutzer erreicht ist, löschen sie einen anderen Benutzer. |
| 260012 | Das im Dialog "Kennwort ändern" eingegebenen Kennwort und die Bestätigung stimmen nicht überein.<br>Das Kennwort wird nicht geändert. Der aktuell angemeldete Benutzer wird abgemeldet.         | Sie müssen sich erneut am System anmelden. Anschließend geben Sie zwei identische Kennwörter ein, um das Kennwort zu ändern.            |
| 260013 | Das im Dialog "Kennwort ändern" eingegebene Kennwort wurde bereits verwendet und ist daher nicht gültig.<br>Das Kennwort wird nicht geändert. Der aktuell angemeldete Benutzer wird abgemeldet. | Sie müssen sich erneut am System anmelden. Anschließend geben Sie ein neues Kennwort ein, das bisher noch nicht verwendet wurde.        |
| 260014 | Sie haben dreimal in Folge versucht sich mit einem falschen Kennwort anzumelden. Sie werden gesperrt und der Gruppe Nr. 0 zugewiesen.   | Sie können sich mit Ihrem richtigen Kennwort am System anmelden. Die Gruppenzuordnung kann nur ein Administrator ändern.                |

| Nummer | Wirkung/Ursachen   | Abhilfe  |
|--------|--|--|
| 260024 | Das eingegebene Kennwort entspricht nicht den erforderlichen Sicherheitsrichtlinien.   | Geben Sie ein Kennwort ein, das aus mindestens einer Zahl besteht.   |
| 260025 | Das eingegebene Kennwort entspricht nicht den erforderlichen Sicherheitsrichtlinien.   | Geben Sie ein Kennwort ein, das mindestens drei Zeichen beinhaltet.  |
| 260028 | Es wurde versucht beim Systemstart, Anmeldeversuch oder beim Versuch das Passwort eines SIMATIC Logon Benutzers zu ändern, auf den SIMATIC Logon Server zuzugreifen.<br>Beim Anmeldeversuch wird der neue Benutzer nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet. | Überprüfen Sie die Verbindung zum SIMATIC Logon Server und dessen Konfiguration, z. B.:<br>1. Portnummer<br>2. IP-Adresse<br>3. Servername<br>4. Funktionsfähiges Transferkabel.<br>Oder verwenden Sie einen lokalen Benutzer. |
| 260030 | Der SIMATIC Logon Benutzer konnte sein Kennwort nicht auf dem SIMATIC Logon Server ändern. Möglicherweise entspricht das neue Kennwort nicht den Kennwortregeln am Server oder der Benutzer hat nicht das Recht, sein Kennwort zu ändern.<br>Das alte Kennwort bleibt und der Benutzer wird abgemeldet.      | Melden Sie sich erneut mit einem anderen Kennwort an. Überprüfen Sie die Kennwortregeln am SIMATIC Logon Server.   |
| 260033 | Die Aktion Kennwort ändern oder Benutzer anmelden konnte nicht durchgeführt werden.  | Überprüfen Sie die Verbindung zum SIMATIC Logon Server und dessen Konfiguration, z. B.:<br>1. Portnummer<br>2. IP-Adresse<br>3. Servername<br>4. Funktionsfähiges Transferkabel<br>Oder verwenden Sie einen lokalen Benutzer.  |
| 260034 | Der letzte Anmeldevorgang wurde noch nicht abgeschlossen. Eine Benutzeraktion oder ein Anmeldedialog kann daher nicht aufgerufen werden.<br>Der Anmeldedialog wird nicht aufgeblendet. Die Benutzeraktion wird nicht ausgeführt.   | Warten Sie, bis der Anmeldevorgang abgeschlossen ist.  |
| 260035 | Der letzte Versuch, das Kennwort zu ändern, wurde noch nicht abgeschlossen. Eine Benutzeraktion oder ein Anmeldedialog kann daher nicht aufgerufen werden.<br>Der Anmeldedialog wird nicht aufgeblendet. Die Benutzeraktion wird nicht ausgeführt.   | Warten Sie, bis der Vorgang abgeschlossen ist.   |
| 260036 | Auf dem SIMATIC Logon Server fehlen Lizenzen. Die Anmeldung wird noch zugelassen.  | Kontrollieren Sie die Lizenzierung auf dem SIMATIC Logon Server.   |
| 260037 | Auf dem SIMATIC Logon Server ist keine Lizenz verfügbar. Eine Anmeldung ist nicht möglich.<br>Es ist keine Anmeldung über den SIMATIC Logon Server möglich, sondern nur noch über lokale Benutzer.   | Kontrollieren Sie die Lizenzierung auf dem SIMATIC Logon Server.   |

| Nummer | Wirkung/Ursachen   | Abhilfe   |
|--------|--|---|
| 260040 | Es wurde versucht beim Anmeldeversuch oder beim Versuch das Passwort eines SIMATIC Logon Benutzers zu ändern, auf den SIMATIC Logon Server zuzugreifen.<br>Beim Anmeldeversuch wird der neue Benutzer nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet.              | Überprüfen Sie die Verbindung zu der Domäne und deren Konfiguration in Editor Runtime-Sicherheitseinstellungen.<br>Oder verwenden Sie einen lokalen Benutzer. |
| 260043 | Es war nicht möglich den Benutzer am SIMATIC Logon Server anzumelden. Möglicherweise war der Benutzername oder das Kennwort falsch oder dem Benutzer fehlen die Rechte, sich anzumelden.<br>Der neue Benutzer wird nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet. | Versuchen Sie es erneut. Gegebenenfalls prüfen Sie die Benutzerdaten am SIMATIC Logon Server.   |
| 260044 | Es war nicht möglich den Benutzer am SIMATIC Logon Server anzumelden, da sein Account gesperrt ist.<br>Der neue Benutzer wird nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet.  | Überprüfen Sie die Benutzerdaten am SIMATIC Logon Server.   |
| 260045 | Der SIMATIC Logon Benutzer gehört keiner oder mehreren Gruppen an.<br>Der neue Benutzer wird nicht angemeldet. Wenn zuvor ein anderer Benutzer angemeldet war, wird dieser abgemeldet.   | Überprüfen Sie die Benutzerdaten am SIMATIC Logon Server und die Konfiguration in Ihrem WinCC Projekt. Ein Benutzer darf nur einer Gruppe zugeordnet sein.    |

## 270000 - Meldungen Meldungen System

## 270000 - Meldungen Meldungen System

| Nummer | Wirkung/Ursachen   | Abhilfe   |
|--------|--|---|
| 270000 | In der Meldung wird eine Variable nicht dargestellt, da sie auf eine ungültige Adresse in der Steuerung zugreift.  | Überprüfen Sie, ob der Datenbereich für die Variable in der Steuerung existiert, ob die projektierte Adresse richtig ist, ob der Wertebereich der Variablen stimmt. |
| 270001 | Es gibt eine geräteabhängige Anzahl von Meldungen, die maximal gleichzeitig anstehen dürfen, um angezeigt werden zu können (siehe Betriebsanleitungen). Diese Anzahl ist überschritten.<br>Die Anzeige enthält nicht mehr alle Meldungen. Im Meldepuffer werden jedoch alle Meldungen eingetragen. | --  |
| 270002 | Es werden Meldungen aus einem Archiv angezeigt, zu denen es im aktuellen Projekt keine Daten gibt.<br>Für die Meldungen werden Platzhalter ausgegeben.   | Löschen Sie gegebenenfalls alte Archivdaten.  |

| Nummer | Wirkung/Ursachen  | Abhilfe   |
|--------|---|---|
| 270003 | Der Dienst kann nicht eingerichtet werden, da zu viele Geräte diesen Dienst einrichten wollen. Maximal können vier Geräte diese Aktion ausführen. | Schließen Sie weniger Bediengeräte an, die den Dienst nutzen sollen.  |
| 270004 | Zugriff auf persistenten Meldepuffer nicht möglich. Meldungen können nicht wiederhergestellt und auch nicht gesichert werden.                     | Wenn beim nächsten Wiederanlauf erneut Probleme auftreten, wenden Sie sich bitte an den Customer Support (Flash löschen).                 |
| 270005 | Persistenter Meldepuffer beschädigt: Meldungen können nicht wiederhergestellt werden.   | Wenn beim nächsten Wiederanlauf erneut Probleme auftreten, wenden Sie sich bitte an den Customer Support (Flash löschen).                 |
| 270006 | Projekt geändert: Meldungen können aus dem persistenten Meldepuffer nicht wiederhergestellt werden.   | Projekt wurde übersetzt und neu auf das Bediengerät geladen; Der Fehler sollte beim nächsten Start des Bediengeräts nicht mehr auftreten. |
| 270007 | Ein Konfigurationsproblem verhindert die Wiederherstellung (z.B. DLL gelöscht, Verzeichnis umbenannt o.ä.).                                       | Bitte aktualisieren Sie das Betriebssystem und laden Sie Ihr Projekt neu auf das Bediengerät.   |

## 290000 - Meldungen Rezeptursystem

## 290000 - Meldungen Rezeptursystem

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 290000 | Die Rezepturvariable konnte nicht gelesen oder geschrieben werden. Sie wird mit dem Startwert belegt.<br>Die Meldung wird gegebenenfalls für bis zu vier weitere fehlerhafte Variablen im Meldepuffer eingetragen. Danach wird die Meldung Nr. 290003 ausgegeben.                                   | Überprüfen Sie in der Projektierung, ob die Adresse in der Steuerung eingerichtet ist.                 |
| 290001 | Es wurde versucht, der Rezepturvariablen einen Wert zuzuweisen, der außerhalb des Wertebereichs liegt, der für diesen Typ zulässig ist.<br>Die Meldung wird gegebenenfalls für bis zu vier weitere fehlerhafte Variablen im Meldepuffer eingetragen. Danach wird die Meldung Nr. 290004 ausgegeben. | Beachten Sie den Wertebereich des Variablentyps.   |
| 290002 | Es ist nicht möglich, den Wert von einem Quellformat in ein Zielformat zu wandeln.<br>Die Meldung wird gegebenenfalls für bis zu vier weitere fehlerhafte Rezepturvariablen im Meldepuffer eingetragen. Danach wird die Meldung Nr. 290005 ausgegeben.  | Kontrollieren Sie den Wertebereich oder den Typ der Variablen.   |
| 290003 | Diese Meldung werden ausgegeben, wenn die Meldung Nr. 290000 mehr als fünfmal ausgelöst wurde.<br>In diesem Fall wird keine Einzelmeldungen mehr erzeugt.   | Überprüfen Sie in der Projektierung, ob die Adressen der Variablen in der Steuerung eingerichtet sind. |

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 290004 | Diese Meldung wird ausgegeben, wenn die Meldung Nr. 290001 mehr als fünfmal ausgelöst wurde.<br>In diesem Fall werden keine Einzelmeldungen mehr erzeugt.   | Beachten Sie den Wertebereich des Variablentyps.   |
| 290005 | Diese Meldung wird ausgegeben, wenn die Meldung Nr. 290002 mehr als fünfmal ausgelöst wurde.<br>In diesem Fall werden keine Einzelmeldungen mehr erzeugt.   | Kontrollieren Sie den Wertebereich oder den Typ der Variablen.   |
| 290006 | Die projektierten Grenzwerte der Variablen wurden durch Werteingabe verletzt.   | Beachten Sie die projektierten oder aktuellen Grenzwerte der Variablen.  |
| 290007 | Es besteht ein Unterschied zwischen Quell- und Zielstruktur in der momentan bearbeiteten Rezeptur. Die Zielstruktur enthält eine zusätzliche Rezepturvariable, die in der Quellstruktur nicht vorhanden ist.<br>Die angegebene Rezepturvariable wird mit ihrem Startwert belegt.  | Fügen Sie die angegebene Rezepturvariable in der Quellstruktur ein.  |
| 290008 | Es besteht ein Unterschied zwischen Quell- und Zielstruktur in der momentan bearbeiteten Rezeptur. Die Quellstruktur enthält eine zusätzliche Rezepturvariable, die in der Zielstruktur nicht vorhanden ist und deshalb nicht zugeordnet werden kann.<br>Der Wert wird verworfen. | Entfernen Sie aus Ihrer Projektierung die angegebene Rezepturvariable in der angegebenen Rezeptur.   |
| 290010 | Der für die Rezeptur projektierte Ablageort ist nicht zulässig.<br>Mögliche Ursachen:<br>Unzulässige Zeichen, Schreibschutz, Datenträger voll oder nicht vorhanden.   | Überprüfen Sie den projektierten Ablageort.  |
| 290011 | Der Datensatz mit der angegebenen Nummer existiert nicht.   | Überprüfen Sie die Quelle für die Nummer (Konstante oder Variablenwert).   |
| 290012 | Die Rezeptur mit der angegebenen Nummer existiert nicht.  | Überprüfen Sie die Quelle für die Nummer (Konstante oder Variablenwert).   |
| 290013 | Es wurde versucht, einen Datensatz unter einer bereits vorhandenen Datensatznummer abzuspeichern.<br>Der Vorgang wird nicht ausgeführt.   | Zur Abhilfe stehen folgende Möglichkeiten zur Verfügung: <ul style="list-style-type: none"> <li>• Überprüfen Sie die Quelle für die Nummer (Konstante oder Variablenwert).</li> <li>• Löschen Sie vorher den Datensatz.</li> <li>• Ändern Sie den Funktionsparameter "Überschreiben".</li> </ul> |
| 290014 | Die angegebene zu importierende Datei konnte nicht gefunden werden.   | Überprüfen Sie Folgendes: <ul style="list-style-type: none"> <li>• Überprüfen Sie den Dateinamen.</li> <li>• Vergewissern Sie sich, dass die Datei im angegebenen Verzeichnis liegt.</li> </ul>  |
| 290020 | Rückmeldung, dass die Übertragung von Datensätzen vom Bediengerät zur Steuerung gestartet wurde.  | --   |

| Nummer | Wirkung/Ursachen   | Abhilfe   |
|--------|--|---|
| 290021 | Rückmeldung, dass die Übertragung von Datensätzen vom Bediengerät zur Steuerung fehlerfrei beendet wurde.  | --  |
| 290022 | Rückmeldung, dass die Übertragung von Datensätzen vom Bediengerät zur Steuerung mit Fehler abgebrochen wurde.  | Überprüfen Sie in der Projektierung Folgendes: <ul style="list-style-type: none"> <li>• Sind die Adressen der Variablen in der Steuerung eingerichtet?</li> <li>• Existiert die Rezepturnummer?</li> <li>• Existiert die Datensatznummer?</li> <li>• Ist der Funktionsparameter "Überschreiben" eingestellt?</li> </ul> |
| 290023 | Rückmeldung, dass die Übertragung von Datensätzen von der Steuerung zum Bediengerät gestartet wurde.   | --  |
| 290024 | Rückmeldung, dass die Datensätze von der Steuerung zum Bediengerät fehlerfrei übertragen wurden.   | ---   |
| 290025 | Rückmeldung, dass die Übertragung von Datensätzen von der Steuerung zum Bediengerät mit Fehler abgebrochen wurde.  | Überprüfen Sie in der Projektierung Folgendes: <ul style="list-style-type: none"> <li>• Sind die Adressen der Variablen in der Steuerung eingerichtet?</li> <li>• Existiert die Rezepturnummer?</li> <li>• Existiert die Datensatznummer?</li> <li>• Ist der Funktionsparameter "Überschreiben" eingestellt?</li> </ul> |
| 290026 | Es wird versucht, einen Datensatz zu lesen/schreiben, obwohl das Datenfach momentan nicht frei ist.<br>Dieser Fehler kann bei Rezepturen auftreten, für die eine Übertragung mit Synchronisation projektiert wurde.  | Setzen Sie im Datenfach den Status auf Null.  |
| 290027 | Momentan kann keine Verbindung zur Steuerung hergestellt werden. Deshalb kann der Datensatz nicht gelesen oder geschrieben werden.<br>Mögliche Ursachen:<br>Keine physikalische Verbindung zur Steuerung (kein Kabel gesteckt, Kabel defekt) oder Steuerung ausgeschaltet. | Überprüfen Sie die Verbindung zur Steuerung.  |
| 290030 | Diese Meldung wird nach Wiederanwahl eines Bildes ausgegeben, das eine Rezepturanzeige enthält, in der bereits ein Datensatz ausgewählt ist.   | Laden Sie den auf dem Ablageort vorhanden Datensatz erneut oder behalten Sie die aktuellen Werte bei.   |
| 290031 | Beim Speichern wurde erkannt, dass bereits ein Datensatz mit der angegebenen Nummer existiert.   | Überschreiben Sie den Datensatz oder brechen Sie den Vorgang ab.  |
| 290032 | Beim Exportieren von Datensätzen wurde erkannt, dass bereits eine Datei mit dem angegebenen Namen existiert.   | Überschreiben Sie die Datei oder brechen Sie den Vorgang ab.  |
| 290033 | Sicherheitsabfrage vor dem Löschen von Datensätzen.  | --  |

| Nummer | Wirkung/Ursachen  | Abhilfe  |
|--------|---|--|
| 290040 | Ein nicht näher zu spezifizierender Datensatzfehler mit dem Fehlercode %1 ist aufgetreten.<br>Die Aktion wird abgebrochen.<br>Möglicherweise ist in der Steuerung das Datenfach nicht korrekt eingerichtet.   | Überprüfen Sie den Ablageort, den Datensatz, den Bereichszeiger "Datensatz" und gegebenenfalls die Verbindung zur Steuerung.<br>Stoßen Sie nach einer kurzen Wartezeit die Aktion erneut an.<br>Tritt der Fehler weiterhin auf, so wenden Sie sich bitte an den Customer Support. Geben Sie dabei den aufgetretenen Fehlercode an. |
| 290041 | Das Abspeichern eines Datensatzes oder einer Datei ist nicht möglich, da der Ablageort voll ist.  | Löschen Sie nicht mehr benötigte Dateien.  |
| 290042 | Es wurde versucht, gleichzeitig mehrere Rezepturaktionen auszuführen. Die letzte Aktion wird nicht ausgeführt.  | Stoßen Sie nach einer kurzen Wartezeit die Aktion erneut an.   |
| 290043 | Sicherheitsabfrage vor dem Speichern von Datensätzen.   | --   |
| 290044 | Die Datenablage für die Rezeptur ist zerstört und wird gelöscht.  | --   |
| 290050 | Rückmeldung, dass der Export von Datensätzen gestartet wurde.   | --   |
| 290051 | Rückmeldung, dass der Export von Datensätzen fehlerfrei beendet wurde.  | --   |
| 290052 | Rückmeldung, dass der Export von Datensätzen mit Fehler abgebrochen wurde.  | Stellen Sie sicher, dass die Struktur der Datensätze im Ablageort und die aktuelle Rezeptur-Struktur am Bediengerät identisch sind.  |
| 290053 | Rückmeldung, dass der Import von Datensätzen gestartet wurde.   | --   |
| 290054 | Rückmeldung, dass der Import von Datensätzen fehlerfrei beendet wurde.  | --   |
| 290055 | Rückmeldung, dass der Import von Datensätzen mit Fehler abgebrochen wurde.  | Stellen Sie sicher, dass die Struktur der Datensätze im Ablageort und die aktuelle Rezeptur-Struktur am Bediengerät identisch sind.  |
| 290056 | Der Wert in der angegebenen Zeile/Spalte konnte nicht fehlerfrei gelesen/geschrieben werden.<br>Die Aktion wird abgebrochen.  | Überprüfen Sie die angegebene Zeile/Spalte.  |
| 290057 | Die Rezepturvariablen der angegebenen Rezeptur wurden von der Betriebsart "Offline" nach "Online" umgeschaltet.<br>Jede Änderung einer Variablen dieser Rezeptur wird nun sofort in die Steuerung übertragen.   | --   |
| 290058 | Die Rezepturvariablen der angegebenen Rezeptur wurden von der Betriebsart "Online" nach "Offline" umgeschaltet.<br>Änderungen von Variablen dieser Rezeptur werden nicht mehr sofort in die Steuerung übertragen, sondern müssen gegebenenfalls über eine Datensatzübertragung explizit in die Steuerung übertragen werden. | --   |
| 290059 | Rückmeldung, dass der angegebene Datensatz erfolgreich gespeichert wurde.   | --   |



| Nummer | Wirkung/Ursachen  | Abhilfe   |
|--------|---|---|
| 290060 | Rückmeldung, dass der Datensatzspeicher erfolgreich gelöscht wurde.   | --  |
| 290061 | Rückmeldung, dass das Löschen des Datensatzspeichers mit Fehler abgebrochen wurde.  | --  |
| 290062 | Die max. Datensatznummer liegt über 65536. Dieser Datensatz kann nicht angelegt werden.   | Wählen Sie eine andere Nummer.  |
| 290063 | Tritt auf bei Systemfunktion "ExportiereDatensaeetze" mit Parameter "Überschreiben" auf "Nein". Es wurde versucht eine Rezeptur unter einem Dateinamen zu speichern, der bereits vorhanden ist. Exportieren wird abgebrochen. | Überprüfen Sie die Parameter der Systemfunktion "ExportiereDatensaeetze".                             |
| 290064 | Rückmeldung, dass das Löschen von Datensätzen gestartet wurde.  | --  |
| 290065 | Rückmeldung, dass das Löschen von Datensätzen fehlerfrei beendet wurde.   | --  |
| 290066 | Sicherheitsabfrage vor dem Löschen von Datensätzen.   | --  |
| 290068 | Sicherheitsabfrage, ob wirklich alle Datensätze der Rezeptur gelöscht werden sollen.  | --  |
| 290069 | Sicherheitsabfrage, ob wirklich alle Datensätze der Rezeptur gelöscht werden sollen.  | --  |
| 290070 | Der spezifizierte Datensatz ist in der Import-Datei nicht vorhanden.  | Überprüfen sie die Quelle der Datensatznummer oder des Datensatznamens (Konstante oder Variablenwert) |
| 290071 | Beim Editieren von Datensatzwerten wurde ein Wert eingegeben, der den unteren Grenzwert der Rezepturvariable unterschreitet. Die Eingabe wird verworfen.  | Geben Sie einen Wert innerhalb der Grenzwerte der Rezepturvariable ein.                               |
| 290072 | Beim Editieren von Datensatzwerten wurde ein Wert eingegeben, der den oberen Grenzwert der Rezepturvariable überschreitet. Die Eingabe wird verworfen.  | Geben Sie einen Wert innerhalb der Grenzwerte der Rezepturvariable ein.                               |
| 290073 | Eine Aktion (z.B. Speichern eines Datensatzes) konnte aus unbekanntem Gründen nicht durchgeführt werden. Der Fehler entspricht der Statusmeldung IDS_OUT_CMD_EXE_ERR der großen Rezeptur-Anzeige.                             | --  |
| 290074 | Beim Speichern wurde erkannt, dass bereits ein Datensatz mit der angegebenen Nummer, aber unter einem anderen Namen existiert.  | Überschreiben Sie den Datensatz, ändern Sie die Datensatznummer oder brechen Sie den Vorgang ab.      |
| 290075 | Ein Datensatz mit diesem Namen ist bereits vorhanden. Das Speichern des Datensatzes wird abgebrochen.   | Bitte wählen Sie einen anderen Datensatznamen.  |

| Nummer | Wirkung/Ursachen  | Abhilfe   |
|--------|---|---|
| 290110 | Setzen der Standardwerte wegen Fehler abgebrochen.  | --  |
| 290111 | Das Subsystem Rezepturen kann nicht benutzt werden. Rezepturviews haben keinen Inhalt und rezepturbezogene Funktionen werden nicht ausgeführt.<br>Mögliche Ursachen: <ul style="list-style-type: none"><li>• Beim Laden der Rezepturen ist ein Fehler aufgetreten.</li><li>• In ES wurden die Rezepturen strukturell verändert. Beim erneuten Download des Projekts wurden die Rezepturen nicht mit übertragen. Somit passen die neuen Projektierungsdaten nicht mehr zu den alten Rezepturen im Gerät.</li></ul> | Laden Sie erneut das Projekt inklusive Rezepturen (entsprechende Checkbox im Ladedialog muss aktiviert sein) auf das Gerät. |

### 10.3.5 Systemdiagnose projektieren

#### 10.3.5.1 Grundlagen der Systemdiagnose

##### Einleitung

Mit der Systemdiagnose können Sie Meldungen aus dem Diagnosepuffer aller integrierten Verbindungen anzeigen lassen.

##### System-Diagnoseanzeige

Die System-Diagnoseanzeige ist ein Bedien- und Anzeigeobjekt, das Sie in einem Bild verwenden.

Sie navigieren direkt zur Fehlerursache und zur zugehörigen Verbindung. Sie haben Zugriff zu allen integrierten Verbindungen, die Sie im Editor "Geräte & Netze" konfiguriert haben.

#### 10.3.5.2 Ansichten in der Systemdiagnose

##### Einleitung

In der einfachen System-Diagnoseanzeige stehen drei verschiedene Ansichten zur Verfügung.

- Geräteansicht
- Diagnosepuffer-Ansicht
- Detailansicht

## Geräteansicht

Die Geräteansicht wird nur angezeigt, wenn mehr als eine integrierte Verbindung projiziert wurde.


Die Geräteansicht zeigt in tabellarischer Sicht alle verfügbaren Verbindungen. Durch einen Doppelklick auf eine Verbindung öffnen sich die Diagnosepuffer-Ansicht.

|   |                  |    |
|---|------------------|----|
| 1 | HMI_connection   | ▲  |
| 2 | HMI_connection_1 | ▲▲ |
| 3 | HMI_connection_3 |    |
| 4 | HMI_connection_4 |    |

Navigationssymbole: ←, →, ↻, ↓, ▼

## Diagnosepuffer-Ansicht

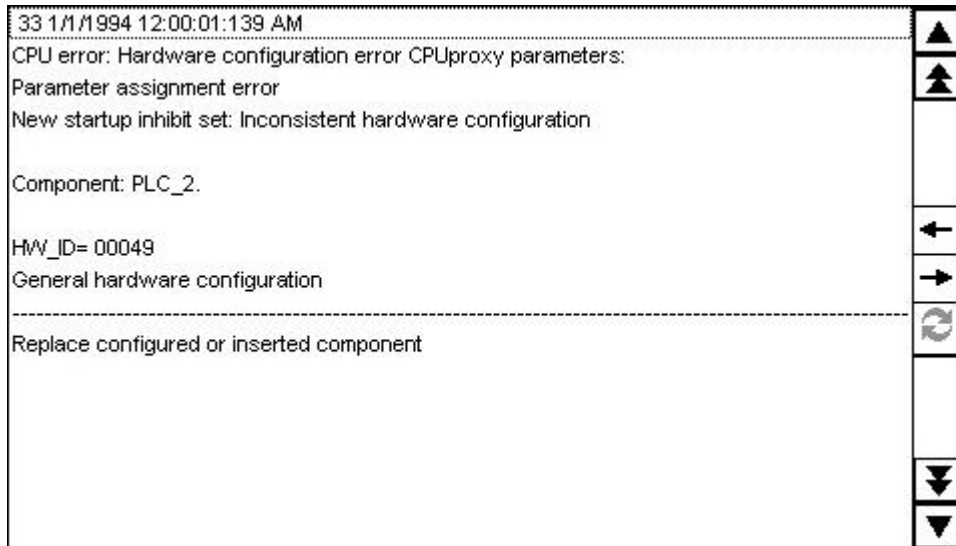
Die Diagnosepuffer-Ansicht zeigt die aktuellen Daten aus dem Diagnosepuffer.

Um die Diagnosepuffer-Ansicht zu aktualisieren, wählen Sie die Schaltfläche .

|    |  |    |
|----|--|----|
| 1  | 1/1/1994 2:18:49 AM Mode transition from STARTUP to RUN                                  | ▲  |
| 2  | 1/1/1994 2:18:48 AM Request for manual warm restart                                      | ▲▲ |
| 3  | 1/1/1994 2:18:48 AM Mode transition from STOP to STARTUP                                 |    |
| 4  | 1/1/1994 2:18:48 AM New startup information in STOP mode                                 |    |
| 5  | 1/1/1994 2:18:37 AM New startup information in STOP mode                                 |    |
| 6  | 1/1/1994 2:18:37 AM STOP caused by stop switch being activated                           |    |
| 7  | 1/1/1994 12:09:09 AM Distributed I/Os: station return                                    | ←  |
| 8  | 1/1/1994 12:09:09 AM I/O access error when transferring the process image to the outp... | →  |
| 9  | 1/1/1994 12:09:09 AM I/O access error when transferring the process image to the outp... | ↻  |
| 10 | 1/1/1994 12:09:09 AM I/O access error when transferring the process image to the outp... |    |
| 11 | 1/1/1994 12:09:09 AM I/O access error when transferring the process image to the outp... |    |
| 12 | 1/1/1994 12:09:09 AM I/O access error when transferring the process image to the outp... |    |
| 13 | 1/1/1994 12:09:09 AM I/O access error when transferring the process image to the outp... |    |
| 14 | 1/1/1994 12:09:09 AM I/O access error when transferring the process image to the outp... | ↓  |
| 15 | 1/1/1994 12:09:09 AM I/O access error when transferring the process image to the outp... | ▼  |

### Detailansicht

Die Detailansicht zeigt ausführliche Informationen über die selektierte Verbindung und die anliegenden Fehler. In der Detailansicht prüfen Sie, ob die Daten korrekt sind.






#### Hinweis

##### Inhalte in der Detailansicht

Die Inhalte der Detailansicht stehen nur für integrierte Verbindungen mit den Steuerungen S7 1200 und S7 1500 zur Verfügung.

### Navigationsschaltflächen

| Schaltfläche  | Taste                                    | Funktion  |
|---|--|---|
|  | Enter-Taste                              | In der Geräteansicht:<br>Öffnet die Diagnosepuffer-Ansicht des gewählten Geräts.<br>In der Diagnosepuffer-Ansicht:<br>Öffnet die Detailansicht. |
|  | Esc-Taste                                | In der Diagnosepuffer-Ansicht :<br>Öffnet die Geräteansicht.<br>In der Detailansicht:<br>Öffnet die Diagnosepuffer-Ansicht.                     |
|  | Projektierte Funktionstaste<br>z. B. F1. | Aktualisiert die Diagnosepuffer-Ansicht..   |

## Siehe auch

System-Diagnoseanzeige (Seite 3154)

System-Diagnoseanzeige projektieren (Seite 3271)

### 10.3.5.3 System-Diagnoseanzeige projektieren

#### Einleitung

Um einen Überblick über alle integrierten Verbindungen zu haben, fügen Sie eine System-Diagnoseanzeige in Ihrem Projekt ein.

#### Voraussetzungen

- Eine Steuerung ist angelegt.
- Ein Basic Panel ist angelegt.
- Eine integrierte Verbindung wurde im Editor "Geräte & Netze" erstellt.
- Ein Bild ist angelegt.
- Das Inspektorfenster ist geöffnet.

#### Vorgehen

1. Doppelklicken Sie in der Task Card "Werkzeuge" auf das Objekt "System-Diagnoseanzeige". Das Objekt wird im Bild eingefügt.
2. Klicken Sie im Inspektorfenster "Eigenschaften > Darstellung".
3. Wählen Sie für "Zeilen pro Eintrag" z. B. 5.

#### Ergebnis

Die System-Diagnoseanzeige ist im Bild eingefügt.

Um die aktuellen Meldungen zu erhalten, aktualisieren Sie den Diagnosepuffer.

## Siehe auch

Ansichten in der Systemdiagnose (Seite 3268)

## 10.4 Mit Rezepturen arbeiten

### 10.4.1 Grundlagen

#### 10.4.1.1 Definition und Einsatzgebiete

##### Einleitung

In Rezepturen werden zusammengehörige Daten, z. B. Maschinenparametrierungen oder Produktionsdaten, zusammengefasst.

Beispiele:

- Maschinenparametrierungen, die notwendig sind, um die Produktion auf eine andere Produktvariante umzustellen.
- Produktbestandteile, die in unterschiedlichen Zusammensetzungen unterschiedliche Endprodukte ergeben.

Eine Rezeptur zeichnet sich durch eine feste Datenstruktur aus. Bei der Projektierung wird die Struktur einer Rezeptur festgelegt. Eine Rezeptur enthält Rezepturdatensätze, die sich in den Werten, nicht aber in der Struktur unterscheiden.

Rezepturen werden auf dem Bediengerät abgelegt. Ein Rezepturdatensatz wird grundsätzlich komplett in einem Arbeitsschritt zwischen Bediengerät und Steuerung übertragen.

---

##### **Hinweis**

##### **Einschränkungen beim Import/Export**

Das Exportieren bzw. Importieren von Rezepturen ist für Basic Panels nicht möglich.

Komplette Rezepturdaten, jedoch nicht einzelne Rezepturdatensätze, können mit ProSave in das CSV-Format exportiert und importiert und auf das Bediengerät übertragen werden. Dabei wird die Runtime angehalten.

---

## Einsatz von Rezepturen

Für den Einsatz von Rezepturen sind z. B. folgende Fälle möglich:

- **Manuelle Produktion**  
Sie wählen die gewünschten Rezepturdaten und zeigen die Rezepturdaten am Bediengerät an. Sie ändern bei Bedarf die Rezepturdaten und speichern Sie auf dem Bediengerät ab. Sie übertragen die Rezepturdaten zur Steuerung.
- **Automatische Produktion**  
Das Steuerungsprogramm stößt die Übertragung der Rezepturdaten zwischen Steuerung und Bediengerät an. Sie können die Übertragung auch durch eine Bedienung am Bediengerät starten. Die Produktion wird damit automatisch umgestellt. Eine Anzeige oder Änderung der Daten ist nicht zwingend notwendig.
- **Teach-In-Betrieb**  
An der Anlage optimieren Sie Produktionsdaten von Hand, z. B. Achspositionen oder Füllmengen. Die ermittelten Werte werden zum Bediengerät übertragen und in einem Rezepturdatensatz gespeichert. Zu einem späteren Zeitpunkt übertragen Sie die gespeicherten Rezepturdaten wieder zur Steuerung.

## Eingeben und Ändern der Rezepturdaten

Sie geben die Daten in den einzelnen Rezepturdatensätzen ein und passen sie bei Bedarf an. Folgende Möglichkeiten stehen Ihnen dazu zur Verfügung:

- **Eingabe der Daten bei der Projektierung**  
Wenn die Produktionsdaten bereits feststehen, geben Sie die Daten bereits während der Projektierung der Rezeptur im Editor "Rezepturen" ein.
- **Eingabe der Daten in Runtime**  
Wenn Sie Produktionsdaten häufig anpassen müssen, ändern Sie die Daten folgendermaßen direkt in Runtime:
  - Sie geben die Daten direkt am Bediengerät ein.
  - Sie stellen die Parameter direkt an der Maschine ein. Sie übertragen dann die Daten von der Steuerung zum Bediengerät und speichern Sie in der Rezeptur.

### 10.4.1.2 Beispiele zum Einsatz von Rezepturen

Rezepturen werden z. B. in der Fertigungsindustrie und im Maschinenbau eingesetzt. Die folgenden Beispiele zeigen typische Anwendungsfälle, die Sie mit der Rezepturfunktionalität von WinCC umsetzen können:

- **Maschinenparametrierung**  
Ein Einsatzgebiet von Rezepturen ist die Maschinenparametrierung in der Fertigungsindustrie: Eine Maschine schneidet Holzplatten auf eine bestimmte Größe zu und bohrt Löcher. Je nach Holzplattengröße werden z. B. die Führungsschienen und Bohrer in neue Positionen gefahren. Die benötigten Positionsdaten sind als Datensätze in einer Rezeptur gespeichert. Im Teach-In-Betrieb parametrieren Sie die Maschine neu, wenn z. B. eine neue Holzplattengröße verarbeitet werden soll. Sie übertragen die neuen Positionsdaten direkt aus der Steuerung zum Bediengerät und speichern sie als neuen Datensatz.
- **Chargenorientierte Produktion**  
Ein anderes Einsatzgebiet von Rezepturen ist die chargenorientierte Produktion in der Nahrungsmittelindustrie: Eine Abfüllstation einer Fruchtsaftanlage produziert Saft, Nektar und Getränk in verschiedenen Geschmacksrichtungen. Die verwendeten Zutaten sind immer dieselben und unterscheiden sich nur in den Mischungsverhältnissen. Jede Geschmacksrichtung entspricht einer Rezeptur. Jedes Mischungsverhältnis entspricht einem Datensatz. Alle notwendigen Daten eines Mischungsverhältnisses können Sie mit einem Knopfdruck an die Maschinensteuerung übertragen.

### 10.4.1.3 Aufbau von Rezepturen

#### Einleitung

Der prinzipielle Aufbau einer Rezeptur wird anhand der Abfüllstation einer Fruchtsaftanlage dargestellt.

In einem Bediengerät können mehrere unterschiedliche Rezepturen vorhanden sein. Sie können eine Rezeptur z. B. vergleichen mit einem Karteikasten, der mehrere Karteikarten enthält. In diesem Karteikasten sind mehrere Herstellungsvarianten einer Produktfamilie enthalten. Die Daten von jeweils einer Herstellungsvariante sind komplett auf einer Karteikarte enthalten.

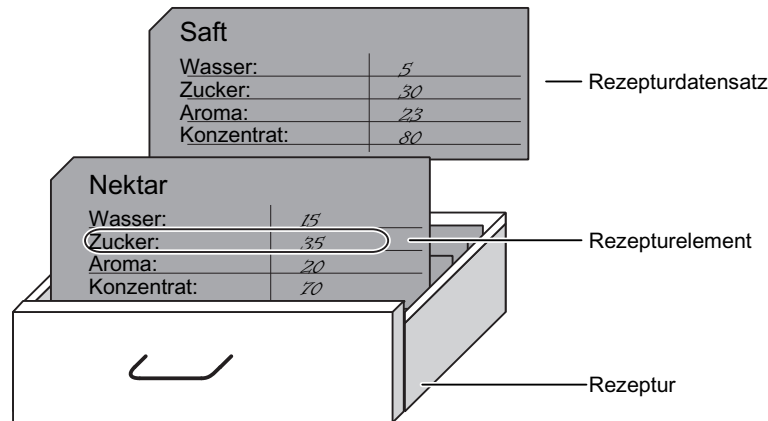
Beispiel:

Bei einer Anlage zur Getränkeherstellung wird eine Rezeptur für unterschiedliche Getränkevarianten benötigt. Es gibt z. B. die Getränkevarianten Fruchtsaftgetränk, Saft, Nektar.



## Rezeptur

Die Rezeptur enthält alle Rezepturdatensätze, z. B. für die unterschiedlichen Getränkevarianten:



## Rezepturdatensätze

Jede Karteikarte repräsentiert einen Rezepturdatensatz, der für die Herstellung einer Produktvariante notwendig ist.

## Rezepturelemente

Jede Karteikarte einer Schublade hat den gleichen Aufbau. Alle Karteikarten enthalten Felder für die verschiedenen Zutaten. Jedes Feld entspricht einem Rezepturelement. Alle Datensätze einer Rezeptur enthalten damit die gleichen Elemente. Die Datensätze unterscheiden sich jedoch im Wert der einzelnen Elemente.

Beispiel:

Alle Getränke enthalten die folgenden Bestandteile:

- Wasser
- Konzentrat
- Zucker
- Aroma

Die Datensätze für Fruchtsaftgetränk, Saft oder Nektar unterscheiden sich jedoch z. B. in der Menge des Zuckers, der bei der Herstellung verwendet wird.

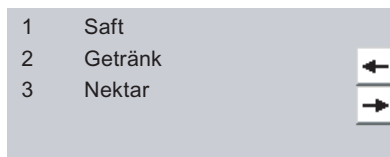
### 10.4.1.4 Anzeige der Rezepturen

#### Einleitung

Um Rezepturen anzuzeigen, projektieren Sie die Rezepturanzeige. In der Rezepturanzeige ändern Sie die Werte einer Rezeptur und beeinflussen so z. B. den Fertigungsprozess oder eine Maschine.

#### Rezepturanzeige

Die Rezepturanzeige ist ein fertiges Anzeige- und Bedienobjekt von WinCC zum Verwalten von Rezepturdatensätzen. Die Rezepturanzeige ist immer ein Bestandteil eines Bildes. Die Rezepturanzeige zeigt Rezepturdatensätze in Tabellenform. Die Darstellung und die Bedienungsmöglichkeiten passen Sie den jeweiligen Bedürfnissen an.



Wenn Sie in Ihrem Projekt Rezepturen mit einer Rezepturanzeige bearbeiten, werden die Werte in Rezepturdatensätzen gespeichert. Die Werte werden erst dann zwischen Bediengerät und Steuerung übertragen, wenn Sie das entsprechende Bedienobjekt bedienen.

### 10.4.1.5 Datenfluss bei Rezepturen

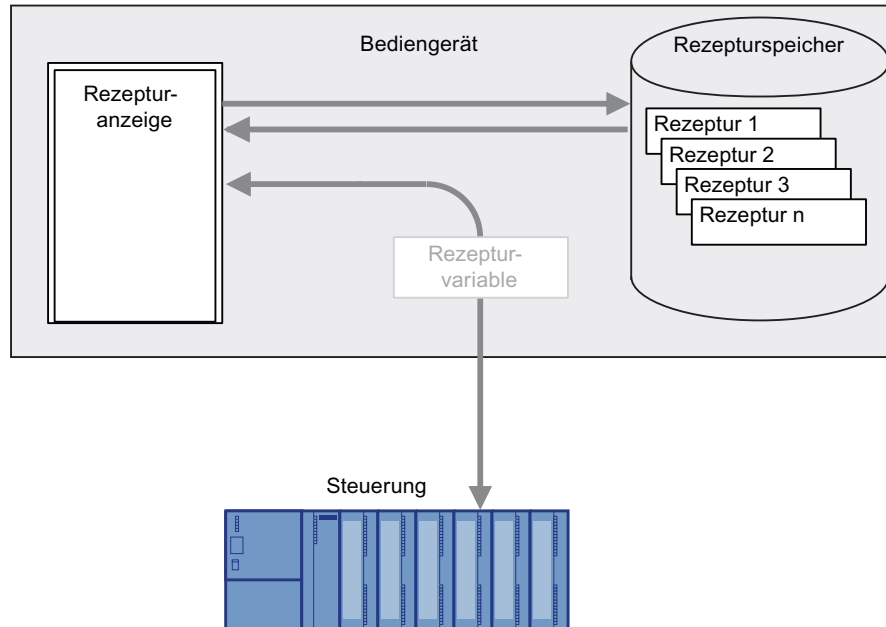
#### Zusammenwirken der Komponenten

In Runtime wirken folgende Komponenten zusammen:

- **Rezepturanzeige**  
Am Bediengerät werden Rezepturen in der Rezepturanzeige angezeigt und bearbeitet. In der Rezepturanzeige werden die Rezepturdatensätze aus dem internen Speicher des Bediengeräts angezeigt und bearbeitet.
- **Rezepturspeicher des Bediengeräts**  
Im Rezepturspeicher des Bediengeräts werden Rezepturen in Form von Rezepturdatensätzen gespeichert.
- **Rezepturvariablen**  
Die Rezepturvariablen enthalten Rezepturdaten.

## Übersicht des Datenflusses

Das folgende Bild stellt den Datenfluss bei Rezepturen dar:



Um Rezepturdatensätze in die Steuerung zu übertragen, nutzen Sie die Schaltfläche "Zur Steuerung" in der Rezepturanzeige oder ein Bedienobjekt mit der Systemfunktion "RezepturanzeigeSchreibeDatensatzInSteuerung".

Über Rezepturvariablen werden Daten mit der Steuerung ausgetauscht. Bei Basic Panels können Sie Rezepturvariablen nicht außerhalb einer Rezeptur verwenden z. B. nicht in E/A-Feldern.

### 10.4.1.6 Synchronisation von Rezepturdatensätzen mit der Steuerung

#### Übersicht

Bei der Übertragung von Rezepturdatensätzen zwischen Bediengerät und Steuerung greifen beide Kommunikationspartner wechselseitig auf gemeinsame Kommunikationsbereiche zu.

Rezepturdatensätze werden immer direkt übertragen. Die Werte der Variablen werden direkt, ohne Zwischenablage, in die projektierten Adressen geschrieben, bzw. aus den projektierten Adressen gelesen.

## Übertragungsarten

Für die Übertragung von Rezepturdatensätzen zwischen Bediengerät und Steuerung gibt es zwei Möglichkeiten:

- Übertragung ohne Koordinierung
- Koordinierte Übertragung über den Bereichszeiger "Datensatz".

---

### Hinweis

#### Koordinierte Übertragung

Bei der koordinierten Übertragung verhindern Sie in Ihrem Steuerungsprogramm ein unkontrolliertes gegenseitiges Überschreiben der Daten.

---

## Voraussetzungen für die koordinierte Übertragung

Folgende Voraussetzungen gelten für die koordinierte Übertragung:

- Im Editor "Kommunikation > Verbindungen" ist der Bereichszeiger "Datensatz" für die gewünschte Verbindung eingerichtet.
- In den Eigenschaften der Rezeptur ist "Koordinierte Übertragung der Datensätze" aktiviert.
- In den Eigenschaften der Rezeptur ist die Verbindung zur Steuerung angegeben, mit der das Bediengerät die Übertragung koordiniert.

## Koordinierte Übertragung

Bei der koordinierten Übertragung setzen sowohl Steuerung als auch Bediengerät Status-Bits im gemeinsam benutzten Datenfach.

Die koordinierte Übertragung von Rezepturdatensätzen bietet sich in folgenden Fällen an:

- Die Steuerung ist der "aktive Partner" bei der Übertragung von Rezepturdatensätzen.
- Die Steuerung wertet Informationen über Rezepturnummer und Rezepturname, sowie Rezepturdatensatznummer und Rezepturdatensatzname aus.
- Die Übertragung von Rezepturdatensätzen wird durch folgende Steuerungsaufträge gestartet:
  - "Datensatz\_in\_Steuerung\_schreiben"
  - "Datensatz\_aus\_Steuerung\_lesen"

## 10.4.2 Elemente und Grundeinstellungen

### 10.4.2.1 Editor "Rezepturen"

#### Einleitung

Im Editor "Rezepturen" erstellen, konfigurieren und bearbeiten Sie Rezepturen, Rezepturelemente und Rezepturdatensätze. Darüber hinaus können Sie im Editor "Rezepturen" Werte in vorhandene Rezepturdatensätze eingeben.

#### Aufbau des Editors "Rezepturen"

Im oberen Teil des tabellarischen Editors legen Sie Rezepturen an und konfigurieren sie dort oder im Inspektorfenster.

Im unteren Teil des tabellarischen Editors finden Sie folgende Registerkarten:

- **Elemente**  
Hier definieren Sie über die Tabellenzellen die Rezepturelemente der ausgewählten Rezeptur. Rezepturelemente verschieben Sie innerhalb der Tabelle über die Kontextmenü-Einträge "Aufwärts" und "Abwärts".
- **Datensätze**  
Hier definieren Sie über die Tabellenzellen die Werte der Rezepturdatensätze der ausgewählten Rezeptur.

| Rezepturen   |             |        |                     |          |                     |                   |  |
|--------------|-------------|--------|---------------------|----------|---------------------|-------------------|--|
| Name         | Anzeigename | Nummer | Version             | Typ      | Maximale Anzahl D.. | Kommunikationsart |  |
| Rezeptur_1   | recipe_1    | 1      | 04.01.2011 17:40:07 | Begrenzt | 20                  | Variablen         |  |
| Rezeptur_2   | recipe_2    | 2      | 04.01.2011 17:40:12 | Begrenzt | 20                  | Variablen         |  |
| <Hinzufügen> |             |        |                     |          |                     |                   |  |

| Elemente          |                  |          |          |            |              |             |  |
|-------------------|------------------|----------|----------|------------|--------------|-------------|--|
| Name              | Anzeigename      | Variable | Datentyp | Datenlänge | Standardwert | Minimalwert |  |
| Rezepturelement_1 | recipe element_1 | <Kein>   |          | 0          |              |             |  |
| <Hinzufügen>      |                  |          |          |            |              |             |  |

Im Inspektorfenster konfigurieren Sie die jeweils ausgewählte Rezeptur, das Rezepturelement oder den Rezepturdatensatz. Weitere Hinweise zur Projektierung der Bestandteile einer Rezeptur finden Sie unter "Rezepturen projektieren".

## Einstellungen Rezepturen

Für die Rezepturen sind folgende Einstellungen möglich:

| Einstellung                           | Beschreibung   |
|---------------------------------------|--|
| Name der Rezeptur                     | Identifiziert die Rezeptur innerhalb des Bediengeräts eindeutig.   |
| Anzeigename                           | Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie aussagekräftige Namen oder Bezeichnungen, die der Bediener einer Rezeptur direkt zuordnen kann, z. B. "Fruchtsaft-Getränk". |
| Rezepturnummer                        | Identifiziert die Rezeptur innerhalb des Bediengeräts eindeutig.   |
| Version                               | Informationen zur Rezeptur. Voreingestellt ist das Datum und der Zeitpunkt der letzten Änderung der Rezeptur.  |
| Pfad                                  | Legt den Ablageort für die Rezepturen fest. Die Rezepturen werden als Datei abgelegt.  |
| Größentyp<br>[unveränderlich]         | Die Rezepturdatensätze sind standardmäßig auf eine vorgegebene Anzahl begrenzt.  |
| Anzahl Datensätze<br>[unveränderlich] | Maximale Anzahl der Rezepturdatensätze einer Rezeptur in Runtime. Die Anzahl wird durch den Rezepturspeicher des Bediengeräts begrenzt.  |
| Kommunikationsart<br>[unveränderlich] | Die Rezepturdatensätze werden direkt in die Adressen der Rezepturvariablen geschrieben oder von dort gelesen.  |
| Tooltip                               | Tooltip zur Rezeptur, der dem Bediener in Runtime angezeigt wird.  |

### Hinweis

#### Pfad

Der Ablageort hängt davon ab, mit welchen Speichermedien das Bediengerät ausgestattet ist.

Basic Panels und OP77A, TP177A (Portrait)

Diese Bediengeräte besitzen keinen externen Speicher. Rezepturen werden immer im internen Flash-Speicher abgelegt.. Die Einstellung "Pfad" steht daher nicht zur Verfügung.

## Einstellungen Rezepturelemente

In der Registerkarte "Elemente" sind folgende Einstellungen möglich:

| Einstellung               | Beschreibung  |
|---------------------------|---|
| Name des Rezepturelements | Identifiziert ein Rezepturelement innerhalb der Rezeptur eindeutig. Vergeben Sie aussagekräftige Namen oder Bezeichnungen, die Sie eindeutig zuordnen können, z. B. Achsenbezeichnungen an einer Maschine oder Zutaten wie "Aroma". |
| Anzeigename               | Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie aussagekräftige Namen oder Bezeichnungen, die der Bediener direkt zuordnen kann, z. B. "Fruchtsaft- Aroma".  |
| Rezepturvariable          | Eine zugeordnete Variable speichert in Runtime den aktuellen Wert des Rezepturelements im Rezepturdatensatz.  |

| Einstellung                     | Beschreibung   |
|---------------------------------|--|
| Datentyp                        | Datentyp der Rezepturvariable.   |
| Datenlänge [unveränderlich]     | Datenlänge der Rezepturvariable, abhängig vom Datentyp.  |
| Textliste                       | In einer Textliste wird einem Wert oder Wertebereich Text zugeordnet. Diesen Text können Sie z. B. in einem Ausgabefeld anzeigen lassen. Die zugeordnete Rezepturvariable muss den Datentyp einer Zahl besitzen. Der Variablenwert muss sich innerhalb des Wertebereichs der Textliste befinden. |
| Standardwert                    | Wird als Standardeintrag verwendet, wenn Sie einen neuen Rezepturdatensatz anlegen.  |
| Minimaler Wert [unveränderlich] | Kleinster darstellbarer Wert einer zahlenbasierten Rezepturvariable, abhängig vom Datentyp.  |
| Maximaler Wert [unveränderlich] | Größter darstellbarer Wert einer zahlenbasierten Rezepturvariable, abhängig vom Datentyp.  |
| Nachkommastellen                | Bestimmt, auf wie viele Stellen genau ein Dezimalwert gerundet wird z. B. 3 Nachkommastellen und umgekehrt mit welcher Zehnerpotenz ein Ganzzahlwert multipliziert wird z. B. 1000.  |
| Tooltip                         | Tooltip zum Rezepturelement, der dem Bediener in Runtime angezeigt wird.   |

### Einstellungen Rezepturdatensätze

In der Registerkarte "Datensätze" sind folgende Einstellungen möglich:

| Einstellung                  | Beschreibung   |
|------------------------------|--|
| Name des Rezepturdatensatzes | Identifiziert den Rezepturdatensatz innerhalb einer Rezeptur eindeutig.  |
| Anzeigename                  | Erscheint in Runtime z. B. in der Rezepturanzeige. Sie können den Anzeigenamen mehrsprachig projektieren. Vergeben Sie aussagekräftige Namen oder Produktnummern, die der Bediener einem Produkt direkt zuordnen kann, z. B. "Gelber Fruchtsaft E231".   |
| Rezepturdatensatznummer      | Identifiziert den Rezepturdatensatz innerhalb einer Rezeptur eindeutig.  |
| Rezepturelemente 1 bis n     | Zu jedem Rezepturelement können Sie bereits während der Projektierung verschiedene Werte hinterlegen. Ein Wert bildet zusammen mit den Werten der anderen Rezepturelemente jeweils einen Rezepturdatensatz. Sie können mehrere Rezepturdatensätze hinterlegen.<br><br>Wenn in den Transfereinstellungen aktiviert, werden beim Laden des Projekts auf das Bediengerät die Rezepturdatensätze mit übertragen und auf dem Bediengerät bereits existierende Datensätze überschrieben. |
| Kommentar                    | Kommentar zum Rezepturdatensatz  |

## 10.4.3 Anzeige und Bearbeitung von Rezepturen in Runtime

### 10.4.3.1 Einfache Rezepturanzeige

#### Rezepturanzeige

Die einfache Rezepturanzeige ist ein vorgefertigtes Anzeige- und Bedienobjekt, das zum Verwalten der Rezepturdatensätze verwendet wird. Die Rezepturanzeige zeigt Rezepturdatensätze in Tabellenform.

Die angezeigten Schaltflächen und Informationen in den Spalten sind einstellbar.

Die in der Rezepturanzeige angezeigten oder eingegebenen Werte werden in Rezepturdatensätzen gespeichert. Über Schaltflächen kann der angezeigte Rezepturdatensatz in die Steuerung geschrieben oder Werte von der Steuerung eingelesen werden.

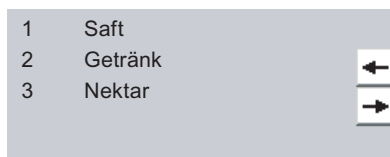
#### Aufbau der Anzeige

Die einfache Rezepturanzeige besteht aus drei Anzeigebereichen:

- Rezepturliste
- Datensatzliste
- Elementliste

Jeder Anzeigebereich der einfachen Rezepturanzeige wird separat am Bediengerät dargestellt. Abhängig von der Projektierung startet die einfache Rezepturanzeige mit der Rezepturliste oder der Datensatzliste.

Das folgende Bild zeigt ein Beispiel für die Datensatzliste.



|   |         |
|---|---------|
| 1 | Saft    |
| 2 | Getränk |
| 3 | Nektar  |



## Wert anzeigen

---

### Hinweis

#### **Bearbeiteter Rezepturdatensatz wird im Hintergrund geändert**

Gilt nur für Basic Panels: Wenn ein Bediener einen Rezepturdatensatz geändert hat und ein Steuerungsauftrag einen beliebigen Rezepturdatensatz dieser Rezeptur lesen oder schreiben möchte, wird der Steuerungsauftrag angehalten und eine Systemmeldung ausgegeben. Hingegen wird der geänderte Wert sofort angezeigt, wenn nur der Steuerungsauftrag und kein Bediener Rezepturdaten geändert hat.

Gilt nicht für Basic Panels: Wenn ein Bediener einen Rezepturdatensatz geändert hat und ein Steuerungsauftrag Werte des betreffenden Rezepturdatensatzes ändert, wird die Rezepturanzeige nicht automatisch aktualisiert. Um die Rezepturanzeige zu aktualisieren, müssen Sie den betreffenden Rezepturdatensatz neu anwählen.

---

## Siehe auch

Rezepturanzeige (Seite 3149)

### 10.4.3.2 Verhalten der Rezepturanzeige in Runtime

#### Bildwechsel

Wenn Sie zu einem anderen Bild wechseln und in der Rezepturanzeige Änderungen an Rezepturdaten noch nicht gespeichert haben, werden Sie aufgefordert, die Rezepturdaten zu sichern. Damit Sie feststellen können, welche Rezepturdaten noch nicht gesichert waren, werden Ihnen der Rezepturname und der Name des Rezepturdatensatzes angezeigt.

#### Rezepturdatensatz neu anlegen, ändern, kopieren oder löschen

Wenn Sie einen Rezepturdatensatz neu anlegen wollen und bereits ein Rezepturdatensatz vorhanden ist, wird am Bildschirm eine Systemmeldung ausgegeben.

#### Rezepturanzeige mit Funktionstasten bedienen

Sie können die Rezepturanzeige über Funktionstasten bedienen, z. B., wenn das Bediengerät keine Touch-Funktionalität besitzt. An die Funktionstasten des Bediengeräts können Funktionen, z. B. "SpeichereDatensatz", projiziert sein.

## Anzeige nach Import von Rezepturdaten

---

### Hinweis

#### Verfügbarkeit

Der Import und Export von Rezepturdaten steht für Basic Panels und OP77A, TP177A (Portrait) nicht zur Verfügung.

---

Wenn Sie während eines Imports von Rezepturdaten die Rezepturanzeige öffnen, werden nur die Rezepturdaten angezeigt, die bereits vollständig importiert wurden. Die Rezepturanzeige wird bei einem Datenimport nicht automatisch aktualisiert. Um eine vollständige Anzeige aller Rezepturdaten zu erhalten, öffnen Sie die Rezepturanzeige erst, nachdem die Systemmeldung über den erfolgreichen Import der Rezepturdaten eingegangen ist. Alternativ aktualisieren Sie die Rezepturanzeige nach erfolgreichem Abschluss des Importvorgangs.

## Variable für Rezepturen und Rezepturdatensätze aktualisieren

---

### Hinweis

#### Verfügbarkeit

Variablen für Rezepturen und Rezepturdatensätze stehen für Basic Panels und OP77A, TP177A (Portrait) nicht zur Verfügung.

---

Der aktuelle Rezepturdatensatz bzw. dessen Nummer kann abhängig von der Projektierung in einer Variablen gespeichert werden. Die Variable wird unter folgenden Bedingungen aktualisiert:

- Das Laden des Rezepturdatensatzes ist beendet.
- Das Bild mit der Rezepturanzeige wurde während des Ladens nicht verlassen.

Dieser Vorgang kann einige Zeit dauern.

## 10.4.4 Rezepturen projektieren

### 10.4.4.1 Allgemeine Vorgehensweise beim Projektieren

Wenn Sie eine Rezeptur neu anlegen, führen Sie folgende Projektierungsschritte aus:

| Schritt | Beschreibung  |
|---------|---|
| 1       | Legen Sie die Struktur der Rezeptur fest.   |
| 2       | Legen Sie entsprechend der Rezepturstruktur Variablen an.<br>Weisen Sie diesen Variablen technologische Namen zu. |
| 3       | Legen Sie die Rezeptur an.  |

| Schritt | Beschreibung   |
|---------|--|
| 4       | Geben Sie die gewünschten Eigenschaften der Rezeptur ein: <ul style="list-style-type: none"> <li>• Sprachabhängiger Anzeigenname der Rezeptur</li> <li>• Option "Koordinierte Übertragung der Datensätze"</li> </ul> Nicht für Basic Panels: <ul style="list-style-type: none"> <li>• Ablageort der Rezepturen</li> <li>• Option "Rezepturanzeige und Rezepturvariablen abgleichen"</li> <li>• Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".</li> </ul> |
| 5       | Legen Sie die Rezepturelemente an und geben Sie die gewünschten Eigenschaften ein: <ul style="list-style-type: none"> <li>• Sprachabhängige Anzeigenamen der Rezepturelemente</li> <li>• Variablen-Anbindung der Rezepturelemente</li> <li>• Standardwerte und Nachkommastelle (Zehnerpotenz) für die Rezepturelemente</li> </ul>  |
| 6       | Legen Sie die Rezepturdatensätze an.<br>Geben Sie die sprachabhängigen Anzeigenamen der Rezepturdatensätze ein.  |
| 7       | Projektieren Sie ein Bild mit Rezepturanzeige oder ein Rezepturbild.   |

**Hinweis****Basic Panels und OP77A, TP177A (Portrait)**

Die Auswahl des Ablageorts steht für diese Geräte nicht zur Verfügung. Die Rezepturen werden immer im internen Flash-Speicher abgelegt.

Rezepturvariablen können Sie nicht außerhalb einer Rezeptur verwenden z. B. nicht in E/A-Feldern, nicht in Meldungen als Triggervariablen, nicht in Systemfunktionen als Parameter usw.

**Hinweis****Einschränkungen Rezepturanzeige und Rezepturbild**

Bei Basic Panels und OP77A, TP177A ist nur die einfache Rezepturanzeige verfügbar. Bei Basic Panels und OP73, OP77A, TP177A (Portrait) sind Rezepturbilder nicht verfügbar.

**10.4.4.2 Rezepturen anlegen und bearbeiten****Neue Rezeptur anlegen****Einleitung**

Zum Anlegen einer vollständigen Rezeptur legen Sie zunächst eine neue Rezeptur an, weisen der Rezeptur Rezepturelemente zu und legen in einem Rezepturdatensatz die entsprechenden Werte fest.

## Voraussetzung

- Die Variablen für die Rezeptur sind angelegt.
- Der Editor "Rezepturen" ist geöffnet.

## Rezeptur anlegen

Um eine Rezeptur anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Editor "Rezepturen" in der ersten freien Tabellenzeile auf "Hinzufügen".

Die neue Rezeptur wird angelegt und in einer Zeile angezeigt.

The screenshot shows the 'Allgemein' configuration window. On the left is a sidebar with a menu containing 'Allgemein', 'Kommunikation', 'Synchronisation', and 'Tooltip'. The main area is titled 'Allgemein' and contains two sections: 'Einstellungen' and 'Größe'. The 'Einstellungen' section has four fields: 'Name' (text box with 'Rezeptur\_1'), 'Anzeigename' (text box with 'recipe\_1'), 'Version' (text box with '04.01.2011 17:40:07'), and 'Nummer' (spin box with '1'). The 'Größe' section has a 'Typ' dropdown menu set to 'Begrenzt' and an 'Anzahl Datensätze' spin box set to '20'.

2. Geben Sie im Bereich "Allgemein" unter "Name" einen aussagekräftigen Namen für die Rezeptur ein.  
Der Name identifiziert die Rezeptur innerhalb eines Bediengeräts eindeutig.
3. Geben Sie unter "Anzeigename" den sprachabhängigen Namen ein, der in der Rezepturanzeige erscheint.
4. Wählen Sie unter "Nummer" eine Rezepturnummer aus.  
Die Nummer identifiziert die Rezeptur innerhalb des Bediengeräts eindeutig.  
Die Rezeptur erhält automatisch eine Version, die das Datum und den Zeitpunkt der letzten Änderung anzeigt. Alternativ geben Sie hier beliebige Informationen zur Rezeptur ein.
5. Legen Sie unter "Datenablage" den Ablageort für die Rezepturdatensätze fest. Die angebotenen Möglichkeiten sind abhängig vom verwendeten Bediengerät.

---

### Hinweis

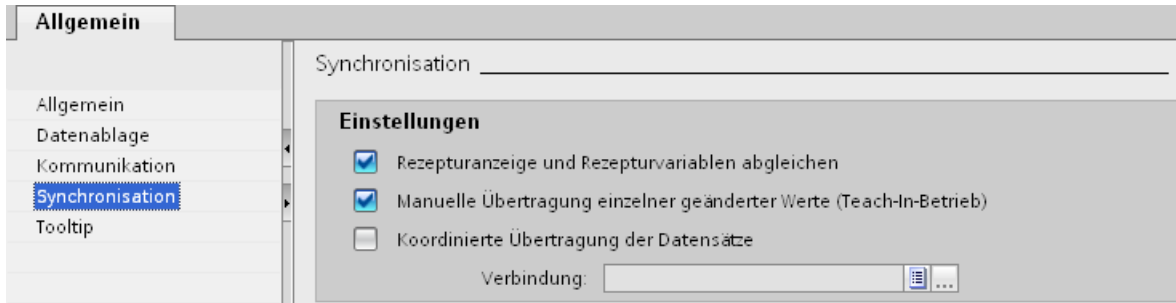
#### Basic Panels und OP77A, TP177A (Portrait)

Die Auswahl des Ablageorts steht für diese Geräte nicht zur Verfügung. Die Rezepturen werden immer im internen Flash-Speicher abgelegt.

Rezepturvariablen können Sie nicht außerhalb einer Rezeptur verwenden z. B. nicht in E/A-Feldern, nicht in Meldungen als Triggervariablen, nicht in Systemfunktionen als Parameter usw.

6. Geben Sie einen Tooltip ein, der dem Bediener in Runtime angezeigt wird.

7. Um in Runtime Rezepturvariablen, die in EA-Feldern projiziert sind, mit der Rezepturanzeige abzugleichen, aktivieren Sie im Inspektorfenster unter "Eigenschaften > Synchronisation" "Rezepturanzeige und Rezepturvariablen abgleichen".



### Hinweis

#### Basic Panels und OP77A, TP177A (Portrait)

Da bei Basic Panels die Rezepturvariablen nicht zusätzlich in EA-Feldern in Bildern verwendet werden können, steht die Option "Rezepturanzeige und Rezepturvariablen abgleichen" nicht zur Verfügung, ebenso im Folgenden nicht die Option "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".


8. Um festzulegen, dass die Rezepturvariablen automatisch beim Editieren in den EA-Feldern in die Steuerung übertragen werden, deaktivieren Sie "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".
9. Um in Runtime die Übertragung der Rezepturdaten über Bereichszeiger zu überwachen, aktivieren Sie "Koordinierte Übertragung der Datensätze".
10. Wählen Sie unter "Synchronisieren mit" die entsprechende Verbindung zur Steuerung für die koordinierte Übertragung.

### Rezepturelement anlegen





Um Rezepturelemente anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Registerkarte "Elemente".
2. Klicken Sie in der ersten freien Zeile des tabellarischen Editors auf "Hinzufügen". Ein neues Rezepturelement wird angelegt.
3. Geben Sie unter "Name" einen aussagekräftigen Namen für das Element ein. Der Name identifiziert das Element innerhalb der Rezeptur eindeutig.
4. Geben Sie unter "Anzeigename" einen sprachabhängigen Anzeigenamen für das Element ein. Der Anzeigename erscheint in Runtime, z. B. in der Rezepturanzeige.

- Wählen Sie unter "Variable" die Variable aus, die Sie mit dem Rezepturelement verbinden wollen.  
In dieser Variablen wird in Runtime der Wert des Rezepturelements gespeichert, der in einem Rezepturdatensatz abgelegt ist.

| Elemente  |              | Datensätze  |            |              |                  |           |
|---|--------------|-------------|------------|--------------|------------------|-----------|
|   | Name         | Anzeigename | Variable   | Standardwert | Nachkommastellen | Hilfetext |
|  | Wasser       | Wasser      | LitreWater | 0            | 0                |           |
|   | <Hinzufügen> |             |            |              |                  |           |


- Geben Sie einen Tooltip ein.  
Der Tooltip wird dem Bediener in Runtime angezeigt.
- Geben Sie unter "Standardwert" den Wert ein, den Sie als Standardeintrag verwenden wollen, wenn Sie einen neuen Rezepturdatensatz anlegen.
- Wenn Sie einem Wert oder einem Wertebereich Text zuordnen wollen, wählen Sie hier die entsprechende Textliste aus. Die zugeordnete Rezepturvariable muss den Datentyp einer Zahl besitzen. Der Variablenwert muss sich innerhalb des Wertebereichs der Textliste befinden.  
Der in der Textliste hinterlegte Text wird z. B. in einem Ausgabefeld in Runtime angezeigt.
- Legen Sie in der Spalte "Nachkommastellen" fest, auf wie viele Stellen genau ein Dezimalwert gerundet wird z. B. 3 Nachkommastellen und umgekehrt mit welcher Zehnerpotenz ein Ganzzahlwert multipliziert wird z. B. 1000.  
Beispiele bei 3 Nachkommastellen: Die Eingabe "5" bei einem Rezepturelement mit dem Datentyp "Integer" liefert den Wert "5000". Die Eingabe "5,6789" bei einem Rezepturelement mit dem Datentyp "Real" liefert den Wert "5,679".
- Legen Sie so viele Rezepturelemente an, wie Sie für die Rezeptur benötigen. Die maximal mögliche Anzahl von Rezepturelementen hängt vom verwendeten Bediengerät ab.

| Elemente  |              | Datensätze  |                 |              |                  |           |
|---|--------------|-------------|-----------------|--------------|------------------|-----------|
|   | Name         | Anzeigename | Variable        | Standardwert | Nachkommastellen | Hilfetext |
|  | Wasser       | Wasser      | LitreWater      | 0            | 0                |           |
|  | Konzentrat   | Konzentrat  | LitreConcentrat | 0            | 0                |           |
|  | Zucker       | Zucker      | KiloSugar       | 0            | 0                |           |
|  | Aroma        | Aroma       | GramAroma       | 0            | 0                |           |
|   | <Hinzufügen> |             |                 |              |                  |           |


### Rezepturdatensatz mit bekannten Rezepturwerten anlegen

Um Rezepturelemente anzulegen, gehen Sie folgendermaßen vor:




1. Klicken Sie auf die Registerkarte "Datensätze".
2. Klicken Sie in der ersten freien Zeile des tabellarischen Editors auf "Hinzufügen". Ein neuer Rezepturdatensatz wird angelegt. Für jedes in der Rezeptur angelegte Rezepturelement enthält der Rezepturdatensatz eine eigene Spalte.

| Elemente  |                     | Datensätze          |        |        |            |        |       |           |
|---|---------------------|---------------------|--------|--------|------------|--------|-------|-----------|
|   | Name                | Anzeigename         | Nummer | Wasser | Konzentrat | Zucker | Aroma | Kommentar |
|  | Rezepturdatensatz_1 | Rezepturdatensatz_1 | 1      | 0      | 0          | 0      | 0     |           |
|   | <Hinzufügen>        |                     |        |        |            |        |       |           |

3. Geben Sie unter "Namen" einen aussagekräftigen Namen ein. Der Name identifiziert den Datensatz innerhalb der Rezeptur eindeutig.
4. Geben Sie unter "Anzeigenamen" einen sprachabhängigen Anzeigenamen ein. Der Anzeigename erscheint in Runtime, z. B. in der Rezepturanzeige.
5. Geben Sie unter "Nummer" eine Rezepturdatensatznummer ein. Die Rezepturdatensatznummer identifiziert den Rezepturdatensatz innerhalb der Rezeptur eindeutig.
6. Wenn Sie die Rezepturwerte bei der Projektierung bereits wissen, geben Sie für jedes Rezepturelement den entsprechenden Wert ein.

| Elemente  |              | Datensätze  |        |        |            |        |       |           |
|---|--------------|-------------|--------|--------|------------|--------|-------|-----------|
|   | Name         | Anzeigename | Nummer | Wasser | Konzentrat | Zucker | Aroma | Kommentar |
|  | Getränk      | Getränk     | 1      | 30     | 70         | 45     | 600   |           |
|   | <Hinzufügen> |             |        |        |            |        |       |           |

7. Legen Sie so viele Datensätze an, wie Sie für die Rezeptur benötigen.

| Elemente  |              | Datensätze  |        |        |            |        |       |           |
|---|--------------|-------------|--------|--------|------------|--------|-------|-----------|
|   | Name         | Anzeigename | Nummer | Wasser | Konzentrat | Zucker | Aroma | Kommentar |
|  | Getränk      | Getränk     | 1      | 30     | 70         | 45     | 600   |           |
|  | Nektar       | Nektar      | 2      | 50     | 50         | 10     | 300   |           |
|  | Saft         | Saft        | 3      | 5      | 95         | 3      | 100   |           |
|   | <Hinzufügen> |             |        |        |            |        |       |           |

## Eingabe der Werte in Runtime

Um Werte in Runtime in die Rezepturdatensätze einzugeben, haben Sie folgende Möglichkeiten:

- Übertragung von Daten direkt aus der Steuerung (Teach-In-Betrieb)
- Import der Werte aus einer CSV-Datei
- Eingabe der Werte am Bediengerät

---

### Hinweis

#### Basic Panels und OP77A ,TP177A (Portrait)

Der Import von Werten steht für diese Geräte nicht zur Verfügung.

---

## Ergebnis

Die komplette Rezeptur ist konfiguriert.

## Rezepturdatensätze mit Datums- oder Zeitangaben

Wenn Sie Datums- oder Zeitangaben verwenden, achten Sie darauf, dass die System-Einstellung für Uhrzeit und Datum auf dem Projektierungsrechner mit der auf dem Zielsystem übereinstimmt. Beispiel: Sie laden um 13:55 einen Rezepturdatensatz auf das Zielsystem, in dem die Bearbeitungszeit 14 Uhr gespeichert ist. Wenn es auf dem Zielrechner bereits 14:05 ist, wird die Rezeptur nicht mehr bearbeitet. Wenn ein Bediener die Rezeptur bearbeitet, werden Änderungsinformationen nicht korrekt in die Datenbank zurückgeschrieben.

Überprüfen Sie nach dem Laden auf das Zielsystem Rezepturen mit Datums- oder Zeitangaben auf dem Zielsystem.

## Rezeptur bearbeiten

### Zweck

Sie wollen eine Rezeptur ändern, erweitern oder Teile löschen.

### Voraussetzung

- Mindestens eine Rezeptur ist angelegt.
- Der Editor "Rezepturen" ist geöffnet.

## Rezeptureinstellungen ändern

Um die Rezeptureinstellungen zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor "Rezepturen" die Rezeptur, die Sie ändern wollen. Das Inspektorfenster wird geöffnet.
2. Ändern Sie die Konfiguration der Rezeptur im Inspektorfenster.



Rezepturelemente und Rezepturdatensätze ändern Sie auf die gleiche Weise.

### Rezepturwerte ändern

Um Rezepturwerte zu ändern, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Rezeptur, deren Werte Sie ändern wollen.
2. Klicken Sie auf die Registerkarte "Datensätze".
3. Geben Sie in den Wertespalten die neuen Werte ein.

### Rezepturelement hinzufügen

Um einer Rezeptur weitere Rezepturelemente hinzuzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Editor "Rezepturen" die Rezeptur, der Sie weitere Elemente hinzufügen wollen.
2. Klicken Sie auf die Registerkarte "Elemente".
3. Klicken Sie in der ersten freien Zeile auf "Hinzufügen".  
Das Rezepturelement wird angelegt.
4. Konfigurieren Sie das Rezepturelement.

Rezepturdatensätze fügen Sie auf die gleiche Weise hinzu.

### Rezepturen verwalten

#### Voraussetzung

- Eine Rezeptur mit Rezepturelementen und Rezepturdatensatz ist angelegt.
- Der Editor "Rezepturen" ist geöffnet.

### Rezepturen umbenennen

Bei Rezepturen, Rezepturelementen und Rezepturdatensätzen wird zwischen den internen Namen und den Anzeigenamen unterschieden.

Um Rezepturen umzubenennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Rezeptur, die Sie umbenennen wollen.  
Das Inspektorfenster wird geöffnet.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie den neuen Namen ein.  
Rezepturelemente und Rezepturdatensätze benennen Sie auf der entsprechenden Registerkarte auf die gleiche Weise um.

---

**Hinweis**

Die Anzeigenamen im Editor "Rezepturen" können Sie auch unter "Sprachen & Ressourcen > Projekttexte" umbenennen. Diese Möglichkeit bietet sich an, wenn Sie z. B. bereits in mehreren Sprachen projiziert haben.

---

### Rezepturen kopieren und einfügen

Um Rezepturen zu kopieren und einzufügen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Rezeptur, die Sie kopieren wollen.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Wählen Sie in der ersten freien Tabellenzeile im Kontextmenü den Befehl "Einfügen".

Die kopierte Rezeptur wird in die Tabelle eingefügt. Mit der Rezeptur werden in der entsprechenden Registerkarte auch die Rezepturelemente und Rezepturdatensätze mitkopiert.

Zusätzlich kopieren Sie Rezepturelemente und Rezepturdatensätze auf der entsprechenden Registerkarte auf die gleiche Weise.

Wenn bereits ein Rezepturdatensatz mit dem selben Namen existiert, wird der Name des kopierten Rezepturdatensatzes um eine Ziffer erweitert. Damit ist die Eindeutigkeit der Namen gewährleistet. Rezepturdatensätze können nur innerhalb einer Rezeptur kopiert und eingefügt werden.

### Rezeptur löschen

Um eine Rezeptur zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Rezeptur, die Sie löschen wollen.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".  
Die Rezeptur wird gelöscht.

Rezepturelemente und Rezepturdatensätze löschen Sie auf der entsprechenden Registerkarte auf die gleiche Weise.

---

**Hinweis**

Beim Löschen einer Rezeptur werden die darin enthaltenen Rezepturdatensätze gelöscht.

---

---

### Hinweis

Beim Löschen eines Rezepturelements werden die dazugehörigen Werte in den Rezepturdatensätzen ebenfalls gelöscht. Die zugeordneten Variablen bleiben erhalten.

---

## 10.4.4.3 Anzeige der Rezeptur projektieren

### Einfache Rezepturanzeige projektieren

#### Voraussetzung

- Die Rezeptur ist angelegt.
- Der Editor "Bilder" ist geöffnet.
- Das Bild ist angelegt und geöffnet.

| <b>ACHTUNG</b>   |
|--|
| <p><b>Datenverlust bei mehreren Rezepturanzeigen im Bild</b></p> <p>Gilt nur für Basic Panels, OP73, OP77A, TP177A und TP177A (Portrait): Wenn in einem Bild zwei oder mehr Rezepturanzeigen die gleiche Rezeptur anzeigen, kommt es zu einem Konflikt beim Datenzugriff.</p> <p>Datenverlust und unvorhersehbarer Zustand von Rezepturdaten sind die Folge.</p> <p>Stellen Sie sicher, dass der Bediener in unterschiedlichen Rezepturanzeigen nicht die gleiche Rezeptur auswählen und ändern kann.</p> <ul style="list-style-type: none"><li>• Zeigen Sie in einer Rezepturanzeige jeweils nur eine Rezeptur an.</li><li>• Zeigen Sie in jeder Rezepturanzeige jeweils eine andere Rezeptur an.</li></ul> |

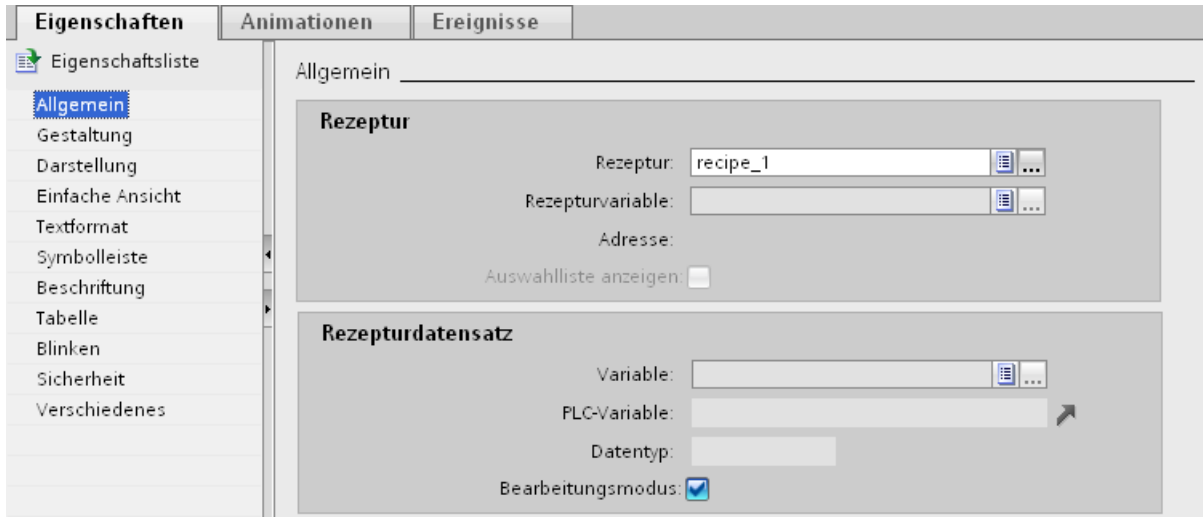


#### Vorgehen

Um eine einfache Rezepturanzeige zu projektieren, gehen Sie folgendermaßen vor:

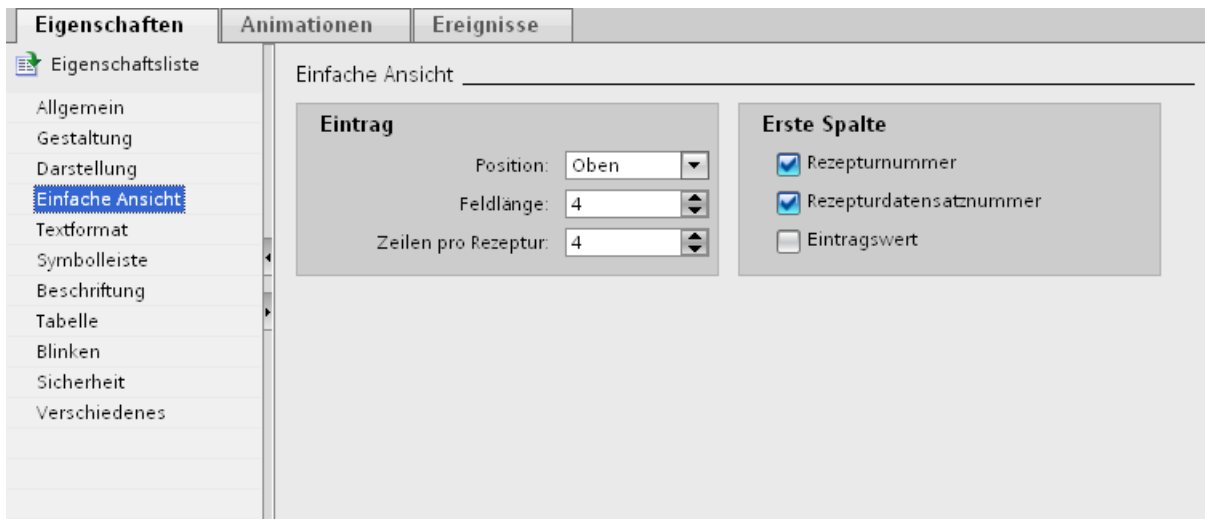
1. Fügen Sie die Rezepturanzeige in das Bild ein. Die Rezepturanzeige finden Sie in der Task Card "Werkzeuge" unter "Controls".
2. Nur bei Geräten, welche auch die erweiterte Rezepturanzeige unterstützen: Aktivieren Sie unter "Eigenschaften > Darstellung > Modus" "Einfache Ansicht".

3. Wenn Sie in der Rezepturanzeige nur die Rezepturdatensätze einer bestimmten Rezeptur anzeigen wollen, wählen Sie unter "Eigenschaften > Allgemein > Rezeptur" die gewünschte Rezeptur aus.

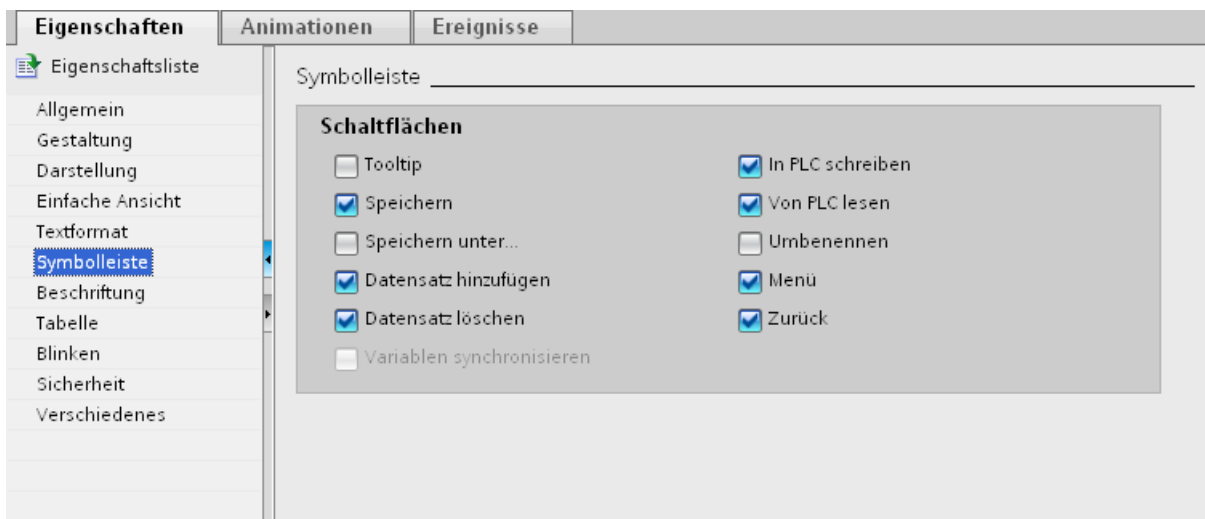


4. Wenn Sie in der Rezepturanzeige die Rezepturdaten nur anzeigen wollen, deaktivieren Sie im Bereich "Rezepturdatensatz" "Bearbeitungsmodus".
5. Legen Sie unter "Eigenschaften > Gestaltung" und "Eigenschaften > Darstellung" zusätzliche Anzeigoptionen der Rezepturanzeige fest.

6. Wählen Sie unter "Eigenschaften > Einfache Ansicht" die Position, Feldlänge und die gewünschte Zeilenanzahl aus.  
Bei "Position > Oben" erscheint der Rezepturwert in der ersten Zeile des Rezeptureintrags.  
Bei "Position > Unten" erscheint der Rezepturwert in der letzten Zeile des Rezeptureintrags.



7. Legen Sie unter "Eigenschaften > Symbolleiste" fest, welche Menübefehle in Runtime in der Rezepturanzeige verfügbar sind.



## Ergebnis

Die einfache Rezepturanzeige ist projiziert. In Runtime können Sie die Rezepturanzeige zum Anzeigen oder Bearbeiten der Rezepturdaten verwenden.

Das Deaktivieren des Bearbeitungsmodus unter "Eigenschaften > Eigenschaften > Allgemein" hat keine Auswirkung auf die Schaltflächen in der Symbolleiste. Die unter "Eigenschaften > Symbolleiste" aktivierten Schaltflächen sind auch bei deaktiviertem Bearbeitungsmodus bedienbar.

## 10.4.5 Rezepturen in Runtime bedienen

### 10.4.5.1 Einfache Rezepturanzeige bedienen

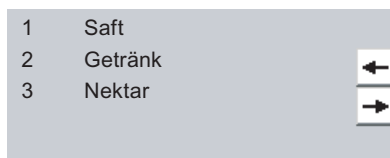
#### Beschreibung Einfache Rezepturanzeige

##### Darstellung

Die einfache Rezepturanzeige besteht aus den folgenden Anzeigebereichen:

- Rezepturliste
- Datensatzliste
- Elementliste

Das folgende Bild zeigt eine Datensatzliste:



Jeder Anzeigebereich wird von der einfachen Rezepturanzeige separat am Bediengerät dargestellt. Jeden dieser Anzeigebereiche bedienen Sie mit einem kontextspezifischen Menü.

Standardmäßig startet die einfache Rezepturanzeige mit der Rezepturliste.

##### Bedienung


Für die Bedienung der einfachen Rezepturanzeige haben Sie abhängig von der Projektierung folgende Möglichkeiten:

- Rezepturdatensatz neu anlegen, ändern, kopieren oder löschen
- Rezepturdatensatz aus Steuerung lesen oder zur Steuerung übertragen

#### Anzeigebereich und Kontextmenü bedienen



Zur Bedienung der einfachen Rezepturanzeige wechseln Sie zwischen den Anzeigebereichen und den Kontextmenüs.

Die folgende Tabelle zeigt die Bedienung des Anzeigebereichs.


| Schaltfläche  | Taste   | Funktion   |
|---|---------|--|
|   | <Enter> | Der nächstniedrige Anzeigebereich, d. h. die Datensatzliste oder die Elementliste wird geöffnet. |
|  | <Esc>   | Der vorherige Anzeigebereich wird geöffnet.  |

| Schaltfläche | Taste                | Funktion  |
|--------------|----------------------|---|
|              | <INS>                | Legt einen neuen Datensatz zur selektierten Rezeptur an, wenn die Liste der Rezepturen oder Rezepturdatensätze angezeigt wird. Wechselt dann zur Liste der Rezepturelemente.<br>Voraussetzung: "Eigenschaften >Allgemein > Bearbeitungsmodus" ist aktiviert.<br>Die Schaltfläche kann mit der Funktion "Taste SimulateSystemKey" simuliert werden auch auf Geräten ohne Tasten. |
|              | <DEL>                | Löscht den selektierten Rezepturdatensatz in der Liste der Rezepturdatensätze.<br>Voraussetzung: "Eigenschaften >Allgemein > Bearbeitungsmodus" ist aktiviert.  |
|              | <Auf>/<Ab>           | Wählt den vorherigen/nächsten Eintrag aus.  |
|              | <Bild Auf>/<Bild Ab> | Eine Anzeigeseite wird nach oben oder unten geblättert.   |
|              | <Pos 1>/<Ende>       | Wählt den ersten/letzten Eintrag aus. Der erste/letzte Eintrag wird markiert.   |

Die folgende Tabelle zeigt die Bedienung der Kontextmenüs:

| Schaltfläche  | Taste                              | Funktion  |
|---|------------------------------------|---|
|  | <Rechts>                           | Das Kontextmenü des Anzeigebereichs wird geöffnet.              |
|  | <Esc>                              | Das Menü wird geschlossen.<br>Der Anzeigebereich wird geöffnet. |
|   | Eingabe der Nummer des Menübefehls | Der Menübefehl wird ausgeführt.                                 |

### Kontextspezifische Menüs der einfachen Rezepturanzeige

Zu jedem Anzeigebereich können Sie mit der Schaltfläche  eine Befehlsauswahl aufrufen. In der Befehlsauswahl werden die Befehle angezeigt, die für den Anzeigebereich verfügbar sind. Jedem Befehl ist eine Nummer zugeordnet. Der Befehl wird ausgeführt, sobald Sie diese Nummer eingeben. Alternativ selektieren Sie den Befehl und betätigen Sie die Taste <Return>.

### Kontextspezifische Menüs der Rezepturliste

| Menübefehl       | Funktion   |
|------------------|--|
| Neu              | Für die markierte Rezeptur wird ein neuer Rezepturdatensatz erstellt.<br>Wenn ein Startwert projektiert wurde, wird dieser im Eingabefeld angezeigt. |
| Tooltip anzeigen | Der für die Rezeptur projektierte Tooltip wird angezeigt.  |
| Öffnen           | Die Datensatzliste der markierten Rezeptur wird geöffnet.  |

### Kontextspezifische Menüs der Rezepturdatensatzliste

| Menübefehl      | Funktion  |
|-----------------|---|
| Neu             | Ein neuer Rezepturdatensatz wird erstellt.<br>Wenn ein Startwert projektiert wurde, wird dieser im Eingabefeld angezeigt. |
| Löschen         | Der markierte Datensatz wird gelöscht.  |
| Speichern unter | Der markierte Datensatz wird unter einem anderen Namen gespeichert. Für die Eingabe des Namens wird ein Dialog geöffnet.  |
| Umbenennen      | Der markierte Datensatz wird umbenannt. Für die Eingabe des Namens wird ein Dialog geöffnet.                              |
| Öffnen          | Die Elementliste des markierten Datensatzes wird geöffnet.  |
| Zurück          | Die Rezepturliste wird geöffnet.  |

### Kontextspezifische Menüs der Rezepturelementliste

| Menübefehl       | Funktion   |
|------------------|--|
| Speichern        | Der markierte Datensatz mit dem Rezepturelement wird gespeichert.  |
| Zur Steuerung    | Die angezeigten Werte des markierten Datensatzes werden vom Bediengerät in die Steuerung übertragen.                   |
| Von Steuerung    | Die Rezepturwerte aus der Steuerung werden am Bediengerät in der Rezepturanzeige angezeigt.                            |
| Speichern unter  | Der markierte Datensatz wird unter einem neuen Namen gespeichert. Für die Eingabe des Namens wird ein Dialog geöffnet. |
| Tooltip anzeigen | Der für das Rezepturelement projektierte Tooltip wird angezeigt.   |
| Umbenennen       | Das markierte Rezepturelement wird umbenannt. Für die Eingabe des Namens wird ein Dialog geöffnet.                     |
| Zurück           | Die Datensatzliste wird geöffnet.  |

### Kontextspezifische Menüs der Datensatzliste

---

#### Hinweis

#### Bediengeräteabhängigkeit

Bei Basic Panels wie auch bei OP 77A, TP 177A, TP 177A (Portrait) und TB 177B sind zusätzlich die folgenden Menübefehle projektiert.

---

| Menübefehl    | Funktion   |
|---------------|--|
| Zur Steuerung | Die angezeigten Werte des markierten Datensatzes werden vom Bediengerät in die Steuerung übertragen. |
| Von Steuerung | Die Rezepturwerte aus der Steuerung werden am Bediengerät in der Rezepturanzeige angezeigt.          |



## Rezepturdatensätze verwalten

### Rezepturdatensätze verwalten

Zur Verwaltung der einfachen Rezepturanzeige haben Sie abhängig von der Projektierung folgende Möglichkeiten:

- Neue Rezepturdatensätze erstellen
- Rezepturdatensätze kopieren
- Rezepturdatensätze ändern
- Rezepturdatensätze löschen

### Neuen Rezepturdatensatz erstellen

Um einen neuen Rezepturdatensatz zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen neuen Rezepturdatensatz erstellen wollen.
2. Wählen Sie im Kontextmenü der Rezepturliste den Befehl "Neu".  
Ein neuer Datensatz mit der nächsten freien Nummer wird angelegt.  
Die Elementliste des neuen Rezepturdatensatzes wird geöffnet.
3. Geben Sie die Werte für die Elemente des Rezepturdatensatzes ein.  
Abhängig von der Projektierung sind die Werte des Rezepturdatensatzes bereits mit Standardwerten vorbelegt.
4. Wählen Sie im Kontextmenü der Elementliste den Befehl "Speichern".  
Der Dialog "Speichern unter" wird geöffnet.
5. Geben Sie Name und Nummer des Rezepturdatensatzes ein.
6. Klicken Sie auf die Schaltfläche "OK".

### Ergebnis

Der neue Rezepturdatensatz wird in der ausgewählten Rezeptur gespeichert. Wenn der Rezepturdatensatz bereits vorhanden ist, wird am Bildschirm eine Systemmeldung ausgegeben.

### Rezepturdatensatz kopieren

Um einen Rezepturdatensatz zu kopieren, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz kopieren wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, von dem Sie eine Kopie speichern wollen.
3. Wählen Sie im Kontextmenü der Datensatzliste den Befehl "Speichern unter".  
Der Dialog "Speichern unter" wird geöffnet. Der Rezepturdatensatz erhält automatisch die nächste freie Rezepturdatensatznummer.

4. Geben Sie bei Bedarf Name und Nummer des Rezepturdatensatzes ein.
5. Klicken Sie auf die Schaltfläche "OK".

### Ergebnis

Der Rezepturdatensatz wird unter dem neuen Namen angelegt.

### Rezepturdatensatz ändern

Um einen Rezepturdatensatz zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, in der Sie einen vorhandenen Rezepturdatensatz ändern wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie ändern wollen.
3. Wählen Sie den gewünschten Rezepturdatensatz.  
Die Elementliste des Rezepturdatensatzes wird angezeigt.
4. Ersetzen Sie die vorhandenen Werte durch die neuen Werte.
5. Wählen Sie im Kontextmenü der Elementliste den Befehl "Speichern".

### Ergebnis

Die geänderten Werte werden in den Rezepturdatensatz übernommen.

### Rezepturdatensatz löschen

Um einen Rezepturdatensatz zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie am Bediengerät die Rezeptur aus, aus der Sie einen vorhandenen Rezepturdatensatz löschen wollen.
2. Wählen Sie am Bediengerät den Rezepturdatensatz aus, den Sie löschen wollen.
3. Wählen Sie im Kontextmenü der Datensatzliste den Befehl "Löschen".
4. Bestätigen Sie die Sicherheitsabfrage, um den Datensatz zu löschen.

### Ergebnis

Der Rezepturdatensatz ist gelöscht.

### Rezepturdatensatz aus Steuerung lesen

### Einleitung

In Runtime können Sie direkt in der Anlage die Werte ändern, die auch in den Rezepturen im Bediengerät hinterlegt sind. Dies ist z. B. der Fall, wenn ein Ventil direkt an der Anlage weiter geöffnet wurde, als in der Rezeptur hinterlegt ist. Die Werte der im Bediengerät gespeicherten

Rezepturdatensätze stimmen dann möglicherweise nicht mehr mit den Werten in der Steuerung überein.

Sie können die Werte der Rezepturvariablen aus der Steuerung lesen und in einen Rezepturdatensatz schreiben.

Die eingelesenen Werte werden in den Rezepturdatensatz geschrieben, der gerade am Bediengerät angezeigt wird.

## Vorgehen

Um einen Rezepturdatensatz aus der Steuerung zu lesen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die Rezeptur am Bediengerät.  
Die Datensatzliste wird geöffnet.
2. Öffnen Sie die Elementliste des Rezepturdatensatzes, in den Sie die Werte aus der Steuerung übernehmen wollen.
3. Wählen Sie im Kontextmenü der Elementliste den Befehl "Von Steuerung".  
Die Werte werden aus der Steuerung gelesen und im aktuellen Rezepturdatensatz angezeigt.
4. Wenn Sie die Werte speichern wollen, wählen Sie den Befehl "Speichern" oder "Speichern unter".

## Ergebnis

Die Werte werden aus der Steuerung gelesen, am Bediengerät dargestellt und im Rezepturdatensatz gespeichert.

## Rezepturdatensatz zur Steuerung übertragen

### Einleitung

Damit die Werte eines mit der Rezepturanzeige geänderten Datensatzes in der Steuerung gültig sind, müssen Sie die Werte zur Steuerung übertragen.

Es werden immer die in der Rezepturanzeige angezeigten Werte zur Steuerung übertragen.

### Vorgehen

Um einen Rezepturdatensatz zur Steuerung zu übertragen, gehen Sie folgendermaßen vor:

1. Öffnen Sie die gewünschte Rezeptur.  
Die Datensatzliste wird geöffnet.
2. Öffnen Sie die Elementliste des Rezepturdatensatzes, dessen Werte Sie zur Steuerung übertragen wollen.
3. Wählen Sie im Kontextmenü der Elementliste den Befehl "Zur Steuerung".

## Ergebnis

Die Werte des Rezepturdatensatzes werden an die Steuerung übertragen.

### 10.4.6 Beispiel

#### 10.4.6.1 Beispiel für das Erstellen einer Rezeptur

## Aufgabe

In diesem Beispiel erstellen Sie drei Rezepturen für eine Fruchtsaftmischanlage. Mit der Fruchtsaftmischanlage werden die Geschmacksrichtungen Orange, Apfel und Tropical produziert. Für jede Geschmacksrichtung erstellen Sie jeweils eine Rezeptur.

Jede Rezeptur enthält für folgende Mischungsverhältnisse jeweils einen Rezepturdatensatz:

- Getränk
- Nektar
- Saft

## Einstellungen

Die Einstellungen beziehen sich auf ein Bediengerät, das an eine SIMATIC S7-300 oder SIMATIC S7-400 gekoppelt ist.

Für das Beispiel benötigen Sie folgende Variablen, Rezepturen, Rezepturelemente und Rezepturdatensätze:

### Variablen:

Tabelle 10-11

| Name             | Steuerungsanbindung | Adresse        | Typ     |
|------------------|---------------------|----------------|---------|
| LitreWater       | Ja                  | DB 120, DBW 0  | Integer |
| LitreConcentrate | Ja                  | DB 120, DBW 4  | Integer |
| KiloSugar        | Ja                  | DB 120, DBW 8  | Integer |
| GramAroma        | Ja                  | DB 120, DBW 12 | Integer |

### Rezepturen:

- Orange
- Apfel
- Tropical

### Rezepturelemente:

| Rezepturelement  | Verbundene Variable |
|------------------|---------------------|
| Liter Wasser     | LitreWater          |
| Liter Konzentrat | LitreConcentrate    |
| Kilo Zucker      | KiloSugar           |
| Gramm Aroma      | GramAroma           |

**Rezepturdatensätze Getränk, Nektar und Saft:**

| Datensatzname | Liter Wasser | Liter Konzentrat | Kilo Zucker | Gramm Aroma |
|---------------|--------------|------------------|-------------|-------------|
| Getränk       | 30           | 70               | 45          | 600         |
| Nektar        | 50           | 50               | 10          | 300         |
| Saft          | 5            | 95               | 3           | 100         |

**Vorgehen**

Um die Rezeptur zu erstellen, gehen Sie folgendermaßen vor:

1. Erstellen Sie die Variablen "LitreWater", "LitreConcentrate", "KiloSugar" und "GramAroma" mit den oben genannten Einstellungen.
2. Erstellen Sie die Rezepturen "Orange", "Apfel" und "Tropical" mit den oben genannten Einstellungen. Legen Sie in jeder Rezeptur die Rezepturelemente an.

| Elemente  |              | Datensätze  |                 |              |                  |           |
|---|--------------|-------------|-----------------|--------------|------------------|-----------|
|   | Name         | Anzeigename | Variable        | Standardwert | Nachkommastellen | Hilfetext |
|  | Wasser       | Wasser      | LitreWater      | 0            | 0                |           |
|  | Konzentrat   | Konzentrat  | LitreConcentrat | 0            | 0                |           |
|  | Zucker       | Zucker      | KiloSugar       | 0            | 0                |           |
|  | Aroma        | Aroma       | GramAroma       | 0            | 0                |           |
|   | <Hinzufügen> |             |                 |              |                  |           |

3. Nicht bei Basic Panels: Konfigurieren Sie jede Rezeptur so, dass die Rezepturdatensätze zwischen Rezepturbild und Rezepturanzeige abgeglichen werden. Die Werte der Rezepturvariablen sollen nicht automatisch an die Steuerung übertragen werden. Dazu sind im Eigenschaftsfenster der jeweiligen Rezeptur folgende Einstellungen notwendig:  
 Unter "Eigenschaften > Extras":
  - Aktivieren Sie "Rezepturanzeige und Rezepturvariablen abgleichen".
  - Aktivieren Sie "Manuelle Übertragung einzelner geänderter Werte (Teach-In-Betrieb)".
4. Legen Sie in jeder Rezeptur die oben genannten Datensätze an. Geben Sie in jeden Datensatz die oben genannten Werte ein.

| Elemente |              | Datensätze  |        |        |            |        |       |           |  |
|----------|--------------|-------------|--------|--------|------------|--------|-------|-----------|--|
|          | Name         | Anzeigename | Nummer | Wasser | Konzentrat | Zucker | Aroma | Kommentar |  |
|          | Getränk      | Getränk     | 1      | 30     | 70         | 45     | 600   |           |  |
|          | Nektar       | Nektar      | 2      | 50     | 50         | 10     | 300   |           |  |
|          | Saft         | Saft        | 3      | 5      | 95         | 3      | 100   |           |  |
|          | <Hinzufügen> |             |        |        |            |        |       |           |  |
|          |              |             |        |        |            |        |       |           |  |
|          |              |             |        |        |            |        |       |           |  |

## Ergebnis

Die drei Rezepturen "Orange", "Apfel" und "Tropical" sind angelegt.

## 10.5 Eine Benutzerverwaltung projektieren

### 10.5.1 Einsatzgebiet der Benutzerverwaltung

#### Prinzip

Der Zugriffsschutz regelt in Runtime den Zugriff auf Daten und Funktionen. So schützen Sie Ihre Anwendungen vor unbefugtem Bedienen. Sie schränken bereits beim Erstellen Ihres Projekts sicherheitsrelevante Bedienungen auf spezielle Benutzergruppen ein. Dazu richten Sie Benutzer und Benutzergruppen ein, die Sie mit charakteristischen Zugriffsrechten ausstatten, den Berechtigungen. An sicherheitsrelevanten Objekten projektieren Sie dann die zur Bedienung erforderlichen Berechtigungen. Bediener haben z. B. nur Zugriff auf bestimmte Bedienobjekte. Inbetriebsetzer haben z. B. uneingeschränkten Zugriff in Runtime.

#### Definition

Sie administrieren Benutzer, Benutzergruppen und Berechtigungen zentral in der Benutzerverwaltung von WinCC. Sie übertragen Benutzer und Benutzergruppen zusammen mit dem Projekt auf das Bediengerät. Die Benutzer und Kennwörter verwalten Sie an dem Bediengerät über die Benutzeranzeige.

## Anwendungsbeispiel

Sie projektieren eine Berechtigung "Service", damit nur Servicetechniker Zugriff auf die Einstellparameter haben. Die Berechtigung ordnen Sie der Benutzergruppe "Servicetechniker" zu. Alle Mitglieder dieser Gruppe bedienen so die geschützten Einstellparameter.

### ACHTUNG

Der Zugriffsschutz schützt nicht vor Fehlbedienungen. Sie müssen selbst dafür sorgen, dass nur entsprechend ausgebildetes und autorisiertes Personal Anlagen und Maschinen konstruiert, in Betrieb setzt, bedient und wartet.

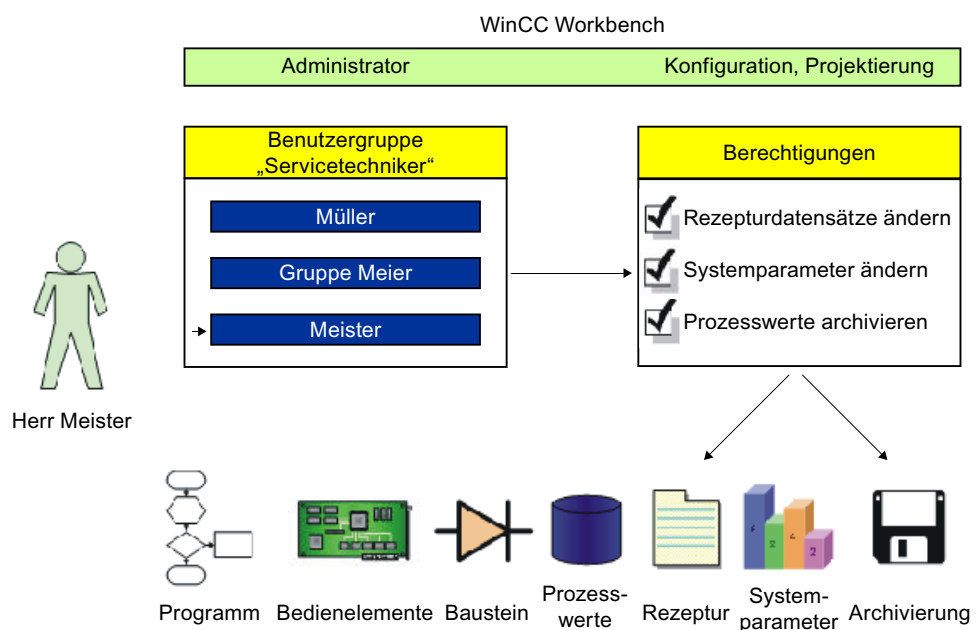
Der Zugriffsschutz ist nicht dazu geeignet, Arbeitsabläufe zu definieren und deren Einhaltung zu überwachen.

## 10.5.2 Aufbau der Benutzerverwaltung

### Einleitung

Bei einem Projekt in der Fertigungstechnik unterscheidet sich die Umgebung beim Maschinenhersteller von der Umgebung beim Endkunden als Anlagenbetreiber.

Der Maschinenhersteller gestattet dem Benutzer z. B. Herr Meister einen bestimmten Zugriff innerhalb der Applikation bzw. dem Bediengerät. Beim Endkunden existiert jedoch kein Benutzer Meister. Die Benutzer des Endkunden und ihre Aufgaben sind während der Projektierung beim Maschinenhersteller noch nicht bekannt. Die endgültigen Benutzer werden in der Regel erst nach der Inbetriebnahme beim Endkunden festgelegt.



## Prinzip

Um den Aufwand bei der Verwaltung gering zu halten, werden Berechtigungen nicht direkt einem einzelnen Benutzer zugewiesen, sondern sind über Benutzergruppen entkoppelt.

Eine Benutzergruppe fasst projektierte Berechtigungen aufgabengerecht zusammen. Z. B. werden alle notwendigen Berechtigungen für einen Serviceeinsatz in einer Gruppe "Servicetechniker" zusammengefasst. Wenn Sie einen Benutzer anlegen, der für den Service zuständig ist, weisen Sie ihn lediglich der Gruppe "Servicetechniker" zu.

Um Benutzer in Runtime zu verwalten, steht Ihnen die Benutzeranzeige zur Verfügung. Die Benutzeranzeige ermöglicht Ihnen in Runtime Benutzer anzulegen, zu löschen und ihnen eine Berechtigung zuzuweisen.

Die Benutzerverwaltung trennt die Administration der Benutzer von der Projektierung der Berechtigungen. Dadurch bietet sie Flexibilität beim Zugriffsschutz.

Eine Vorbelegung der Benutzerverwaltung kann bereits während der Projektierung im Engineering System erfolgen.

## 10.5.3 Grundlagen

### 10.5.3.1 Benutzer

#### Einleitung

Benutzer legen Sie im Editor "Benutzerverwaltung" unter der Registerkarte "Benutzer" an und ordnen sie den Benutzergruppen zu. Die Registerkarte "Benutzer" ist Teil der Benutzerverwaltung von WinCC.

#### Öffnen

Sie öffnen die Registerkarte "Benutzer", indem Sie im Projektfenster auf "Benutzerverwaltung" doppelklicken.

#### Arbeitsbereich

Im Arbeitsbereich verwalten Sie die Benutzer:

- Sie erstellen oder löschen Benutzer.
- Sie ordnen Benutzer Benutzergruppen zu.

---

#### Hinweis

Sie können einen Benutzer nur einer Benutzergruppe zuordnen.

---



## Inspektorfenster

Wenn Sie einen Benutzer markieren, können Sie in der Gruppe "Allgemein" das Kennwort ändern. Unter "Automatisches Abmelden" legen Sie fest, ob der Benutzer automatisch von dem Bediengerät abgemeldet wird, wenn nach der eingegebenen Zeit keine Eingabe erfolgt.


### 10.5.3.2 Arbeitsbereich Benutzer

#### Einleitung



Der Arbeitsbereich "Benutzer" zeigt tabellarisch die Benutzer und Benutzergruppen. Sie administrieren die Benutzer und ordnen sie einer Benutzergruppe zu.

#### Prinzip

Der Arbeitsbereich besteht aus den Tabellen "Benutzer" und "Benutzergruppen".

| Benutzer  |                               |          |                                     |             |  |
|---|-------------------------------|----------|-------------------------------------|-------------|--|
|   | Name                          | Kennwort | Automatisches Abmelden              | Abmeldezeit | Kommentar  |
|  | Administrator<br><Hinzufügen> | *****    | <input checked="" type="checkbox"/> | 5           | Benutzer 'Administrator' wird der Gruppe 'Administratoren' zugeordnet. |

| Gruppen   |                         |  |
|---|-------------------------|--|
| Mitglied von  | Name                    | Kommentar  |
|  | • Administratorengruppe | Gruppe 'Administratoren' hat anfangs alle Rechte.          |
|  | • Benutzer              | Gruppe 'Benutzer' hat anfangs die Berechtigung 'Bedienen'. |
| <Hinzufügen>  |                         |  |

Die Tabelle "Benutzer" zeigt die angelegten Benutzer. Wenn Sie in dieser Tabelle einen Benutzer markieren, zeigt die Tabelle "Gruppen", welcher Benutzergruppe der Benutzer angehört.

#### Hinweis

Für den Benutzer "Administrator" ist das Kennwort "administrator" voreingestellt. Aus Sicherheitsgründen sollten Sie das Kennwort dieses Benutzers ändern.

### 10.5.3.3 Benutzergruppen

#### Einleitung

Benutzergruppen und Berechtigungen legen Sie im Editor "Benutzerverwaltung" unter der Registerkarte "Benutzergruppen" an. Die Registerkarte "Benutzergruppen" ist Teil der Benutzerverwaltung von WinCC.

#### Öffnen

Doppelklicken Sie im Projektfenster auf "Benutzerverwaltung". Wählen Sie die Registerkarte "Benutzergruppen".

#### Arbeitsbereich

Im Arbeitsbereich verwalten Sie die Benutzergruppen und Berechtigungen:

- Sie legen neue Benutzergruppen und Berechtigungen an oder löschen diese.
- Sie weisen den Benutzergruppen die Berechtigungen zu.

#### Inspektorfenster

Wenn Sie eine Benutzergruppe oder eine Berechtigung markieren, können Sie in der Gruppe "Allgemein" den Namen editieren. Darüber hinaus fügen Sie in der Gruppe "Kommentar" eine kurze Beschreibung ein.

### 10.5.3.4 Arbeitsbereich Benutzergruppen

#### Einleitung

Der Arbeitsbereich "Benutzergruppen" zeigt tabellarisch die Gruppen und ihre Berechtigungen. Sie administrieren die Benutzergruppen und weisen ihnen Berechtigungen zu.

## Prinzip

Der Arbeitsbereich besteht aus den Tabellen "Gruppen" und "Berechtigungen".

| Gruppen   |  |  |  |  |  |
|---|--|--|--|--|--|
| Name  | Kommentar  |  |  |  |  |
|  Administratorengruppe | Gruppe 'Administratoren' hat anfangs alle Rechte.          |  |  |  |  |
|  Benutzer              | Gruppe 'Benutzer' hat anfangs die Berechtigung 'Bedienen'. |  |  |  |  |
| <Hinzufügen>  |  |  |  |  |  |

| Berechtigungen  |                    |                    |        |  |  |
|---|--------------------|--------------------|--------|--|--|
| Aktiv   | Name               | Anzeigename        | Nummer | Kommentar  |  |
|  <input checked="" type="checkbox"/> | Bedienen           | Bedienen           | 3      | Berechtigung 'Bedienen'.   |  |
|  <input checked="" type="checkbox"/> | Benutzerverwaltung | Benutzerverwaltung | 1      | Berechtigung 'Benutzerverwaltung' für Verwaltung von Benutzern in der Benutze... |  |
|  <input checked="" type="checkbox"/> | Überwachen         | Überwachen         | 2      | Berechtigung 'Überwachen'.   |  |
| <Hinzufügen>  |                    |                    |        |  |  |

Die Tabelle "Gruppen" zeigt die angelegten Benutzergruppen. Wenn Sie in dieser Tabelle eine Benutzergruppe markieren, zeigt die Tabelle "Berechtigungen" in der Spalte "Aktiv", welche Berechtigungen zugewiesen wurden.

Die Nummer der Benutzergruppe und der Berechtigung vergibt die Benutzerverwaltung. Die Bezeichnungen und Beschreibungen vergeben Sie.

Die Nummern der vordefinierten Berechtigungen sind vorgegeben. Berechtigungen, die Sie erstellen, sind frei editierbar. Achten Sie bei der Vergabe jedoch auf die Eindeutigkeit der Nummern.

### 10.5.3.5 Einstellungen für die Benutzerverwaltung

#### Einleitung

Im Editor "Runtime-Einstellungen > Benutzerverwaltung" projektieren Sie Sicherheitseinstellungen für die Benutzer und ihre Kennwörter in Runtime.

#### Öffnen

Doppelklicken Sie im Projektfenster auf den Editor "Runtime-Einstellungen". Klicken Sie auf "Benutzerverwaltung".

#### Arbeitsbereich

Im Arbeitsbereich nehmen Sie Einstellungen für die Gültigkeit von Kennwörtern in Runtime vor. Sie bestimmen z. B. die Komplexität des Kennworts.

## Auswirkungen in Runtime

Abhängig von der Konfiguration haben die Sicherheitseinstellungen folgende Auswirkungen in Runtime.

- Gruppe "Allgemein"
  - Kontrollkästchen "Grenze für Login-Versuche aktivieren" aktiviert  
Die im Feld "Anzahl ungültiger Login-Versuche" eingetragene Zahl bestimmt, wie viele Anmeldeversuche ein Benutzer hat, bevor er der Gruppe "Unberechtigt" zugeordnet wird.  
Kontrollkästchen "Grenze für Login-Versuche aktivieren" nicht aktiviert  
Der Benutzer hat in Runtime unendlich viele Anmeldeversuche.
  - Feld "Anzahl ungültiger Login-Versuche"  
Wenn Sie im Feld z. B. "4" eingetragen haben und der vierte Anmeldeversuch des Benutzers fehlschlägt, wird er automatisch der Gruppe "Unberechtigt" zugewiesen. Sie können 1 - 9 Versuche eintragen.
  - Kontrollkästchen "Login nur mit Kennwort möglich"  
Wenn das Kontrollkästchen aktiv ist, findet eine Authentifizierung der Benutzer über das Kennwort statt. Der Benutzername ist nicht erforderlich.  
Um eine Zuordnung zu ermöglichen, dürfen Sie daher die Kennwörter nicht doppelt projektieren.
- Gruppe "Hierarchieebene"
  - Kontrollkästchen "Gruppenspezifische Rechte für die Benutzerverwaltung"  
Mit aktiven Kontrollkästchen verwalten Administratoren nur die Benutzer, deren Gruppennummer kleiner oder gleich ihrer eigenen ist.  
Ein Administrator, dessen Gruppennummer z. B. 5 ist, kann nur Benutzer verwalten, deren Gruppennummer kleiner oder gleich 5 ist. D. h. auch, dass der Administrator Benutzer nur Gruppen zuordnen kann, die eine Gruppennummer kleiner oder gleich 5 haben.

- Gruppe "Kennwort"
  - Kontrollkästchen "Kennwortalterung aktivieren" aktiviert  
Das Kennwort läuft nach der im Feld "Gültigkeit für Kennwort (Tage)" eingetragenen Anzahl von Tagen ab.  
Im Editor "Benutzergruppen" wird zusätzlich die Spalte "Kennwortalterung" aktiviert. Damit bestimmen Sie gruppenspezifisch, ob Kennwörter ablaufen und ob Kennwortgenerationen gespeichert werden. Für Gruppen, deren Kennwortalterung nicht aktiviert ist, laufen die Kennwörter niemals ab.
  - Feld "Vorwarnzeit (Tage)"  
Der Benutzer wird die eingegebene Anzahl von Tagen vor Ablauf seines Kennworts davon unterrichtet, dass sein Kennwort abläuft.
  - Feld "Kennwortgenerationen"  
Wenn der Benutzer sein Kennwort ändert, muss sich das neue Kennwort von der eingegebenen Anzahl vorheriger Kennwörter unterscheiden. Die mögliche Anzahl der Kennwortgenerationen liegt zwischen 1 und 5.
- Gruppe "Kennwortkomplexität"
  - Kontrollkästchen "Muss Sonderzeichen enthalten" aktiviert  
Der Benutzer muss ein Kennwort mit mindestens einem Sonderzeichen eingeben, unabhängig an welcher Position.
  - Kontrollkästchen "Muss Ziffer enthalten" aktiviert  
Der Benutzer muss ein Kennwort mit mindestens einer Ziffer eingeben, unabhängig an welcher Position.
  - Feld "Minimale Kennwortlänge"  
Der Benutzer muss ein Kennwort mit einer Mindestlänge, von der im Feld "Minimale Kennwortlänge" eingetragenen Zahl, eingeben.  
Die minimale Länge des Kennworts ist 3 Zeichen.

## 10.5.4 Benutzerverwaltung aufbauen

### 10.5.4.1 Grundlagen zur Benutzerverwaltung

#### Prinzip

Das vorliegende Kapitel richtet sich an vier Zielgruppen. Entsprechend sind die Themen geordnet. Die Zielgruppen stehen beispielhaft für unterschiedliche Personengruppen, die mit der Benutzerverwaltung arbeiten.

1. Administrator OEM
2. Administrator RT
3. Projekteur
4. Bediener

Als Administrator OEM erstellen Sie z. B. beim Maschinenhersteller im Engineering System die Benutzergruppen, Benutzer und Berechtigungen für Runtime.

Als Administrator RT verwalten Sie in Runtime Benutzer über die "Benutzeranzeige".

Als Projekteur weisen Sie im Engineering System die Berechtigungen den Benutzergruppen zu. Des Weiteren projektieren Sie Berechtigungen an Objekte.

Als Bediener melden Sie sich in Runtime an. Sie haben nur Zugriff auf ein geschütztes Objekt, wenn Sie die erforderliche Berechtigung besitzen.

---

#### **Hinweis**

Die Zielgruppe Administrator RT ist in der Runtime Benutzerverwaltung als vordefinierte Benutzergruppe "Administratorengruppe" bereits angelegt. Im Folgenden werden zur besseren Veranschaulichung die vordefinierten Benutzergruppen und Berechtigungen nicht verwendet.

---

### **10.5.4.2 Benutzer für Runtime administrieren**

#### **Berechtigung erstellen**

##### **Einleitung**

Sie erstellen eine Berechtigung und weisen sie einer oder mehreren Benutzergruppen zu.

##### **Voraussetzungen**

Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

##### **Vorgehensweise**

1. Doppelklicken Sie in der Tabelle "Berechtigungen" auf "Hinzufügen".
2. Geben Sie als Name der Berechtigung "Runtime beenden" ein.
3. Geben Sie als "Kommentar" eine kurze Beschreibung ein.

#### **Benutzergruppe anlegen**

##### **Einleitung**

Damit Sie nicht jedem Benutzer eine Berechtigung zuweisen, erstellen Sie Benutzergruppen. Sie legen eine Benutzergruppe an, weisen ihr Berechtigungen zu und ordnen ihr Benutzer zu.

---

#### **Hinweis**

Der Name der Benutzergruppe muss innerhalb des Projekts eindeutig sein. Sonst wird die Eingabe nicht übernommen.

---

---

### Hinweis

#### Verwenden von SIMATIC Logon

Stellen Sie sicher, dass die Namen der Benutzergruppen in Windows und WinCC exakt übereinstimmen.

---

### Voraussetzungen

Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

### Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Gruppen" auf "Hinzufügen".
2. Geben Sie als "Name" der Benutzergruppe "Bediener" ein.
3. Bei Bedarf ändern Sie die "Nummer" der Benutzergruppe.
4. Geben Sie als "Anzeigename" der Benutzergruppe "Bediener" an.
5. Geben Sie als "Kommentar" eine kurze Beschreibung ein.

In Runtime zeigt die Benutzeranzeige den Anzeigenamen der Benutzergruppe an. Der Anzeigename der Benutzergruppe ist sprachabhängig. Sie projektieren die Bezeichnung in mehreren Sprachen und schalten die Sprache in Runtime um.

### Siehe auch

Berechtigung zuweisen (Seite 3313)

Benutzer anlegen (Seite 3314)

Benutzer einer Benutzergruppe zuordnen (Seite 3315)

Benutzergruppen verwalten (Seite 3317)

Benutzer verwalten (Seite 3316)

### Berechtigung zuweisen

### Einleitung

Wenn Sie einer Benutzergruppe eine Berechtigung zuweisen, besitzen alle zugeordneten Benutzer diese Berechtigung.

### Voraussetzungen

- Eine Berechtigung "Runtime beenden" ist erstellt.
- Eine Benutzergruppe "Bediener" ist angelegt.
- Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

### Vorgehensweise

1. Klicken Sie in der Tabelle "Gruppen" auf die Benutzergruppe "Bediener". Die Tabelle "Berechtigungen" zeigt alle Berechtigungen.
2. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Runtime beenden".

---

#### Hinweis

Die Berechtigung "Runtime beenden" ist nur eine Bezeichnung und hat keinen Bezug zur Funktion "StoppeRuntime". Sie müssen diesen Bezug selbst herstellen. Projektieren Sie dazu an eine Schaltfläche die Systemfunktionen "StoppeRuntime" und wählen als "Berechtigung" "Runtime beenden".

---

### Siehe auch

Benutzergruppe anlegen (Seite 3312)

### Benutzer anlegen

### Einleitung

Sie legen einen Benutzer an, damit sich nach dem Laden auf das Gerät Personen mit ihrem Benutzernamen in Runtime anmelden.

Alternativ können Sie Benutzer über die "Benutzeranzeige" in Runtime anlegen und ändern.

Damit ein angelegter Benutzer Berechtigungen besitzt, müssen Sie ihn einer Benutzergruppe zuordnen und der Benutzergruppe Berechtigungen zuweisen.

Wenn der bei der Anmeldung eingegebene Benutzername mit einem Benutzer in Runtime übereinstimmt, ist die Anmeldung erfolgreich. Darüber hinaus muss das eingegebene Kennwort mit dem hinterlegten Kennwort des Benutzers übereinstimmen.

---

#### Hinweis

Beachten Sie die Groß- und Kleinschreibung.

---

### Voraussetzungen

Der Arbeitsbereich "Benutzer" ist geöffnet.



## Vorgehensweise



1. Doppelklicken Sie in der Tabelle "Benutzer" auf "Hinzufügen".
2. Geben Sie als Benutzername "Meister" ein.

---

### Hinweis

Der Benutzername muss innerhalb des Projekts eindeutig sein. Sonst wird die Eingabe nicht übernommen.

---

3. Klicken Sie in der Spalte "Kennwort" auf die Schaltfläche . Ein Dialog zur Eingabe des Kennworts wird geöffnet.
4. Geben Sie das Kennwort des Benutzers ein.
5. Geben Sie zur Bestätigung das Kennwort im darunter liegenden Feld ein zweites Mal ein.
6. Schließen Sie den Dialog über das Symbol .
7. Wenn der Benutzer nach einer bestimmten Zeitspanne abgemeldet werden soll, aktivieren Sie "Automatisches Abmelden".
8. Klicken Sie in die Spalte "Abmeldezeit". Als "Abmeldezeit" sind 5 Minuten voreingestellt.
9. Geben Sie als "Kommentar" eine kurze Beschreibung ein.

## Siehe auch

Benutzergruppe anlegen (Seite 3312)

## Benutzer einer Benutzergruppe zuordnen

### Einleitung

Wenn Sie einen Benutzer einer Benutzergruppe zuordnen, besitzt er die Berechtigungen der Benutzergruppe.

---

### Hinweis

Sie müssen einen Benutzer genau einer Benutzergruppe zuordnen. Bei der Konsistenzprüfung und Generierung des Projekts wird die Zuordnung geprüft.

---

## Voraussetzungen

- Der Benutzer "Meister" ist angelegt.
- Eine Benutzergruppe "Bediener" ist angelegt.
- Der Arbeitsbereich "Benutzer" ist geöffnet.

### Vorgehensweise

1. Klicken Sie in der Tabelle "Benutzer" auf den Benutzer "Meister". Die Tabelle "Gruppen" zeigt alle Benutzergruppen.
2. Aktivieren Sie in der Tabelle "Gruppen" die Benutzergruppe "Bediener".

### Siehe auch

Benutzergruppe anlegen (Seite 3312)

### Benutzer verwalten

#### Einleitung

Im Arbeitsbereich verwalten Sie Benutzer und ordnen Sie Benutzergruppen zu.

#### Voraussetzungen


Der Arbeitsbereich "Benutzer" ist geöffnet.

#### Benutzername ändern

1. Doppelklicken Sie in der Tabelle "Benutzer" in der Spalte "Name" auf das Feld des zu ändernden Benutzernamens.
2. Ändern Sie den Benutzernamen.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie den Benutzer im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Benutzernamen.

#### Kennwort des Benutzers ändern

1. Klicken Sie in der Tabelle "Benutzer" in der Spalte "Kennwort" auf die Schaltfläche . Ein Dialog zur Eingabe des Kennworts wird geöffnet.
2. Geben Sie im Feld "Kennwort eingeben" das neue Kennwort ein.
3. Geben Sie das neue Kennwort im Feld "Kennwort bestätigen" nochmals ein.
4. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie den Benutzer im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" das Kennwort.

#### Nicht sichtbare Spalten anzeigen lassen

1. Bewegen Sie den Mauszeiger auf die Kopfzeile der Tabelle "Benutzer".
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und aktivieren Sie die Anzeige der Spalte z. B. "Abmeldezeit".

### Abmeldezeit des Benutzers ändern

1. Doppelklicken Sie im Bereich "Benutzer" in der Spalte "Abmeldezeit" auf das Feld der zu ändernden Abmeldezeit.
2. Ändern Sie die Abmeldezeit.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie den Benutzer im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Automatisches Abmelden" die Abmeldezeit.

### Benutzer löschen

1. Markieren Sie die Zeile des zu löschenden Benutzers.
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und wählen Sie den Befehl "Löschen".

---

#### Hinweis

Vordefinierte Benutzer können Sie nicht löschen.

---

### Siehe auch

Benutzergruppe anlegen (Seite 3312)

### Benutzergruppen verwalten

### Einleitung

Im Arbeitsbereich verwalten Sie die Benutzergruppen und vergeben Berechtigungen für die Verwendung in Runtime.

### Voraussetzungen

Der Arbeitsbereich "Benutzergruppen" ist geöffnet.

### Namen der Benutzergruppe ändern

1. Doppelklicken Sie in der Tabelle "Gruppen" in der Spalte "Name" auf das Feld des zu ändernden Namens der Benutzergruppe.
2. Ändern Sie den Namen der Benutzergruppe.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie die Benutzergruppe im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Namen.

---

**Hinweis**

Vordefinierte Benutzergruppen können nicht gelöscht werden.

---

### Nicht sichtbare Spalten anzeigen lassen

1. Bewegen Sie den Mauszeiger auf die Kopfzeile der Tabelle "Gruppen".
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und aktivieren Sie die Anzeige der Spalte z. B. "Anzeigename".

### Anzeigename der Benutzergruppe ändern

1. Doppelklicken Sie in die Tabelle "Gruppen" in der Spalte "Anzeigename" auf das Feld des zu ändernden Anzeigenamens der Benutzergruppe.
2. Ändern Sie den Anzeigenamen der Benutzergruppe.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie die Benutzergruppe im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Anzeigenamen.

### Benutzergruppe löschen

1. Markieren Sie die Zeile der zu löschenden Benutzergruppe
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und wählen Sie den Befehl "Löschen".

---

**Hinweis**

Vordefinierte Benutzergruppen können nicht gelöscht werden.

---

### Namen der Berechtigung ändern

1. Doppelklicken Sie in der Tabelle "Berechtigungen" in der Spalte "Name" auf das Feld des zu ändernden Namens der Berechtigung.
2. Ändern Sie den Namen der Berechtigung.
3. Bestätigen Sie die Eingabe mit der Taste <Return>.

Alternativ markieren Sie die Berechtigung im Arbeitsbereich. Ändern Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein" den Namen.

## Berechtigung löschen

1. Markieren Sie die Zeile der zu löschenden Berechtigung
2. Öffnen Sie das Kontextmenü über die rechte Maustaste und wählen Sie den Befehl "Löschen".

---

### Hinweis

Vordefinierte Berechtigungen können nicht gelöscht werden.

---

## Siehe auch

Benutzergruppe anlegen (Seite 3312)

### 10.5.4.3 Benutzer in Runtime administrieren

## Benutzer in Runtime

### Prinzip

Sie legen Benutzer und Benutzergruppen an und weisen ihnen Berechtigungen zu. Sie projektieren Objekte mit Berechtigungen. Nach dem Laden auf das Bediengerät sind in Runtime alle Objekte vor unberechtigtem Zugriff geschützt, die mit einer Berechtigung projiziert wurden.

## Benutzeranzeige

Wenn Sie im Engineering System eine Benutzeranzeige projektieren, administrieren Sie nach dem Laden auf dem Bediengerät in der Benutzeranzeige die Benutzer.

|                |
|----------------|
| <b>ACHTUNG</b> |
|----------------|

|  |
|--|
| Änderungen in der Benutzeranzeige in Runtime sind sofort wirksam. Änderungen in Runtime werden im Engineering System nicht aktualisiert. Beim Laden der Benutzerverwaltung auf das Bediengerät werden alle Änderungen in der Benutzeranzeige auf Nachfrage und in Abhängigkeit von den Einstellungen beim Laden überschrieben. |
|--|

Benutzer, welche die Berechtigung "Benutzerverwaltung" haben, besitzen uneingeschränkten Zugriff auf die Benutzeranzeige. Damit administrieren sie alle Benutzer. Um sich selbst zu administrieren, hat jeder andere Benutzer nur einen eingeschränkten Zugriff auf die Benutzeranzeige.

## Benutzeranzeige

### Zweck

Um auch in Runtime die Benutzer zu administrieren, projektieren Sie eine Benutzeranzeige im Engineering System.

### Aufbau

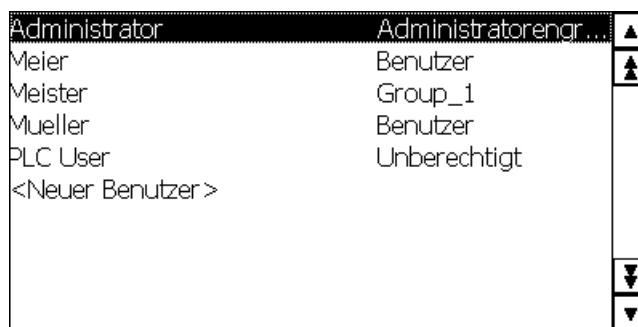
Die Benutzeranzeige zeigt in jeder Zeile:

- Den Benutzer,
- Die zugehörige Benutzergruppe.

Wenn kein Benutzer angemeldet ist, ist die Benutzeranzeige leer. Nach dem Anmelden werden die Inhalte der einzelnen Felder angezeigt.



### Benutzeranzeige eines Administrators



Wenn ein Administrator angemeldet ist, zeigt die Benutzeranzeige alle Benutzer. Der Administrator ändert den Benutzernamen und das Kennwort. Der Administrator legt neue Benutzer an und ordnet sie einer bestehenden Benutzergruppe zu.

## Benutzeranzeige eines Benutzers



Wenn kein Administrator angemeldet ist, zeigt die Benutzeranzeige nur den angemeldeten Benutzer. Der Benutzer kann sein Kennwort ändern.

## Benutzeranzeige projektieren

### Einleitung

Um die Benutzer auch im Runtime zu administrieren, projektieren Sie eine Benutzeranzeige im Engineering System.

### Voraussetzungen

Ein Bild ist erstellt.

### Vorgehensweise

1. Markieren Sie im Werkzeugfenster in der Kategorie "Controls" das Objekt "Benutzeranzeige"
2. Ziehen Sie per Drag&Drop das Objekt "Benutzeranzeige" in das Bild.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften".
4. Legen Sie das Erscheinungsbild der "Benutzeranzeige" fest.
5. Wählen Sie z. B. "Darstellung > Größenanpassung > Objekt an Inhalt anpassen".

### Ergebnis

Sie haben eine Benutzeranzeige im Bild angelegt.

## Benutzer anlegen

### Einleitung

Sie legen einen Benutzer an, damit sich Personen unter dem Benutzernamen in Runtime anmelden.

Alternativ können Sie Benutzer im Engineering System anlegen und auf das Bediengerät laden.

Nur wenn der bei der Anmeldung eingegebene Benutzername mit einem Benutzer in Runtime übereinstimmt, ist die Anmeldung erfolgreich. Darüber hinaus muss das bei der Anmeldung eingegebene Kennwort bei dem Benutzer übereinstimmen.

---

**Hinweis**

Beachten Sie die Groß- und Kleinschreibung.

---

Sie ordnen den Benutzer einer Benutzergruppe zu. Der Benutzer besitzt dann die Berechtigungen der Benutzergruppe.

---

**Hinweis**

In Runtime müssen Sie einen Benutzer einer Benutzergruppe zuordnen. Die Benutzergruppe wird im Engineering System angelegt. Die Bezeichnung der Benutzergruppe ist sprachabhängig.

---

## Voraussetzungen

- Die Benutzeranzeige ist geöffnet.
- Eine Benutzergruppe "Group 2" ist angelegt.

## Vorgehensweise

1. Klicken Sie in der Benutzeranzeige <Neuer Benutzer>. Ein Dialog wird geöffnet.
2. Geben Sie als Benutzernamen "Meister" ein.
3. Betätigen Sie die Taste <Return>.
4. Klicken Sie auf "Kennwort".
5. Geben Sie das Kennwort des Benutzers ein.
6. Betätigen Sie die Taste <Return>. Das Kennwort ist unsichtbar.
7. Klicken Sie in die Spalte "Gruppe".



8. Wählen Sie als "Gruppe" "Group 2".

| Benutzer      | Kennwort | Gruppe      | Abmeldezeit |
|---------------|----------|-------------|-------------|
| Administrator | *****    | Administ... | 5           |
| Johnson       | *****    | Administ... | 5           |
| Meister       | *****    | Group 2 ▾   | 5           |
| PLC User      | *****    | Unberec...  | 5           |
|               |          |             |             |
|               |          |             |             |

9. Betätigen Sie die Taste <Return>.

10. Klicken Sie in die Spalte "Abmeldezeit".

11. Geben Sie die Zeit ein, nachdem der Benutzer automatisch abgemeldet wird.

## Benutzer in der einfachen Benutzeranzeige verwalten

### Einleitung

Wenn Sie im Engineering System eine Benutzeranzeige projektieren haben, lassen sich die Benutzer und Benutzergruppen in Runtime verwalten.

#### ACHTUNG

Änderungen in der Benutzeranzeige in Runtime sind sofort wirksam. Änderungen in Runtime werden im Engineering System nicht aktualisiert. Beim Laden der Benutzerverwaltung auf das Bediengerät werden alle Änderungen in der Benutzeranzeige auf Nachfrage und in Abhängigkeit von den Einstellungen überschrieben.

### Voraussetzungen

- Runtime ist aktiviert.
- Die einfache Benutzeranzeige ist angelegt.
- Das Bild mit der einfachen Benutzeranzeige ist geöffnet.
- Sie besitzen die vordefinierte Berechtigung "Benutzerverwaltung".

#### Hinweis

Wenn Sie die Berechtigung "Benutzerverwaltung" nicht besitzen, haben Sie nur das Recht Ihr Kennwort und Ihre Abmeldezeit zu ändern.

### Benutzernamen ändern

1. Klicken Sie in die Zeile des Benutzers, dessen Namen Sie ändern wollen. Ein Dialog wird geöffnet.
2. Geben Sie einen neuen Benutzernamen ein.
3. Bestätigen Sie die Eingabe mit "OK".

---

#### Hinweis

Der Benutzer kann sich nicht mehr mit seinem bisherigen Benutzernamen in Runtime anmelden. Wenn Sie den Namen löschen und die Taste <Return> betätigen, wird der Benutzer gelöscht.

---

### Kennwort des Benutzers ändern in der einfachen Benutzeranzeige

1. Klicken Sie in die Zeile des Benutzers, dessen Kennwort Sie ändern wollen. Ein Dialog wird geöffnet.
2. Geben Sie ein neues Kennwort ein.
3. Bestätigen Sie die Eingabe mit "OK".

---

#### Hinweis

Der Benutzer kann sich nicht mehr mit seinem bisherigen Kennwort in Runtime anmelden. Wenn Sie in der einfachen Benutzeranzeige das Kennwort löschen und mit <Return> betätigen, wird eine Meldung ausgegeben. Die Meldung besagt, dass das Kennwort nicht in den angegebenen Grenzen liegt.

---

### Abmeldezeit des Benutzers ändern

1. Klicken Sie in die Zeile des Benutzers, dessen Anmeldezeit Sie ändern wollen. Ein Dialog wird geöffnet.
2. Geben Sie eine neue Abmeldezeit ein.
3. Bestätigen Sie die Eingabe mit der "OK".

### Benutzer löschen

1. Klicken Sie auf den Namen des zu löschenden Benutzers.
2. Löschen Sie den Namen.
3. Betätigen Sie die Taste <Return>.

---

#### Hinweis

Der Benutzer kann sich nicht mehr in Runtime anmelden.

---

## Benutzer einer anderen Benutzergruppe zuordnen

1. Klicken Sie in die Zeile des Benutzers, dessen Benutzergruppe Sie ändern wollen. Ein Dialog wird geöffnet.
2. Klicken Sie in das Feld "Benutzergruppe".
3. Wählen Sie eine Benutzergruppe aus.
4. Bestätigen Sie die Auswahl mit der "OK".

## Gesperrte Benutzer entsperren

### Gesperrte Benutzer entsperren

In den "Runtime-Einstellungen > Benutzerverwaltung" ist das Kontrollkästchen "Grenze für Login-Versuche aktivieren" aktiviert.

Im Feld "Anzahl der ungültigen Login-Versuche" ist die Zahl 3 eingetragen.

Wenn ein Benutzer bei der Anmeldung drei Fehlversuche erzeugt, z. B. durch eine falsche Kennworteingabe, wird er der Gruppe "Unberechtigt" zugewiesen. Der Benutzer verliert alle Berechtigungen. Der Benutzer kann sich zwar noch anmelden, hat aber keinerlei Berechtigungen mehr. Nur ein Benutzer mit Administratorrechten weist den unberechtigten Benutzer erneut einer Benutzergruppe zu.

## Sich als Benutzer anmelden

### Einleitung

Sie melden sich als Benutzer in der Regel über eine spezielle Schaltfläche an. Dazu erscheint der Anmeldedialog.

Der Anmeldedialog erscheint standardmäßig beim Zugriff auf ein geschütztes Objekt, wenn

- Kein Benutzer in Runtime angemeldet ist
- Der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt.

---

#### Hinweis

Auf den Bediengeräten OP 73, OP 77A, TP 177A und Basic Panels wird beim Bedienen einer Schaltfläche mit Zugriffsschutz immer der Anmeldedialog aufgeblendet:

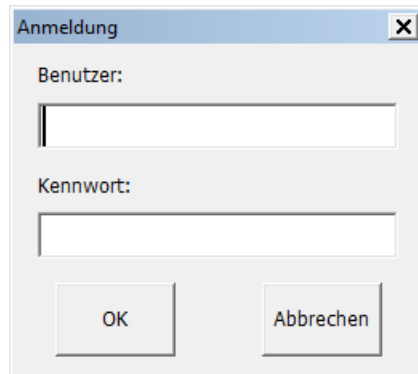
---

### Voraussetzungen

- Unter "Runtime-Einstellungen > Benutzerverwaltung" ist
  - das Kontrollkästchen "Grenze für Login-Versuche aktivieren" aktiviert.
  - im Feld "Anzahl der ungültigen Login-Versuche" die Zahl 3 eingetragen.
- Die Systemfunktion "ZeigeAnmeldedialog" ist an eine Schaltfläche mit der Beschriftung "Logon" projiziert.

## Vorgehensweise

1. Klicken Sie auf die Schaltfläche "Logon". Der Anmeldedialog erscheint.



2. Geben Sie Ihren Benutzernamen ein, wie in der Benutzerverwaltung vergeben, z. B. "Meister".  
Wenn sich vor Ihnen ein Benutzer angemeldet hat, wird dessen Benutzername angezeigt.
3. Geben Sie das zugehörige Kennwort ein. Die Eingabe erfolgt verdeckt.
4. Schließen Sie den Dialog über die Schaltfläche "OK".

## Anmeldung war erfolgreich

Wenn Sie als Benutzernamen "Meister" eingegeben haben und das eingegebene Kennwort mit dem hinterlegten Kennwort übereinstimmt, dann sind Sie in Runtime als Benutzer "Meister" angemeldet. Sie besitzen die Berechtigungen des Benutzers "Meister".

Nur wenn der Benutzer "Meister" beim Zugriff auf ein geschütztes Objekt, z. B. die Schaltfläche "Archivierung", die erforderliche Berechtigung besitzt, wird der Zugriff frei gegeben. Die projektierte Funktion wird sofort ausgeführt.

Wenn Sie nach der erfolgreichen Anmeldung nicht die erforderliche Berechtigung besitzen, erscheint eine entsprechende Fehlermeldung. Sie bleiben aber in Runtime angemeldet.

## Anmeldung war nicht erfolgreich

Eine Fehlermeldung erscheint.

Aus Sicherheitsgründen besitzen Sie oder der zuvor angemeldete Benutzer keine Berechtigungen mehr. Der Zugriff auf nicht geschützte Objekte ist jedoch möglich. Die Benutzeranzeige zeigt jedoch keine Einträge. Benutzeranzeige und Berechtigungen ändern sich bei der nächsten erfolgreichen Anmeldung.

Wenn der dritte Anmeldeversuch fehlgeschlagen ist, wird der Benutzer der vordefinierten Gruppe "Unberechtigt" zugeordnet. Projektieren Sie daher keine Benutzergruppen mit diesem Anzeigenamen.

Wenn die Funktion "Abmelden" aufgerufen wird oder die Abmeldezeit des Benutzers abgelaufen ist, dann wird der Benutzer abgemeldet.

#### 10.5.4.4 Zugriffsschutz projektieren

### Zugriffsschutz

#### Einleitung

Um ein Objekt vor Zugriff zu schützen, projektieren Sie an das Objekt eine Berechtigung. Alle angemeldeten Benutzer, die diese Berechtigung besitzen, greifen dann auf das Objekt zu. Wenn ein Benutzer nicht die Bedienberechtigung eines Objekts besitzt, wird automatisch der Anmeldedialog aufgeblendet.

---

#### Hinweis

Zur Verarbeitung von Benutzer, Kennwort und Benutzergruppe, z. B. in der Steuerung, stehen Ihnen unter "Benutzerverwaltung" mehrere Systemfunktionen zur Verfügung.

---

### Bedienberechtigung projektieren (Basic,Advanced; Professional)

#### Einleitung

Sie projektieren die Berechtig "Runtime beenden" an eine Schaltfläche. Damit haben nur Benutzer darauf Zugriff, die diese Berechtigung besitzen, z. B. alle Benutzer der Benutzergruppe "Bediener".

Dadurch ist der Zugriff auf die Schaltfläche geschützt. Wenn ein angemeldeter Benutzer auf die Schaltfläche klickt, welcher der Benutzergruppe "Bediener" angehört und die erforderliche Berechtigung besitzt, wird Runtime beendet.

Wie Sie eine Schaltfläche mit Zugriffsschutz projektieren, ist in einem Beispiel ausführlich beschrieben.

#### Voraussetzungen

- Die Benutzergruppe "Bediener" ist angelegt.
- Die Berechtigung "Runtime beenden" ist erstellt.
- Ein Bild ist erstellt und geöffnet.
- Das Bild enthält eine Schaltfläche.

#### Vorgehensweise

1. Klicken Sie im Bild auf die Schaltfläche.
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Sicherheit".
3. Wählen Sie als "Berechtigung" Runtime beenden".

4. Wählen Sie im Inspektorfenster "Eigenschaften > Ereignisse > Klicken"
5. Wählen Sie eine Systemfunktion aus der Funktionsliste, z.B. "StoppeRuntime".

---

**Hinweis**

Die Ereignisse "Aktivieren" und "Deaktivieren" dienen ausschließlich zur Erkennung, ob ein Objekt ausgewählt bzw. abgewählt wurde. Die Ereignisse lösen aber keine Kennwortabfrage aus.

Verwenden Sie daher nicht das Ereignis "Aktivieren" bzw. "Deaktivieren", wenn Sie an den Funktionsaufruf des Objekts einen Zugriffsschutz projektieren wollen.

---

## 10.5.5 Referenz

### 10.5.5.1 Objekte mit Zugriffsschutz

#### Einleitung

Folgende Objekte sind mit einer Berechtigung projektierbar:

- Datum/Uhrzeit-Feld
- E/A-Feld
- Grafisches E/A-Feld
- Rezepturanzeige
- Schalter
- Schaltfläche
- Symbolisches E/A-Feld
- System-Diagnoseanzeige

### 10.5.5.2 Vordefinierte Benutzergruppen und Berechtigungen

#### Prinzip

Die vordefinierten Benutzergruppen und Berechtigungen haben folgende Nummern:

| Benutzergruppe          | Nummer |
|-------------------------|--------|
| "Administratorengruppe" | 1      |
| "Benutzer"              | 2      |

| Berechtigung         | Nummer |
|----------------------|--------|
| "Benutzerverwaltung" | 1      |
| "Überwachen"         | 2      |
| "Bedienen"           | 3      |

## 10.5.6 Beispiele

### 10.5.6.1 Beispiel: Schaltfläche mit Anmeldedialog projektieren

#### Aufgabe

In folgendem Beispiel projektieren Sie die Funktion "ZeigeAnmeldedialog" an eine Schaltfläche. Z. B. beim Schichtwechsel meldet sich ein anderer Benutzer in Runtime an. Dabei wird der zuvor angemeldete Benutzer abgemeldet.

---

#### Hinweis

In Runtime erscheint standardmäßig erst dann der Anmeldedialog, wenn Sie auf ein geschütztes Objekt zugreifen. Entweder ist kein Benutzer angemeldet oder der angemeldete Benutzer besitzt nicht die erforderliche Berechtigung.

---

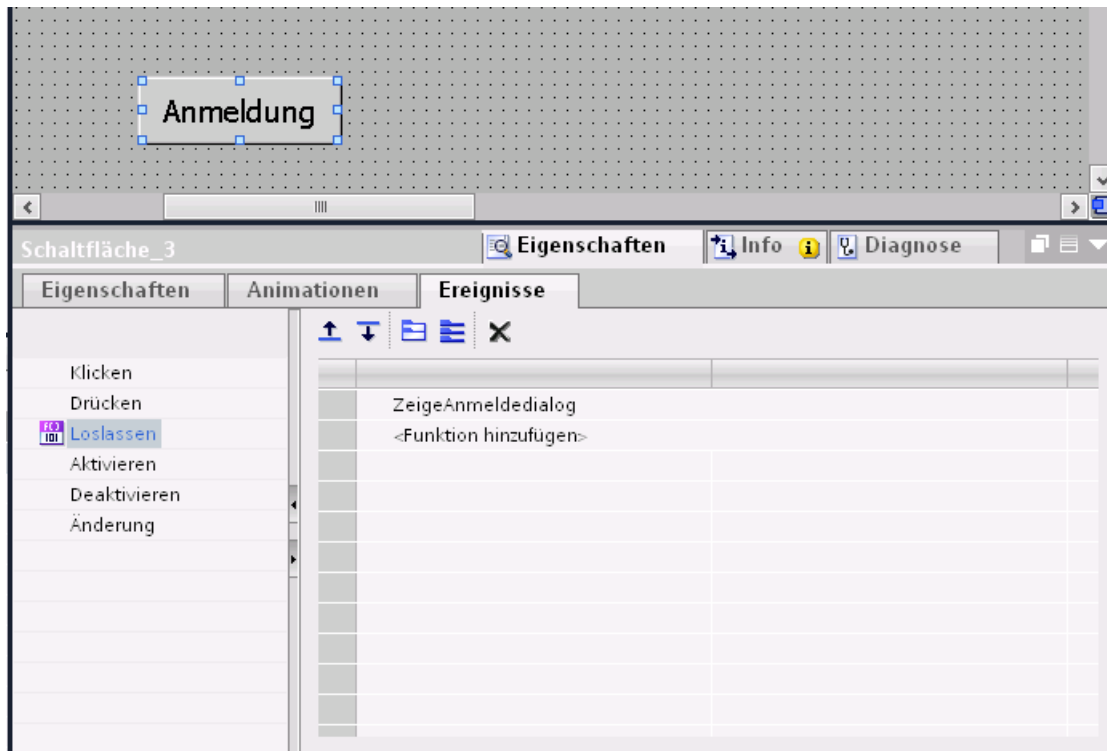
#### Voraussetzungen

- Ein Bild ist erstellt.
- Eine Schaltfläche ist im Bild neu erstellt.

#### Vorgehensweise

1. Klicken Sie im Bild auf die Schaltfläche.
2. Klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Loslassen".

3. Klicken Sie in der Tabelle "Funktionsliste" auf den Eintrag "Funktion hinzufügen".
4. Wählen Sie in der Gruppe "Benutzerverwaltung" die Systemfunktion "ZeigeAnmeldedialog".



## Ergebnis

Wenn der Benutzer in Runtime auf die Schaltfläche klickt, wird die Funktion "ZeigeAnmeldedialog" aufgerufen. Beim Aufruf der Funktion "ZeigeAnmeldedialog" erscheint der Anmeldedialog. Der Benutzer meldet sich mit dem Benutzernamen und seinem Kennwort an.

### 10.5.6.2 Beispiel: Anmelden und Abmelden protokollieren

## Aufgabe

In folgendem Beispiel projektieren Sie die Funktion "VerfolgeBenutzeränderung" an das Ereignis "Benutzerwechsel".

## Prinzip

Wenn sich ein Benutzer anmeldet oder abmeldet, wird die Funktion "VerfolgeBenutzeränderung" aufgerufen. Wenn eine Funktion aufgerufen wird, wird eine Systemmeldung mit Angabe des betreffenden Benutzers ausgegeben.

Diese Systemmeldung können Sie archivieren. Beim Archivieren wird die Systemmeldung mit einem Datumstempel und Zeitstempel versehen. Dadurch können Sie zurückverfolgen, welcher Benutzer zu welcher Zeit und wie lange am Bediengerät angemeldet war.



## Voraussetzungen

- Ein Bediengerät mit Runtime Advanced ist angelegt.
- Das Inspektorfenster ist geöffnet.

## Vorgehensweise

1. Doppelklicken Sie im Projektfenster auf "Aufgabenplaner".
2. Doppelklicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen".
3. Geben Sie als "Name" "Logon-Protocol" ein.
4. Wählen Sie als "Trigger" "Benutzerwechsel".
5. Öffnen Sie im Inspektorfenster "Eigenschaften > Ereignisse".
6. Klicken Sie in der Tabelle "Funktionsliste" auf den Eintrag "Funktion hinzufügen".
7. Wählen Sie die Systemfunktion "VerfolgeBenutzeränderung".

## Ergebnis

Wenn sich ein Benutzer anmeldet oder abmeldet, wird eine Systemmeldung ausgegeben.

### 10.5.6.3 Beispiel einer Benutzerverwaltung

#### Beispiel: Aufbau einer Benutzerverwaltung

#### Aufgabe

Im folgenden Beispiel richten Sie eine Benutzerverwaltung für unterschiedliche Benutzer und Benutzergruppen ein. Das Beispiel orientiert sich an einem typischen Anforderungsprofil aus der Fertigungstechnik.

#### Prinzip

Bei einer Anlage, einem Projekt sind ganz unterschiedliche Personengruppen beteiligt. Jede Personengruppe schützt jeweils ihre Daten und Funktionen vor dem Zugriff anderer. Dazu werden Benutzer angelegt und einer Benutzergruppe zugeordnet.

Sie können unterschiedliche Sichten durch Benutzergruppen abbilden.

Beispiel:

- Organisatorische Sicht: Inbetriebsetzer, Bediener, Schicht I, Schicht II
- Technologische Sicht: Achsensteuerung, Werkzeugwechsler, Anlage Nord, Anlage Süd

Das nachfolgende Beispiel orientiert sich an der organisatorischen Sicht.

Jede Benutzergruppe hat charakteristische Anforderungen an den Zugriffsschutz: Eine Benutzergruppe besitzt Berechtigungen für bestimmte Anwendungsfälle. Ein Programmierer ändert z. B. Rezepturdatensätze.

Im Beispiel werden die Benutzer Müller, Gruppe Meier und Meister angelegt und unterschiedlichen Benutzergruppen zugeordnet.

Frau Müller arbeitet als Programmierer mit dem Engineering System. Bei der Gruppe Meier handelt es sich um Inbetriebsetzer. Herr Meister ist Bediener.

### Voraussetzungen

- Ein neues Projekt ist angelegt.
- Der Editor "Benutzerverwaltung" ist geöffnet.

### Handlungsübersicht

Das Arbeiten mit der Benutzerverwaltung hat im Beispiel folgenden Ablauf:

1. Berechtigungen erstellen: Der Projekteur legt fest, welche Berechtigungen für den Zugriffsschutz benötigt werden.
2. Berechtigungen projektieren: Der Projekteur legt fest, welche Objekte bedient und welche Funktionen ausgeführt werden dürfen.
3. Benutzergruppen anlegen und Berechtigungen zuweisen: Der Administrator legt zusammen mit dem Projekteur die Benutzergruppen an. Der Projekteur legt über die Berechtigungen fest, wer Objekte bedienen und Änderungen an Parametern durchführen darf.
4. Benutzer anlegen und einer Benutzergruppe zuordnen: Der Administrator verwaltet die Benutzer.

### Ergebnis

Das Ziel ist folgender Aufbau der Benutzerverwaltung aus Benutzern, Benutzergruppen und Berechtigungen:

| Benutzer |       |         | Benutzergruppen       | Berechtigungen                    |                                |                                 |           |
|----------|-------|---------|-----------------------|-----------------------------------|--------------------------------|---------------------------------|-----------|
| Müller   | Meier | Meister | Rollen                | Rezeptur-<br>datensätze<br>ändern | System-<br>parameter<br>ändern | Prozess-<br>parameter<br>ändern | Verwalten |
|          |       |         | Administratorengruppe |                                   |                                |                                 | x         |
| X        |       |         | Programmierer         | X                                 |                                |                                 |           |
|          | X     |         | Inbetriebsetzer       | X                                 | X                              | X                               |           |
|          |       | X       | Bediener              | x                                 |                                |                                 |           |

Der Benutzer "Meister", welcher der Benutzergruppe "Bediener" angehört, hat Zugriff auf die projektierte Schaltfläche "Zur Rezepturanzeige".

#### Hinweis

Alternativ können Sie als Bediener mehrere Benutzergruppen anlegen mit unterschiedlichen Bedienberechtigungen z. B. Bediener Stufe1, Bediener Stufe 2.

## Beispiel: Berechtigung erstellen und projektieren

### Aufgabe

Im folgenden Beispiel erstellen Sie die Berechtigungen.

### Vorgehensweise

1. Öffnen Sie den Arbeitsbereich "Benutzergruppen".
2. Doppelklicken Sie in der Tabelle "Berechtigungen" auf "Hinzufügen".
3. Geben Sie als "Name" der Berechtigung "Rezepturdatensätze ändern" ein.
4. Wiederholen Sie die Schritte 2 und 3, um die weiteren Berechtigungen zu erstellen: "Systemparameter ändern", "Prozessparameter ändern".

### Ergebnis

The screenshot shows the 'Benutzergruppen' (User Groups) tab in the WinCC Basic user management interface. It contains two tables: 'Gruppen' (Groups) and 'Berechtigungen' (Permissions).

**Gruppen (Groups):**

| Name                  | Nummer | Anzeigename           | Kennwortalterung         | Kommentar                      |
|-----------------------|--------|-----------------------|--------------------------|--------------------------------|
| Administratorengruppe | 1      | Administratorengruppe | <input type="checkbox"/> | Gruppe 'Administratoren' ha... |
| Benutzer              | 2      | Benutzer              | <input type="checkbox"/> | Gruppe 'Benutzer' hat anfan... |
| <Hinzufügen>          |        |                       |                          |                                |

**Berechtigungen (Permissions):**

| Aktiv                               | Name                      | Anzeigename               | Nummer | Kommentar                    |
|-------------------------------------|---------------------------|---------------------------|--------|------------------------------|
| <input type="checkbox"/>            | Benutzerverwaltung        | Benutzerverwaltung        | 1      | Berechtigung 'Benutzerver... |
| <input type="checkbox"/>            | Überwachen                | Überwachen                | 2      | Berechtigung 'Überwache...   |
| <input checked="" type="checkbox"/> | Bedienen                  | Berechtigung              | 3      | Berechtigung 'Bedienen'.     |
| <input type="checkbox"/>            | Rezepturdatensätze ändern | Rezepturdatensätze ändern | 4      |                              |
| <input type="checkbox"/>            | Systemparameter ändern    | Systemparameter ändern    | 5      |                              |
| <input type="checkbox"/>            | Prozessparameter ändern   | Prozessparameter ändern   | 6      |                              |
|                                     | <Hinzufügen>              |                           |        |                              |

## Beispiel: Schaltfläche mit Zugriffsschutz projektieren



### Aufgabe

Im folgenden Beispiel erstellen Sie mithilfe einer Systemfunktion eine Schaltfläche für einen Bildwechsel. Sie schützen die Schaltfläche "Zur Rezepturanzeige" vor unbefugter Bedienung. Dazu projektieren Sie die Berechtigung "Rezepturdatensätze ändern" an die Schaltfläche "Zur Rezepturanzeige".

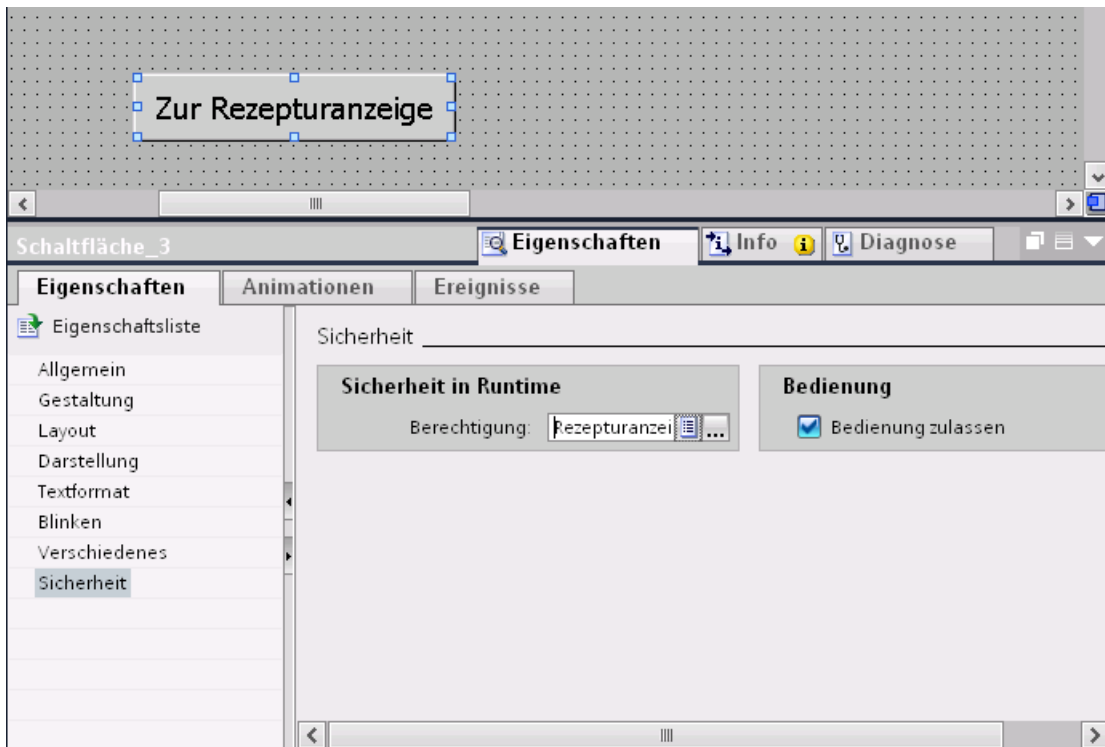
### Voraussetzungen

- Eine Berechtigung "Rezepturdatensätze ändern" ist erstellt.
- Ein Bild "Rezepturen" ist erstellt.
- Ein Bild "Start" ist erstellt und geöffnet.
- Eine Schaltfläche ist im Bild "Start" angelegt und markiert.

### Vorgehensweise

1. Klicken Sie im Inspektorfenster auf "Eigenschaften > Eigenschaften > Allgemein".
2. Geben Sie als Text "Zur Rezepturanzeige" ein.
3. Klicken Sie im Inspektorfenster auf "Eigenschaften > Ereignisse > Klicken".
4. Klicken Sie in der Tabelle "Funktionsliste" in der ersten Zeile auf den Eintrag "Funktion hinzufügen".
5. Wählen Sie in der Gruppe "Bilder" die Systemfunktion "AktiviereBild".
6. Klicken Sie im Feld "Bildname" auf die Schaltfläche . Ein Dialog zur Auswahl des Bildes wird geöffnet.
7. Wählen Sie das Bild "Rezepturen" und schließen Sie den Dialog über die Schaltfläche .
8. Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Sicherheit".
9. Wählen Sie als "Berechtigung" "Rezepturdatensätze ändern".

## Ergebnis



Der Zugriff auf die Schaltfläche "Zur Rezepturanzeige" ist geschützt. Wenn z. B. der Benutzer "Meier" in Runtime auf die Schaltfläche klickt, wird das Bild "Rezepturanzeige" geöffnet. Vorausgesetzt der Benutzer "Meier" ist korrekt angemeldet und hat die erforderliche Berechtigung. Das Bild "Rezepturen" enthält eine Rezepturanzeige und andere Bildobjekte.

Wenn der angemeldete Benutzer nicht die erforderliche Berechtigung besitzt oder kein Benutzer angemeldet ist, erscheint der Anmeldedialog.

## Beispiel: Benutzergruppen anlegen und Berechtigungen zuweisen

### Aufgabe

Im folgenden Beispiel legen Sie die Benutzergruppen an und weisen ihnen Berechtigungen zu.

### Vorgehensweise

1. Öffnen Sie den Arbeitsbereich "Benutzergruppen".
2. Doppelklicken Sie in der Tabelle "Gruppen" auf "Hinzufügen".
3. Geben Sie als "Name" "Programmierer" ein.
4. Wiederholen Sie die Schritte 2 und 3, um die Benutzergruppen "Inbetriebsetzer" und "Bediener" anzulegen.

5. Klicken Sie in der Tabelle "Gruppen" auf "Administratorengruppe".
6. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Systemparameter ändern".

### Zwischenergebnis

The screenshot shows a software interface with two main sections: "Gruppen" and "Berechtigungen".

**Gruppen**

| Name                  | Nummer | Anzeigename           | Kennwortalterung         | Kommentar                   |
|-----------------------|--------|-----------------------|--------------------------|-----------------------------|
| Administratorengruppe | 1      | Administratorengruppe | <input type="checkbox"/> | Gruppe 'Administratoren' ha |
| Benutzer              | 2      | Benutzer              | <input type="checkbox"/> | Gruppe 'Benutzer' hat anfan |
| Programmierer         | 3      | Programmierer         | <input type="checkbox"/> |                             |
| Inbetriebsetzer       | 4      | Inbetriebsetzer       | <input type="checkbox"/> |                             |
| Bediener              | 5      | Bediener              | <input type="checkbox"/> |                             |

**Berechtigungen**

| Aktiv                               | Name                      | Anzeigename               | Nummer | Kommentar                |
|-------------------------------------|---------------------------|---------------------------|--------|--------------------------|
| <input checked="" type="checkbox"/> | Benutzerverwaltung        | Benutzerverwaltung        | 1      | Berechtigung 'Benutzerve |
| <input checked="" type="checkbox"/> | Überwachen                | Überwachen                | 2      | Berechtigung 'Überwache  |
| <input checked="" type="checkbox"/> | Bedienen                  | Berechtigung              | 3      | Berechtigung 'Bedienen'. |
| <input type="checkbox"/>            | Rezepturdatensätze ändern | Rezepturdatensätze ändern | 4      |                          |
| <input checked="" type="checkbox"/> | Systemparameter ändern    | Systemparameter ändern    | 5      |                          |
| <input type="checkbox"/>            | Prozessparameter ändern   | Prozessparameter ändern   | 6      |                          |

### Vorgehensweise

1. Klicken Sie in der Tabelle "Gruppen" auf "Bediener".
2. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Rezepturdatensätze ändern".
3. Klicken Sie in der Tabelle "Gruppen" auf "Inbetriebsetzer".
4. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Rezepturdatensätze ändern", "Systemparameter ändern" und "Prozessparameter ändern".
5. Klicken Sie in der Tabelle "Gruppen" auf "Programmierer".
6. Aktivieren Sie in der Tabelle "Berechtigungen" die Berechtigung "Rezepturdatensätze ändern".

## Ergebnis



| Benutzer                            |                           | Benutzergruppen           |                          |                              |  |
|-------------------------------------|---------------------------|---------------------------|--------------------------|------------------------------|--|
| <b>Gruppen</b>                      |                           |                           |                          |                              |  |
| Name                                | Nummer                    | Anzeigename               | Kennwortalterung         | Kommentar                    |  |
| Administratorengruppe               | 1                         | Administratorengruppe     | <input type="checkbox"/> | Gruppe 'Administratoren' ha  |  |
| Benutzer                            | 2                         | Benutzer                  | <input type="checkbox"/> | Gruppe 'Benutzer' hat anfan. |  |
| Programmierer                       | 3                         | Programmierer             | <input type="checkbox"/> |                              |  |
| Inbetriebsetzer                     | 4                         | Inbetriebsetzer           | <input type="checkbox"/> |                              |  |
| Bediener                            | 5                         | Bediener                  | <input type="checkbox"/> |                              |  |
| <b>Berechtigungen</b>               |                           |                           |                          |                              |  |
| Aktiv                               | Name                      | Anzeigename               | Nummer                   | Kommentar                    |  |
| <input type="checkbox"/>            | Benutzerverwaltung        | Benutzerverwaltung        | 1                        | Berechtigung 'Benutzerver    |  |
| <input type="checkbox"/>            | Überwachen                | Überwachen                | 2                        | Berechtigung 'Überwache      |  |
| <input type="checkbox"/>            | Bedienen                  | Berechtigung              | 3                        | Berechtigung 'Bedienen'.     |  |
| <input checked="" type="checkbox"/> | Rezepturdatensätze ändern | Rezepturdatensätze ändern | 4                        |                              |  |
| <input type="checkbox"/>            | Systemparameter ändern    | Systemparameter ändern    | 5                        |                              |  |
| <input type="checkbox"/>            | Prozessparameter ändern   | Prozessparameter ändern   | 6                        |                              |  |
| <Hinzufügen>                        |                           |                           |                          |                              |  |

## Beispiel: Benutzer anlegen und einer Benutzergruppe zuordnen

### Aufgabe

Im folgenden Beispiel legen Sie die Benutzer an und ordnen sie ihren Benutzergruppen zu. Der Benutzer wird gleich nach der Eingabe des Namens alphabetisch einsortiert.

### Vorgehensweise

1. Öffnen Sie den Arbeitsbereich "Benutzer".
2. Doppelklicken Sie in der Tabelle "Benutzer" auf "Hinzufügen".
3. Geben Sie als Benutzernamen "Mueller" ein.
4. Klicken Sie in der Spalte "Kennwort" auf die Schaltfläche . Der Dialog zur Eingabe des Kennworts wird geöffnet.
5. Geben Sie als Kennwort "mueller" ein.
6. Geben Sie zur Bestätigung das Kennwort im darunter liegenden Feld ein zweites Mal ein.
7. Schließen Sie den Dialog über das Symbol .
8. Aktivieren Sie in der Tabelle "Gruppen" die Benutzergruppe "Programmierer".

### Zwischenergebnis

The screenshot shows a software interface with two tabs: 'Benutzer' (selected) and 'Benutzergruppen'. Below the tabs are two tables.

**Benutzer**

|  | Name          | Kennwort | Automatisches Abmel...              | Abmeldezeit | Nummer | Kommentar                |
|--|---------------|----------|-------------------------------------|-------------|--------|--------------------------|
|  | Administrator | *****    | <input checked="" type="checkbox"/> | 5           | 1      | Benutzer 'Administrator' |
|  | Mueller       | ***      | <input checked="" type="checkbox"/> | 5           | 2      |                          |
|  | <Hinzufügen>  |          |                                     |             |        |                          |

**Gruppen**

|  | Mitglied von                     | Name                  | Nummer | Anzeigename           | Kennwortalterung         | Kommentar    |
|--|----------------------------------|-----------------------|--------|-----------------------|--------------------------|--------------|
|  | <input type="radio"/>            | Administratorengruppe | 1      | Administratorengruppe | <input type="checkbox"/> | Gruppe 'Adm  |
|  | <input type="radio"/>            | Benutzer              | 2      | Benutzer              | <input type="checkbox"/> | Gruppe 'Benu |
|  | <input checked="" type="radio"/> | Programmierer         | 3      | Programmierer         | <input type="checkbox"/> |              |
|  | <input type="radio"/>            | Inbetriebsetzer       | 4      | Inbetriebsetzer       | <input type="checkbox"/> |              |
|  | <input type="radio"/>            | Bediener              | 5      | Bediener              | <input type="checkbox"/> |              |
|  | <Hinzufügen>                     |                       |        |                       |                          |              |

### Vorgehensweise

1. Doppelklicken Sie in der Tabelle "Benutzer" auf "Hinzufügen".
2. Geben Sie als Benutzernamen "Meier" ein.
3. Klicken Sie in der Spalte "Kennwort" auf die Schaltfläche . Der Dialog zur Eingabe des Kennworts wird geöffnet.
4. Geben Sie als Kennwort "meier" ein.
5. Geben Sie zur Bestätigung das Kennwort im darunter liegenden Feld ein zweites Mal ein.
6. Schließen Sie den Dialog über das Symbol .
7. Aktivieren Sie in der Tabelle "Gruppen" die Benutzergruppe "Inbetriebsetzer".
8. Wiederholen Sie die Schritte 2 bis 6 für den Benutzer "Meister".
9. Aktivieren Sie in der Tabelle "Gruppen" die Benutzergruppe "Bediener".



## Ergebnis

The screenshot displays the WinCC user management interface. At the top, there are two tabs: 'Benutzer' (selected) and 'Benutzergruppen'. Below the tabs, there are two main sections: 'Benutzer' and 'Gruppen'.

**Benutzer**

| Name          | Kennwort | Automatisches Abmel...              | Abmeldezeit | Nummer | Kommentar            |
|---------------|----------|-------------------------------------|-------------|--------|----------------------|
| Administrator | *****    | <input checked="" type="checkbox"/> | 5           | 1      | Benutzer 'Administra |
| Mueller       | ***      | <input checked="" type="checkbox"/> | 5           | 2      |                      |
| Meier         | ***      | <input checked="" type="checkbox"/> | 5           | 3      |                      |
| Meister       | ***      | <input checked="" type="checkbox"/> | 5           | 4      |                      |

**Gruppen**

| Mitglied von                     | Name                  | Nummer | Anzeigename           | Kennwortalterung         | Kommentar   |
|----------------------------------|-----------------------|--------|-----------------------|--------------------------|-------------|
| <input type="radio"/>            | Administratorengruppe | 1      | Administratorengruppe | <input type="checkbox"/> | Gruppe 'Adm |
| <input type="radio"/>            | Benutzer              | 2      | Benutzer              | <input type="checkbox"/> | Gruppe 'Ben |
| <input type="radio"/>            | Programmierer         | 3      | Programmierer         | <input type="checkbox"/> |             |
| <input type="radio"/>            | Inbetriebsetzer       | 4      | Inbetriebsetzer       | <input type="checkbox"/> |             |
| <input checked="" type="radio"/> | Bediener              | 5      | Bediener              | <input type="checkbox"/> |             |
| <Hinzufügen>                     |                       |        |                       |                          |             |

## 10.6 Mit Systemfunktionen arbeiten

### 10.6.1 Grundlagen

#### 10.6.1.1 Systemfunktionen

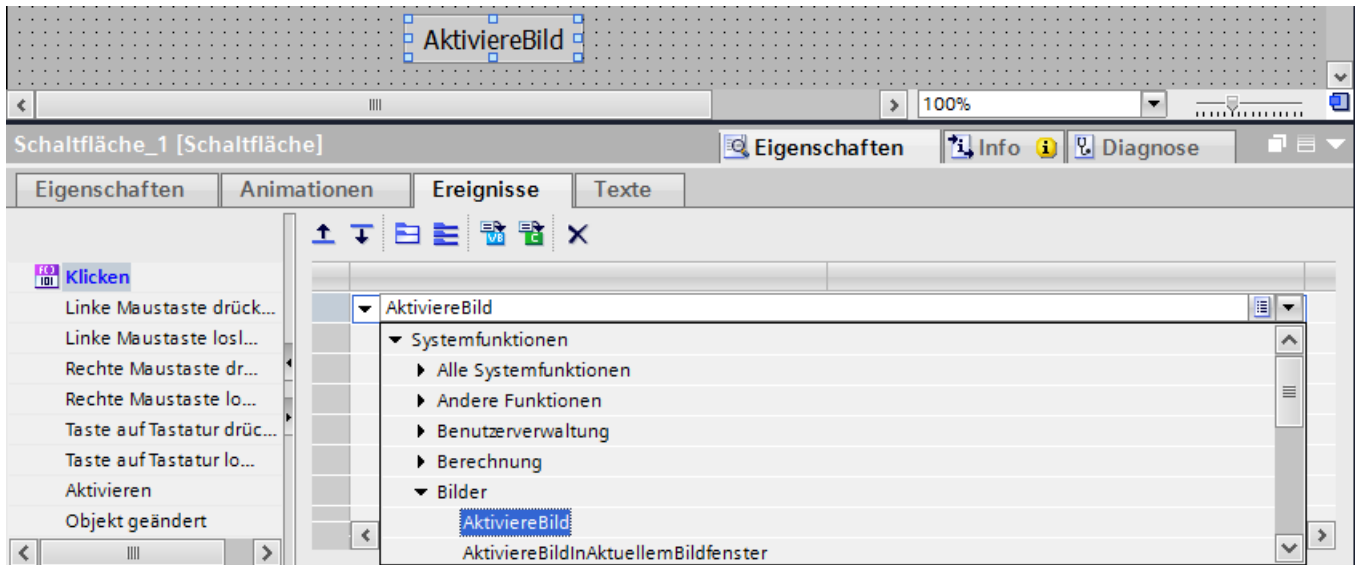
### Einleitung

Systemfunktionen sind von WinCC mitgelieferte Funktionen. Sie sind vordefiniert und können nicht geändert werden. Mit den Systemfunktionen lassen sich auch ohne Programmierkenntnisse viele Aufgaben in Runtime realisieren, z. B.:

- Berechnungen, z. B. das Erhöhen eines Variablenwerts um einen bestimmten oder variablen Betrag.
- Archivfunktionen, z. B. das Starten eines Prozesswertarchivs.
- Einstellungen, z. B. das Wechseln der Steuerung oder das Setzen eines Bits in der Steuerung.
- Meldungen, z. B. nach dem Ändern des Benutzers.

## Verwendung

Systemfunktionen verwenden Sie in einer Funktionsliste. Wenn Sie eine Funktionsliste projektieren, wählen Sie die Systemfunktionen aus einer nach Kategorien geordneten Auswahlliste aus:



In WinCC ist jede Systemfunktion mit einem Objekt und einem Ereignis verknüpft. Sobald das Ereignis eintritt, wird die Systemfunktion ausgelöst.

## Sprachabhängigkeit

Die Namen der Systemfunktionen sind abhängig von der eingestellten Projektiersprache. Die Funktionalität ist damit für den Projektteur sofort erkennbar.

## Verfügbarkeit

In einer Funktionsliste projektieren Sie nur die Systemfunktionen, die vom ausgewählten Bediengerät unterstützt werden. Wenn Sie ein Projekt für mehrere Bediengeräte verwenden, werden die Systemfunktionen farblich gekennzeichnet, die von einem Bediengerät nicht unterstützt werden.

## Ereignisse

Welches Ereignis als Auslöser für die Ausführung einer Systemfunktion definiert werden kann, hängt vom jeweiligen Objekt und von der gewählten Funktion ab.

So gehören z. B. zu dem Objekt "Variable" die Ereignisse "Wertänderung", "Untergrenze unterschritten" und "Obergrenze überschritten". Zu dem Objekt "Bild" gehören die Ereignisse "Aufgebaut" und "Abgebaut".

## 10.6.1.2 Einsatz von Systemfunktionen

### Einleitung

Wenn ein projektiertes Ereignis eintritt, wird in Runtime eine Funktionsliste ausgeführt. Der Bediener triggert ein Ereignis, indem er z. B. eine Funktionstaste am Bediengerät drückt. Ein Ereignis kann auch vom System getriggert werden, z. B. wenn ein Prozesswert einen Grenzwert unterschreitet.

### Einsatzmöglichkeiten

Systemfunktionen können Sie an alle Objekte projektieren, die auf ein Ereignis reagieren können. Systemfunktionen können Sie direkt in Funktionslisten einsetzen und damit den Ablauf steuern.

- Funktionsliste  
In einer Funktionsliste werden Systemfunktionen Zeile für Zeile abgearbeitet. Um Wartezeiten zu vermeiden, werden bei WinCC Runtime Systemfunktionen mit längerer Laufzeit parallel abgearbeitet. Eine nachfolgende Systemfunktion wird z. B. bereits ausgeführt, obwohl die vorhergehende Systemfunktion noch nicht beendet ist.

Ein Beispiel für die Projektierung einer Funktionsliste finden Sie unter "Wechsel der Betriebsart am Bediengerät mit aktueller Anzeige".

## 10.6.2 Arbeiten mit Funktionslisten

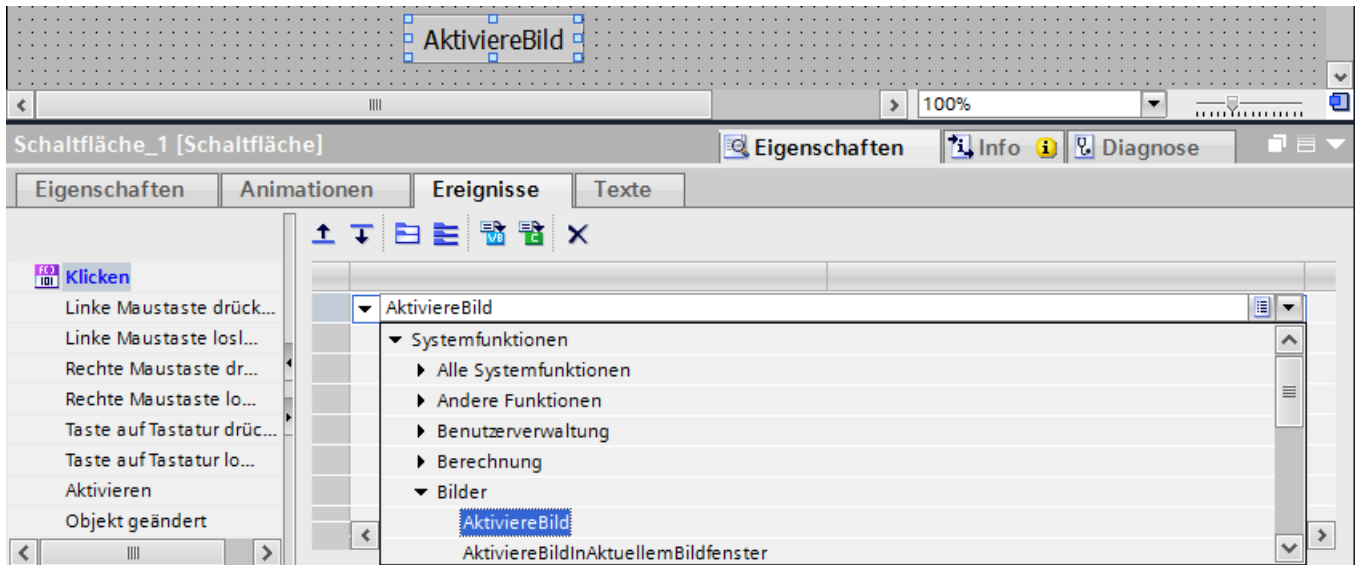
### 10.6.2.1 Grundlagen zur Funktionsliste

#### Einleitung

Mit der Funktionsliste führen Sie beim Eintreten des projektierten Ereignisses mehrere Systemfunktionen aus.

## Prinzip

Die Funktionsliste projektieren Sie an ein Ereignis eines Objekts, z. B. ein Bildobjekt oder eine Variable. Welche Ereignisse zur Verfügung stehen, hängt vom ausgewählten Objekt und Bediengerät ab.



Ereignisse treten nur ein, während das Projekt in Runtime ist. Ereignisse sind z. B.:

- Wertänderung einer Variablen
- Drücken einer Schaltfläche
- Aktivieren von Runtime

An jedes Ereignis projektieren Sie genau eine Funktionsliste.

---

### Hinweis

Die Auswahl der projektierbaren Systemfunktionen in einer Funktionsliste ist vom ausgewählten Bediengerät abhängig.

---

## 10.6.2.2 Eigenschaften einer Funktionsliste

### Statusinformation

Während der Projektierung werden die Projektdaten im Hintergrund geprüft.

Bei folgenden Ursachen wird die Funktionsliste nicht in Runtime ausgeführt und die fehlerhaften Einträge rot markiert:

- Mindestens eine Systemfunktion ist nicht vollständig mit Parametern versorgt.
- Mindestens eine Systemfunktion ist enthalten, die vom gewählten Bediengerät nicht unterstützt wird, z. B. durch Wechsel des Gerätetyps

## Abarbeitung von Systemfunktionen

Systemfunktionen in einer Funktionsliste werden in Runtime sequenziell von oben nach unten abgearbeitet. Um Wartezeiten zu vermeiden, werden Systemfunktionen mit längerer Laufzeit (z. B. Dateioperationen) parallel abgearbeitet. Eine nachfolgende Systemfunktion wird z. B. bereits ausgeführt, obwohl die vorhergehende Systemfunktion noch nicht beendet ist.

Um nicht sequenzielle und bedingte Abläufe zu programmieren, verwenden Sie ein Skript mit Schleifen, bedingten Anweisungen und Abbruchbedingungen.

---

### Hinweis

#### Geräteabhängigkeit

Benutzerdefinierte Funktionen stehen auf Basic Panels nicht zur Verfügung.

---

### 10.6.2.3 Funktionsliste projektieren

#### Einleitung

Sie projektieren eine Funktionsliste, indem Sie die Systemfunktionen aus einer Klappliste auswählen. Die Systemfunktionen sind in der Klappliste nach Kategorien geordnet.

#### Voraussetzung

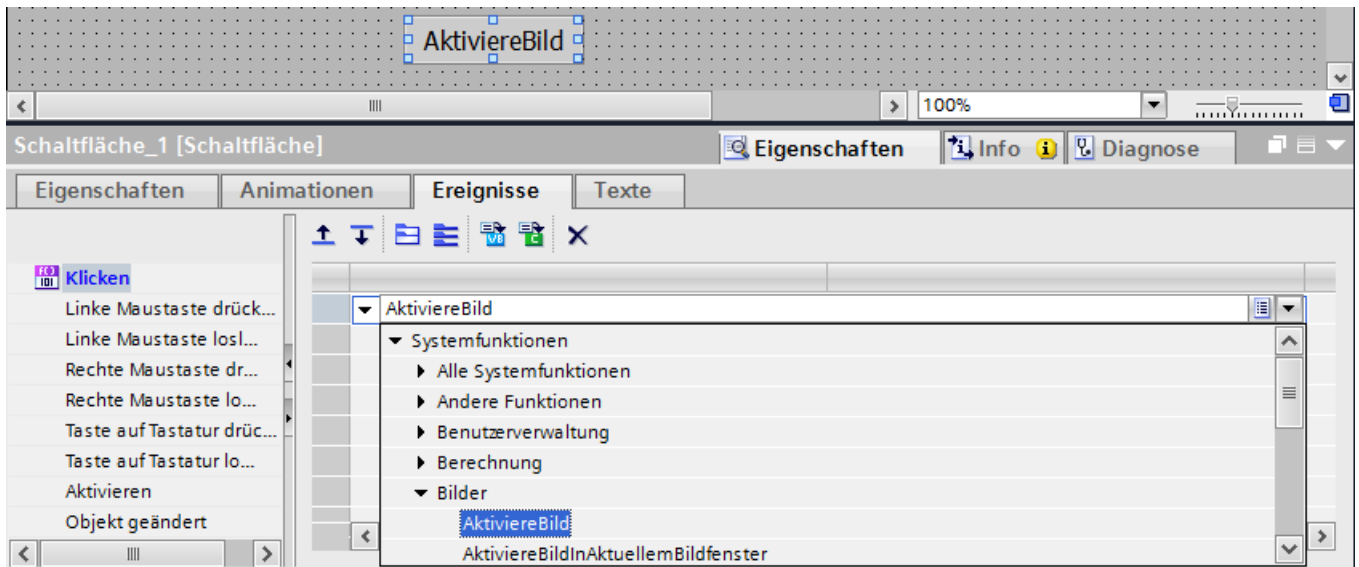
Objekt besitzt mindestens ein projektierbares Ereignis.

#### Vorgehen

Um eine Funktionsliste zu projektieren, gehen Sie folgendermaßen vor:

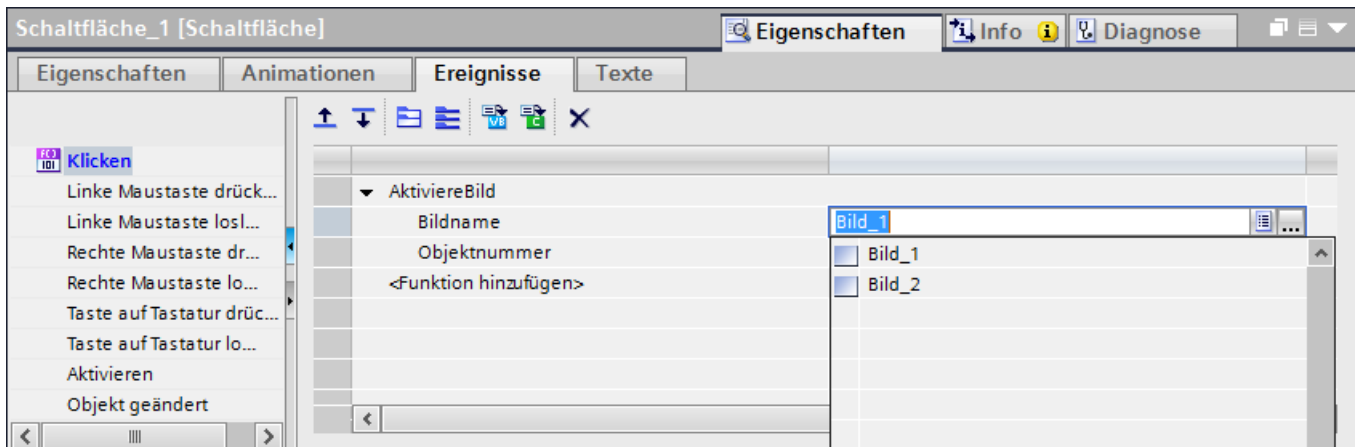
1. Öffnen Sie in WinCC den Editor, in dem sich das Objekt befindet.
2. Markieren Sie das Objekt.
3. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse". Markieren Sie das Ereignis, an das Sie die Funktionsliste projektieren wollen.
4. Markieren Sie im Inspektorfenster in der Klappliste den Eintrag "<Keine Funktion>".
5. Wählen Sie die gewünschte Systemfunktion.

6. Alternativ geben Sie den Namen der Systemfunktion ein.



Die Systemfunktion wird in die Funktionsliste eingetragen.

7. Wenn die Systemfunktion Parameter besitzt, wählen Sie für die Parameter die entsprechenden Werte aus.



8. Wenn Sie der Funktionsliste weitere Systemfunktionen oder Funktionen hinzufügen wollen, dann wiederholen Sie die Schritte vier bis sieben.

### Ergebnis

Die Funktionsliste ist projiziert. Im Inspektorfenster wird neben dem projizierten Ereignis der Status der Funktionsliste angezeigt. Wenn in Runtime das projizierte Ereignis eintritt, wird die Funktionsliste von oben nach unten abgearbeitet.



### 10.6.2.5 Abarbeitung der Funktionsliste in Runtime

#### Prinzip

Eine Funktionsliste wird in Runtime von oben nach unten abgearbeitet. Damit es während der Abarbeitung nicht zu Wartezeiten kommt, wird zwischen synchroner Abarbeitung und asynchroner Abarbeitung unterschieden. Die Unterscheidung wird vom System getroffen, indem es die verschiedenen Laufzeiten der Systemfunktionen auswertet. Benutzerdefinierte Funktionen werden unabhängig von der Laufzeit stets synchron abgearbeitet. Wenn eine Systemfunktion einen Fehlerstatus zurückgibt, wird die Abarbeitung der Funktionsliste abgebrochen.

#### Synchrone Abarbeitung

Bei der synchronen Abarbeitung werden die Systemfunktionen in einer Funktionsliste nacheinander ausgeführt. Damit eine Systemfunktion ausgeführt wird, muss die vorhergehende Systemfunktion beendet sein.

#### Asynchrone Abarbeitung

Systemfunktionen, die Dateioperationen wie Speichern oder Lesen ausführen, haben eine längere Laufzeit als Systemfunktionen, die z. B. einen Variablenwert setzen.

Systemfunktionen mit längerer Laufzeit werden deshalb asynchron ausgeführt. Während eine Systemfunktion z. B. einen Rezepturdatensatz auf ein Speichermedium schreibt, wird bereits die nächste Systemfunktion ausgeführt. Durch die parallele Abarbeitung der Systemfunktionen werden Wartezeiten am Bediengerät vermieden.

### 10.6.3 Beispiel

#### 10.6.3.1 Wechsel der Betriebsart am Bediengerät mit aktueller Anzeige

#### Aufgabe

In diesem Beispiel verwenden Sie die Systemfunktion "SetzeBetriebsartGerät", um am Bediengerät zwischen den Betriebsarten "Online" und "Offline" zu wechseln. Sie lassen die aktuell eingestellte Betriebsart zusätzlich am Bediengerät anzeigen.

#### Voraussetzungen

Ein Prozessbild ist erstellt.

#### Einstellungen

Für das Beispiel benötigen Sie eine HMI-Variable und eine Textliste mit folgenden Einstellungen:



HMI-Variable:

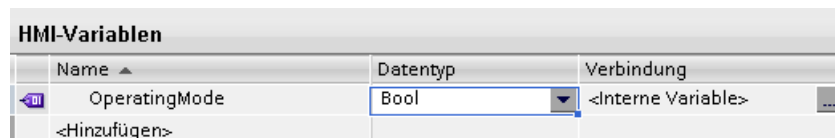
| Name          | Steuerungsanbindung | Typ  |
|---------------|---------------------|------|
| OperatingMode | nein                | Bool |

Textliste:

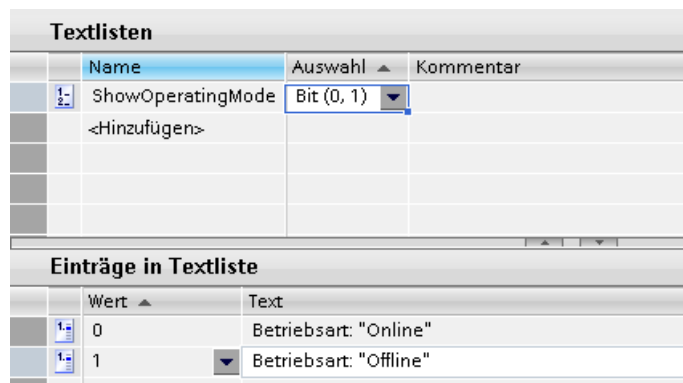
| Name              | Enthält   | Werte   |
|-------------------|-----------|---|
| ShowOperatingMode | Bit (0/1) | 1: Betriebsart: "Online"<br>0: Betriebsart: "Offline" |

## Vorgehen

1. Erstellen Sie die oben genannte HMI-Variable "OperatingMode".

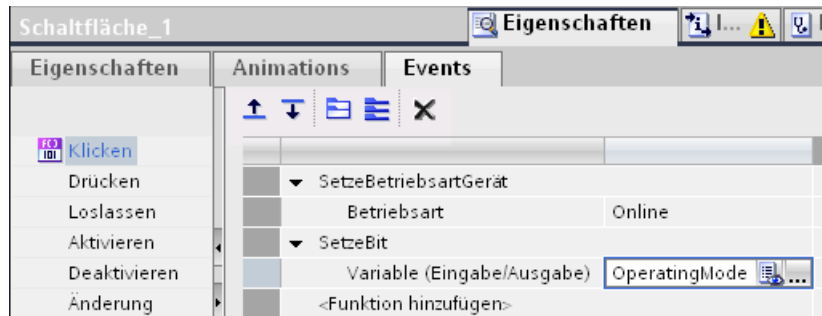


2. Erstellen Sie die oben genannte Textliste "ShowOperatingMode".

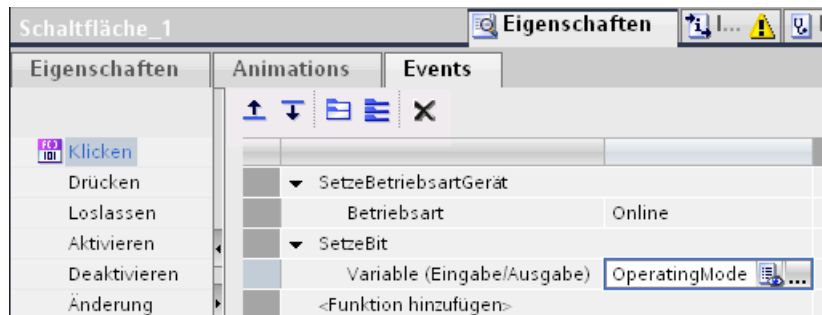


3. Öffnen Sie das Bild und fügen Sie eine Schaltfläche ein, an die Sie den Betriebsartenwechsel nach "Online" projektieren.
4. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse". Markieren Sie das Ereignis "Drücken".
5. Projektieren Sie die Systemfunktion "SetzeBetriebsartGerät" an das Ereignis "Drücken". Die Systemfunktion finden Sie in der Auswahlliste unter "Einstellungen".
6. Wählen Sie für den Parameter "Betriebsart" den Eintrag "Online" aus.
7. Projektieren Sie die Systemfunktion "SetzeBit" an das Ereignis "Drücken". Die Systemfunktion finden Sie in der Auswahlliste unter "Bitverarbeitung".

- 8. Wählen Sie für den Parameter "Variable" aus der Auswahlliste die HMI-Variable "OperatingMode" aus.



- 9. Fügen Sie im Prozessbild eine Schaltfläche ein, an die Sie den Betriebsartenwechsel nach "Offline" projektieren.
- 10. Wiederholen Sie die Schritte vier bis sieben. Wählen Sie für den Parameter "Betriebsart" den Eintrag "Offline". Projektieren Sie die Systemfunktion "RücksetzeBit" anstelle der Systemfunktion "SetzeBit".



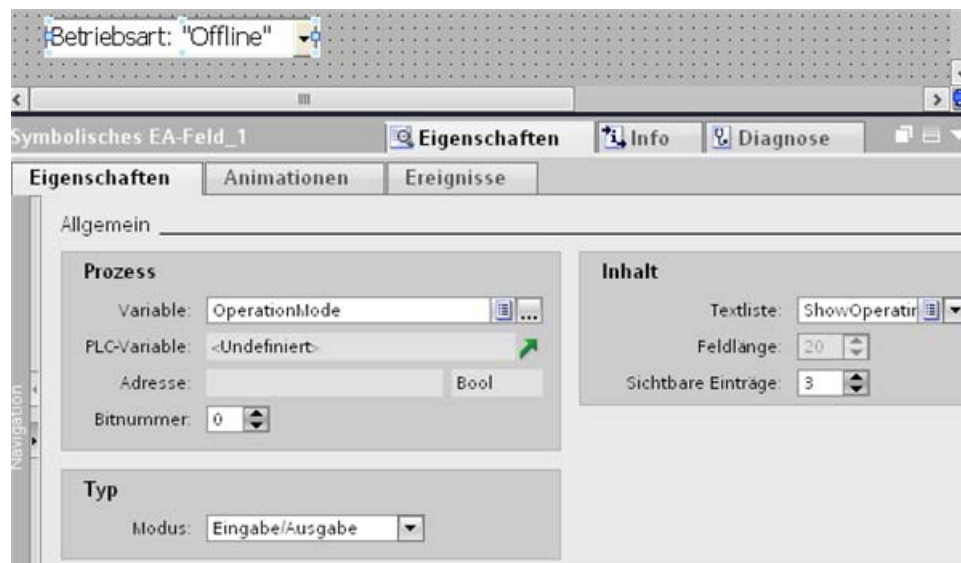
### Zwischenergebnis

Sie können in Runtime mit den beiden Schaltflächen die Betriebsart des Bediengeräts umschalten.

Sie wollen die aktuell eingestellte Betriebsart in einem Ausgabefeld am Bediengerät anzeigen.

## Vorgehen

1. Erstellen Sie im Prozessbild ein "Symbolisches E/A-Feld". Klicken Sie im Inspektorfenster "Eigenschaften > Eigenschaften".
2. Nehmen Sie in der Gruppe "Allgemein" folgende Einstellungen vor:
  - Wählen Sie als "Modus" "Ausgabe".
  - Wählen Sie als "Textliste" die Textliste "ShowOperatingMode".
  - Wählen Sie als "Variable" "OperatingMode".



## Ergebnis

Wenn Sie mit den Schaltflächen die Bedienart wechseln, wird stets die aktuell eingestellte Betriebsart am Bediengerät angezeigt.

## 10.6.4 Referenz

### 10.6.4.1 Funktionsliste

## Geräteabhängigkeit von Systemfunktionen

## Verfügbarkeit der Systemfunktionen

Die nachfolgenden Tabellen zeigen die Verfügbarkeit von Systemfunktionen und benutzerdefinierten Funktionen auf den Bediengeräten.

Technische Änderungen vorbehalten.

Übersicht

|   | KP300<br>Basic<br>PN | KTP400<br>Basic PN | KTP 600<br>Basic DP | KTP 600<br>Basic PN | KTP<br>1000<br>Basic DP | KTP<br>1000<br>Basic PN | TP 1500<br>Basic<br>PN |
|---|----------------------|--------------------|---------------------|---------------------|-------------------------|-------------------------|------------------------|
| Benutzerdefinierte Funktionen                 | nein                 | nein               | nein                | nein                | nein                    | nein                    | nein                   |
| Abmelden (Seite 3352)                         | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| ÄndereKontrast (Seite 3353)                   | ja                   | ja                 | nein                | ja <sup>1)</sup>    | ja                      | ja                      | ja                     |
| AktiviereBild (Seite 3354)                    | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| AktiviereBildMitNummer (Seite 3355)           | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| AktivierePutzbild (Seite 3356)                | nein                 | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| AktiviereVorherigesBild (Seite 3357)          | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| AktualisiereVariable (Seite 3358)             | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| Anmelden (Seite 3358)                         | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| BearbeiteMeldung (Seite 3359)                 | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| BildobjektCursorAbwärts (Seite 3360)          | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| BildobjektCursorAufwärts<br>(Seite 3360)      | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| BildobjektSeiteAbwärts (Seite 3361)           | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| BildobjektSeiteAufwärts (Seite 3361)          | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| ErhöheFokussiertenWert<br>(Seite 3362)        | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| ErhöheVariable (Seite 3362)                   | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| GeheZumAnfang (Seite 3363)                    | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| GeheZumEnde (Seite 3364)                      | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| InvertiereBit (Seite 3365)                    | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| InvertiereBitInVariable (Seite 3366)          | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| InvertiereLineareSkalierung<br>(Seite 3367)   | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KalibriereTouchscreen (Seite 3368)            | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KurvenanzeigeBlättereVorwärts<br>(Seite 3369) | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KurvenanzeigeBlättereZurück<br>(Seite 3370)   | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KurvenanzeigeErweitern (Seite 3370)           | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KurvenanzeigeKomprimieren<br>(Seite 3371)     | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KurvenanzeigeLinealRechts<br>(Seite 3372)     | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KurvenanzeigeLinealLinks<br>(Seite 3371)      | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KurvenanzeigeSetzeLinealmodus<br>(Seite 3372) | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| KurvenanzeigeStartenStoppen<br>(Seite 3373)   | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |

|  | KP300<br>Basic<br>PN | KTP400<br>Basic PN | KTP 600<br>Basic DP | KTP 600<br>Basic PN | KTP<br>1000<br>Basic DP | KTP<br>1000<br>Basic PN | TP 1500<br>Basic<br>PN |
|--|----------------------|--------------------|---------------------|---------------------|-------------------------|-------------------------|------------------------|
| KurvenanzeigeZurückZumAnfang<br>(Seite 3373)                 | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| LeseBenutzername (Seite 3374)                                | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| LeseGruppennummer (Seite 3375)                               | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| LeseKennwort (Seite 3375)                                    | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| LineareSkalierung (Seite 3377)                               | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| LöscheMeldepuffer (Seite 3378)                               | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| LöscheMeldepufferProTool<br>(Seite 3379)                     | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| MeldeanzeigeBearbeiteMeldung<br>(Seite 3380)                 | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| MeldeanzeigeQuittiereMeldung<br>(Seite 3381)                 | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| MeldeanzeigeZeigeHilfetext<br>(Seite 3381)                   | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| QuittiereMeldung (Seite 3382)                                | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeDatensatzNeu<br>(Seite 3383)                  | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeLeseDatensatzAusS<br>teuerung (Seite 3383)    | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeLöscheDatensatz<br>(Seite 3384)               | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeMenü (Seite 3384)                             | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeÖffnen (Seite 3385)                           | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeSchreibeDatensatzI<br>nSteuerung (Seite 3385) | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeSpeichereDatensatz<br>(Seite 3386)            | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeSpeichernUnterDate<br>nsatz (Seite 3386)      | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeUmbenennenDaten<br>satz (Seite 3387)          | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeZeigeHilfetext<br>(Seite 3387)                | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RezepturanzeigeZurück (Seite 3388)                           | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RücksetzeBit (Seite 3388)                                    | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| RücksetzeBitInVariable (Seite 3389)                          | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| SchaltflächeDrücken (Seite 3391)                             | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| SchaltflächeLoslassen (Seite 3391)                           | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| SchiebenUndMaskieren (Seite 3392)                            | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| SeiteAbwärts (Seite 3394)                                    | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| SeiteAufwärts (Seite 3394)                                   | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| SetzeBetriebsartGerät (Seite 3395)                           | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |

|  | KP300<br>Basic<br>PN | KTP400<br>Basic PN | KTP 600<br>Basic DP | KTP 600<br>Basic PN | KTP<br>1000<br>Basic DP | KTP<br>1000<br>Basic PN | TP 1500<br>Basic<br>PN |
|--|----------------------|--------------------|---------------------|---------------------|-------------------------|-------------------------|------------------------|
| SetzeBit (Seite 3396)                            | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| SetzeBitInVariable (Seite 3397)                  | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| SetzeBitWährendTasteGedrückt<br>(Seite 3398)     | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| SetzeFarbeHintergrundbeleuchtung<br>(Seite 3400) | ja                   | nein               | nein                | nein                | nein                    | nein                    | nein                   |
| SetzeSprache (Seite 3402)                        | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| SetzeVariable (Seite 3403)                       | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| SetzeVerbindungsmodus<br>(Seite 3404)            | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| SimuliereSystemtaste (Seite 3406)                | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| SimuliereVariable (Seite 3407)                   | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| StoppeRuntime (Seite 3408)                       | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| VerfolgeBenutzeränderung<br>(Seite 3409)         | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| VerringereFokussiertenWert<br>(Seite 3409)       | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | nein                   |
| VerringereVariable (Seite 3410)                  | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| WechseleVerbindung (Seite 3411)                  | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| ZeigeAnmeldedialog (Seite 3412)                  | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| ZeigeHilfetext (Seite 3413)                      | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |
| ZeigeMeldefenster (Seite 3414)                   | ja                   | ja                 | ja                  | ja                  | ja                      | ja                      | ja                     |

<sup>1)</sup> Nur für KTP600 Basic mono PN

## Systemfunktionen

### Abmelden

### Beschreibung

Meldet den aktuellen Benutzer am Bediengerät ab.

### Verwendung in der Funktionsliste

Abmelden

### Verwendung in benutzerdefinierten Funktionen

Logoff

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

--

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

Anmelden (Seite 3358)

### ÄndereKontrast

### Beschreibung

Ändert den Kontrast des Displays am Bediengerät um eine Stufe.

### Verwendung in der Funktionsliste

ÄndereKontrast (Änderung)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Änderung

Legt fest, wie der Kontrast geändert wird:

0 (hmiDecrease) = Verringern: Verringert den Kontrast um eine Stufe.

1 (hmiIncrease) = Erhöhen: Erhöht den Kontrast um eine Stufe.

### Anwendungsbeispiel

#### Ziel

Sie wollen über jeweils eine Schaltfläche den Kontrast des Bildschirms erhöhen und verringern.

#### Hinweise zur Projektierung

Projektieren Sie zwei Schaltflächen und projektieren Sie jeweils an das Ereignis "Drücken" die Systemfunktion "ÄndereKontrast". Vergeben Sie als Parameter einmal "Verringern" und einmal "Erhöhen".

### Ablauf am Bediengerät

Wenn Sie in Runtime eine der beiden Schaltflächen drücken, wird der Kontrast um eine Stufe erhöht oder verringert.

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

Abmelden (Seite 3352)

### AktiviereBild

#### Beschreibung

Führt einen Bildwechsel zu dem angegebenen Bild aus.

Um vom Grundbild in das Permanentfenster oder umgekehrt zu wechseln, verwenden Sie die Systemfunktion "AktiviereBildMitNummer".

#### Verwendung in der Funktionsliste

AktiviereBild (Bildname, Objektnummer)

#### Verwendung in benutzerdefinierten Funktionen

ActivateScreen (Screen\_name, Object\_number)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

#### Parameter

##### Bildname

Name des Bildes, zu dem gewechselt wird.

##### Objektnummer

Das Bedienelement, das nach dem Bildwechsel im angegebenen Bild den Fokus erhält. Die Nummer des Bedienelements legen Sie während der Projektierung durch die Tab-Reihenfolge fest.

Wenn Sie "0" angeben:



- Ist der Fokus beim Aufruf der Systemfunktion im Permanentfenster, behält das Permanentfenster den Fokus.
- Ist der Fokus beim Aufruf der Systemfunktion im Grundbild, erhält das erste Bedienelement im angegebenen Bild den Fokus.

---

**Hinweis**

Wenn die Systemfunktion "AktiviereBild" an das Ereignis "Rand erreicht" projiziert wird, so ist für den Parameter "Objektnummer" nur der Wert 0 zulässig. Das aktive Objekt wird nicht anhand der Objektnummer festgelegt, sondern anhand der X-Position des aktiven Objekts vor dem Bildwechsel.

---

**Beispiel**

Der folgende Programmcode aktiviert mit der Funktion ActivateScreen beim Klicken auf eine Taste das Bild "Screen\_2".

```
{  
  
// User defined code  
// i.e. when pressing a button  
ActivateScreen ("Screen_2", 0);  
...  
}
```

**Siehe auch**

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

AktiviereBildMitNummer (Seite 3355)

**AktiviereBildMitNummer****Beschreibung**

Führt abhängig von einem Variablenwert einen Bildwechsel zu einem Bild aus.

Das Bild wird über seine Bildnummer identifiziert.

**Verwendung in der Funktionsliste**

AktiviereBildMitNummer (Bildnummer, Objektnummer)

**Verwendung in benutzerdefinierten Funktionen**

ActivateScreenByNumber (Screen\_number, Object\_number)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

### Bildnummer

Variable, welche die Bildnummer enthält, zu dem gewechselt wird.

Wenn Sie vom Grundbild ins Permanentfenster oder umgekehrt wechseln wollen, geben Sie "0" oder "-1" an:

0 = Wechselt vom Grundbild ins Permanentfenster.

-1 = Wechselt vom Permanentfenster ins Grundbild

### Objektnummer

Nummer des Bildobjekts, das nach dem Bildwechsel im angegebenen Bild den Fokus erhält. Die Nummer des Bedienelements legen Sie während der Projektierung durch die Tab-Reihenfolge fest.

Wenn Sie "0" angeben:

- Ist der Fokus beim Aufruf der Systemfunktion im Permanentfenster, behält das Permanentfenster den Fokus.
- Ist der Fokus beim Aufruf der Systemfunktion im Grundbild, erhält das erste Bedienelement im angegebenen Bild den Fokus.

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

AktiviereBild (Seite 3354)

## AktivierePutzbild

### Beschreibung

Aktiviert das Putzbild am Bediengerät. Dabei wird das Display des Bediengeräts für den angegebenen Zeitraum deaktiviert.

Während das Display des Bediengeräts deaktiviert ist, können Sie es reinigen, ohne unbeabsichtigt Touch-Funktionen auszulösen.

### Verwendung in der Funktionsliste

AktivierePutzbild (Zeitraum)

### Verwendung in benutzerdefinierten Funktionen

--

## Parameter

### Zeitraum

Zeitraum, in dem das Display deaktiviert ist. Die verbleibende Zeit wird als Fortschrittsbalken angezeigt.

Wertebereich in Sekunden von 10 bis 300.

---

### Hinweis

Die Systemfunktion AktivierePutzbild kann nicht simuliert werden.

---

## Siehe auch

Abmelden (Seite 3352)

## AktiviereVorherigesBild

### Beschreibung

Führt einen Bildwechsel zu dem Bild aus, das vor dem aktuellen Bild aktiviert war. Der Bildwechsel wird nicht ausgeführt, wenn zuvor kein Bild aktiviert war.

Die 10 zuletzt aufgerufenen Bilder werden gespeichert. Wenn Sie zu einem Bild wechseln, das nicht mehr gespeichert ist, wird eine Systemmeldung ausgegeben.

---

### Hinweis

Wenn Sie die Systemfunktion verwenden wollen, muss das Bild zu dem sie wechseln in der Navigationsstruktur verwendet werden.

---

## Verwendung in der Funktionsliste

AktiviereVorherigesBild

## Verwendung in benutzerdefinierten Funktionen

ActivatePreviousScreen

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

--

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## AktualisiereVariable

### Beschreibung

Liest für die Variable mit der angegebenen Aktualisierungskennung den aktuellen Wert aus der Steuerung.

### Verwendung in der Funktionsliste

AktualisiereVariable (Aktualisierungskennung)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Aktualisierungskennung**

Aktualisierungskennung, die der zu aktualisierenden Variablen zugeordnet ist.

## Siehe auch

Abmelden (Seite 3352)

## Anmelden

### Beschreibung

Meldet den aktuellen Benutzer am Bediengerät an.

### Verwendung in der Funktionsliste

Anmelden (Kennwort, Benutzername)

### Verwendung in benutzerdefinierten Funktionen

Logon (Password, User\_name)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

### **Kennwort**

Die Variable, aus der das Kennwort des anzumeldenden Benutzers gelesen wird.  
Wenn der Benutzer angemeldet ist, wird das Kennwort in der Variablen gelöscht.

### **Benutzername**

Die Variable, aus welcher der Benutzername des anzumeldenden Benutzers gelesen wird.

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)  
Abmelden (Seite 3352)

## BearbeiteMeldung

### Beschreibung

Löst das Ereignis "Bearbeiten" für alle gewählten Meldungen aus.  
Wenn Sie die zu bearbeitenden Meldungen noch nicht quittiert haben, erfolgt mit Aufruf dieser Systemfunktion automatisch die Quittierung.  
Diese Systemfunktion kann nur für Funktionstasten verwendet werden.

### Verwendung in der Funktionsliste

BearbeiteMeldung

### Verwendung in benutzerdefinierten Funktionen

EditAlarm

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

--

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## BildobjektCursorAufwärts

### Beschreibung

Führt im angegebenen Bildobjekt zu einer zeilenweisen Cursorbewegung nach oben.

Diese Systemfunktion ist für folgende Bildobjekte verwendbar:

- Benutzeranzeige
- Meldeanzeige
- Rezepturanzeige

### Verwendung in der Funktionsliste

BildobjektCursorAufwärts (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name des Bildobjekts, in dem die Tastenfunktion ausgelöst wird.

## BildobjektCursorAbwärts

### Beschreibung

Führt im angegebenen Bildobjekt zu einer zeilenweisen Cursorbewegung nach unten.

Diese Systemfunktion ist für folgende Bildobjekte verwendbar:

- Benutzeranzeige
- Meldeanzeige
- Rezepturanzeige

### Verwendung in der Funktionsliste

BildobjektCursorAbwärts (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

## Parameter

### **Bildobjekt**

Name des Bildobjekts, in dem die Tastenfunktion ausgelöst wird.

## **BildobjektSeiteAufwärts**

### **Beschreibung**

Führt im angegebenen Bildobjekt zu einer seitenweisen Cursorbewegung nach oben.

Diese Systemfunktion ist für folgende Bildobjekte verwendbar:

- Benutzeranzeige
- Meldeanzeige
- Rezepturanzeige

### **Verwendung in der Funktionsliste**

BildobjektSeiteAufwärts (Bildobjekt)

### **Verwendung in benutzerdefinierten Funktionen**

-

## Parameter

### **Bildobjekt**

Name des Bildobjekts, in dem die Tastenfunktion ausgelöst wird.

## **BildobjektSeiteAbwärts**

### **Beschreibung**

Führt im angegebenen Bildobjekt zu einer seitenweisen Cursorbewegung nach unten.

Diese Systemfunktion ist für folgende Bildobjekte verwendbar:

- Benutzeranzeige
- Meldeanzeige
- Rezepturanzeige

### **Verwendung in der Funktionsliste**

BildobjektSeiteAbwärts (Bildobjekt)

## Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name des Bildobjekts, in dem die Tastenfunktion ausgelöst wird.

## ErhöheFokussiertenWert

### Beschreibung

Addiert den angegebenen Wert zum Wert der Variablen, die mit dem Eingabefeld (symbolisches Auswahlfeld, grafisches Auswahlfeld, Schieber) verbunden ist, das aktuell den Fokus besitzt.

Diese Systemfunktion kann nur an Funktionstasten verwendet werden.

### Verwendung in der Funktionsliste

ErhöheFokussiertenWert (Wert)

## Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Wert**

Der Wert, der zum Variablenwert addiert wird.

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## ErhöheVariable

### Beschreibung

Addiert den angegebenen Wert zum Wert der Variablen.



$X = X + a$

---

#### Hinweis

Die Systemfunktion verwendet als Eingangswert und Ausgangswert dieselbe Variable. Wenn Sie diese Systemfunktion zum Umrechnen eines Werts verwenden, müssen Sie mit einer Hilfsvariablen arbeiten. Der Hilfsvariablen können Sie mit der Systemfunktion "SetzeVariable" den Variablenwert zuweisen.

---

Wenn Sie die Systemfunktion an Ereignisse einer Meldung projektieren und die Variable wird im aktuellen Bild nicht verwendet, ist nicht sichergestellt, dass der tatsächliche Wert der Variablen in der Steuerung verwendet wird. Dies kann durch Setzen der Erfassungsart "Zyklisch fortlaufend" verbessert werden.

#### Verwendung in der Funktionsliste

ErhöheVariable (Variable, Wert)

#### Verwendung in benutzerdefinierten Funktionen

IncreaseTag (Tag, Value)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

#### Parameter

##### Variable

Die Variable, zu welcher der angegebene Wert addiert wird.

##### Wert

Der Wert, der addiert wird.

#### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

SetzeVariable (Seite 3403)

#### GeheZumAnfang

#### Beschreibung

Führt die Tastenfunktion <Pos1> am Bediengerät aus.

Verwenden Sie diese Systemfunktion, wenn das Bediengerät diese Funktionalität nicht standardmäßig besitzt.

Diese Systemfunktion ist nur für Funktionstasten verwendbar.

### Verwendung in der Funktionsliste

GeheZumAnfang

### Verwendung in benutzerdefinierten Funktionen

GoToHome

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

--

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

### GeheZumEnde

### Beschreibung

Führt die Tastenfunktion <Ende> am Bediengerät aus.

Verwenden Sie diese Systemfunktion, wenn das Bediengerät diese Funktionalität nicht standardmäßig besitzt.

Diese Systemfunktion ist nur für Funktionstasten verwendbar.

### Verwendung in der Funktionsliste

GeheZumEnde

### Verwendung in benutzerdefinierten Funktionen

GoToEnd

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

--

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## InvertiereBit

### Beschreibung

Invertiert den Wert der angegebenen Variable vom Typ "Bool":

- Hat die Variable den Wert 1 (TRUE), wird sie auf 0 (FALSE) gesetzt.
- Hat die Variable den Wert 0 (FALSE), wird sie auf 1 (TRUE) gesetzt.

### Verwendung in der Funktionsliste

InvertiereBit (Variable)

### Verwendung in benutzerdefinierten Funktionen

InvertBit (Tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Variable

Die Variable, dessen Bit gesetzt wird.

### Beispiel

Der folgende Programmcode invertiert den Wert der booleschen Variable b\_value und gibt das Ergebnis zusammen mit dem ursprünglichen Wert b\_saved aus.

```
{  
  BOOL b_value = 0;  
  BOOL b_saved = b_value;  
  
  //Invert variable  
  invertBit(b_value);  
  
  //print current and saved value  
  printf ("Current value: %d\r\n, Saved value: %d\r\n",b_value, b_saved);  
  ...  
}
```

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

InvertiereBitInVariable (Seite 3366)

## InvertiereBitInVariable

### Beschreibung

Invertiert ein Bit in der angegebenen Variablen:

- Hat das Bit in der Variablen den Wert 1 (TRUE), wird es auf 0 (FALSE) gesetzt.
- Hat das Bit in der Variablen den Wert 0 (FALSE), wird es auf 1 (TRUE) gesetzt.

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

---

#### Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie statt dessen die Systemfunktion "InvertiereBit".

---

### Verwendung in der Funktionsliste

InvertiereBitInVariable (Variable, Bit)

### Verwendung in benutzerdefinierten Funktionen

InvertBitInTag (Tag, Bit)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Tag

Die Variable, in der das angegebene Bit gesetzt wird.

#### Bit

Die Nummer des Bits, das gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in einer Variablen von rechts nach links gezählt. Die Zählung beginnt mit 0.

### Beispiel

Der folgende Programmcode invertiert ein Bit an der angegebenen Position bitposition in der Variable bvalue und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bsaved aus.

```
{
```

```
BYTE bvalue;  
BYTE bsaved = bvalue;  
BYTE bitposition = 2;  
  
//Invert bit in bitposition  
InvertBitInTag (bvalue, bitposition);  
//print current and saved value  
printf ("Current value: %d\r\n, Saved value: %d\r\n",bvalue, bsaved);  
    ...  
}
```

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

InvertiereBit (Seite 3365)

## InvertiereLineareSkalierung

### Beschreibung

Weist der Variablen X einen Wert zu, der aus dem Wert der angegebenen Variablen Y über die lineare Funktion  $X = (Y - b) / a$  berechnet wird.

Die Variablen X und Y dürfen nicht identisch sein. Diese Systemfunktion ist die Umkehrfunktion zur Systemfunktion "LineareSkalierung".

---

#### Hinweis

Die Variablen X und Y dürfen nicht identisch sein. Wenn Sie eine Variable auf sich selbst umrechnen wollen, müssen Sie eine Hilfsvariable verwenden.

Mit der Systemfunktion "SetzeVariable" können Sie den Wert der umzurechnenden Variablen der Hilfsvariablen zuweisen.

---

### Verwendung in der Funktionsliste

InvertiereLineareSkalierung (X, Y, b, a)

### Verwendung in benutzerdefinierten Funktionen

InverseLinearScaling (X, Y, b, a)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

**X**

Die Variable, welcher der berechnete Wert aus der linearen Gleichung zugewiesen wird.

**Y**

Die Variable, deren Wert zur Berechnung verwendet wird.

**b**

Der Wert, der subtrahiert wird.

**a**

Der Wert, durch den dividiert wird.

## Beispiel

Der folgende Programmcode weist der Variable varX einen Wert mit der Funktion InverseLinearScaling zu.

```
{
BYTE varX;
BYTE Yvalue = 10;
BYTE bvalue = 3;
BYTE avalue = 4;

//Inverse linear scaling
InverseLinearScaling (varX, Yvalue, bvalue, avalue);

printf ("varX = %d\r\n", varX);
...
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

LineareSkalierung (Seite 3377)

## KalibriereTouchscreen

### Beschreibung

Ruft ein Programm zur Kalibrierung des Touch-Displays auf.

Während des Kalibriervorgangs werden Sie aufgefordert, fünf Positionen auf dem Touch-Display zu berühren. Um den Kalibriervorgang zu bestätigen, berühren Sie innerhalb von 30 Sekunden das Touch-Display. Wenn Sie das Touch-Display innerhalb der Zeitspanne nicht

berühren, werden die Kalibriereinstellungen verworfen. Die Bedienerführung ist in englischer Sprache.

Verwenden Sie diese Systemfunktion, wenn Sie das Bediengerät zum ersten Mal in Betrieb nehmen.

### Verwendung in der Funktionsliste

KalibriereTouchscreen

### Verwendung in benutzerdefinierten Funktionen

CalibrateTouchScreen

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

--

---

#### Hinweis

Die Systemfunktion KalibriereTouchscreen kann nicht simuliert werden.

---

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

### KurvenanzeigeBlättereVorwärts

#### Beschreibung

Blättert in der Kurvenanzeige um eine Anzeigenbreite vorwärts.

### Verwendung in der Funktionsliste

KurvenanzeigeBlättereVorwärts (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

## Parameter

### **Bildobjekt**

Name der Kurvenanzeige, in welcher um eine Anzeigenbreite vorwärts geblättert wird.

## KurvenanzeigeBlättereZurück

### Beschreibung

Blättert in der Kurvenanzeige um eine Anzeigenbreite rückwärts.

### Verwendung in der Funktionsliste

KurvenanzeigeBlättereZurück (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

## Parameter

### **Bildobjekt**

Name der Kurvenanzeige, in der um eine Anzeigenbreite rückwärts geblättert wird.

## KurvenanzeigeErweitern

### Beschreibung

Verkleinert den Zeitabschnitt, der in der Kurvenanzeige dargestellt wird.

### Verwendung in der Funktionsliste

KurvenanzeigeErweitern (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

## Parameter

### **Bildobjekt**

Name der Kurvenanzeige, in welcher der dargestellte Zeitabschnitt verkleinert wird.



## KurvenanzeigeKomprimieren

### Beschreibung

Vergrößert den Zeitabschnitt, der in der Kurvenanzeige dargestellt wird.

### Verwendung in der Funktionsliste

KurvenanzeigeKomprimieren (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Bildobjekt

Name der Kurvenanzeige, in welcher der dargestellte Zeitabschnitt vergrößert wird.

## KurvenanzeigeLinealLinks

### Beschreibung

Bewegt in der Kurvenanzeige die Leselinie rückwärts (nach links).

---

#### Hinweis

Damit Sie die Leselinie bewegen können, müssen Sie die Leselinie eingeschaltet haben. Verwenden Sie dazu die Systemfunktion "KurvenanzeigeSetzeLinealmodus".

---

### Verwendung in der Funktionsliste

KurvenanzeigeLinealLinks (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Bildobjekt

Name der Kurvenanzeige, in der die Leselinie rückwärts bewegt wird.

## Siehe auch

KurvenanzeigeSetzeLinealmodus (Seite 3372)

## KurvenanzeigeLinealRechts

### Beschreibung

Bewegt in der Kurvenanzeige die Leselinie vorwärts (nach rechts).

---

#### Hinweis

Damit Sie die Leselinie bewegen können, müssen Sie die Leselinie eingeschaltet haben. Verwenden Sie dazu die Systemfunktion "KurvenanzeigeSetzeLinealmodus".

---

### Verwendung in der Funktionsliste

KurvenanzeigeLinealRechts (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Bildobjekt

Name der Kurvenanzeige, in der die Leselinie vorwärts bewegt wird.

## Siehe auch

KurvenanzeigeSetzeLinealmodus (Seite 3372)

## KurvenanzeigeSetzeLinealmodus

### Beschreibung

Blendet in der Kurvenanzeige die Leselinie ein oder aus. Die Leselinie zeigt den zu einem X-Wert gehörigen Y-Wert an.

---

#### Hinweis

Damit das Lineal eingeblendet wird, müssen Sie in den Eigenschaften der Kurvenanzeige die Einstellung "Lineal anzeigen" aktivieren.

---

### Verwendung in der Funktionsliste

KurvenanzeigeSetzeLinealmodus (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name der Kurvenanzeige, in der die Leselinie eingeblendet oder ausgeblendet wird.

### Siehe auch

KurvenanzeigeLinealLinks (Seite 3371)

KurvenanzeigeLinealRechts (Seite 3372)

### KurvenanzeigeStartenStoppen

### Beschreibung

Hält in der Kurvenanzeige die Kurvenaufzeichnung an oder setzt die Kurvenaufzeichnung fort.

### Verwendung in der Funktionsliste

KurvenanzeigeStartenStoppen (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name der Kurvenanzeige, in der die Kurvenaufzeichnung gestartet oder gestoppt wird.

### KurvenanzeigeZurückZumAnfang

### Beschreibung

Blättert in der Kurvenanzeige zum Beginn des Darstellungsbereichs zurück.

### Verwendung in der Funktionsliste

KurvenanzeigeZurückZumAnfang (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name der Kurvenanzeige, in der zum Beginn des Darstellungsbereichs geblättert wird.

### LeseBenutzername

### Beschreibung

Schreibt den Benutzernamen des gerade am Bediengerät angemeldeten Benutzers in die angegebene Variable.

Wenn die angegebene Variable eine Steuerungsanbindung besitzt, steht der Benutzername auch in der Steuerung zur Verfügung. Mit dieser Systemfunktion können Sie z.B. eine benutzerabhängige Freischaltung bestimmter Funktionalitäten realisieren.

### Verwendung in der Funktionsliste

LeseBenutzername (Variable)

### Verwendung in benutzerdefinierten Funktionen

GetUserName (Tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### **Variable**

Die Variable, in die der Benutzername geschrieben wird.

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## LeseGruppennummer

### Beschreibung

Liest die Nummer der Gruppe, welcher der am Bediengerät angemeldete Benutzer angehört, und schreibt sie in die angegebene Variable.

### Verwendung in der Funktionsliste

LeseGruppennummer (Variable)

### Verwendung in benutzerdefinierten Funktionen

GetGroupNumber (Tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Variable

Die Variable, in welche die Nummer der Gruppe geschrieben wird.

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## LeseKennwort

### Beschreibung

Schreibt das Kennwort des gerade am Bediengerät angemeldeten Benutzers in die angegebene Variable.

---

#### Hinweis

Stellen Sie sicher, dass der Wert der angegebenen Variable nicht an anderer Stelle im Projekt angezeigt wird.

---

#### Hinweis

Kennwörter von SIMATIC Logon-Benutzern können nicht ausgelesen werden.

---

### Verwendung in der Funktionsliste

LeseKennwort (Variable)

## Verwendung in benutzerdefinierten Funktionen

GetPassword (Tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

### Variable

Die Variable, in die das Kennwort geschrieben wird.

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## LesePLCModus

## Beschreibung

Wertet den aktuellen Zustand der verbundenen Steuerung aus.

Sie Systemfunktion "LesePLCModus" ist nur für folgende Steuerungen projektierbar:

- SIMATIC S7 1200
- SIMATIC S7 1500

## Verwendung in der Funktionsliste

LesePLCModus (Verbindung, Modus)

## Verwendung in benutzerdefinierten Funktionen

GetPLCMode (Connection, Mode)

## Parameter

### Verbindung

Verbindung von Steuerung und Bediengerät.

### Modus

Wertet den Zustand der verbundenen Steuerung aus.

Wählen Sie für die Auswertung die Variable "@DiagnosticsIndicatorTag" aus.

## LineareSkalierung

### Beschreibung

Weist der Variablen Y einen Wert zu, der aus dem Wert der angegebenen Variablen X über die lineare Funktion  $Y = (a * X) + b$  berechnet wird.

Die Umkehrfunktion ist die Systemfunktion "InvertiereLineareSkalierung".

---

#### Hinweis

Die Variablen X und Y dürfen nicht identisch sein. Wenn Sie eine Variable auf sich selbst umrechnen wollen, müssen Sie eine Hilfsvariable verwenden.

Mit der Systemfunktion "SetzeVariable" können Sie den Wert der umzurechnenden Variablen der Hilfsvariablen zuweisen.

---

### Verwendung in der Funktionsliste

LineareSkalierung (Y, a, X, b)

### Verwendung in benutzerdefinierten Funktionen

LinearScaling (Y, a, X, b)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Y

Die Variable, welcher der berechnete Wert aus der linearen Gleichung zugewiesen wird.

#### a

Der Wert, mit dem multipliziert wird.

#### X

Die Variable, deren Wert zur Berechnung verwendet wird.

#### b

Der Wert, der addiert wird.

### Beispiel

Der folgende Programmcode weist mit der Funktion LinearScaling der Variable Yvar einen Wert zu.

```
{  
BYTE Yvar;  
BYTE Xvalue = 10;  
BYTE bvalue = 3;  
BYTE avalue = 4;  
  
// linear scaling  
LinearScaling ( Yvar, avalue, Xvalue, bvalue);  
  
printf ("Yvar = %d\r\n, Yvar);  
...  
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

InvertiereLineareSkalierung (Seite 3367)

### LöscheMeldepuffer

#### Beschreibung

Löscht Meldungen aus dem Meldepuffer am Bediengerät.

---

#### Hinweis

Noch nicht quittierte Meldungen werden ebenfalls gelöscht.

---

#### Verwendung in der Funktionsliste

LöscheMeldepuffer (Nummer der Meldeklasse)

#### Verwendung in benutzerdefinierten Funktionen

ClearAlarmBuffer (Alarm\_class\_number)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

#### Parameter

##### Nummer der Meldeklasse

Legt fest, welche Meldungen aus dem Meldepuffer gelöscht werden:

0 (hmiAll) = Alle Meldungen



- 1 (hmiAlarms) = Meldungen der Meldeklasse "Errors"
- 2 (hmiEvents) = Meldungen der Meldeklasse "Warnings"
- 3 (hmiSystem) = Meldungen der Meldeklasse "System"
- 4 (hmiS7Diagnosis) = Meldungen der Meldeklasse "Diagnosis Events"

---

**Hinweis**

**Geräteabhängigkeit**

Meldungen der Meldeklasse "Diagnosis Events" stehen auf Basic Panels nicht zur Verfügung.

---

**Siehe auch**

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

**LöscheMeldepufferProTool**

**Beschreibung**

Diese Systemfunktion existiert aus Kompatibilitätsgründen.

Sie hat die gleiche Funktionalität wie die Systemfunktion "LöscheMeldepuffer", verwendet aber die alte ProTool-Nummerierung.

**Verwendung in der Funktionsliste**

LöscheMeldepufferProTool (Nummer der Meldeklasse)

**Verwendung in benutzerdefinierten Funktionen**

ClearAlarmBufferProtoolLegacy (Alarm\_class\_number)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

**Parameter**

**Nummer der Meldeklasse**

Nummer der Meldeklasse, deren Meldungen gelöscht werden sollen:

- 1 (hmiAllProtoolLegacy) = Alle Meldungen
- 0 (hmiAlarmsProtoolLegacy) = Meldungen der Meldeklasse "Errors"
- 1 (hmiEventsProtoolLegacy) = Meldungen der Meldeklasse "Warnings"
- 2 (hmiSystemProtoolLegacy) = Meldungen der Meldeklasse "System"

3 (hmiS7DiagnosisProtoolLegacy) = Meldungen der Meldeklasse "Diagnosis Events"

---

**Hinweis**

**Geräteabhängigkeit**

Meldungen der Meldeklasse "Diagnosis Events" stehen auf Basic Panels nicht zur Verfügung.

---

**Siehe auch**

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

LöscheMeldepuffer (Seite 3378)

**MeldeanzeigeBearbeiteMeldung**

**Beschreibung**

Löst das Ereignis "Bearbeiten" für alle Meldungen aus, die in der angegebenen Meldeanzeige ausgewählt sind.

Verwenden Sie diese Systemfunktion, wenn die integrierte Schaltfläche des ActiveX-Controls nicht benutzt werden soll.

An das Ereignis "Bearbeiten" können Sie wiederum eine Systemfunktion projektieren. Beispielsweise können Sie zu dem Prozessbild wechseln, in dem die Meldung aufgetreten ist.

---

**Hinweis**

Wenn Sie die zu bearbeitenden Meldungen noch nicht quittiert haben, erfolgt mit Aufruf dieser Systemfunktion automatisch die Quittierung.

---

**Verwendung in der Funktionsliste**

MeldeanzeigeBearbeiteMeldung (Bildobjekt)

**Verwendung in benutzerdefinierten Funktionen**

-

## Parameter

### **Bildobjekt**

Name der Meldeanzeige, in der das Ereignis ausgelöst wird.

---

### **Hinweis**

Folgende Bediengeräte unterstützen diese Systemfunktion nicht für das Objekt "Bild": OP 73, OP 77A, TP 177A.

---

## MeldeanzeigeQuittiereMeldung

### **Beschreibung**

Quittiert die Meldungen, die in der angegebenen Meldeanzeige ausgewählt sind.

Verwenden Sie diese Systemfunktion, wenn die integrierte Schaltfläche des ActiveX-Controls nicht benutzt werden soll.

### **Verwendung in der Funktionsliste**

MeldeanzeigeQuittiereMeldung (Bildobjekt)

### **Verwendung in benutzerdefinierten Funktionen**

-

## Parameter

### **Bildobjekt**

Name der Meldeanzeige, in der das Ereignis ausgelöst wird.

---

### **Hinweis**

Folgende Bediengeräte unterstützen diese Systemfunktion nicht für das Objekt "Bild": OP 73, OP 77A, TP 177A.

---

## MeldeanzeigeZeigeHilfetext

### **Beschreibung**

Zeigt den projizierten Tooltip der Meldung an, die in der angegebenen Meldeanzeige ausgewählt ist.

### Verwendung in der Funktionsliste

MeldeanzeigeZeigeHilfetext (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name der Meldeanzeige, in der das Ereignis ausgelöst wird.

---

#### **Hinweis**

Folgende Bediengeräte unterstützen diese Systemfunktion nicht für das Objekt "Bild": OP 73, OP 77A, TP 177A.

---

### QuittiereMeldung

#### Beschreibung

Quittiert alle ausgewählten Meldungen.

Verwenden Sie diese Systemfunktion, wenn das Bediengerät keine ACK-Taste besitzt oder die integrierte Taste der Meldeanzeige nicht genutzt werden soll.

Diese Systemfunktion kann nur für Funktionstasten verwendet werden.

### Verwendung in der Funktionsliste

QuittiereMeldung

### Verwendung in benutzerdefinierten Funktionen

AcknowledgeAlarm

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

--

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## RezepturanzeigeDatensatzNeu

### Beschreibung

Erstellt in der angegebenen Rezepturanzeige einen neuen Datensatz.

### Verwendung in der Funktionsliste

RezepturanzeigeDatensatzNeu (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Bildobjekt

Name der Rezepturanzeige, in welcher der neue Rezepturdatensatz angelegt wird.

## RezepturanzeigeLeseDatensatzAusSteuerung

### Beschreibung

Überträgt den Datensatz, der gerade in der Steuerung geladen ist, an das Bediengerät und zeigt ihn in der Rezepturanzeige an.

### Verwendung in der Funktionsliste

RezepturanzeigeLeseDatensatzAusSteuerung (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Bildobjekt

Name der Rezepturanzeige, in welcher der aus der Steuerung übertragene Rezepturdatensatz angezeigt wird.

## RezepturanzeigeLöscheDatensatz

### Beschreibung

Löscht den Datensatz, der gerade in der Rezepturanzeige angezeigt wird.

### Verwendung in der Funktionsliste

RezepturanzeigeLöscheDatensatz (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name der Rezepturanzeige, in welcher der angezeigte Rezepturdatsatz gelöscht wird.

## RezepturanzeigeMenü

### Beschreibung

Öffnet das Menü der angegebenen einfachen Rezepturanzeige.

Verwenden Sie diese Systemfunktion nur bei einer einfachen Rezepturanzeige.

### Verwendung in der Funktionsliste

RezepturanzeigeMenü (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name der Rezepturanzeige, in der das Menü geöffnet werden soll.

## RezepturanzeigeÖffnen

### Beschreibung

Zeigt in der angegebenen Rezepturanzeige die Datensatzwerte an oder wechselt zum nächsten Auswahlfeld. Die Systemfunktion hat keine Wirkung, wenn am Bediengerät das Auswahlfeld für die Rezepturdatensatzwerte angezeigt wird.

Bedienreihenfolge der Auswahlfelder in Runtime:

- Rezepturname
- Datensatzname
- Rezepturdatensatzwerte

Verwenden Sie diese Systemfunktion, wenn Sie die einfache Rezepturanzeige projiziert haben. In der einfachen Rezepturanzeige wird jeweils nur ein Auswahlfeld am Bediengerät angezeigt. Verwenden Sie die Systemfunktion "RezepturanzeigeZurück", um das vorherige Auswahlfeld anzuzeigen.

### Verwendung in der Funktionsliste

RezepturanzeigeÖffnen (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### **Bildobjekt**

Name der Rezepturanzeige, in welcher der Befehl ausgelöst wird.

### Siehe auch

RezepturanzeigeZurück (Seite 3388)

## RezepturanzeigeSchreibeDatensatzInSteuerung

### Beschreibung

Überträgt den Rezepturdatensatz, der gerade in der Rezepturanzeige angezeigt wird, zur Steuerung.

### Verwendung in der Funktionsliste

RezepturanzeigeSchreibeDatensatzInSteuerung (Bildobjekt)

## Verwendung in benutzerdefinierten Funktionen

-

## Parameter

### Bildobjekt

Name der Rezepturanzeige, aus welcher der Rezepturdatensatz zur verbundenen Steuerung übertragen wird.

## RezepturanzeigeSpeichereDatensatz

## Beschreibung

Speichert den Rezepturdatensatz, der gerade in der Rezepturanzeige angezeigt wird.

## Verwendung in der Funktionsliste

RezepturanzeigeSpeichereDatensatz (Bildobjekt)

## Verwendung in benutzerdefinierten Funktionen

-

## Parameter

### Bildobjekt

Name der Rezepturanzeige, in welcher der Rezepturdatensatz gespeichert wird.

## RezepturanzeigeSpeichernUnterDatensatz

## Beschreibung

Speichert den Datensatz, der gerade in der Rezepturanzeige angezeigt wird, unter einem neuen Namen.

## Verwendung in der Funktionsliste

RezepturanzeigeSpeichernUnterDatensatz (Bildobjekt)

## Verwendung in benutzerdefinierten Funktionen

-



## Parameter

### **Bildobjekt**

Speichert den Datensatz, der in der Rezepturanzeige angezeigt wird, unter einem neuen Namen und/oder eine andere Nummer.

## RezepturanzeigeUmbenennenDatensatz

### Beschreibung

Benennt in der angegebenen Rezepturanzeige den ausgewählten Datensatz um.  
Verwenden Sie diese Systemfunktion nur bei einer einfachen Rezepturanzeige.

### Verwendung in der Funktionsliste

RezepturanzeigeUmbenennenDatensatz (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

## Parameter

### **Bildobjekt**

Name der Rezepturanzeige, in welcher der Rezepturdatsatz umbenannt wird.

## RezepturanzeigeZeigeHilfetext

### Beschreibung

Zeigt den projizierten Tooltip der angegebenen Rezepturanzeige an.

### Verwendung in der Funktionsliste

RezepturanzeigeZeigeHilfetext (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

## Parameter

### **Bildobjekt**

Name der Rezepturanzeige, deren projizierter Tooltip angezeigt wird.

## RezepturanzeigeZurück

### Beschreibung

Kehrt in der angegebenen Rezepturanzeige zum vorherigen Auswahlfeld zurück.

Die Systemfunktion hat keine Wirkung, wenn am Bediengerät das Auswahlfeld für die Rezeptur angezeigt wird. Bedienreihenfolge der Auswahlfelder in Runtime:

- Rezepturname
- Datensatzname
- Rezepturdatensatzwerte

Verwenden Sie diese Systemfunktion, wenn Sie die einfache Rezepturanzeige projiziert haben. In der einfachen Rezepturanzeige wird jeweils nur ein Auswahlfeld am Bediengerät angezeigt. Verwenden Sie die Systemfunktion "RezepturanzeigeÖffnen", um die Rezepturdatensatzwerte oder das nächste Auswahlfeld anzuzeigen.

### Verwendung in der Funktionsliste

RezepturanzeigeZurück (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Bildobjekt

Name der Rezepturanzeige, in welcher der Befehl ausgelöst wird.

## RücksetzeBit

### Beschreibung

Setzt den Wert einer Variablen vom Typ "Bool" auf 0 (FALSE).

### Verwendung in der Funktionsliste

RücksetzeBit (Variable)

### Verwendung in benutzerdefinierten Funktionen

ResetBit (Tag)

Verwendbar, sofern das projizierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

### Variable

Die Variable vom Typ BOOL, die auf 0 (FALSE) gesetzt wird.

## Beispiel

Der folgende Programmcode setzt den Wert der boolschen Variable b\_value mit der Funktion ResetBit auf 0 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert b\_saved aus.

```
{  
  BOOL b_value = 1;  
  BOOL b_saved = b_value;  
  
  //Reset bit  
  ResetBit (b_value);  
  
  //print current and saved value  
  printf ("Current value: %d\r\n, Saved value: %d\r\n",b_value, b_saved);  
  ...  
}
```

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)  
RücksetzeBitInVariable (Seite 3389)

## RücksetzeBitInVariable

### Beschreibung

Setzt ein Bit in der angegebenen Variablen auf 0 (FALSE).

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

---

### Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie stattdessen die Systemfunktion "RücksetzeBit".

---

## Verwendung in der Funktionsliste

RücksetzeBitInVariable (Variable, Bit)

## Verwendung in benutzerdefinierten Funktionen

ResetBitInTag (Tag, Bit)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

### Variable

Die Variable, in der ein Bit auf 0 (FALSE) gesetzt wird.

### Bit

Die Nummer des Bits, das auf 0 (FALSE) gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in der angegebenen Variablen unabhängig von der verwendeten Steuerung von rechts nach links gezählt. Die Zählung beginnt mit 0.

## Beispiel

Der folgende Programmcode setzt ein Bit an der angegebenen Position bitposition der Variablen bvalue auf 0 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert bsaved aus.

```
{  
BYTE bvalue;  
BYTE bsaved = bvalue;  
BYTE bitposition = 2;  
  
//Reset bit in bitposition  
ResetBitInTag (bvalue, bitposition);  
  
//print current and saved value  
printf ("Current value: %d\r\n, Saved value: %d\r\n",bvalue, bsaved);  
    ...  
}
```

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

RücksetzeBit (Seite 3388)

## SchaltflächeDrücken

### Beschreibung

Die Systemfunktion kann nur an die Funktionstasten eines Bediengeräts projiziert werden und löst am angegebenen Bildobjekt das Ereignis "Taste Drücken" aus.

Verwenden Sie diese Systemfunktion, wenn Sie z. B. eine Schaltfläche im Bild über eine Funktionstaste des Bediengeräts bedienen wollen.

---

### Hinweis

Sie müssen die Systemfunktionen "SchaltflächeDrücken" und "SchaltflächeLoslassen" immer gemeinsam projektieren. Wenn Sie also an das Ereignis "Taste Drücken" einer Funktionstaste die Systemfunktion "SchaltflächeDrücken" projektieren, dann projektieren Sie an das Ereignis "Loslassen" derselben Funktionstaste die Systemfunktion "SchaltflächeLoslassen".

---

### Verwendung in der Funktionsliste

SchaltflächeDrücken (Bildobjekt)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Bildobjekt

Name des Bildobjekts, an dem das Ereignis ausgelöst wird.

## SchaltflächeLoslassen

### Beschreibung

Die Systemfunktion kann nur an die Funktionstasten eines Bediengeräts projiziert werden und löst am angegebenen Bildobjekt das Ereignis "Taste Loslassen" aus.

Verwenden Sie diese Systemfunktion, wenn Sie z. B. eine Schaltfläche im Bild über eine Funktionstaste des Bediengeräts bedienen wollen.

---

**Hinweis**

Sie müssen die Systemfunktionen "SchaltflächeDrücken" und "SchaltflächeLoslassen" immer gemeinsam projektieren. Wenn Sie also an das Ereignis "Taste Drücken" einer Funktionstaste die Systemfunktion "SchaltflächeDrücken" projektieren, dann projektieren Sie an das Ereignis "Taste Loslassen" derselben Funktionstaste die Systemfunktion "SchaltflächeLoslassen".

---

**Verwendung in der Funktionsliste**

SchaltflächeLoslassen (Bildobjekt)

**Verwendung in benutzerdefinierten Funktionen**

-

**Parameter**

**Bildobjekt**

Name des Bildobjekts, an dem das Ereignis ausgelöst wird.

**SchiebenUndMaskieren**

**Beschreibung**

Die Systemfunktion wandelt das Eingangs-Bitmuster der Quellvariablen in ein Ausgangs-Bitmuster der Zielvariablen. Dabei werden Bits verschoben und maskiert.

---

**Hinweis**

Wenn Quellvariable und Zielvariable eine unterschiedliche Anzahl von Bits haben, kann die Anwendung der Systemfunktion in der Zielvariablen zu einer Wertebereichsüberschreitung führen.

---

**Verwendung in der Funktionsliste**

SchiebenUndMaskieren (Quellvariable, Zielvariable, Zu schiebende Bits, Zu maskierende Bits)

**Verwendung in benutzerdefinierten Funktionen**

ShiftAndMask (Source\_tag, Target\_tag, Bits\_to\_shift, Bits\_to\_mask)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

### Quellvariable

Die Variable enthält das Eingangs-Bitmuster, erlaubt sind Variablen von ganzzahligem Typ z. B. "Byte", "Char", "Int", "UInt", "Long" und "ULong".

Beispiel: Die Quellvariable vom Typ 16 Bit Integer hat den aktuellen Wert 72:  
000000001001000.

### Zielvariable

In die Variable wird das Ausgangs-Bitmuster gespeichert. Erlaubt sind Variablen von ganzzahligem Typ z. B. "Byte", "Char", "Int", "UInt", "Long" und "ULong".

Beispiel: Das verschobene Eingangs-Bitmuster wird mit der Bit-Maske multipliziert, Bit für Bit UND-verknüpft: 000000000001001. Das Ergebnis hat den Dezimalwert "8" und wird in der Zielvariablen gespeichert.

Achten Sie auf Folgendes:

- Quell- und Zielvariable haben die gleiche Anzahl Bits.
- Die Anzahl der zu schiebenden Bits ist kleiner als die Anzahl der Bits der Quell- und Zielvariablen.
- Zu maskierende Bits hat nicht mehr Bits als Quell- und Zielvariable.

### Zu schiebende Bits

Anzahl der Bits, um die das Eingangs-Bitmuster nach rechts verschoben wird. Ein negativer Wert verschiebt das Eingangs-Bitmuster nach links.

Beispiel: "Zu schiebende Bits" hat den Wert "+3". Dann wird beim Aufruf der Systemfunktion das Eingangs-Bitmuster um 3 Bits nach rechts verschoben: 000000000001001.

Links wird mit "0" aufgefüllt. Rechts werden 3 Bits abgeschnitten. Neuer Dezimalwert ist "9".

---

### Hinweis

Wenn die Quellvariable einen vorzeichenbehafteten Datentyp Integer besitzt mit dem Vorzeichen "-", ist das linke Bit "1". Beim Verschieben nach rechts wird dieses Vorzeichen-Bit mit "0" aufgefüllt. Das Vorzeichen wechselt nach "+".

---

### Zu maskierende Bits

Eine Integer-Zahl dient als Bit-Maske. Mit deren Bitmuster wird das verschobene Eingangs-Bitmuster multipliziert. Beispiel: Zahl "2478" mit dem Bitmuster "0000100110101110".

Die Bit-Maske geben Sie auf 3 verschiedene Arten ein:

- Hexadezimal: Geben Sie zuerst als Präfix "0h" oder "0H" ein, gefolgt von einem optionalen Leerzeichen zur besseren Lesbarkeit. Anschließend gruppieren Sie das Bitmuster in Viererblöcken (0000)(1001)(1010)(1110) und stellen jeden Block im 16er System dar: (0)(9)(A)(E). Bei der Eingabe sind nur die Zeichen 0-9, A-F, a-f erlaubt: "0h 09AE".
- Binär: Geben Sie zuerst als Präfix "0b" oder "0B" ein, gefolgt von einem optionalen Leerzeichen zur besseren Lesbarkeit. Anschließend gruppieren Sie das binäre Bitmuster zur Kontrolle in Viererblöcken 0000 1001 1010 1110 mit Leerzeichen dazwischen. Bei der Eingabe sind nur die Zeichen "0" oder "1" erlaubt: "0b 0000 1001 1010 1110".
- Dezimal: Geben Sie direkt ohne Präfix den Wert "2478" ein.

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

### SeiteAufwärts

### Beschreibung

Führt die Tastenfunktion <Bild Auf> am Bediengerät aus.

Diese Systemfunktion kann nur für Funktionstasten und für Aufgaben mit zeitlichen Trigger verwendet werden.

### Verwendung in der Funktionsliste

SeiteAufwärts

### Verwendung in benutzerdefinierten Funktionen

PageUp

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

-

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

### SeiteAbwärts

### Beschreibung

Führt die Tastenfunktion <Bild Ab> am Bediengerät aus.



Diese Systemfunktion kann nur für Funktionstasten verwendet werden.

### Verwendung in der Funktionsliste

SeiteAbwärts

### Verwendung in benutzerdefinierten Funktionen

PageDown

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

-

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

### SetzeBetriebsartGerät

### Beschreibung

Schaltet die Betriebsart am Bediengerät um. Folgende Betriebsarten sind möglich: "Online", "Offline" und "Laden".

### Verwendung in der Funktionsliste

SetzeBetriebsartGerät (Betriebsart)

### Verwendung in benutzerdefinierten Funktionen

SetDeviceMode (Operating\_mode)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### **Betriebsart**

Legt die Betriebsart des Bediengeräts fest:

0 (hmiOnline) = Online: Die Verbindung zur Steuerung wird hergestellt.

1 (hmiOffline) = Offline: Die Verbindung zur Steuerung wird getrennt.

2 (hmiTransfer) = Laden: Sie können ein Projekt vom Projektierungsrechner auf das Bediengerät übertragen.

---

**Hinweis**

Wenn Sie als Bediengerät einen PC verwenden, wird beim Umschalten der Betriebsart nach "Laden" die Runtime-Software beendet.

---

**Siehe auch**

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

SetzeVerbindungsmodus (Seite 3404)

**SetzeBit**

**Beschreibung**

Setzt den Wert einer Variablen vom Typ "Bool" auf 1 (TRUE).

**Verwendung in der Funktionsliste**

SetzeBit (Variable)

**Verwendung in benutzerdefinierten Funktionen**

SetBit (Tag)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

**Parameter**

**Variable**

Die Variable vom Typ BOOL, die auf 1 (TRUE) gesetzt wird.

**Beispiel**

Der folgende Programmcode setzt den Wert der boolschen Variable b\_value mit der Funktion SetBit auf 1 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert b\_saved aus.

```
{  
  BOOL b_value = 0;  
  BOOL b_saved = b_value;  
  
  //Set bit
```

```
SetBit (b_value);  
  
//print current and saved value  
printf ("Current value: %d\r\n, Saved value: %d\r\n",b_value, b_saved);  
...  
}
```

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## SetzeBitInVariable

### Beschreibung

Setzt ein Bit in der angegebenen Variablen auf 1 (TRUE).

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde.

---

#### Hinweis

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie statt dessen die Systemfunktion "SetzeBit".

---

### Verwendung in der Funktionsliste

SetzeBitInVariable (Variable, Bit)

### Verwendung in benutzerdefinierten Funktionen

SetBitInTag(Tag, Bit)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Variable

Die Variable, in der ein Bit auf 1 (TRUE) gesetzt wird.

#### Bit

Die Nummer des Bits, das auf 1 (TRUE) gesetzt wird.

Wenn Sie diese Systemfunktion in einer benutzerdefinierten Funktion verwenden, werden die Bits in der angegebenen Variablen unabhängig von der verwendeten Steuerung von rechts nach links gezählt. Die Zählung beginnt mit 0.

---

#### Hinweis

Eine notwendige Bedingung für eine zuverlässige Funktionalität ist eine garantierte Aktualisierung der verwendeten Variablen mit den aktuellen Prozesswerten. Projektieren Sie die Variable deswegen in einem E/A-Feld oder die Systemfunktion an einem Bildobjekt z. B. einer Schaltfläche.

Wenn Sie die Systemfunktion an ein kurzzeitiges Ereignis z. B. Kommen einer Meldung projiziert haben, dann erreichen Sie aktuelle Prozesswerte nur, wenn Sie die Variable auf dauerhaft lesen setzen.

---

#### Beispiel

Der folgende Programmcode setzt ein Bit an der angegebenen Position `bitposition` der Variablen `bvalue` auf 1 und gibt das Ergebnis zusammen mit dem ursprünglichen Wert `bsaved` aus.

```
{
BYTE bvalue;
BYTE bsaved = bvalue;
BYTE bitposition = 2;

//Reset bit in bitposition
SetBitInTag (bvalue, bitposition);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bvalue, bsaved);
    ...
}
```

#### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

SetzeBit (Seite 3396)

#### SetzeBitWährendTasteGedrückt

##### Beschreibung

Setzt ein Bit der angegebenen Variablen auf 1 (TRUE), solange der Benutzer die projizierte Taste gedrückt hält.

Die Systemfunktion überträgt nach der Änderung des angegebenen Bits die gesamte Variable wieder an die Steuerung. Es wird nicht geprüft, ob sich zwischenzeitlich andere Bits in der

Variablen geändert haben. Bediener und Steuerung dürfen auf die angegebene Variable nur lesend zugreifen, bis die Variable wieder an die Steuerung übertragen wurde. Um Probleme mit sich zeitlich überschneidenden Zugriffen auf dieselbe Variable zu vermeiden, sollten Sie mit dieser Systemfunktion ausschließlich auf Variablen vom Typ BOOL zugreifen.

---

#### Hinweis

Durch einen für eine Taste projektierten Bildwechsel werden alle Funktionen am Ereignis "Loslassen" sofort ausgeführt, auch wenn die Taste noch gedrückt ist.

Wenn bei einer Funktionstaste die Systemfunktion "SetzeBitWährendTasteGedrückt" projektiert ist, wird sofort nach dem Ausführen des Bildwechsels das Bit zurückgesetzt. Dieses Verhalten ist notwendig, da sich die Tastenbelegung nach dem Bildwechsel ändert.

Verwenden Sie diese Systemfunktion nicht, wenn die Steuerung BOOL-Variablen unterstützt. Verwenden Sie statt dessen die Systemfunktion "SetzeBit".

---

### Verwendung in der Funktionsliste

SetzeBitWährendTasteGedrückt (Variable, Bit)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Variable

Die Variable, in der ein Bit zeitweilig auf 1 (TRUE) gesetzt wird. Verwenden Sie, soweit dies die Steuerung zulässt, ausschließlich Variablen vom Typ BOOL.

#### Bit

Die Nummer des Bits, das zeitweilig auf 1 (TRUE) gesetzt wird.

---

#### Hinweis

Eine notwendige Bedingung für eine zuverlässige Funktionalität ist eine garantierte Aktualisierung der verwendeten Variablen mit den aktuellen Prozesswerten. Projektieren Sie die Variable deswegen in einem IO-Feld oder die Funktion an einem Bildelement z. B. einer Schaltfläche.

Wenn Sie die Funktion an ein kurzzeitiges Ereignis z. B. Kommen einer Meldung projektiert haben, dann erreichen Sie aktuelle Prozesswerte nur, wenn Sie die Variable auf dauerhaft lesen setzen.

---

## Siehe auch

SetzeBit (Seite 3396)

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## SetzeFarbeHintergrundbeleuchtung

### Beschreibung

Legt die Farbe für die Hintergrundbeleuchtung fest.

---

#### Hinweis

Beim Neustart des Bediengeräts wird die Konfiguration wieder hergestellt, die beim Ausschalten eingestellt war.

---

### Verwendung in der Funktionsliste

SetzeFarbeHintergrundbeleuchtung (Wert)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Wert

Legt die Farbe für die Hintergrundbeleuchtung fest:

0 (hmiWhite) = Weiß: Keine Farbe

1 (hmiGreen) = Grün: Grüne Farbe

2 (hmiYellow) = Gelb: Gelbe Farbe

3 (hmiRed) = Rot: Rote Farbe

## SetzePLCDatumUhrzeit

### Beschreibung

Ändert das Datum und die Uhrzeit der verbundenen Steuerung.

Sie Systemfunktion "LesePLCDatumUhrzeit" ist nur für folgende Steuerungen projektierbar:

- SIMATIC S7 1200
- SIMATIC S7 1500

### Verwendung in der Funktionsliste

LesePLCDatumUhrzeit (Verbindung, Zeit)

### Verwendung in benutzerdefinierten Funktionen

GetPLCDateTime (Connection, Time)

### Parameter

#### Verbindung

Verbindung von Steuerung und Bediengerät.

#### Zeit

Übergibt das Datum und die Uhrzeit des Bediengeräts an die Steuerung. Die Steuerung übernimmt das Datum und die Uhrzeit des Bediengeräts.

### SetzePLCModus

#### Beschreibung

Schaltet die Betriebsart der Steuerung in einen der folgenden Zustände um:

- RUN
- STOP

Sie Systemfunktion "SetzePLCModus" ist nur für folgende Steuerungen projektierbar:

- SIMATIC S7 1200
- SIMATIC S7 1500

### Verwendung in der Funktionsliste

SetzePLCModus (Verbindung, Modus)

### Verwendung in benutzerdefinierten Funktionen

SetPLCMode (Connection, Mode)

### Parameter

#### Verbindung

Verbindung von Steuerung und Bediengerät.

#### Modus

Legt die Betriebsart der Steuerung fest:

RUN = Die Steuerung wird in den Zustand RUN versetzt. Das Steuerungsprogramm wird abgearbeitet.

STOP = Die Steuerung wird in den Zustand STOP versetzt. Das Steuerungsprogramm wird unterbrochen.

## SetzeSprache

### Beschreibung

Schaltet die Sprache am Bediengerät um. Alle projektierten Texte und Systemmeldungen werden am Bediengerät in der neu eingestellten Sprache dargestellt.

### Verwendung in der Funktionsliste

SetzeSprache (Sprache)

### Verwendung in benutzerdefinierten Funktionen

SetLanguage (Language)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

### Sprache

Legt fest, welche Sprache am Bediengerät eingestellt wird. Es sind folgende Angaben möglich

- -1 (hmiToggle) = Umschalten: Wechselt zur nächsten Sprache. Die Reihenfolge legen Sie während der Projektierung im Editor "Projektsprachen" fest.
- Nummer, die Sie im Editor "Runtime-Einstellungen" unter "Sprache & Schriftart" festgelegt haben. Wechselt zu der Sprache mit der angegebenen Nummer.
- Sprache, die Sie im Editor "Runtime-Einstellungen" unter "Sprache & Schriftart" festgelegt haben.
- Sprachkürzel gemäß der VBScript5-Referenz: Wechselt zur Sprache, die dem angegebenen Sprachkürzel entspricht, z. B. "de-DE" für Deutsch (Deutschland) oder "en-US" für Englisch (Vereinigte Staaten von Amerika).  
Eine Übersicht aller Sprachkürzel finden Sie in den Grundlagen von VBScript unter dem Thema "Gebietsschema-ID (LCID)-Diagramm".

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)



## SetzeVariable

### Beschreibung

Weist der angegebenen Variablen einen neuen Wert zu.

---

#### Hinweis

Abhängig vom Variablentyp können Sie mit dieser Systemfunktion Zeichenketten und Zahlen zuweisen.

---

### Verwendung in der Funktionsliste

SetzeVariable (Variable, Wert)

### Verwendung in benutzerdefinierten Funktionen

SetTag (Tag, Value)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Variable

Die Variable, welcher der angegebene Wert zugewiesen wird.

#### Wert

Der Wert, welcher der angegebenen Variablen zugewiesen wird.

---

#### Hinweis

Die Systemfunktion "SetzeVariable" wird nur nach Aufbau einer Verbindung ausgeführt.

---

### Beispiel

Der folgende Programmcode setzt mit der Funktion SetTag den Wert der Variablen gs\_tag\_bit auf TRUE und speichert den Rückgabewert in der Variable ok.

```
{  
BOOL ok;  
BOOL bvalue;  
  
//Set the tag to true  
ok = SetTag("gs_tag_bit", TRUE);  
//error handling
```

```
if(ok)
{
    // succeeded
    printf ( "Function has run through.\r\n" );
    bvalue = GetTagBit("gs_tag_bit");
    printf ("Value of gs_tag_bit: %d\r\n", bvalue);
}
else
{
    // failed
    printf ( "Error - function failed." );
}
...
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

ErhöheVariable (Seite 3362)

### SetzeVerbindungsmodus

#### Beschreibung

Baut die angegebene Verbindung auf oder ab.

---

#### Hinweis

Eine Verbindung zur Steuerung kann nur dann hergestellt werden, wenn Sie am Bediengerät die Betriebsart ONLINE eingestellt haben. Verwenden Sie dazu die Systemfunktion "SetzeBetriebsartGerät".

---

#### Verwendung in der Funktionsliste

SetzeVerbindungsmodus (Modus, Verbindung)

#### Verwendung in benutzerdefinierten Funktionen

SetConnectionMode (Mode, Connection)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

## Parameter

### Modus

Legt fest, ob die Verbindung mit der Steuerung hergestellt wird oder getrennt wird:

0 (hmiOnline) = Online: Verbindung wird hergestellt.

1 (hmiOffline) = Offline: Verbindung wird getrennt.

### Verbindung

Die Steuerung, die mit dem Bediengerät verbunden wird. Den Namen der Steuerung legen Sie im Verbindungs-Editor fest.

## Mehrfache Verwendung der Systemfunktion in einer benutzerdefinierten Funktion

Wenn Sie die Systemfunktion "SetzeVerbindungsmodus" für verschiedene Verbindungen verwenden, kann es vorkommen, dass nicht alle Systemfunktionen korrekt ausgeführt werden. Um dies zu vermeiden, gehen Sie wie folgt vor:

1. Legen Sie eine Variable vom Datentyp "BOOL" mit dem Startwert "0" an.
2. Projektieren Sie an das Ereignis "Wertänderung" der HMI-Variablen die Systemfunktion "SetzeVerbindungsmodus". Wenn Sie z.B. 3 Verbindungen abbauen wollen, dann müssen die Systemfunktion dreimal projektieren.
3. Wenden Sie in der benutzerdefinierten Funktion die Systemfunktion "InvertBit" auf die HMI-Variable an.

## Anwendungsbeispiel

Typische Anwendungsbeispiele für diese Systemfunktion sind die beiden folgenden:

- **Test**  
Solange keine Steuerung am Bediengerät angeschlossen ist, werden während des Tests am Bediengerät keine Fehlermeldungen ausgegeben. Wenn das Bediengerät an eine Steuerung angeschlossen ist, können Sie die Verbindung zur Steuerung auf Tastendruck herstellen.
- **Inbetriebnahme**  
Für eine Anlage sollen mehrere Steuerungen projektiert werden. Dazu projektieren Sie zunächst alle Steuerungen bis auf eine "offline". Nach der Inbetriebnahme der ersten Steuerung stellen Sie die Verbindung zu jeder weiteren Steuerung über Tastendruck her. Auf diese Weise nehmen Sie die anderen Steuerungen nacheinander in Betrieb.

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

SetzeBetriebsartGerät (Seite 3395)

## SimuliereSystemtaste

### Beschreibung

Simuliert das Verhalten einer Systemtaste. Verwenden Sie diese Systemfunktion, wenn auf dem Bediengerät eine Systemtaste nicht vorhanden ist. Z. B. die "ACK"-Taste, die "Eingabe"-Taste oder der Nummern-Block.

### Verwendung in der Funktionsliste

SimuliereSystemtaste(Systemtaste)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Systemtaste

Systemtaste, deren Verhalten simuliert werden soll.

### Systemtaste "+/-"

Bei der Systemfunktion SimuliereSystemtaste wird die Systemtaste "+/-" ausschließlich bei den folgenden Bediengeräten unterstützt:

- KP300 Basic
- KP400 Basic
- KTP400 Basic mono PN
- KTP400 Basic color PN
- KTP400 Basic color PN Portrait
- KTP600 Basic mono PN
- KTP600 Basic color PN
- KTP600 Basic color DP
- KTP1000 Basic PN
- KTP1000 Basic DP

Bei allen anderen Bediengeräten verwenden Sie die Systemtasten "+" und "-" getrennt.

## SimuliereVariable

### Beschreibung

Simuliert das Verhalten von Variablen und dynamischen Objekten wie Textlisten, ohne dass das Bediengerät an eine Steuerung angeschlossen ist. Die Systemfunktion können Sie z. B. and das Ereignis "Aufgebaut" eines Bildes projektieren.

Verwenden Sie diese Systemfunktion, wenn Sie z. B. die Funktionalität eines Projekts präsentieren wollen.

Zur Simulation können nur Variablen vom Datentyp Integer verwendet werden. Jedoch können bei OP 73, OP 77A, TP 177A sowohl Variablen vom Datentyp Integer als auch Double Integer verwendet werden.

---

#### Hinweis

Wenn Sie auf einem Basic Panel die Systemfunktion "SimuliereVariable" in Verbindung mit einer kleinen Zykluszeit verwenden, kann das Bediengerät überlastet werden.

---

### Verwendung in der Funktionsliste

SimuliereVariable (Variable, Zyklus, Maximaler Wert, Minimaler Wert, Wert)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Variable

Die Variable, deren Wert verändert wird.

#### Zyklus

Der Faktor, mit dem der Basiszyklus von 200 Millisekunden multipliziert wird. Der Zyklus legt fest, wann der Variablenwert um den angegebenen Wert verändert wird. Möglicher Zyklus zwischen 1 und 32767.

#### Maximaler Wert

Der Maximalwert, den der Variablenwert bei der Simulation annehmen kann. Der Maximalwert muss größer als der Minimalwert sein, aber kleiner gleich 32767.

#### Minimaler Wert

Der Minimalwert, den der Variablenwert bei der Simulation annehmen kann. Der Minimalwert muss kleiner als der Maximalwert sein, aber größer gleich -32768.

### Wert

Der Wert, um den der Variablenwert mit jedem Zyklus verändert wird. Möglicher Wert zwischen -32768 und 32767.

- Ein positiver Wert erhöht den Variablenwert. Wenn der Maximalwert erreicht wird, wird der Variablenwert nach dem nächsten Aktualisierungszyklus auf den Minimalwert gesetzt.
- Ein negativer Wert verringert den Variablenwert. Wenn der Minimalwert erreicht wird, wird der Variablenwert nach dem nächsten Aktualisierungszyklus auf den Maximalwert gesetzt.

## StoppeRuntime

### Beschreibung

Beendet die Runtime-Software und damit das laufende Projekt am Bediengerät.

### Verwendung in der Funktionsliste

StoppeRuntime (Modus)

### Verwendung in benutzerdefinierten Funktionen

StopRuntime (Mode)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Modus

Legt fest, ob nach dem Beenden von Runtime auch das Betriebssystem heruntergefahren wird:

0 (hmiStopRuntime) = Runtime: Betriebssystem wird nicht heruntergefahren

1 (hmiStopRuntimeAndOperatingSystem) = Runtime und Betriebssystem: Betriebssystem wird heruntergefahren (bei WinCE nicht möglich)

### Beispiel

Der folgende Programmcode fährt Runtime und Betriebssystem herunter.

```
{  
  
//Stop runtime and shutdown  
StopRuntime (hmiStopRuntimeAndOperationSystem);  
  
}
```

Der gespeicherte Rückgabewert kann im nachfolgenden Code verarbeitet werden.

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## VerfolgeBenutzeränderung

### Beschreibung

Gibt eine Systemmeldung aus, die anzeigt, welcher Benutzer gerade am Bediengerät angemeldet ist.

Diese Systemfunktion kann nur im Aufgabenplaner verwendet werden.

### Verwendung in der Funktionsliste

VerfolgeBenutzeränderung

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

--

## VerringereFokussiertenWert

### Beschreibung

Subtrahiert den angegebenen Wert vom Wert der Variablen, die mit dem Bildobjekt verbunden ist, das aktuell den Fokus besitzt.

Diese Systemfunktion kann nur an Funktionstasten verwendet werden.

### Verwendung in der Funktionsliste

VerringereFokussiertenWert (Wert)

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

#### Wert

Der Wert, der vom Variablenwert subtrahiert wird.

## VerringereVariable

### Beschreibung

Subtrahiert den angegebenen Wert vom Wert der Variablen.

$$X = X - a$$

---

#### Hinweis

Die Systemfunktion verwendet als Eingangswert und Ausgangswert dieselbe Variable. Wenn Sie diese Systemfunktion zum Umrechnen eines Wertes verwenden, müssen Sie mit einer Hilfsvariablen arbeiten. Der Hilfsvariablen weisen Sie mit der Systemfunktion "SetzeVariable" den Variablenwert zu.

---

Wenn Sie die Systemfunktion an Ereignisse einer Meldung projektieren und die Variable wird im aktuellen Bild nicht verwendet, ist nicht sichergestellt, dass der tatsächliche Wert der Variablen in der Steuerung verwendet wird. Dies kann durch Setzen der Erfassungsart "Zyklisch fortlaufend" verbessert werden.

### Verwendung in der Funktionsliste

VerringereVariable (Variable, Wert)

### Verwendung in benutzerdefinierten Funktionen

DecreaseTag (Tag, Value)

Verwendbar, sofern das projizierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Variable

Die Variable, von welcher der angegebene Wert subtrahiert wird.

#### Wert

Der Wert, der subtrahiert wird.

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)



## WechseleVerbindung

### Beschreibung

Trennt die Verbindung zur gerade verwendeten Steuerung und stellt eine Verbindung zu einer Steuerung mit einer anderen Adresse her. Die neu verbundene Steuerung muss dabei zur gleichen Geräteklasse (S7-1200, S7-300, ...usw) gehören. Bei S7-1200 ist die Verwendung der Funktion darüber hinaus nur bei absoluter Addressierung zulässig.

---

#### Hinweis

Achten Sie beim Wechsel zu einer anderen Adresse darauf, dass diese Adresse nicht bereits von einem anderen Bediengerät verwendet wird.

---

Folgende Adresstypen werden unterstützt:

- IP-Adresse
- MPI-Adresse

Folgende Steuerungstypen werden unterstützt:

- SIMATIC S7 300/400
- SIMATIC S7 NC
- SIMOTION

### Verwendung in der Funktionsliste

WechseleVerbindung (Verbindung, Adresse, Steckplatz, Baugruppenträger)

### Verwendung in benutzerdefinierten Funktionen

ChangeConnection (Connection, Address, Slot, Rack)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Verbindung

Name der Verbindung, die getrennt wird. Den Namen legen Sie während der Projektierung z.B. im Editor "Verbindungen" fest.

### **Adresse**

MPI/PROFIBUS- oder IP-Adresse der Steuerung, zu der die Verbindung hergestellt wird.

---

### **Hinweis**

Sie legen die Adresse über eine Variable fest. Die Objektliste zeigt Ihnen Variablen aller Datentypen an. Wählen Sie ausschließlich Variablen folgender Datentypen:

- Ethernet-Verbindung: Datentyp "String"
  - MPI-Verbindung: Datentypen "Int"
- 

### **Steckplatz**

Steckplatz der Steuerung, zu der die Verbindung hergestellt wird.

### **Baugruppenträger**

Baugruppenträger der Steuerung, zu der die Verbindung hergestellt wird.

## **Anwendungsbeispiel**

Sie wollen ein Bediengerät an verschiedenen Maschinen betreiben. Dazu projektieren Sie im Projekt die benötigten Steuerungen, zu denen Sie mit Tastendruck wechseln wollen. Beim Wechsel der Steuerung wird die Verbindung mit der gerade verwendeten Steuerung getrennt. Danach wird die Verbindung zur neuen Steuerung mit anderen Adressparametern wieder aufgebaut. Um auf die Werte der neuen Steuerung zuzugreifen, projektieren Sie für die verwendete Steuerung dieselben Variablen.

Die Steuerung, die Sie beim Anlegen des Projekts angegeben haben, wird standardmäßig verwendet.

1. Geben Sie im Editor "Verbindungen" Namen und Adresse der Steuerung ein.
2. Projektieren Sie im Prozessbild eine Schaltfläche.
3. Projektieren Sie am Ereignis "Drücken" die Systemfunktion "WechseleVerbindung".
4. Geben Sie als Parameter den Namen der Verbindung und die Adresse der Steuerung an.

## **Siehe auch**

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## **ZeigeAnmeldedialog**

### **Beschreibung**

Öffnet einen Dialog am Bediengerät, mit dem sich der Benutzer am Bediengerät anmelden kann.

### Verwendung in der Funktionsliste

ZeigeAnmeldedialog

### Verwendung in benutzerdefinierten Funktionen

-

### Parameter

--

### ZeigeHilfetext

### Verwendung

Zeigt den projizierten Tooltip des ausgewählten Objekts an.

Wenn die Systemfunktion an eine Funktionstaste projiziert ist, wird der Tooltip zu dem Bildobjekt angezeigt, das gerade den Fokus besitzt. Wenn gleichzeitig ein Tooltip zum Bild selbst projiziert ist, kann zu diesem mit der Taste <Enter> oder durch Doppelklick auf das Hilfefenster umgeschaltet werden.

Wenn die Systemfunktion an eine Schaltfläche projiziert ist, wird nur der Tooltip zum aktuellen Bild angezeigt. Wenn gleichzeitig ein Tooltip an die Schaltfläche selbst projiziert ist, wird zuerst der Tooltip zur Schaltfläche angezeigt. Mit der Taste <Enter> oder durch Doppelklick auf das Hilfefenster wird zum Tooltip des aktuellen Bilds umgeschaltet.

---

#### Hinweis

Solange das Hilfefenster geöffnet ist, lässt sich kein anderes Bildobjekt bedienen. Um die Bildobjekte zu bedienen, schließen Sie das Hilfefenster.

---

### Schließen des Hilfefensters

Es gibt folgende Möglichkeiten das Hilfefenster zu schließen:

Bei Tasten-Bedienung:

- Erneutes Drücken der Taste <HELP>
- Drücken der Taste <ESC>

Bei Touch-Bedienung:

- Bedienen der Schaltfläche 

### Verwendung in der Funktionsliste

ZeigeHilfetext (Darstellung)

### Verwendung in benutzerdefinierten Funktionen

ShowOperatorNotes (Display\_mode)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Darstellung

Legt fest, ob der projektierte Tooltip eingeblendet oder ausgeblendet wird:

0 (hmiOff) = Aus: Projektierter Tooltip wird ausgeblendet

1 (hmiOn) = Ein: Projektierter Tooltip wird eingeblendet

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her

### Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

### ZeigeMeldefenster

#### Beschreibung

Blendet das Meldefenster am Bediengerät ein oder aus.

#### Verwendung in der Funktionsliste

ZeigeMeldefenster (Objektnamen, Darstellung)

### Verwendung in benutzerdefinierten Funktionen

ShowAlarmWindow (Object\_name, Display\_mode)

Verwendbar, sofern das projektierte Gerät benutzerdefinierte Funktionen unterstützt. Weitere Informationen finden Sie unter "Geräteabhängigkeit".

### Parameter

#### Objektnamen

Name der Meldeanzeige, die eingeblendet oder ausgeblendet wird.

#### Darstellung

Legt fest, ob das Meldefenster eingeblendet oder ausgeblendet wird:

0 (hmiOff) = Aus: Meldeanzeige wird ausgeblendet

1 (hmiOn) = Ein: Meldeanzeige wird eingeblendet

-1 (hmiToggle) = Umschalten: Schaltet zwischen den beiden Zuständen hin und her

## Siehe auch

Geräteabhängigkeit von Systemfunktionen (Seite 3349)

## 10.6.4.2 Ereignisse





### Übersicht





### Editoren





### Einleitung

Die folgende Tabelle zeigt welche Ereignisse in welchem Editor eintreten.

Technische Änderungen vorbehalten.

| Symbol  | Editor         |
|---|----------------|
|  | Bilder         |
|  | HMI-Meldungen  |
|  | HMI-Variablen  |
|  | Aufgabenplaner |

|    |   |  |  |  |  |
|----|---|--|---|---|---|
| 1  | Abgebaut (Seite 3420)                     | X  | --  | --  | --  |
| 2  | Aktivieren (Seite 3421)                   | --   | --  | --  | --  |
| 3  | Änderung (Seite 3421)                     | --   | --  | --  | --  |
| 4  | Aufgebaut (Seite 3422)                    | X  | --  | --  | --  |
| 5  | Ausführen (Seite 3422)                    | --   | --  | --  | X   |
| 6  | Auswahl geändert (Seite 3422)             | --   | --  | --  | --  |
| 7  | Bei Überschreitung (Seite 3422)           | --   | --  | X   | --  |
| 8  | Bei Unterschreitung (Seite 3423)          | --   | --  | X   | --  |
| 9  | Beim Öffnen eines Dialogs (Seite 3423)    | --   | --  | --  | X   |
| 10 | Beim Schließen eines Dialogs (Seite 3423) | --   | --  | --  | X   |
| 11 | Benutzerwechsel (Seite 3424)              | --   | --  | --  | X   |
| 12 | Bildwechsel (Seite 3424)                  | --   | --  | --  | X   |
| 13 | Deaktivieren (Seite 3424)                 | --   | --  | --  | X   |
| 14 | Doppelklicken (Seite 3425)                | --   | --  | --  | --  |
| 15 | Drücken (Seite 3425)                      | --   | --  | --  | --  |
| 16 | Eingabe abgeschlossen (Seite 3425)        | --   | --  | --  | --  |
| 17 | ESC doppelt drücken (Seite 3426)          | --   | --  | --  | --  |
| 18 | Gegangen (Seite 3426)                     | --   | X   | --  | --  |

|    |  |  |  |  |  |
|----|--|---|---|---|---|
| 19 | Gekommen (Seite 3426)                        | --  | X   | --  | --  |
| 20 | Klicken (Seite 3427)                         | --  | --  | --  | --  |
| 21 | Klicken bei Blinken (Seite 3427)             | --  | --  | --  | --  |
| 22 | Loop-In-Alarm (Seite 3427)                   | --  | X   | --  | --  |
| 23 | Loslassen (Seite 3428)                       | --  | --  | --  | --  |
| 24 | Meldungspufferüberlauf (Seite 3428)          | --  | --  | --  | X   |
| 25 | Quittieren (Seite 3428)                      | --  | X   | --  | --  |
| 26 | Rand erreicht (Seite 3429)                   | --  | --  | --  | --  |
| 27 | Runtime-Stopp (Seite 3429)                   | --  | --  | --  | X   |
| 28 | Taste drücken (Seite 3430)                   | --  | --  | --  | --  |
| 29 | Taste loslassen (Seite 3430)                 | --  | --  | --  | --  |
| 30 | Überlauf (Seite 3430)                        | --  | --  | --  | --  |
| 31 | Umschalten AUS (Seite 3431)                  | --  | --  | --  | --  |
| 32 | Umschalten EIN (Seite 3431)                  | --  | --  | --  | --  |
| 33 | Wenig freier Speicher (Seite 3431)           | --  | --  | --  | --  |
| 34 | Wenig freier Speicher, kritisch (Seite 3431) | --  | --  | --  | --  |
| 35 | Wertänderung (Seite 3432)                    | --  | --  | X   | --  |
| 36 | Zeit abgelaufen (Seite 3432)                 | --  | --  | --  | --  |

## Basisobjekte

### Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchem Objekt eintreten.

Technische Änderungen vorbehalten.

| Symbol  | Objekt        |
|---|---------------|
|  | Linie         |
|  | Ellipse       |
|  | Kreis         |
|  | Rechteck      |
|  | Textfeld      |
|  | Grafikanzeige |








|    |  |  |  |  |  |  |  |
|----|--|---|--|---|---|---|---|
| 1  | Abgebaut (Seite 3420)                        | --  | --   | --  | --  | --  | --  |
| 2  | Aktivieren (Seite 3421)                      | --  | --   | --  | --  | --  | --  |
| 3  | Änderung (Seite 3421)                        | --  | --   | --  | --  | --  | --  |
| 4  | Aufgebaut (Seite 3422)                       | --  | --   | --  | --  | --  | --  |
| 5  | Ausführen (Seite 3422)                       | --  | --   | --  | --  | --  | --  |
| 6  | Auswahl geändert (Seite 3422)                | --  | --   | --  | --  | --  | --  |
| 7  | Bei Überschreitung (Seite 3422)              | --  | --   | --  | --  | --  | --  |
| 8  | Bei Unterschreitung (Seite 3423)             | --  | --   | --  | --  | --  | --  |
| 9  | Beim Öffnen eines Dialogs (Seite 3423)       | --  | --   | --  | --  | --  | --  |
| 10 | Beim Schließen eines Dialogs (Seite 3423)    | --  | --   | --  | --  | --  | --  |
| 11 | Benutzerwechsel (Seite 3424)                 | --  | --   | --  | --  | --  | --  |
| 12 | Bildwechsel (Seite 3424)                     | --  | --   | --  | --  | --  | --  |
| 13 | Deaktivieren (Seite 3424)                    | --  | --   | --  | --  | --  | --  |
| 14 | Doppelklicken (Seite 3425)                   | --  | --   | --  | --  | --  | --  |
| 15 | Drücken (Seite 3425)                         | --  | --   | --  | --  | --  | --  |
| 16 | Eingabe abgeschlossen (Seite 3425)           | --  | --   | --  | --  | --  | --  |
| 17 | ESC doppelt drücken (Seite 3426)             | --  | --   | --  | --  | --  | --  |
| 18 | Gegangen (Seite 3426)                        | --  | --   | --  | --  | --  | --  |
| 19 | Gekommen (Seite 3426)                        | --  | --   | --  | --  | --  | --  |
| 20 | Klicken (Seite 3427)                         | --  | --   | --  | --  | --  | --  |
| 21 | Klicken bei Blinken (Seite 3427)             | --  | --   | --  | --  | --  | --  |
| 22 | Loop-In-Alarm (Seite 3427)                   | --  | --   | --  | --  | --  | --  |
| 23 | Loslassen (Seite 3428)                       | --  | --   | --  | --  | --  | --  |
| 24 | Meldungspufferüberlauf (Seite 3428)          | --  | --   | --  | --  | --  | --  |
| 25 | Quittieren (Seite 3428)                      | --  | --   | --  | --  | --  | --  |
| 26 | Rand erreicht (Seite 3429)                   | --  | --   | --  | --  | --  | --  |
| 27 | Runtime-Stopp (Seite 3429)                   | --  | --   | --  | --  | --  | --  |
| 28 | Taste drücken (Seite 3430)                   | --  | --   | --  | --  | --  | --  |
| 29 | Taste loslassen (Seite 3430)                 | --  | --   | --  | --  | --  | --  |
| 30 | Überlauf (Seite 3430)                        | --  | --   | --  | --  | --  | --  |
| 31 | Umschalten AUS (Seite 3431)                  | --  | --   | --  | --  | --  | --  |
| 32 | Umschalten EIN (Seite 3431)                  | --  | --   | --  | --  | --  | --  |
| 33 | Wenig freier Speicher (Seite 3431)           | --  | --   | --  | --  | --  | --  |
| 34 | Wenig freier Speicher, kritisch (Seite 3431) | --  | --   | --  | --  | --  | --  |
| 35 | Wertänderung (Seite 3432)                    | --  | --   | --  | --  | --  | --  |
| 36 | Zeit abgelaufen (Seite 3432)                 | --  | --   | --  | --  | --  | --  |








## Elemente

## Einleitung

Die folgende Tabelle zeigt welche Ereignisse an welchem Objekt eintreten.

Technische Änderungen vorbehalten.

| Symbol  | Objekt               |
|---|----------------------|
|  | EA-Feld              |
|  | Schaltfläche         |
|  | Symbolisches EA-Feld |
|  | Grafisches EA-Feld   |
|  | Datum/Uhrzeit-Feld   |
|  | Balken               |
|  | Schalter             |

|    |   |  |  |  |  |  |  |  |
|----|---|---|---|--|---|---|---|---|
| 1  | Abgebaut (Seite 3420)                     | --  | --  | --   | --  | --  | --  | --  |
| 2  | Aktivieren (Seite 3421)                   | X   | X   | X  | X   | --  | --  | X   |
| 3  | Änderung (Seite 3421)                     | --  | X   | X  | --  | --  | --  | X   |
| 4  | Aufgebaut (Seite 3422)                    | --  | --  | --   | --  | --  | --  | --  |
| 5  | Ausführen (Seite 3422)                    | --  | --  | --   | --  | --  | --  | --  |
| 6  | Auswahl geändert (Seite 3422)             | --  | --  | --   | --  | --  | --  | --  |
| 7  | Bei Überschreitung (Seite 3422)           | --  | --  | --   | --  | --  | --  | --  |
| 8  | Bei Unterschreitung (Seite 3423)          | --  | --  | --   | --  | --  | --  | --  |
| 9  | Beim Öffnen eines Dialogs (Seite 3423)    | --  | --  | --   | --  | --  | --  | --  |
| 10 | Beim Schließen eines Dialogs (Seite 3423) | --  | --  | --   | --  | --  | --  | --  |
| 11 | Benutzerwechsel (Seite 3424)              | --  | --  | --   | --  | --  | --  | --  |
| 12 | Bildwechsel (Seite 3424)                  | --  | --  | --   | --  | --  | --  | --  |
| 13 | Deaktivieren (Seite 3424)                 | X   | X   | X  | X   | --  | --  | X   |
| 14 | Doppelklicken (Seite 3425)                | --  | --  | --   | --  | --  | --  | --  |
| 15 | Drücken (Seite 3425)                      | --  | X   | --   | --  | --  | --  | --  |
| 16 | Eingabe abgeschlossen (Seite 3425)        | --  | --  | --   | --  | --  | --  | --  |
| 17 | ESC doppelt drücken (Seite 3426)          | --  | --  | --   | --  | --  | --  | --  |
| 18 | Gegangen (Seite 3426)                     | --  | --  | --   | --  | --  | --  | --  |
| 19 | Gekommen (Seite 3426)                     | --  | --  | --   | --  | --  | --  | --  |
| 20 | Klicken (Seite 3427)                      | --  | X   | --   | --  | --  | --  | --  |
| 21 | Klicken bei Blinken (Seite 3427)          | --  | --  | --   | --  | --  | --  | --  |
| 22 | Loop-In-Alarm (Seite 3427)                | --  | --  | --   | --  | --  | --  | --  |
| 23 | Loslassen (Seite 3428)                    | --  | X   | --   | --  | --  | --  | --  |
| 24 | Meldungspufferüberlauf (Seite 3428)       | --  | --  | --   | --  | --  | --  | --  |
| 25 | Quittieren (Seite 3428)                   | --  | --  | --   | --  | --  | --  | --  |
| 26 | Rand erreicht (Seite 3429)                | --  | --  | --   | --  | --  | --  | --  |
| 27 | Runtime-Stopp (Seite 3429)                | --  | --  | --   | --  | --  | --  | --  |
| 28 | Taste drücken (Seite 3430)                | --  | --  | --   | --  | --  | --  | --  |
| 29 | Taste loslassen (Seite 3430)              | --  | --  | --   | --  | --  | --  | --  |
| 30 | Überlauf (Seite 3430)                     | --  | --  | --   | --  | --  | --  | --  |
| 31 | Umschalten AUS (Seite 3431)               | --  | --  | --   | --  | --  | --  | X   |



| 32 | Umschalten EIN (Seite 3431)                  | -- | -- | -- | -- | -- | -- | X  |
|----|--|----|----|----|----|----|----|----|
| 33 | Wenig freier Speicher (Seite 3431)           | -- | -- | -- | -- | -- | -- | -- |
| 34 | Wenig freier Speicher, kritisch (Seite 3431) | -- | -- | -- | -- | -- | -- | -- |
| 35 | Wertänderung (Seite 3432)                    | -- | -- | -- | -- | -- | -- | -- |
| 36 | Zeit abgelaufen (Seite 3432)                 | -- | -- | -- | -- | -- | -- | -- |







## Controls

## Einleitung

Die folgende Tabelle zeigt welche Ereignisse bei welchem Objekt eintreten.  
Technische Änderungen vorbehalten.

| Symbol | Objekt                    |
|--------|---------------------------|
|        | Meldeanzeige/Meldefenster |
|        | Meldeindikator            |
|        | Kurvenanzeige             |
|        | Benutzeranzeige           |
|        | Rezepturanzeige           |
|        | Hilfsindikator            |

| 1  | Abgebaut (Seite 3420)                     | -- | -- | -- | -- | -- | -- |
|----|---|----|----|----|----|----|----|
| 2  | Aktivieren (Seite 3421)                   | X  | -- | X  | X  | -- | -- |
| 3  | Änderung (Seite 3421)                     | -- | -- | -- | -- | -- | -- |
| 4  | Aufgebaut (Seite 3422)                    | -- | -- | -- | -- | -- | -- |
| 5  | Ausführen (Seite 3422)                    | -- | -- | -- | -- | -- | -- |
| 6  | Auswahl geändert (Seite 3422)             | -- | -- | -- | -- | -- | -- |
| 7  | Bei Überschreitung (Seite 3422)           | -- | -- | -- | -- | -- | -- |
| 8  | Bei Unterschreitung (Seite 3423)          | -- | -- | -- | -- | -- | -- |
| 9  | Beim Öffnen eines Dialogs (Seite 3423)    | -- | -- | -- | -- | -- | -- |
| 10 | Beim Schließen eines Dialogs (Seite 3423) | -- | -- | -- | -- | -- | -- |
| 11 | Benutzerwechsel (Seite 3424)              | -- | -- | -- | -- | -- | -- |
| 12 | Bildwechsel (Seite 3424)                  | -- | -- | -- | -- | -- | -- |
| 13 | Deaktivieren (Seite 3424)                 | X  | -- | X  | X  | -- | -- |
| 14 | Doppelklicken (Seite 3425)                | -- | -- | -- | -- | -- | -- |
| 15 | Drücken (Seite 3425)                      | -- | -- | -- | -- | -- | -- |
| 16 | Eingabe abgeschlossen (Seite 3425)        | -- | -- | -- | -- | -- | -- |
| 17 | ESC doppelt drücken (Seite 3426)          | -- | -- | -- | -- | -- | -- |
| 18 | Gegangen (Seite 3426)                     | -- | -- | -- | -- | -- | -- |

|    |  |  |  |  |  |  |  |
|----|--|---|---|---|---|---|---|
| 19 | Gekommen (Seite 3426)                        |   |   |   |   |   |   |
| 20 | Klicken (Seite 3427)                         | --  | X   | --  | --  | --  | --  |
| 21 | Klicken bei Blinken (Seite 3427)             | --  | X   | --  | --  | --  | --  |
| 22 | Loop-In-Alarm (Seite 3427)                   | --  | --  | --  | --  | --  | --  |
| 23 | Loslassen (Seite 3428)                       | --  | --  | --  | --  | --  | --  |
| 24 | Meldungspufferüberlauf (Seite 3428)          | --  | --  | --  | --  | --  | --  |
| 25 | Quittieren (Seite 3428)                      | --  | --  | --  | --  | --  | --  |
| 26 | Rand erreicht (Seite 3429)                   | --  | --  | --  | --  | --  | --  |
| 27 | Runtime-Stopp (Seite 3429)                   | --  | --  | --  | --  | --  | --  |
| 28 | Taste drücken (Seite 3430)                   | --  | --  | --  | --  | --  | --  |
| 29 | Taste loslassen (Seite 3430)                 | --  | --  | --  | --  | --  | --  |
| 30 | Überlauf (Seite 3430)                        | --  | --  | --  | --  | --  | --  |
| 31 | Umschalten AUS (Seite 3431)                  | --  | --  | --  | --  | --  | --  |
| 32 | Umschalten EIN (Seite 3431)                  | --  | --  | --  | --  | --  | --  |
| 33 | Wenig freier Speicher (Seite 3431)           | --  | --  | --  | --  | --  | --  |
| 34 | Wenig freier Speicher, kritisch (Seite 3431) | --  | --  | --  | --  | --  | --  |
| 35 | Wertänderung (Seite 3432)                    | --  | --  | --  | --  | --  | --  |
| 36 | Zeit abgelaufen (Seite 3432)                 | --  | --  | --  | --  | --  | --  |

**Ereignisse**

**Abgebaut**

**Beschreibung**

Tritt ein, wenn das aktive Bild am Bediengerät abgebaut ist.

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

## Aktivieren

### Beschreibung

Tritt ein, wenn der Benutzer ein Anzeige- und Bedienobjekt über die projizierte Tab-Reihenfolge auswählt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

#### Hinweis

Wenn der Benutzer z. B. eine Schaltfläche mit der Maus anklickt, wird das Ereignis "Klicken" ausgelöst. Um das Ereignis "Aktivieren" auszulösen, wählt der Benutzer die Schaltfläche über die Tab-Reihenfolge aus.

---

Das Ereignis "Aktivieren" dient ausschließlich zur Erkennung, ob ein Objekt ausgewählt wurde. Das Ereignis löst aber keine Kennwortabfrage aus.

Verwenden Sie daher nicht das Ereignis "Aktivieren", wenn Sie an den Funktionsaufruf des Objekts einen Zugriffsschutz projektieren wollen.

## Änderung

### Beschreibung

Tritt ein, wenn sich der Zustand eines Anzeige- und Bedienobjekts ändert.

Wenn der Benutzer z. B. einen Schalter betätigt, ändert sich der Zustand eines Objekts.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Aufgebaut

### Beschreibung

Tritt ein, wenn nach einem Bildwechsel alle projizierten Anzeige- und Bedienobjekte im aktiven Bild geladen sind.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

#### Hinweis

Um sicherzustellen, dass die Verbindung zur Steuerung nach dem Einschalten aufgebaut ist, aktivieren Sie einen Bildwechsel.

---

## Ausführen

### Beschreibung

Tritt ein, wenn die geplante Aufgabe ausgeführt wird.

## Auswahl geändert

### Beschreibung

Tritt ein, wenn der Benutzer die Auswahl ändert.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Bei Überschreitung

### Beschreibung

Tritt ein, wenn der obere Grenzwert einer Variablen überschritten wird.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Bei Unterschreitung

### Beschreibung

Tritt ein, wenn der untere Grenzwert einer Variablen unterschritten wird.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Beim Öffnen eines Dialogs

### Beschreibung

Tritt ein, wenn sich ein modaler Dialog öffnet.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Beim Schließen eines Dialogs

### Beschreibung

Tritt ein, wenn sich ein modaler Dialog schließt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Benutzerwechsel

### Beschreibung

Tritt ein, wenn sich ein Benutzer am Bediengerät abmeldet oder sich ein anderer Benutzer am Bediengerät anmeldet.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Bildwechsel

### Beschreibung

Tritt ein, wenn nach einem Bildwechsel alle projizierten Anzeige- und Bedienobjekte im Bild geladen sind.

Um bei einem Bildwechsel zu einem bestimmten Bild andere Systemfunktionen auszuführen, verwenden Sie das Ereignis "Aufgebaut".

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Deaktivieren

### Beschreibung

Tritt ein, wenn der Benutzer den Fokus von einem Anzeige- und Bedienobjekt den Fokus nimmt.

Sie deaktivieren ein Bildobjekt über die projizierte Tab-Reihenfolge oder wenn Sie mit der Maus eine andere Aktion ausführen.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

---

#### Hinweis

Systemfunktionen oder benutzerdefinierte Funktionen am Ereignis "Deaktivieren" eines Bilds werden bei Bildabwahl nicht ausgeführt.

---

Das Ereignis "Deaktivieren" dient ausschließlich zur Erkennung, ob ein Objekt abgewählt wurde. Das Ereignis löst aber keine Kennwortabfrage aus.

Verwenden Sie daher nicht das Ereignis "Deaktivieren", wenn Sie an den Funktionsaufruf des Objekts einen Zugriffsschutz projektieren wollen.

## Doppelklicken

### Beschreibung

Tritt ein, wenn der Benutzer auf ein Objekt aus der Symbolbibliothek doppelklickt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Drücken

### Beschreibung

Tritt ein, wenn der Benutzer auf eine Schaltfläche mit der linken Maustaste klickt, die Taste <RETURN> oder <SPACE> drückt.

Tritt auch ein, wenn der Benutzer auf ein Objekt der Symbolbibliothek mit der rechten Maustaste klickt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Eingabe abgeschlossen

### Beschreibung

Tritt ein, wenn die Eingabe in einem EA-Feld mit der Enter-Taste, mit der Maus oder per Touchbedienung bestätigt wird.

Das Ereignis "Eingabe abgeschlossen" wird auch gestartet, wenn sich der Wert einer Variable nicht ändert, z. B. bei einer Wertüberschreitung, oder wenn ein Benutzer den Dialog zum Quittieren einer quittierpflichtigen Variablen, z. B. beim Optionspaket Audit, abbricht.

Beim Anmelden als Benutzer und bei Eingabefeldern, die mit einer Berechtigung projiziert wurden, wird das Ereignis hingegen nicht ausgelöst.

---

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## ESC doppelt drücken

### Beschreibung

Tritt ein, wenn der Bediener die Taste <ESC> am Bediengerät zweimal drückt.

---

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Gegangen

### Beschreibung

Tritt ein, wenn eine Meldung geht.

---

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Gekommen

### Beschreibung

Tritt ein, wenn eine Meldung ausgelöst wurde und in der Meldeanzeige angezeigt wird.

---

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---



## Klicken

### Beschreibung

Tritt ein, wenn der Benutzer ein Anzeige- und Bedienobjekt mit der Maus anklickt oder am Touch-Gerät mit dem Finger berührt.

Wenn Sie ein falsches Objekt angeklickt haben, verhindern Sie die Abarbeitung der projizierten Funktionsliste wie folgt:

- Bewegen Sie den Mauszeiger mit gedrückter Maustaste vom Objekt weg. Sobald der Mauszeiger das Objekt verlassen hat, lassen Sie die Maustaste los. Die Funktionsliste wird dann nicht abgearbeitet.
- Bei Touch-Geräten berühren Sie das Display so lange mit dem Finger bis eine Aktion ausgeführt wird, z. B. ein Bildwechsel.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Klicken bei Blinken

### Beschreibung

Tritt ein, wenn der Benutzer einen blinkenden Meldeindikator mit der Maus anklickt oder mit dem Finger berührt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Loop-In-Alarm

### Beschreibung

Tritt ein, sobald der Benutzer eine Meldung in der Meldeanzeige markiert und auf die Schaltfläche "Loop-In-Alarm" klickt oder auf die Meldung doppelklickt.

An das Ereignis "Loop-In-Alarm" projizieren Sie Systemfunktionen, z. B. wechseln Sie zu dem Bild, in dem die Meldung aufgetreten ist.

An das Ereignis "Loop-In-Alarm" können Sie bei einer Runtime Professional keine lokalen Skripte projizieren.

---

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

**Loslassen**

**Beschreibung**

Tritt ein, wenn der Benutzer eine Schaltfläche loslässt.

Solange Sie die Schaltfläche gedrückt halten, tritt dieses Ereignis nicht ein.

---

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

**Meldungspufferüberlauf**

**Beschreibung**

Tritt ein, wenn die projektierte Größe des Meldepuffers erreicht ist.

---

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

**Quittieren**

**Beschreibung**

Tritt ein, wenn der Benutzer eine Meldung quittiert.

---

**Hinweis**

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Rand erreicht

### Beschreibung

Tritt ein, wenn der Benutzer den Anfang oder das Ende eines scrollbaren Bereichs erreicht.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

#### Hinweis

An das Ereignis "Rand erreicht" darf keine benutzerdefinierte Funktion projiziert werden.

---

### Projektierbare Objekte

Das Ereignis können Sie nur an die Tasten <Auf> und <Ab> projektieren oder an die Tasten, an die Sie die Systemfunktionen "BildobjektSeiteAufwärts" oder "BildobjektSeiteAbwärts" projiziert haben.

## Runtime-Stopp

### Beschreibung

Tritt ein, wenn der Benutzer die Runtime-Software am Bediengerät beendet.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

#### Hinweis

An das Ereignis "Runtime-Stopp" darf keine benutzerdefinierte Funktion projiziert werden.

---

## Taste drücken

### Beschreibung

Tritt ein, wenn der Benutzer eine Funktionstaste drückt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Taste loslassen

### Beschreibung

Tritt ein, wenn der Benutzer eine Funktionstaste loslässt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Überlauf

### Beschreibung

Tritt ein, wenn die projizierte Größe eines Archivs erreicht ist. Sie verwenden den Archivtyp "Ereignis auslösen".

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Umschalten AUS

### Beschreibung

Tritt ein, wenn der Benutzer das Anzeige- und Bedienobjekt "Schalter" in die Stellung AUS bewegt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Umschalten EIN

### Beschreibung

Tritt ein, wenn der Benutzer das Anzeige- und Bedienobjekt "Schalter" in die Stellung EIN bewegt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Wenig freier Speicher

### Beschreibung

Tritt ein, wenn auf dem Speichermedium, auf dem ein Audit Trail gespeichert wird, weniger Speicherplatz als die konfigurierte Mindestmenge zur Verfügung steht.

## Wenig freier Speicher, kritisch

### Beschreibung

Tritt ein, wenn hardwarebedingt auf dem Speichermedium, auf dem ein Audit Trail gespeichert wird, zu wenig Speicherplatz zur Verfügung steht.

## Wertänderung

### Beschreibung

Tritt ein, wenn sich der Wert einer Variablen oder der Wert eines Arrayelementes ändert.

Die Wertänderung einer Variablen wird von der Steuerung ausgelöst oder vom Benutzer, z. B. wenn er einen neuen Wert eingibt.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## Zeit abgelaufen

### Beschreibung

Tritt ein, wenn die projektierte Zeit im Aufgabenplaner abgelaufen ist.

---

#### Hinweis

Bitte beachten Sie, dass die Verfügbarkeit des Ereignisses abhängig von Bediengerät und Objekttyp ist.

---

## 10.7 Aufgaben planen

### 10.7.1 Einsatzgebiet des Aufgabenplaners

#### Definition

Im Aufgabenplaner projektieren Sie Aufgaben, die unabhängig vom Bild im Hintergrund ausgeführt werden. Sie erstellen die Aufgaben, indem Sie Systemfunktionen oder Skripte an einen Trigger knüpfen. Wenn das auslösende Ereignis eintritt, folgt der Aufruf der verknüpften Funktionen.

#### Anwendungsbeispiel

Der Aufgabenplaner dient dazu, automatisiert ereignisgesteuerte Aufgaben auszuführen. Mit einer Aufgabe automatisieren Sie z. B.:

- Regelmäßige Auslagerung von Archivdaten
- Ausdruck eines Meldeprotokolls bei Meldepufferüberlauf

- Ausdruck eines Protokolls am Schichtende
- Überwachung einer Variablen
- Überwachung eines Benutzerwechsels

---

**Hinweis**

Die Verfügbarkeit der genannten Beispiele ist abhängig vom Bediengerät.

---

**Siehe auch**

Arbeiten mit Aufgaben und Triggern (Seite 3434)

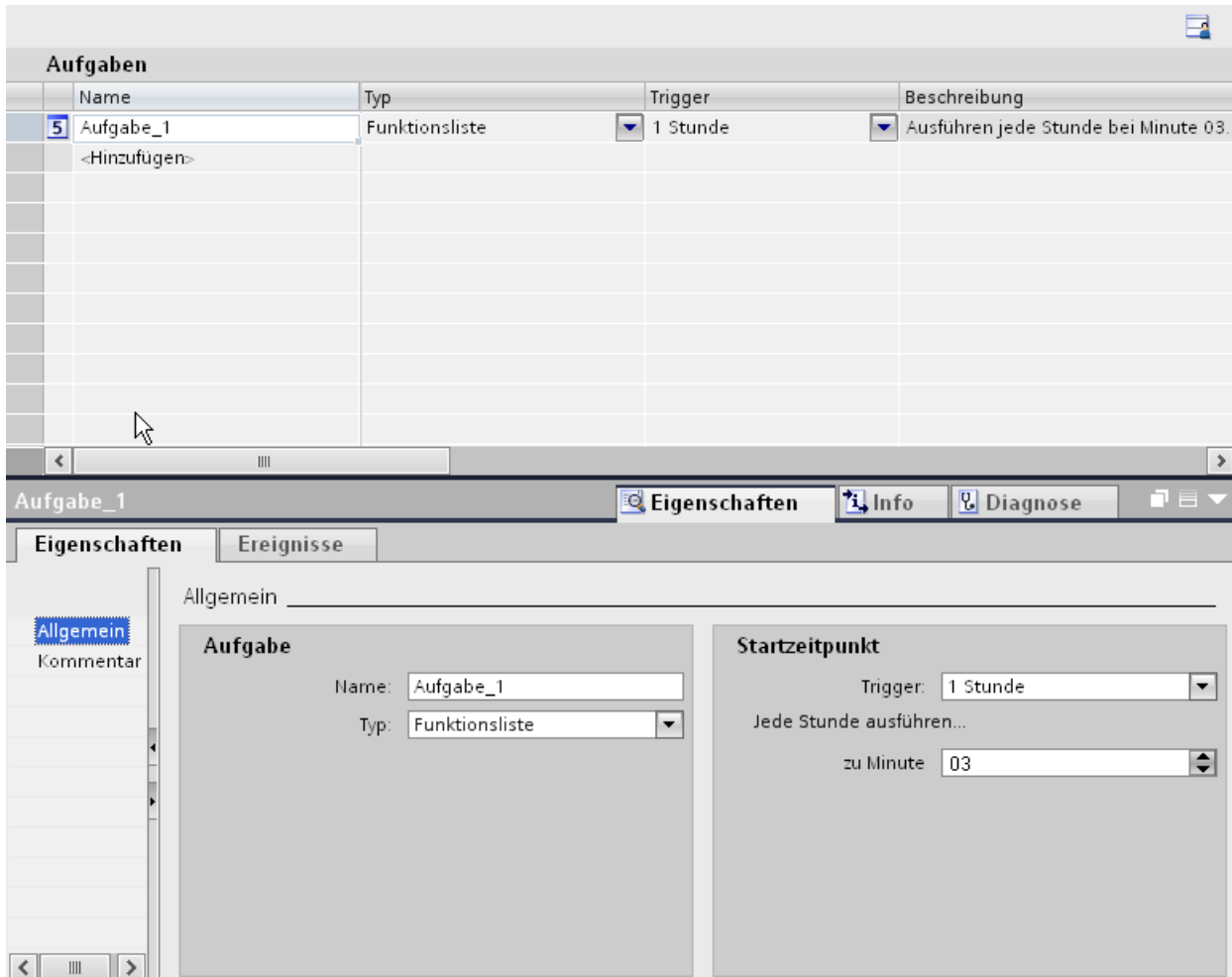
Beispiel: Benutzer bei Benutzerwechsel aktualisieren (Seite 3439)

Arbeitsbereich des Editors Aufgabenplaner (Seite 3435)

### 10.7.2 Arbeiten mit Aufgaben und Triggern

#### Einleitung

Eine Aufgabe besteht aus einem Trigger und einem Aufgabentyp.



#### Starten einer Aufgabe

Gesteuert durch einen Trigger startet der Aufgabenplaner die Aufgabe, die zu dem Trigger gehört.

#### Siehe auch

Einsatzgebiet des Aufgabenplaners (Seite 3432)



## 10.7.3 Grundlagen

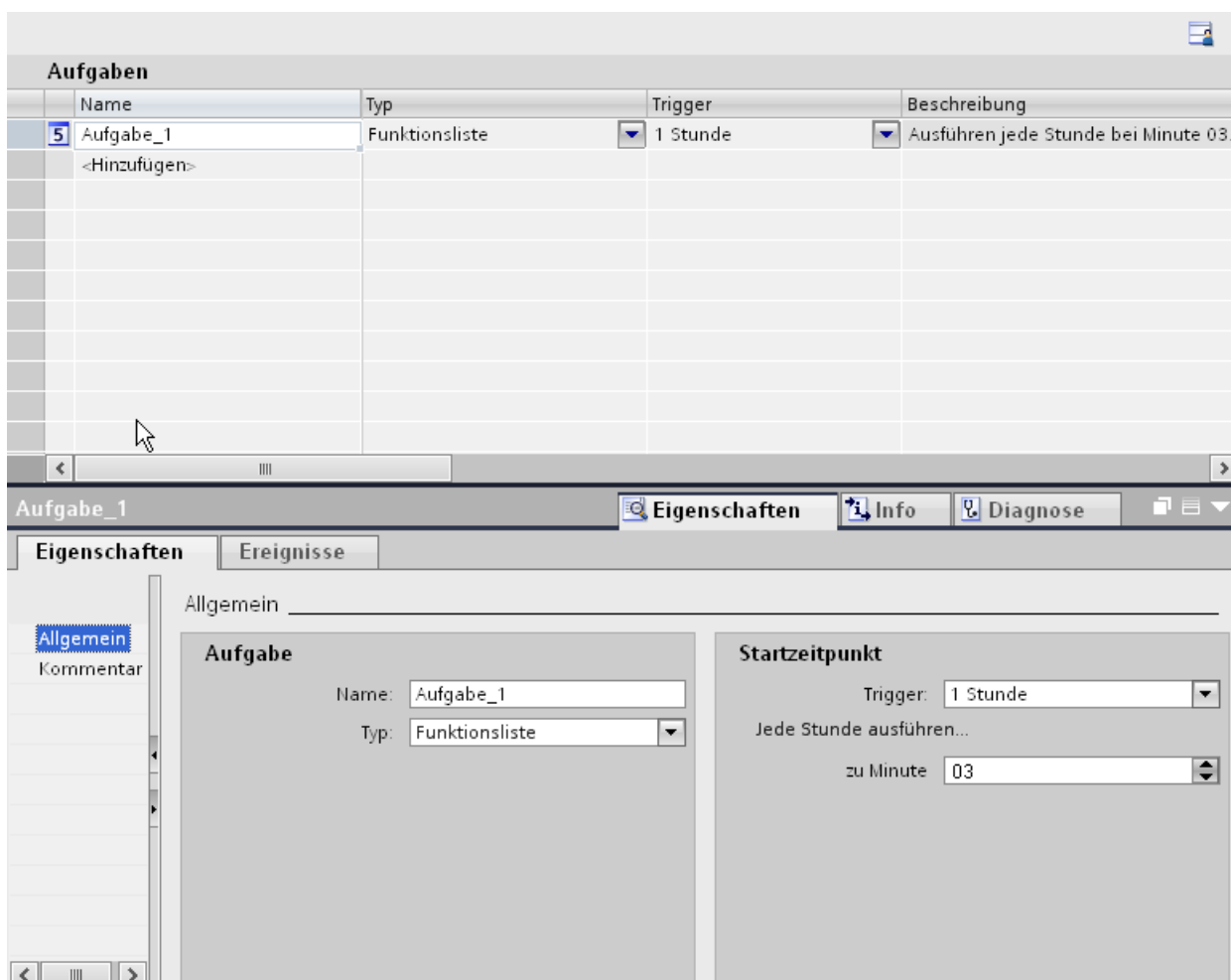
### 10.7.3.1 Arbeitsbereich des Editors Aufgabenplaner

#### Einleitung

Sie öffnen den Aufgabenplaner im Projektfenster durch einen Doppelklick auf "Aufgabenplaner". Der Arbeitsbereich zeigt die geplanten Aufgaben, die aus dem auslösenden Trigger und dem Aufgabentyp bestehen z. B. der Funktionsliste.

#### Aufbau

Der Arbeitsbereich besteht aus der Tabelle der Aufgaben.



Die Tabelle der Aufgaben zeigt die festgelegten Aufgaben mit ihren Eigenschaften z. B. dem auslösenden Trigger. Sie wählen einen Aufgabentypen und einen Trigger aus. Sie vergeben einen Namen und einen Kommentar für die Aufgabe. Die Beschreibung fasst textuell die Aufgabe mit dem geplanten Zeitpunkt zusammen.

## Inspektorfenster

Die Registerkarte "Eigenschaften" des Inspektorfensters teilt sich in zwei Bereiche.

Der Bereich "Aufgabe" zeigt ebenfalls den Namen der Aufgabe und den Aufgabentyp. Der Bereich "Startzeitpunkt" zeigt den auslösenden Trigger. Der Bereich unterscheidet sich je nach Auswahl des Triggers.

In der Registerkarte "Ereignisse" projektieren Sie die Funktionsliste mit Systemfunktionen, die in der Aufgabe ausgeführt werden.

---

### Hinweis

Nähere Informationen zu Objekten in der Oberfläche erhalten Sie über Tooltips. Bewegen Sie dazu den Mauszeiger auf das gewünschte Objekt oder drücken Sie <F1>, wenn das Objekt selektiert ist.

---

## Siehe auch

Einsatzgebiet des Aufgabenplaners (Seite 3432)

Aufgabe planen mit Ereignis-Trigger (Seite 3437)

Trigger (Seite 3437)

Funktionsliste (Seite 3436)

### 10.7.3.2 Funktionsliste

## Funktionsliste

Ein Trigger startet die Funktionsliste. Die Funktionsliste wird Zeile für Zeile abgearbeitet. Jede Zeile enthält eine Systemfunktion. An jede Aufgabe projektieren Sie genau eine Funktionsliste.

---

### Hinweis

Die Auswahl der projektierbaren Systemfunktionen in einer Funktionsliste ist abhängig vom gewählten Trigger und dem Bediengerät.

---

## Siehe auch

Arbeitsbereich des Editors Aufgabenplaner (Seite 3435)

### 10.7.3.3 Trigger

#### Einleitung

Ein Trigger wird mit einer Aufgabe verbunden und bildet somit das auslösende Ereignis für den Aufruf dieser Aufgabe. Wenn der Trigger eintritt, wird die Aufgabe ausgeführt.

#### Ereignis-Trigger

Wenn eine Aufgabe an ein Systemereignis geknüpft ist, dann wird die Aufgabe ereignisgesteuert ausgelöst. Systemereignisse sind z. B. Runtime-Stopp, Bildwechsel, Benutzerwechsel usw.

Jedes Systemereignis ist pro Bediengerät nur einmal projektierbar.

#### Aufgabe deaktivieren

Wenn Sie eine Aufgabe vorübergehend nicht benötigen, deaktivieren Sie die Aufgabe im Engineering System. Mit dem Trigger "Deaktiviert" machen Sie auch ein bereits projektiertes Systemereignis wieder verfügbar.

Beispiel: Sie planen eine Aufgabe "A" mit dem Systemereignis "Runtime-Stopp". Dann ist dieses Systemereignis für eine andere Aufgabe "B" nicht mehr verfügbar. Um das Systemereignis "Runtime-Stopp" wieder verfügbar zu machen, wählen Sie bei der Aufgabe "A" als Trigger "Deaktiviert".

---

#### Hinweis

Welche Trigger Ihnen zur Verfügung stehen ist abhängig vom Bediengerät.

---

#### Siehe auch

Arbeitsbereich des Editors Aufgabenplaner (Seite 3435)

### 10.7.3.4 Aufgabe planen mit Ereignis-Trigger

#### Einleitung

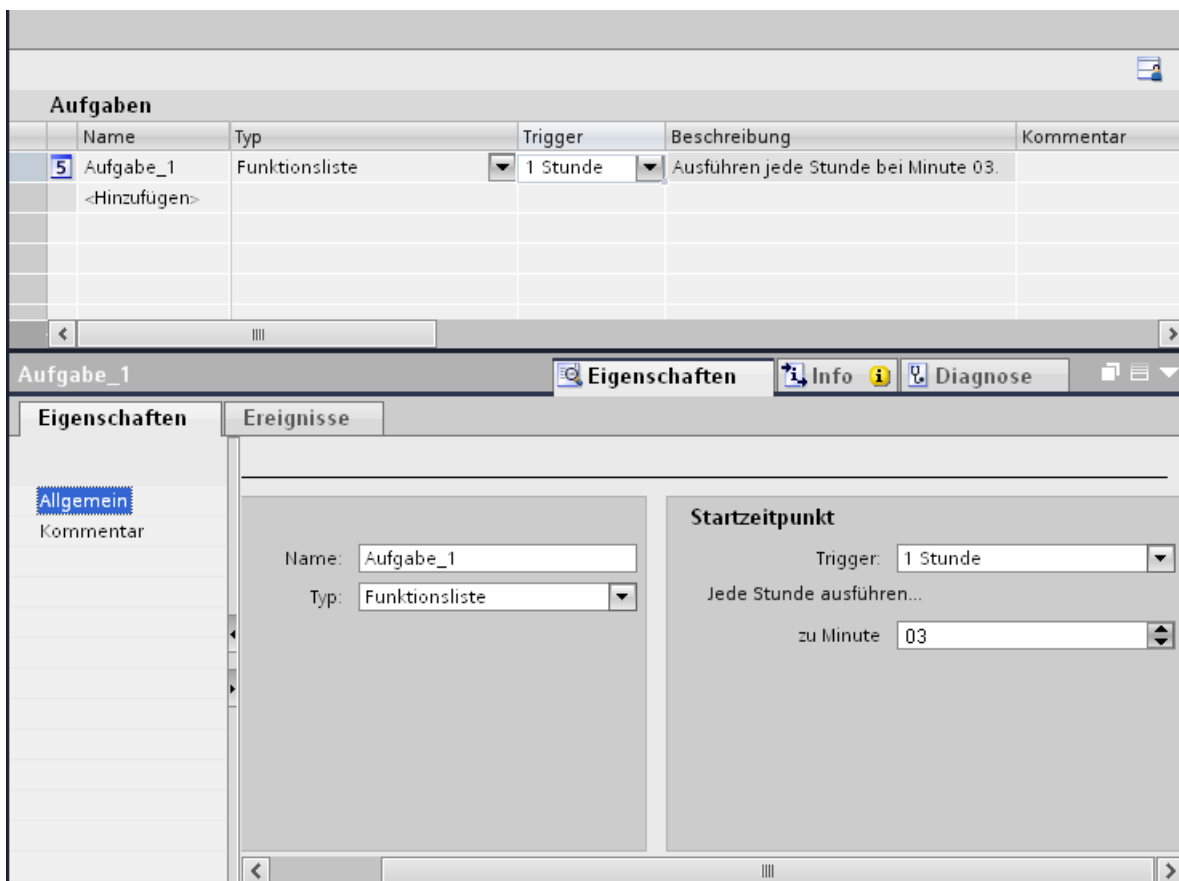
Sie planen eine Aufgabe, die beim Benutzerwechsel einen Bildwechsel auslöst.

#### Voraussetzungen

- Der Arbeitsbereich "Aufgabenplaner" ist geöffnet.
- Das Bild "Start" ist angelegt.

### Vorgehensweise

1. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
2. Geben Sie als "Name" "Bildwechsel bei Benutzerwechsel" ein.
3. Wählen Sie als "Trigger" "Benutzerwechsel".
4. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".
5. Wählen Sie in der Funktionsliste die Systemfunktion "Bilder/AktiviereBild".
6. Wählen Sie im Feld Bildname das Bild "Start".



### Ergebnis

Die Aufgabe wird beim Ereignis "Benutzerwechsel" ausgeführt. Wenn sich ein neuer Benutzer anmeldet, wird das Bild "Start" eingeblendet.

### Siehe auch

Arbeitsbereich des Editors Aufgabenplaner (Seite 3435)

## 10.7.4 Beispiele

### 10.7.4.1 Beispiel: Benutzer bei Benutzerwechsel aktualisieren

#### Aufgabe

Sie projektieren ein E/A-Feld, welches den angemeldeten Benutzer anzeigt. Sie planen eine Aufgabe, die beim Wechsel des angemeldeten Benutzers das E/A-Feld aktualisiert.

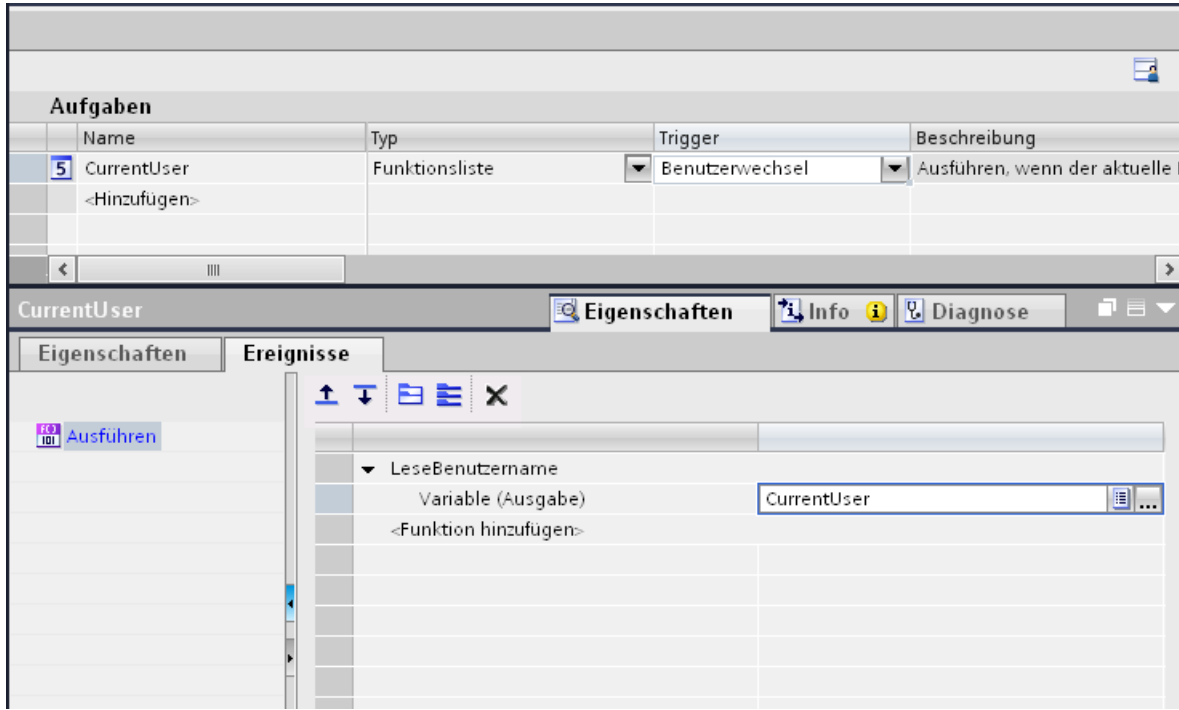
#### Voraussetzungen

- Eine Variable "CurrentUser" vom Typ "String" ist angelegt.
- Ein Bild ist angelegt und geöffnet.
- Ein E/A-Feld ist im Bild angelegt.

#### Vorgehensweise

1. Klicken Sie auf das Objekt "E/A-Feld".
2. Klicken Sie im Inspektorfenster "Eigenschaften >Eigenschaften > Allgemein":
  - Wählen Sie als "Anzeigeformat" "Zeichenkette".
  - Wählen Sie als "Variable" "CurrentUser".
  - Wählen Sie als Modus "Ausgabe".
3. Wechseln Sie in den Arbeitsbereich des Aufgabenplaners.
4. Klicken Sie in der Tabelle des Aufgabenbereichs auf "Hinzufügen...".
5. Geben Sie als "Name" "CurrentUser" ein.
6. Wählen Sie als "Trigger" "Benutzerwechsel".
7. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse".

8. Wählen Sie in der Funktionsliste in der Gruppe "Benutzerverwaltung" die Systemfunktion "LeseBenutzername".
9. Wählen Sie als "Variable" "CurrentUser".



### Ergebnis

Wenn sich ein neuer Benutzer erfolgreich anmeldet, wird die Systemfunktion "LeseBenutzername" aufgerufen. Die Variable "CurrentUser" wird aktualisiert und im E/A-Feld der neue angemeldete Benutzer angezeigt.

Wenn sich ein Benutzer nicht erfolgreich anmeldet, wird der angemeldete Benutzer abgemeldet. Das E/A-Feld zeigt weiterhin den ehemals angemeldeten Benutzer, bis sich ein neuer Benutzer erfolgreich anmeldet.

### Siehe auch

Einsatzgebiet des Aufgabenplaners (Seite 3432)

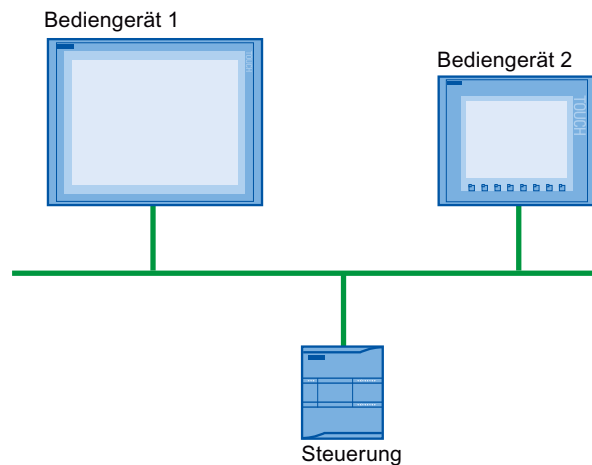
## 10.8 Mit Steuerungen kommunizieren

### 10.8.1 Grundlagen zur Kommunikation

#### 10.8.1.1 Kommunikation zwischen Geräten

##### Kommunikation

Der Datenaustausch zwischen mehreren Geräten wird als Kommunikation bezeichnet. Die Geräte können über eine Direktverbindung oder über ein Netzwerk miteinander verbunden sein. Die miteinander verbundenen Geräte werden bei einer Kommunikation als Kommunikationspartner bezeichnet.



Die übertragenen Daten zwischen den Kommunikationspartnern können verschiedenen Zwecken dienen:

- Prozesse darstellen
- Prozesse bedienen
- Meldungen ausgeben
- Prozessparameter und Maschinenparameter verwalten

##### Kommunikationspartner

Zwischen den folgenden Geräten wird in diesem Kapitel die Kommunikation näher beschrieben:

- Steuerung  
Die Steuerung steuert einen Prozess mithilfe eines Anwenderprogramms.
- Bediengerät  
Mit dem Bediengerät bedienen und beobachten Sie den Prozess.

## Grundlage für jede Kommunikation

Grundlage für jede Art von Kommunikation ist eine Netzkonfiguration. In einer Netzkonfiguration legen Sie fest, welche Verbindung zwischen den projektierten Geräten besteht.

Mit der Netzkonfiguration schaffen Sie außerdem die notwendigen Voraussetzungen für Kommunikation, d. h.:

- Alle Teilnehmer an einem Netz sind mit eindeutigen Adressen versorgt.
- Die Teilnehmer wickeln die Kommunikation mit konsistenten Übertragungseigenschaften ab.

## Automatisierungssystem

Ein Automatisierungssystem beschreibt folgende Eigenschaften:

- Steuerung und Bediengerät sind miteinander verbunden
- Netzwerk zwischen Steuerung und Bediengerät ist konfiguriert

## Kommunikation zwischen Bediengeräten

Für die Kommunikation zwischen Bediengeräten steht Ihnen das HTTP Protokoll zur Verfügung.

Nähere Hinweise finden Sie in der Dokumentation zum SIMATIC HMI HTTP Protocol.

## Kommunikation über eine einheitliche und herstellerunabhängige Schnittstelle

Mit OPC (Openness Productivity Collaboration) verfügt WinCC über eine einheitliche und herstellerunabhängige Software-Schnittstelle. Diese Schnittstelle ermöglicht einen standardisierten Datenaustausch zwischen Anwendungen aus Industrie, Büro und Fertigung.

Nähere Hinweise finden Sie in der Dokumentation zu OPC.

### 10.8.1.2 Geräte und Netze im Automatisierungssystem

#### Einleitung

Um ein Automatisierungssystem aufzubauen, müssen Sie die einzelnen Geräte konfigurieren, parametrieren und miteinander verbinden.

Sie fügen Steuerungen und Bediengeräte auf die gleiche Art und Weise in das Projekt ein. Die Konfiguration beider Geräte führen Sie ebenfalls auf die gleiche Art und Weise durch.

Automatisierungssystem aufbauen:

1. Steuerung in das Projekt einfügen.
2. Bediengerät in das Projekt einfügen.
3. Geräte miteinander vernetzen.
4. Geräte miteinander verbinden.



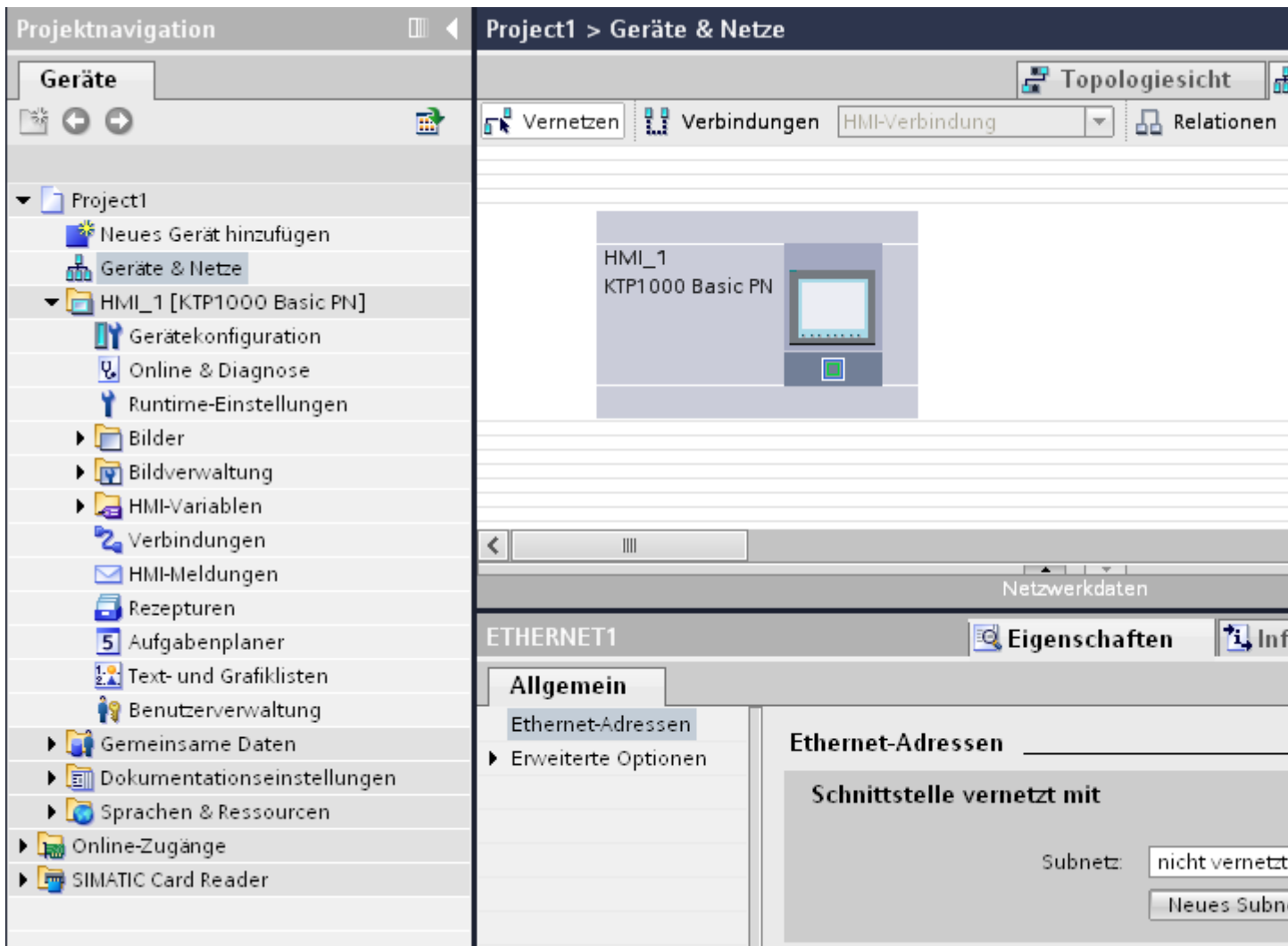
## **Geräte einfügen**

Wenn Sie ein Projekt angelegt haben, können Sie wahlweise in der Portalansicht oder in der Projektansicht ein Gerät hinzufügen.

- Portalansicht

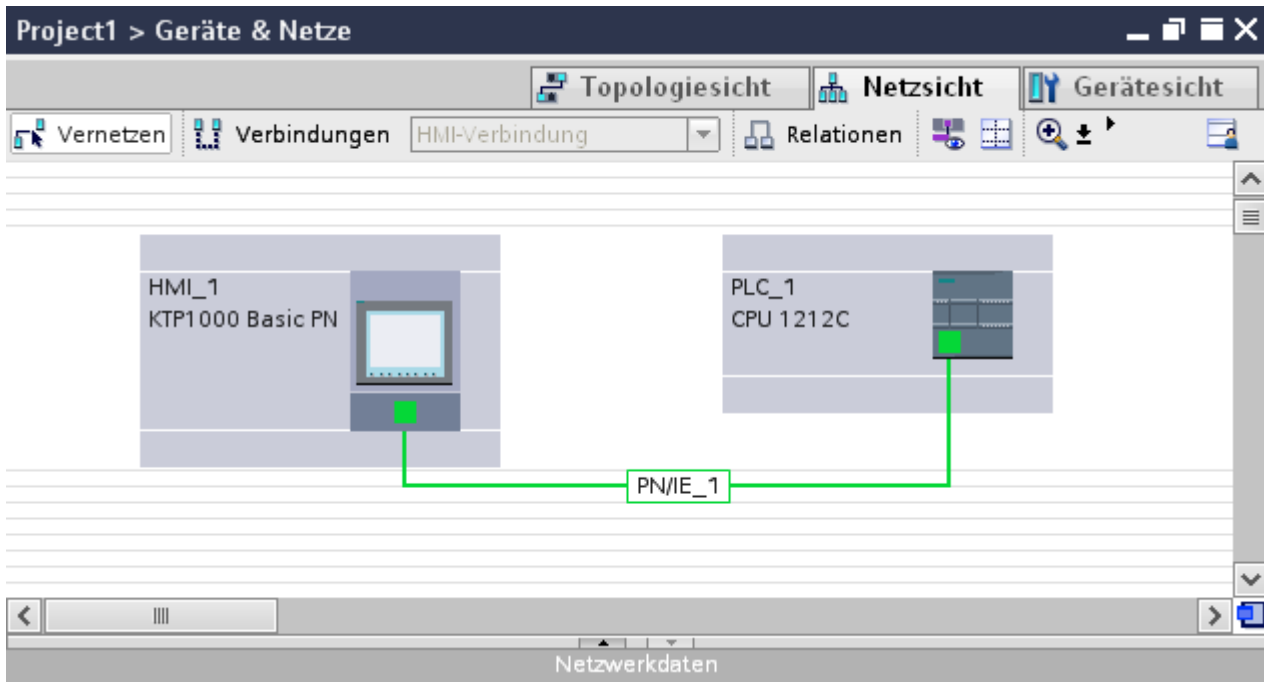


- Projektansicht



### Geräte vernetzen

Im Editor "Geräte und Netze" können Sie die Schnittstellen der kommunikationsfähigen Geräte bequem vernetzen. Mit dem Vernetzen projektieren Sie die physikalische Verbindung der Geräte.

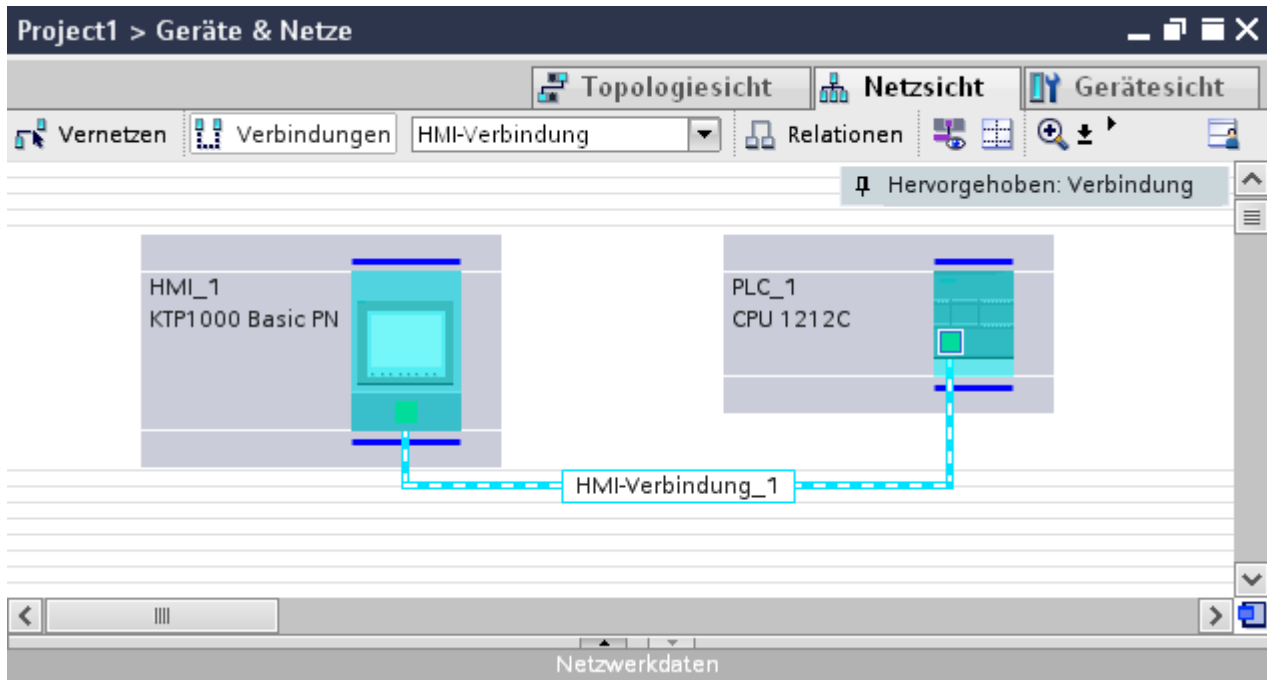


Die tabellarische Netzübersicht ergänzt die grafische Netzsicht mit folgenden Funktionen:

- Sie erhalten Detailinformationen über den Aufbau und die Parametrierung der Geräte.
- Sie können über die Spalte "Subnetz" kommunikationsfähige Komponenten mit angelegten Subnetzen verbinden.

## Geräte verbinden

Nachdem Sie die Geräte miteinander vernetzt haben, projizieren Sie die Verbindung. Für eine Kommunikation mit dem Bediengerät projizieren Sie den Verbindungstyp "HMI-Verbindung".



### 10.8.1.3 Datenaustausch über Variablen

#### Kommunikation über Variablen

Prozesswerte werden in Runtime durch Variablen weitergegeben. Prozesswerte sind Daten, die im Speicher eines der angeschlossenen Automatisierungssysteme abgelegt werden. Sie stellen den Zustand einer Anlage dar, z. B. als Temperaturen, Füllstände oder Schaltzustände. Für die Verarbeitung der Prozesswerte in WinCC definieren Sie externe Variablen.

WinCC arbeitet mit zwei Arten von Variablen:

- Externe Variablen
- Interne Variablen

#### Mit Variablen arbeiten

Nähere Informationen zur Projektierung von Variablen finden Sie im Kapitel "Mit Variablen arbeiten (Seite 3167)".

#### 10.8.1.4 Datenaustausch über Bereichszeiger

##### Kommunikation über Bereichszeiger

Bereichszeiger sind Parameterfelder. Aus diesen Parameterfeldern erhält WinCC in Runtime zur Laufzeit des Projekts Informationen. Diese Informationen beinhalten Daten über die Lage und Größe von Datenbereichen in der Steuerung.

In diese Datenbereiche wird während der Kommunikation wechselseitig von der Steuerung und dem Bediengerät geschrieben und gelesen.

Durch Auswertung der in den Datenbereichen abgelegten Daten lösen Steuerung und Bediengerät gegenseitig fest definierte Aktionen aus.

Die Bereichszeiger werden im Editor "Verbindungen" zentral verwaltet. Die Bereichszeiger dienen zum Austausch von Daten aus bestimmten Anwenderdatenbereichen.

In WinCC verwenden Sie folgende Bereichszeiger:

- Datensatz
- Datum/Uhrzeit
- Koordinierung
- Steuerungsauftrag
- Datum/Uhrzeit PLC
- Projektkennung
- Bildnummer

Die Verfügbarkeit der einzelnen Bereichszeiger ist abhängig vom verwendeten Bediengerät.

#### 10.8.1.5 Kommunikationstreiber

##### Kommunikationstreiber

Ein Kommunikationstreiber ist eine Software-Komponente, die eine Verbindung zwischen Steuerung und Bediengerät aufbaut. Der Kommunikationstreiber ermöglicht damit die Versorgung der HMI-Variablen mit Prozesswerten.

Abhängig vom verwendeten Bediengerät und dem verbundenen Kommunikationspartner kann die verwendete Schnittstelle so wie das Profil und die Übertragungsgeschwindigkeit gewählt werden.

## 10.8.2 Editoren für die Kommunikation

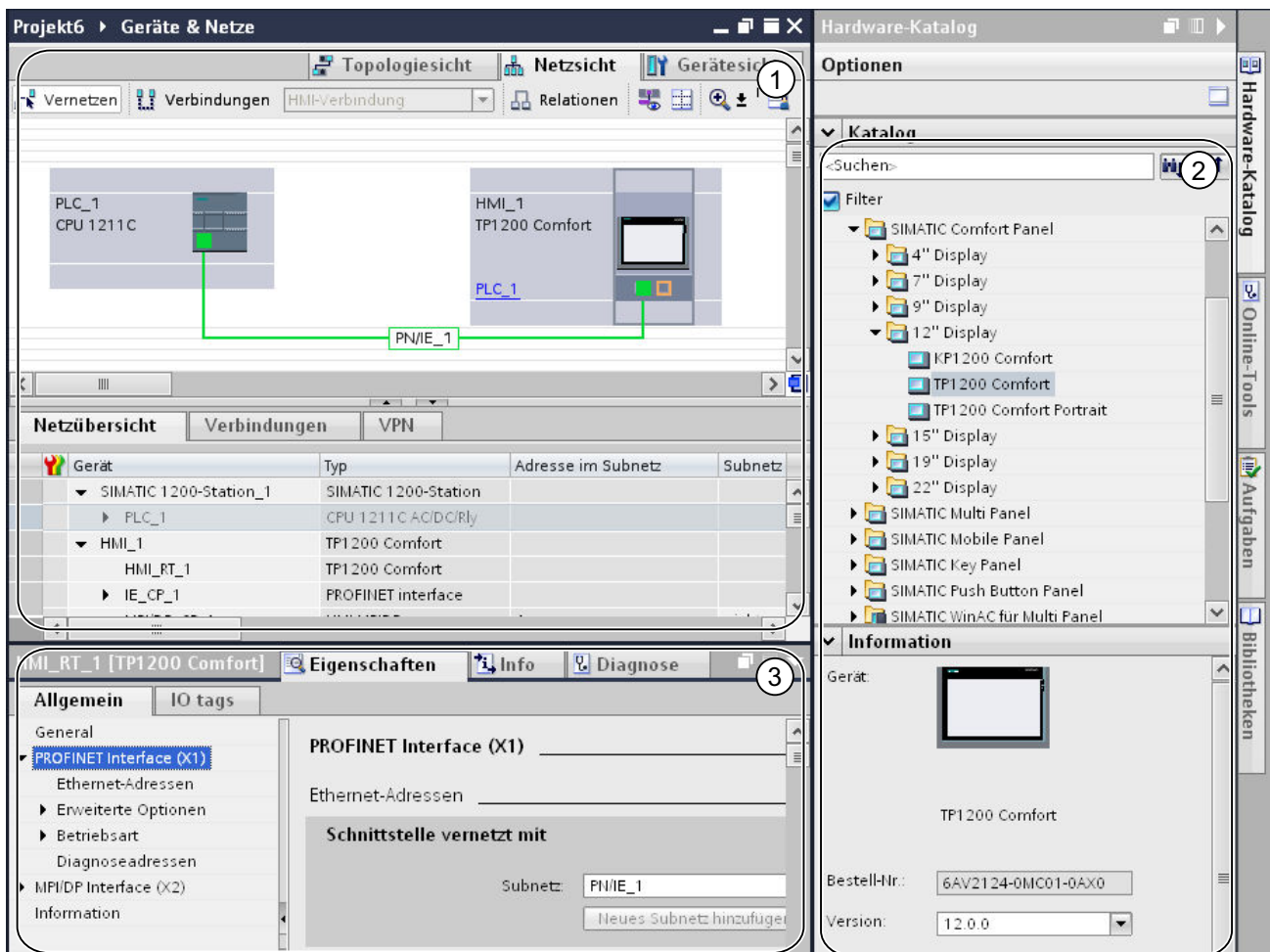
### 10.8.2.1 Editor "Geräte & Netze"

#### Funktion des Hardware- und Netzwerkeditors

Der Editor "Geräte & Netze" ist die Entwicklungsumgebung mit dem Sie Geräte und Baugruppen vernetzen, konfigurieren und parametrieren.

#### Aufbau

Der Editor "Geräte & Netze" besteht aus den folgenden Komponenten:



- 1 Gerätesicht , Netzsicht , Topologiesicht
- 2 Hardware-Katalog
- 3 Inspektorfenster

Der Editor "Geräte & Netze" bietet Ihnen drei unterschiedliche Sichten auf Ihr Projekt. Zwischen diesen drei Sichten können Sie jederzeit umschalten, je nachdem, ob Sie einzelne Geräte und Baugruppen, ganze Netzwerke und Gerätekonfigurationen oder den topologischen Aufbau ihres Projektes erstellen und bearbeiten möchten.

Im Inspektorfenster finden Sie Informationen über das aktuell markierte Objekt. Dort können Sie auch die Einstellungen für das markierte Objekt ändern.

Aus dem Hardware-Katalog ziehen Sie sich die für Ihr Automatisierungssystem benötigten Geräte und Baugruppen in die Geräte-, Netz- oder Topologiesicht.

### 10.8.2.2 Netzsicht

#### Einführung

Die Netzsicht ist ein Arbeitsbereich des Hardware- und Netzwerkeditors.

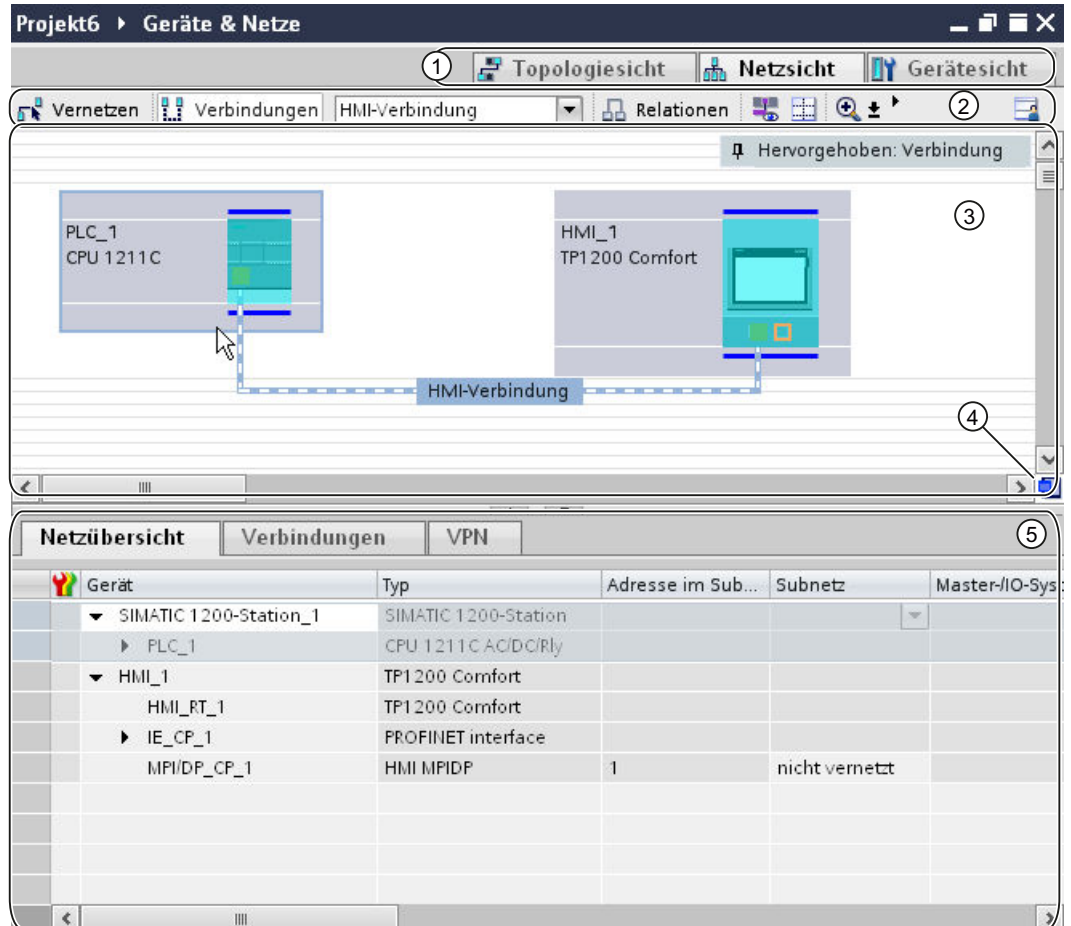
In der Netzsicht führen Sie folgende Aufgaben durch:

- Geräte konfigurieren und parametrieren
- Geräte miteinander vernetzen



## Aufbau

Das folgende Bild zeigt die beiden Bestandteile der Netzsicht:



- ① Umschalter Netzsicht/Gerätesicht/Topologiesicht
- ② Funktionsleiste der Netzsicht
- ③ Grafischer Bereich der Netzsicht
- ④ Übersichtsnavigation
- ⑤ Tabellarischer Bereich der Netzsicht








Die Teilung zwischen grafischem und tabellarischem Bereich der Netzsicht können Sie mit der Maus verändern.

Klicken Sie dazu mit der Maus auf den oberen Rand des tabellarischen Bereichs und ziehen Sie den tabellarischen Bereich bei gedrückter Maustaste größer oder kleiner.

Über die beiden kleinen Pfeiltasten können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

## Funktionsleiste

Über die Funktionsleiste stehen Ihnen die folgenden Funktionen zur Verfügung:

| Symbol  | Bedeutung   |
|---|---|
|  | Modus zum Vernetzen der Geräte.   |
|  | Modus zum Erstellen von Verbindungen.<br>Über die nebenstehende Klappliste können Sie den Verbindungstyp einstellen.  |
|  | Modus zum Erstellen von Relationen.   |
|  | Anzeige der Schnittstellenadressen.   |
|  | Einstellen der Zoomstufe.<br>Über die nebenstehende Klappliste können Sie die Zoomstufe auswählen oder direkt eintragen. Über das Zoomsymbol können Sie die Ansicht auch schrittweise vergrößern oder verkleinern oder einen Rahmen um einen zu vergrößernden Bereich ziehen. |
|  | Zeige Seitenumbrüche<br>Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.  |
|  | Anordnung merken<br>Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.  |

## Grafischer Bereich

Im grafischen Bereich der Netzsicht werden Ihnen netzrelevante Geräte, Netze, Verbindungen und Relationen angezeigt. Hier fügen Sie Geräte aus dem Hardware-Katalog ein, verbinden diese über ihre Schnittstellen miteinander und bearbeiten die Kommunikationseinstellungen.

## Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

## Tabellarischer Bereich

Der tabellarische Bereich der Netzsicht beinhaltet verschiedene Tabellen zu den vorhandenen Geräten, Verbindungen und Kommunikationseinstellungen:

- Netzübersicht
- Verbindungen
- E/A-Kommunikation

Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen.

### 10.8.2.3 Netzwerkdaten

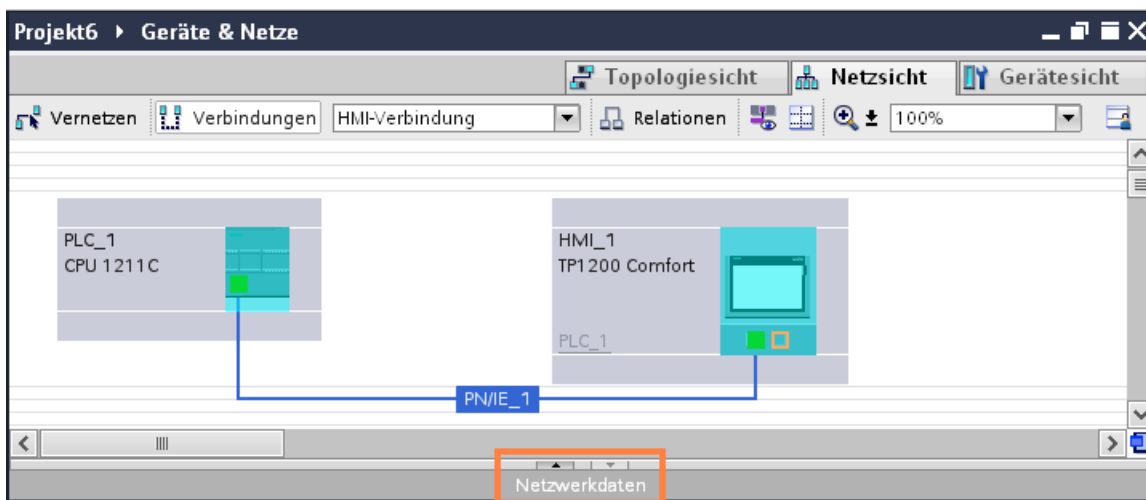
#### Einleitung

Im Editor "Netzwerkdaten" haben Sie neben der grafischen Netzwerksicht auch eine tabellarische Anzeige der "Netzwerkdaten".

Im Editor "Netzwerkdaten" haben Sie folgende Auswahlmöglichkeiten:

- Netzübersicht
- Verbindungen
- VPN
- E/A- Kommunikation

Sie öffnen die "Netzwerkdaten" unterhalb der grafischen Netzübersicht.



#### Grundfunktionen

Die Netzwerkdaten werden in tabellarischer Form angezeigt und unterstützen die folgenden Grundfunktionen zur Bearbeitung einer Tabelle:

- Spalten der Tabelle ein- und ausblenden  
Hinweis: Die für die Konfiguration maßgeblichen Spalten können nicht ausgeblendet werden.
- Spaltenbreite optimieren
- Tabelle sortieren
- Bedeutung einer Spalte, einer Zeile oder eines Feldes über Tooltips anzeigen

### Netzübersicht

Die tabellarische Netzübersicht ergänzt die grafische Netzansicht mit folgenden Funktionen:

- Sie erhalten Detailinformationen über den Aufbau und die Parametrierung der Geräte.
- Sie können über die Spalte "Subnetz" kommunikationsfähige Komponenten mit angelegten Subnetzen verbinden.

Die

| Netzübersicht            |                      |                    |                |                   |  |
|--------------------------|----------------------|--------------------|----------------|-------------------|--|
| Gerät                    | Typ                  | Adresse im Subnetz | Subnetz        | Master-/IO-System |  |
| ▼ SIMATIC 1200-Station_1 | SIMATIC 1200-Station |                    |                |                   |  |
| ▶ PLC_1                  | CPU 1211C AC/DC/Rly  |                    |                |                   |  |
| ▼ HMI_1                  | TP1200 Comfort       |                    |                |                   |  |
| HMI_RT_1                 | TP1200 Comfort       |                    |                |                   |  |
| ▶ IE_CP_1                | PROFINET interface   |                    |                |                   |  |
| MPI/DP_CP_1              | HMI MPIDP            | 1                  | nicht vernetzt |                   |  |

### Verbindungen

Unter "Verbindungen" finden Sie weitere Netzwerkdaten.

| Netzübersicht  |         |                 |                  |         |                |
|--|---------|-----------------|------------------|---------|----------------|
| Verbindungen   |         |                 |                  |         |                |
| Lokaler V  | Partner | Lokale ID (Hex) | Partner-ID (Hex) | Partner | Verbindungstyp |
| <div style="display: flex; justify-content: space-between;"> <div style="border: 1px solid gray; padding: 5px;"> <p>Spalten ein-/ausblenden</p> <p>Spaltenbreite optimieren</p> <p>Breite aller Spalten optimieren</p> </div> <div style="border: 1px solid gray; padding: 5px;"> <p><input checked="" type="checkbox"/> Lokaler Verbindungsname</p> <p><input checked="" type="checkbox"/> Lokaler Endpunkt</p> <p><input checked="" type="checkbox"/> Lokale ID (Hex)</p> <p><input checked="" type="checkbox"/> Partner-ID (Hex)</p> <p><input checked="" type="checkbox"/> Partner</p> <p><input checked="" type="checkbox"/> Verbindungstyp</p> <p><input type="checkbox"/> Einseitig</p> <p><input type="checkbox"/> Verbindungsaufbau</p> <p><input type="checkbox"/> Lokales Subnetz</p> <p><input type="checkbox"/> Partner-Subnetz</p> <p><input type="checkbox"/> Lokale Schnittstelle</p> <p><input type="checkbox"/> Partner-Schnittstelle</p> <p><input type="checkbox"/> Lokale Adresse</p> <p><input type="checkbox"/> Partner-Adresse</p> <p><input type="checkbox"/> Local TSAP</p> <p><input type="checkbox"/> Partner TSAP</p> <p><input type="checkbox"/> Local port</p> <p><input type="checkbox"/> Partner port</p> <p><input type="checkbox"/> Local LSAP</p> <p><input type="checkbox"/> Partner LSAP</p> <p><input type="checkbox"/> Protokoll</p> </div> </div> |         |                 |                  |         |                |



## Verbindungsinformation

Mit der Funktion "Verbindungsinformation" können Sie Diagnosedaten der im Editor "Geräte&Netze" markierten Verbindung anzeigen lassen.

Eine Grafik zeigt die Kommunikationspartner der Verbindung an und über welchen Kommunikationstreiber sie miteinander verbunden sind.



Folgende Daten werden angezeigt:

- Endpunkt
- Schnittstelle
- Subnetz
- Adresse
- TSAP
- Anzahl der HMI-Ressourcen

### 10.8.2.5 Gerätesicht

#### Einführung

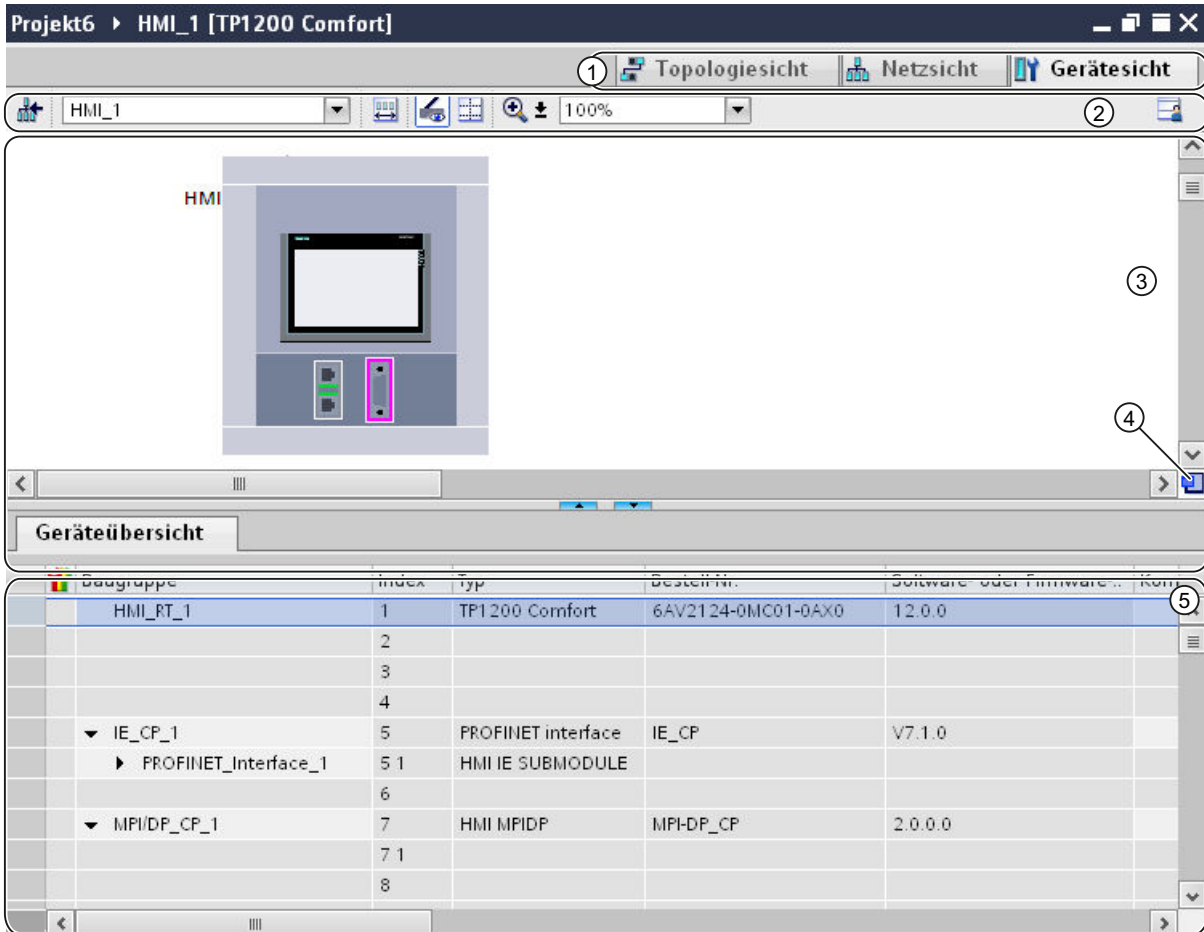
Die Gerätesicht ist ein Arbeitsbereich des Hardware- und Netzwerkeditors.

In der Gerätesicht führen Sie folgende Aufgaben durch:

- Geräte konfigurieren und parametrieren
- Baugruppen konfigurieren und parametrieren

## Aufbau

Das folgende Bild zeigt die beiden Bestandteile der Gerätesicht:









- ① Umschalter Netzsicht/Gerätesicht/Topologiesicht
- ② Funktionsleiste der Gerätesicht
- ③ Grafischer Bereich der Gerätesicht
- ④ Übersichtsnavigation
- ⑤ Tabellarischer Bereich der Gerätesicht

Die Teilung zwischen grafischem und tabellarischem Bereich der Gerätesicht können Sie mit der Maus verändern.

Klicken Sie dazu mit der Maus auf den oberen Rand des tabellarischen Bereichs und ziehen Sie den tabellarischen Bereich bei gedrückter Maustaste größer oder kleiner. Über die beiden kleinen Pfeiltasten können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

## Funktionsleiste

Über die Funktionsleiste stehen Ihnen die folgenden Funktionen zur Verfügung:

| Symbol  | Bedeutung  |
|---|--|
|  | Schaltet in die Netzsicht um.<br>Hinweis: Über die nebenstehende Klappliste kann die Gerätesicht zwischen den vorhandenen Geräten umgeschaltet werden.   |
|  | Anzeige der Ablage nicht gesteckter Baugruppen.  |
|  | Anzeige der Baugruppenbeschriftung.  |
|  | Einstellen der Zoomstufe.<br>Sie können die Zoomstufe in der nebenstehenden Klappliste auswählen oder direkt eintragen. Über das Zoomsymbol können Sie die Ansicht auch schrittweise vergrößern oder verkleinern oder einen Rahmen um einen zu vergrößernden Bereich ziehen.<br>Ab einer Zoomstufe von 200 % können Sie bei Signalmodulen die Adressbezeichnungen der E/A-Kanäle erkennen. |
|  | Zeige Seitenumbrüche<br>Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.   |
|  | Anordnung merken<br>Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.   |

## Grafischer Bereich

Im grafischen Bereich der Gerätesicht werden Ihnen Hardware-Komponenten und ggf. die zugehörigen Baugruppen angezeigt, die über einen oder mehrere Baugruppenträger einander zugeordnet sind. Bei Geräten mit Baugruppenträgern können Sie hier weitere Hardware-Objekte aus dem Hardware-Katalog in die Steckplätze der Baugruppenträger stecken.

## Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

## Tabellarischer Bereich

Der tabellarische Bereich der Gerätesicht gibt Ihnen eine Übersicht über die verwendete Hardware mit den wichtigsten technischen und organisatorischen Daten.

Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen.



## 10.8.2.6 Topologiesicht

### Einführung

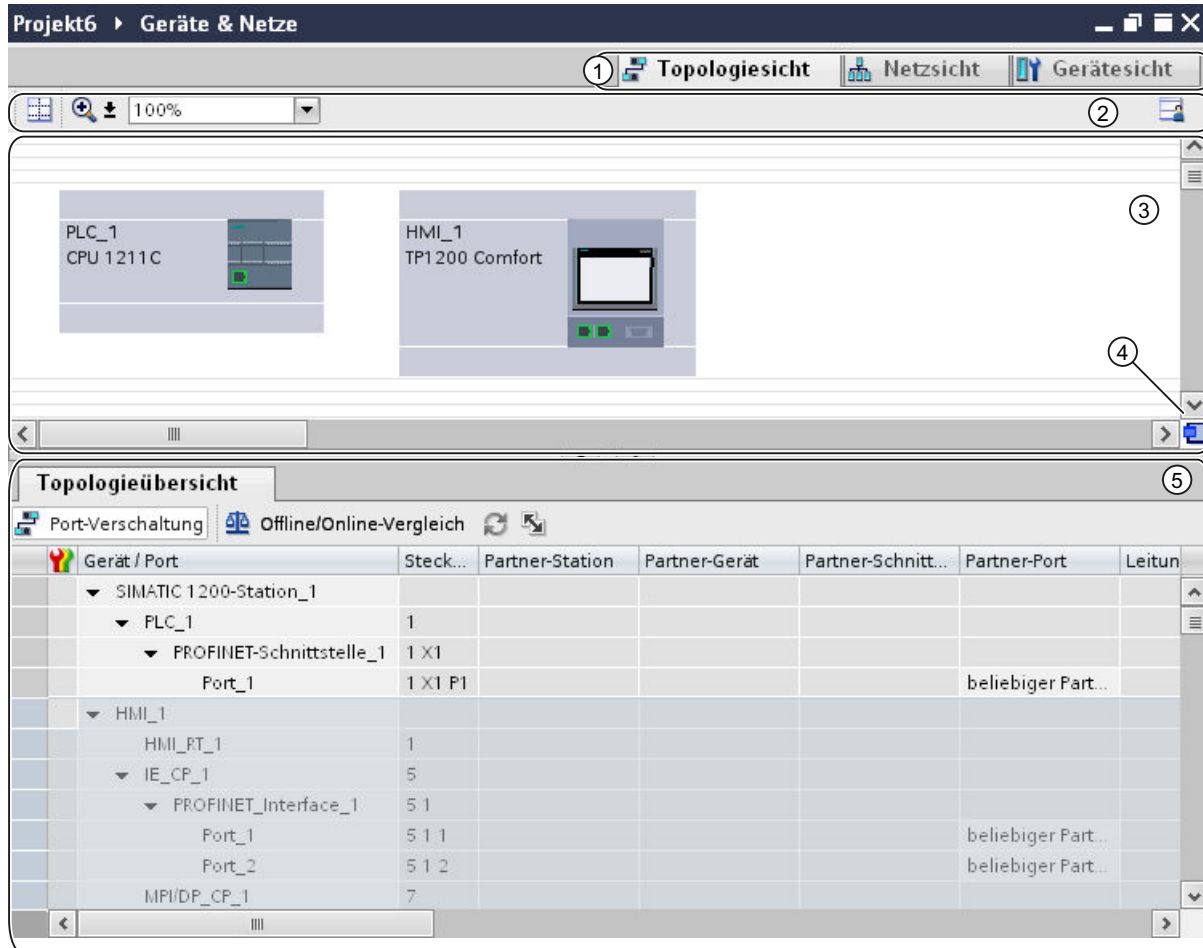
Die Topologiesicht ist ein Arbeitsbereich des Hardware- und Netzwerkeditors.

In der Topologiesicht führen Sie folgende Aufgaben durch:

- Ethernet-Topologie anzeigen
- Ethernet-Topologie projektieren
- Unterschiede zwischen Soll- und Ist-Topologie ermitteln und minimieren

### Aufbau

Das folgende Bild gibt Ihnen einen Überblick über die Topologiesicht.






- ① Umschalter Gerätesicht/Netzsicht/Topologiesicht
- ② Funktionsleiste der Topologiesicht
- ③ Grafischer Bereich der Topologiesicht
- ④ Übersichtsnavigation
- ⑤ Tabellarischer Bereich der Topologiesicht

Die Teilung zwischen grafischem und tabellarischem Bereich der Topologiesicht können Sie mit der Maus verändern.

Klicken Sie dazu mit der Maus auf den oberen Rand des tabellarischen Bereichs und ziehen Sie den tabellarischen Bereich bei gedrückter Maustaste größer oder kleiner. Über die beiden kleinen Pfeiltasten können Sie den tabellarischen Bereich mit einem Klick minimieren, maximieren oder die zuletzt gewählte Teilung wiederherstellen.

## Funktionsleiste

Über die Funktionsleiste stehen Ihnen die folgenden Funktionen zur Verfügung:

| Symbol  | Bedeutung   |
|---|---|
|  | Einstellen der Zoom-Stufe.<br>Über die nebenstehende Klappliste können Sie die Zoom-Stufe auswählen oder direkt eintragen. Über das Zoom-Symbol können Sie die Ansicht auch stufenweise vergrößern oder verkleinern oder einen Rahmen um einen zu vergrößernden Bereich ziehen. |
|  | Zeige Seitenumbrüche<br>Aktiviert die Vorschau des Seitenumbruchs. An den Stellen, an denen die Seite beim späteren Ausdruck umbricht, werden gestrichelte Linien angezeigt.  |
|  | Anordnung merken<br>Speichert die aktuelle Tabellendarstellung. Die Anordnung, Breite und Sichtbarkeit der Spalten in der tabellarischen Sicht wird gespeichert.  |

## Grafischer Bereich

Im grafischen Bereich der Topologiesicht werden Ihnen Ethernet-Baugruppen mit den zugehörigen Ports und Portverschaltungen angezeigt. Hier können Sie weitere Hardware-Objekte mit Ethernet-Schnittstellen hinzufügen.

## Übersichtsnavigation

Klicken Sie mit der Maus in die Übersichtsnavigation, um einen Überblick über die angelegten Objekte im grafischen Bereich zu erhalten. Bei gedrückter Maustaste können Sie in der Übersichtsnavigation schnell zu den gewünschten Objekten navigieren und sich diese im grafischen Bereich anzeigen lassen.

## Tabellarischer Bereich

Hier werden Ihnen die Ethernet- bzw. PROFINET-Baugruppen, deren Ports und Portverschaltungen in tabellarischer Form angezeigt. Diese Tabelle entspricht der Netzübersichtstabelle in der Netzsicht.

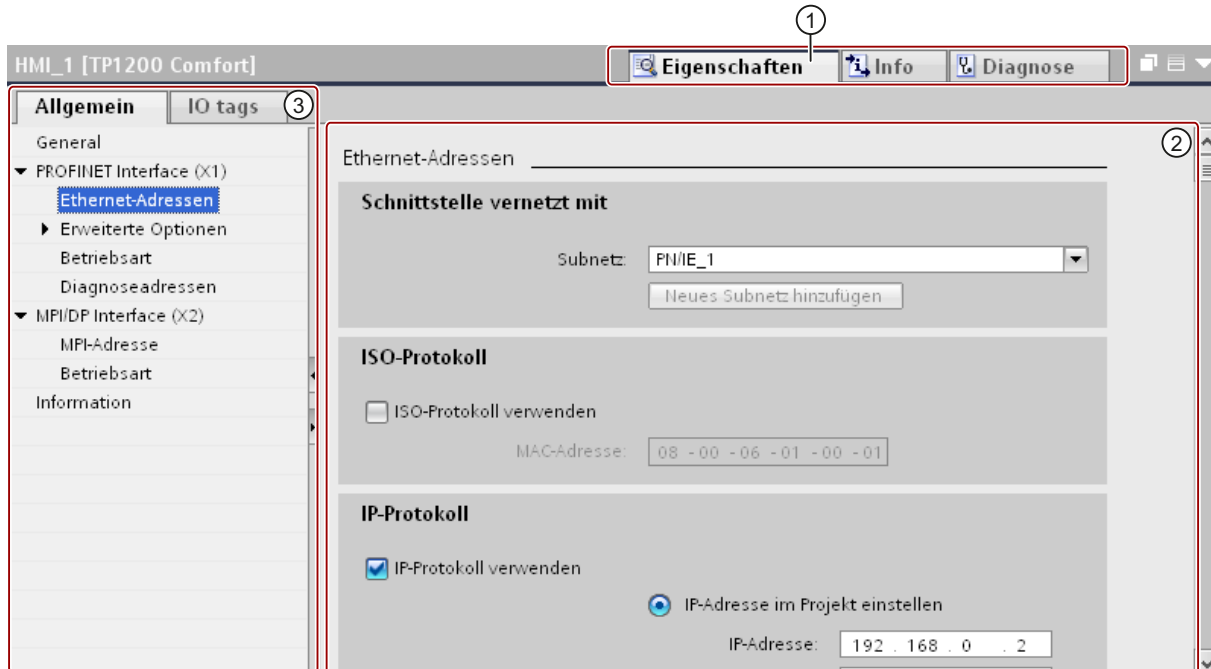
Über das Kontextmenü der Titelleiste der Tabelle können Sie die Tabellendarstellung anpassen.

### 10.8.2.7 Inspektorfenster

Die angezeigten Eigenschaften und Parameter des ausgewählten Objekts können innerhalb des Inspektorfensters bearbeitet werden.

## Aufbau

Das Inspektorfenster besteht aus folgenden Komponenten:



- ① Umschaltung zwischen verschiedenen Informations- und Arbeitsbereichen
- ② Navigation zwischen verschiedenen Informationen und Parametern
- ③ Anzeige der ausgewählten Informationen und Parameter

## Funktion

Die Informationen und Parameter im Inspektorfenster sind in verschiedene Informationstypen aufgeteilt:

- Eigenschaften
- Info
- Diagnose

Für die Anzeige der entsprechenden Informationen und Parameter klicken Sie auf den gewünschten Bereich. Für das Konfigurieren eines Automatisierungssystems ist der Bereich "Eigenschaften" der wichtigste Bereich. Dieser Bereich wird standardmäßig angezeigt.

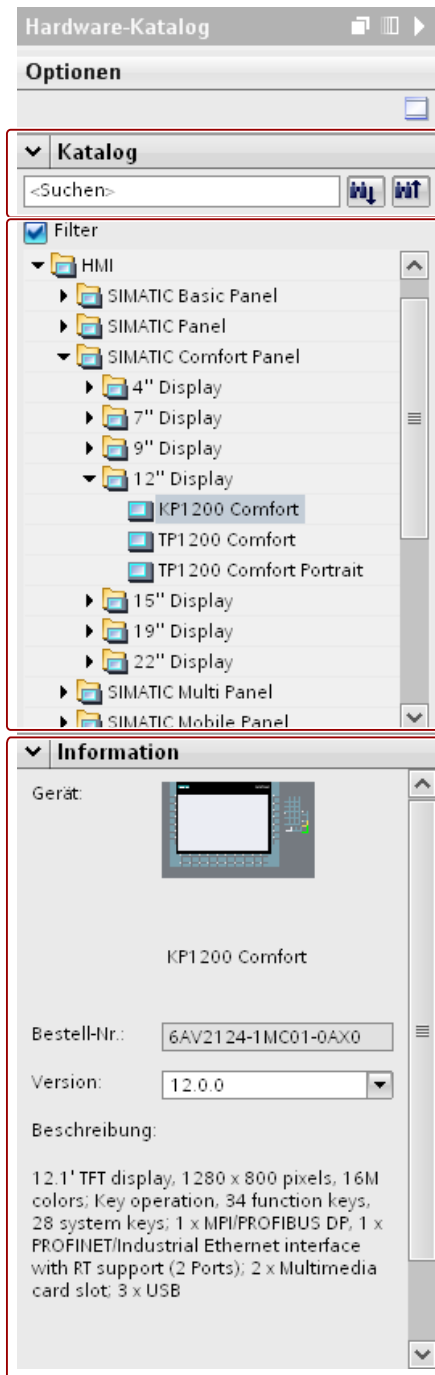
Im linken Teil des Inspektorfensters befindet sich die Bereichsnavigation. Informationen und Parameter sind dort in Gruppen angeordnet. Wenn Sie auf das Pfeilsymbol links neben dem Gruppennamen klicken, können Sie die Gruppe bei vorhandenen Untergruppen aufklappen. Wenn Sie eine Gruppe oder eine Untergruppe auswählen, werden die entsprechenden Informationen und Parameter im rechten Teil des Inspektorfensters angezeigt und können dort auch bearbeitet werden.

### **10.8.2.8 Hardware-Katalog**

Die Task Card "Hardware-Katalog" ermöglicht Ihnen einen einfachen Zugriff auf verschiedenste Hardware-Komponenten.

### Aufbau

Die Task Card "Hardware-Katalog" besteht aus den folgenden Paletten:



- ① Palette "Katalog", Such- und Filterfunktion
- ② Palette "Katalog", Komponentenauswahl
- ③ Palette "Information"

## Such- und Filterfunktion

Die Such- und Filterfunktionen der Palette "Katalog" erlauben Ihnen eine komfortable Suche nach bestimmten Hardware-Komponenten. Über die Filterfunktion können Sie die Anzeige der HW-Komponenten nach bestimmten Kriterien beschränken. Beispielsweise können Sie die Anzeige auf Objekte beschränken, die Sie auch im aktuellen Kontext platzieren können oder die bestimmte Funktionen beinhalten.

Verwendbare Objekte im aktuellen Kontext sind z. B. vernetzbare Objekte in der Netzsicht oder nur zum Gerät kompatible Baugruppen in der Gerätesicht.

## Komponentenauswahl

Die Komponentenauswahl der Palette "Katalog" enthält die installierten Hardware-Komponenten in einer Baumstruktur. Aus dem Katalog können Sie die gewünschten Geräte oder Baugruppen in den grafischen Arbeitsbereich der Geräte- oder Netzsicht ziehen.

Hardware-Komponenten, die zwar installiert sind, aber für die keine Lizenz vorhanden ist, werden grau angezeigt. Nicht-lizenzierte HW-Komponenten können Sie nicht verwenden.

Hardware-Komponenten, die thematisch zu verschiedenen Komponentengruppen gehören, sind zum Teil als verlinktes Objekt ausgeführt. Wenn Sie solch eine verlinkte Hardware-Komponente anklicken, öffnet sich die Katalogstruktur, in der Sie die entsprechende Hardware-Komponente finden.

## Information

In der Palette "Information" sehen Sie Detail-Informationen des im Katalog selektierten Objekts:

- Schematische Darstellung
- Name
- Bestellnummer
- Versionsnummer
- Beschreibung

### 10.8.2.9 Informationen zu Hardware-Komponenten

Im Hardware-Katalog haben Sie die Möglichkeit, sich Informationen von ausgewählten Hardware-Komponenten in der Palette "Information" anzeigen zu lassen. Zusätzlich können Sie sich weitere Informationen zu den ausgewählten Hardware-Komponenten über das Kontextmenü anzeigen lassen.

### Zugriff auf weitere Informationen

Wenn Sie im Hardware-Katalog ein Hardware-Objekt selektieren und das Kontextmenü öffnen, stehen Ihnen neben der "Kopieren"-Funktion drei Optionen für den Zugriff auf Informationen zu Service & Support zu Verfügung:

- Informationen zum Produktsupport
- FAQs
- Handbücher

Die gewünschten Informationen werden Ihnen im Arbeitsbereich des Hardware- und Netzwerkeditors angezeigt.

---

#### Hinweis

Den Zugriff auf Service & Support können Sie nur verwenden, wenn Sie mit dem Internet verbunden sind und die Funktion aktiviert ist. Standardmäßig ist die Funktion deaktiviert.

Um die Funktion zu aktivieren, beachten Sie das Kapitel "Produktsupport aktivieren".

---

### Informationen zum Produktsupport

Hier erhalten Sie Zugang zu allgemeinen Informationen zu Hard- und Software-Komponenten. Die Bestellnummer des selektierten Hardware-Objektes ist in der Suchmaske voreingestellt. Sie können aber auch nach anderen Hard- und Software-Komponenten suchen.

### FAQs

Hier erhalten Sie Zugang zu den "Frequently Asked Questions" (FAQ). Sie können sich verschiedene Beiträge zu Hard- und Software-Fragen ansehen. Über eine detaillierte Suchmaske können Sie das gewünschte Themengebiet herausfiltern.

### Handbücher

Hier erhalten Sie Zugang zu den Handbüchern der verschiedenen Hardware-Komponenten. Dies ist vor allem hilfreich, wenn die vorzunehmende Konfiguration, Adressierung oder Parametrierung genauere Kenntnisse über die verwendete Hardware voraussetzt.



## 10.8.3 Netze und Verbindungen

### 10.8.3.1 SIMATIC Kommunikationsnetze

#### Kommunikationsnetze

##### Überblick

Kommunikationsnetze sind ein zentraler Bestandteil moderner Automatisierungslösungen. Industrielle Netze müssen besondere Anforderungen erfüllen, z. B.:

- Kopplung von Automatisierungssystemen sowie einfachen Sensoren, Aktoren und Rechnern.
- Die Informationen müssen korrekt und zum richtigen Zeitpunkt übertragen werden.
- Robust gegen elektromagnetische Störungen, mechanische Belastung und Verschmutzung
- Flexible Anpassung an die Produktionsanforderungen

Industrielle Netze sind im Bereich der LANs (Local Area Networks) angesiedelt und ermöglichen die Kommunikation auf einem räumlich begrenzten Gebiet.

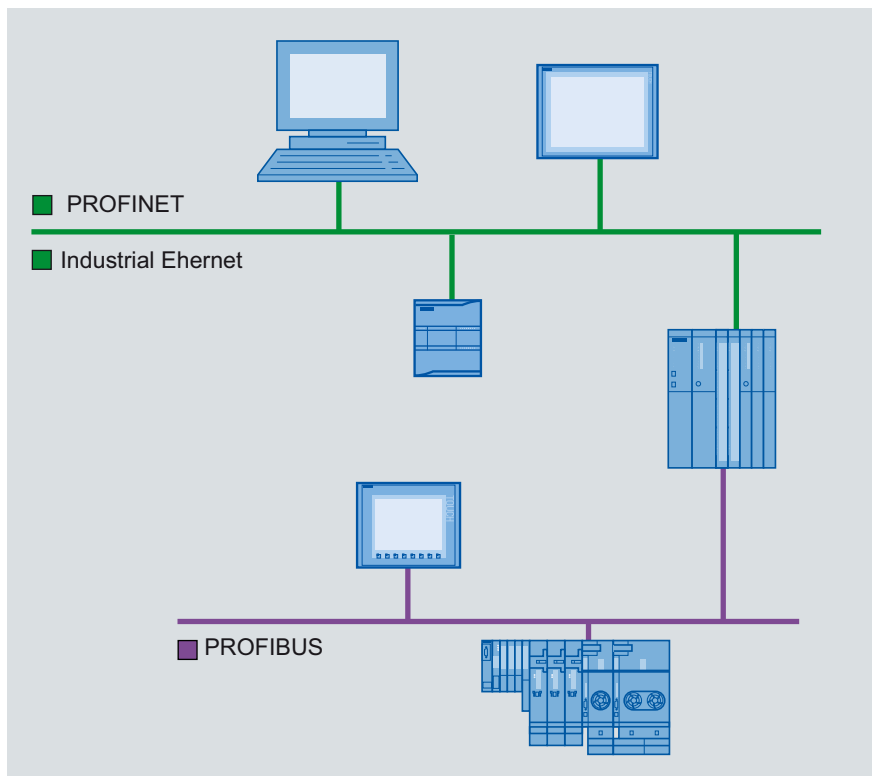
Industrielle Netze erfüllen die folgenden Kommunikationsfunktionen:

- Prozess- und Feldkommunikation der Automatisierungssysteme incl. Sensoren und Aktoren
- Datenkommunikation zwischen Automatisierungssystemen
- IT-Kommunikation zur Einbindung der modernen Informationstechnik

##### Übersicht der Netze

In diesem Kapitel werden die folgenden Netzwerke behandelt:

- **Industrial Ethernet**  
Der industrielle Netzwerkstandard für alle Ebenen
- **PROFINET**  
Der offene Industrial Ethernet Standard für die Automatisierung
- **PROFIBUS**  
Der internationale Standard für den Feldbereich und Marktführer bei den Feldbussen
- **MPI**  
Die integrierte Schnittstelle der SIMATIC-Produkte
- **PPI**  
Die integrierte Schnittstelle speziell für die S7-200



## PROFINET und Ethernet

### Industrial Ethernet

Industrial Ethernet basiert auf der Norm IEEE 802.3 und ermöglicht die Verbindung Ihres Automatisierungssystems mit Ihren Büronetzen. Industrial Ethernet bietet Ihnen IT- Dienste, mit denen Sie aus der Büroumgebung auf die Produktionsdaten zugreifen können.

### Ethernet-Netzwerk

In einem Ethernet-Netzwerk können Sie alle Geräte miteinander verbinden, die über eine integrierte Ethernet-Schnittstelle bzw. einen Kommunikationsmodul an das Netz angeschlossen sind. Hierbei können Sie mehrere Bediengeräte an eine SIMATIC S7-Steuerung und mehrere SIMATIC S7-Steuerungen an ein Bediengerät anschließen. Die maximale Anzahl der Kommunikationspartner, die Sie an ein Bediengerät anschließen können, ist abhängig vom verwendeten Bediengerät. Weitere Informationen dazu finden Sie in der Dokumentation zum jeweiligen Bediengerät.

### PROFINET

PROFINET ist ein offener Standard nach der Norm IEEE 61158 für die Industrieautomatisierung, basierend auf Industrial Ethernet. PROFINET nutzt die IT-Standards bis in die Feldebene und ermöglicht ein anlagenweites Engineering.

Mit PROFINET können Sie Automatisierungslösungen mit hoher Performance realisieren, die harte Echtzeit erfordern.

## PROFIBUS

### PROFIBUS DP

PROFIBUS DP (Dezentrale Peripherie) dient zum Anschluss von folgenden Geräten:

- Steuerungen, PCs, Bediengeräte
- Dezentrale Feldgeräte, z. B. SIMATIC ET 200
- Ventile
- Antriebe

PROFIBUS DP eignet sich mit seinen schnellen Reaktionszeiten vor allem für die Fertigungsindustrie.

Zur Grundfunktionalität gehören der zyklische Datenaustausch von Prozessdaten zwischen Master und PROFIBUS DP-Slaves sowie Diagnose.

### PROFIBUS-Netzwerk

Sie können ein Bediengerät im PROFIBUS-Netz an SIMATIC S7-Baugruppen anschließen, die eine integrierte PROFIBUS oder PROFIBUS DP-Schnittstelle haben. Hierbei können Sie mehrere Bediengeräte an eine SIMATIC S7-Steuerung anschließen und mehrere SIMATIC S7-Steuerungen an ein Bediengerät anschließen.

Die maximale Anzahl der Kommunikationspartner, die Sie an ein Bediengerät anschließen können, ist abhängig vom verwendeten Bediengerät. Weitere Informationen dazu finden Sie in der Dokumentation zum jeweiligen Bediengerät.

Die Steuerung SIMATIC S7-200 projektieren Sie im Netzwerk als passiven Teilnehmer. Sie schließen die SIMATIC S7-200 über den DP-Stecker oder über ein PROFIBUS-Kommunikationsmodul an.

## MPI

### MPI

MPI (Multi-Point Interface, mehrpunktfähige Schnittstelle) ist die integrierte Schnittstelle für SIMATIC Produkte:

- Steuerungen
- Bediengeräte
- PG/PC

Mit MPI werden kleine Subnetze mit folgenden Eigenschaften aufgebaut:

- Geringe Ausdehnung
- Wenig Teilnehmer
- Kleine Datenmengen

## MPI-Netzwerk

Sie schließen das Bediengerät an die MPI-Schnittstelle der SIMATIC S7-Steuerung an. Sie können mehrere Bediengeräte an eine SIMATIC S7-Steuerung und mehrere SIMATIC S7-Steuerungen an ein Bediengerät anschließen. Die maximale Anzahl der Kommunikationspartner, die Sie an ein Bediengerät anschließen können, ist abhängig vom verwendeten Bediengerät. Weitere Informationen dazu finden Sie in der Dokumentation zum jeweiligen Bediengerät.

## Netzwerkarchitekturen

MPI basiert auf dem PROFIBUS-Standard (IEC 61158 und EN 50170) und unterstützt die folgenden Bustopologien:

- Linie
- Stern
- Baum

Ein MPI-Subnetz umfasst max. 127 Teilnehmer und besteht aus mehreren Segmenten. Ein Segment umfasst max. 32 Teilnehmer und wird durch Abschlusswiderstände begrenzt. Segmente werden durch Repeater gekoppelt. Die maximale Leitungslänge ohne Repeater beträgt 50 m.

## PPI

### Einleitung

PPI (Point-to-Point Interface) ist eine integrierte Schnittstelle, die speziell für die SIMATIC S7-200 entwickelt wurde. Ein PPI-Netzwerk verbindet typischerweise S7-200 Steuerungen. Andere SIMATIC Steuerungen (z. B. S7-300 und S7-400) oder Bediengeräte können jedoch im PPI-Netzwerk mit einer SIMATIC S7-200 kommunizieren.

### PPI-Netzwerk

Bei einer Kopplung über PPI handelt es sich um eine Punkt-zu-Punkt Verbindung. Das Bediengerät ist der Master. Die SIMATIC S7-200 ist der Slave.

An ein Bediengerät können Sie maximal eine SIMATIC S7-200 anschließen. Sie schließen das Bediengerät über den seriellen Stecker der CPU an. Sie können mehrere Bediengeräte

an eine SIMATIC S7-200 anschließen. Hierbei ist aus Sicht der SIMATIC S7-200 zu einem Zeitpunkt nur eine Verbindung möglich.

---

#### Hinweis

Im PPI-Netz dürfen neben dem Bediengerät maximal vier Master vorhanden sein. Konfigurieren Sie im PPI-Netz aus Performancegründen nicht mehr als vier Teilnehmer gleichzeitig als Master.

---

### Netzwerkarchitekturen

PPI basiert auf dem PROFIBUS-Standard (IEC 61158 und EN 50170) und unterstützt folgende Bustopologien:

- Linie
- Stern

Mit PPI werden Multi-Master-Netzwerke aufgebaut mit max. 32 Mastern:

- Die Anzahl der Master, die mit jedem Slave kommunizieren können, ist nicht begrenzt.
- Ein Slave kann mehreren Mastern zugeordnet werden.

Mit dem RS 485 Repeater kann das PPI-Netzwerk verlängert werden. Auch Modems können an das PPI-Netzwerk angeschlossen werden.

### 10.8.3.2 Netze und Verbindungen projektieren

#### Vernetzen von Geräten

#### Einleitung

Für die Projektierung einer Verbindung steht Ihnen der Editor "Geräte & Netze" zur Verfügung. In dem Editor vernetzen Sie die Geräte miteinander. Weiterhin konfigurieren und parametrieren Sie die Geräte und Schnittstellen. Zwischen den vernetzten Geräten projektieren Sie anschließend die benötigten Verbindungen.

Im Editor "Geräte & Netze" projektieren Sie HMI-Verbindungen mit den Steuerungen:

- SIMATIC S7 1500
- SIMATIC S7 1200
- SIMATIC S7 300
- SIMATIC S7 400

HMI-Verbindungen mit anderen Steuerungen projektieren Sie im Editor "Verbindungen" des jeweiligen Bediengeräts.

### Geräte vernetzen

In der Netzansicht des Editors "Geräte & Netze" stehen Ihnen für die Vernetzung ein grafischer und ein tabellarischer Bereich zur Verfügung. Im grafischen Bereich vernetzen Sie die im Projekt enthaltenen Geräte per Drag&Drop. Im tabellarischen Bereich erhalten Sie eine Übersicht über die Geräte und deren Komponenten.

Im Editor "Geräte & Netze" können Sie folgende Steuerungen mit Bediengeräten vernetzen:

- SIMATIC S7 1500
- SIMATIC S7 1200
- SIMATIC S7 300
- SIMATIC S7 400

Alle anderen Steuerungen sind im TIA-Portal verfügbar und werden "Nicht-integriert" projiziert. "Nicht-integrierte" Verbindungen projizieren Sie im Editor "Verbindungen" des Bediengeräts.

The screenshot shows the 'Geräte & Netze' editor in 'Netzansicht' (Network View). The top toolbar includes 'Vernetzen', 'Verbindungen', 'HMI-Verbindung', 'Relationen', and a zoom level of 100%. The main workspace displays a network diagram with three devices: HMI\_1 (KTP1000 Basic PN), PLC\_1 (CPU 1214C), and PLC\_2 (CPU 1214C). A green line labeled 'PN/IE\_1' connects the HMI\_1 to both PLC\_1 and PLC\_2. Below the diagram is a table view with the following data:

| Gerät                    | Typ                  | Subnetz-Adresse | Subnetz | Master-/IO-System | Kommentar |
|--------------------------|----------------------|-----------------|---------|-------------------|-----------|
| ▼ HMI_1                  | KTP1000 Basic PN     |                 |         |                   |           |
| HMI_RT_1                 | KTP1000 Basic PN     |                 |         |                   |           |
| ▶ IE_CP_1                | PROFINET interface   |                 |         |                   |           |
| ▼ SIMATIC 1200-Station_1 | SIMATIC 1200-Station |                 |         |                   |           |
| ▶ PLC_1                  | CPU 1214C AC/DC/Rly  |                 |         |                   |           |
| ▼ SIMATIC 1200-Station_2 | SIMATIC 1200-Station |                 |         |                   |           |
| ▶ PLC_2                  | CPU 1214C AC/DC/Rly  |                 |         |                   |           |

Mit der Vernetzung projektieren Sie die physikalische Verbindung der Kommunikationspartner. Die Vernetzung der Geräte wird je nach Schnittstelle durch farbige Linien dargestellt.

## **Integrierte Verbindung im Editor "Geräte & Netze" projektieren**

### **Einleitung**

Eine HMI-Verbindung zwischen Bediengerät und einer SIMATIC S7 Steuerung projektieren Sie im Editor "Geräte & Netze". Diese HMI-Verbindung ist die direkte Verbindung zwischen den Kommunikationspartnern, die Sie in einem Projekt angelegt haben.

### **Integrierte Verbindungen**

Verbindungen von Geräten, die sich innerhalb eines Projekts befinden, werden als integrierte Verbindungen bezeichnet. Bei integrierten Verbindungen können Sie Adressen von Variablen einer Steuerung direkt projektieren.

---

#### **Hinweis**

Eine HMI-Verbindung im Editor "Geräte & Netze" ist nur für folgende Steuerungen projektierbar:

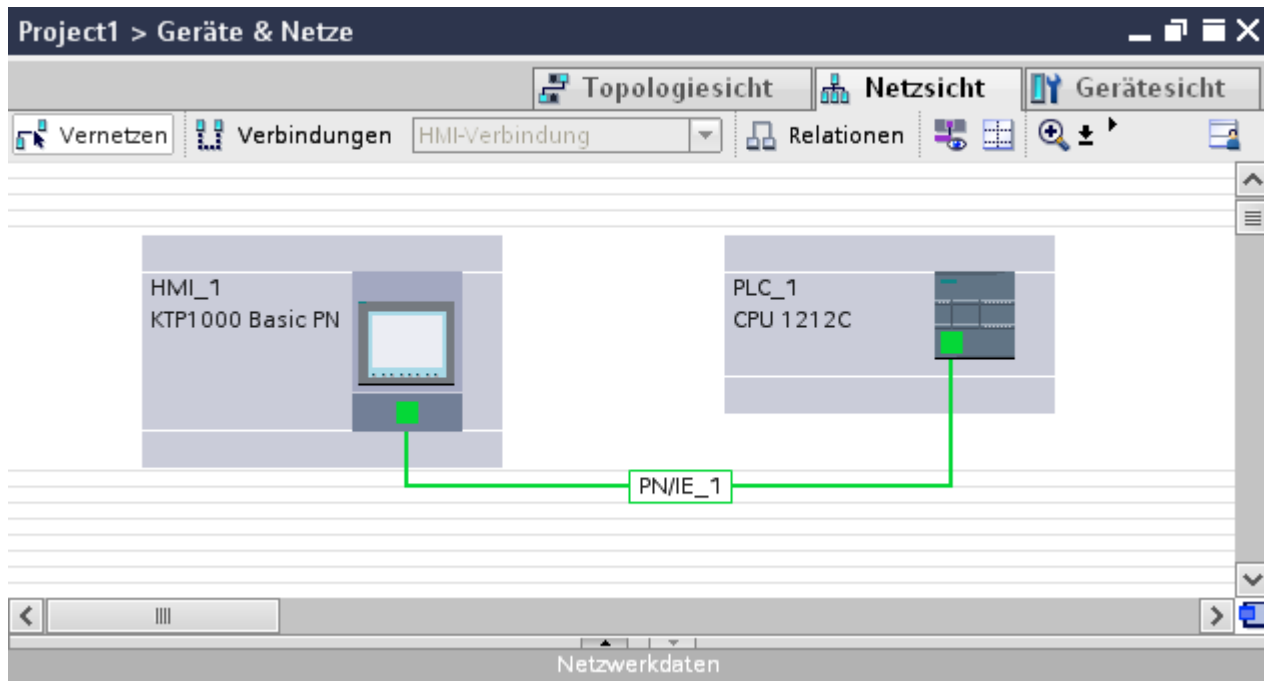
- SIMATIC S7 1500
- SIMATIC S7 1200
- SIMATIC S7 300
- SIMATIC S7 400

HMI-Verbindungen zu allen anderen Steuerungen projektieren Sie im Editor "Verbindungen" des Bediengeräts.

---

## HMI-Verbindung im Editor "Geräte & Netze" projektieren

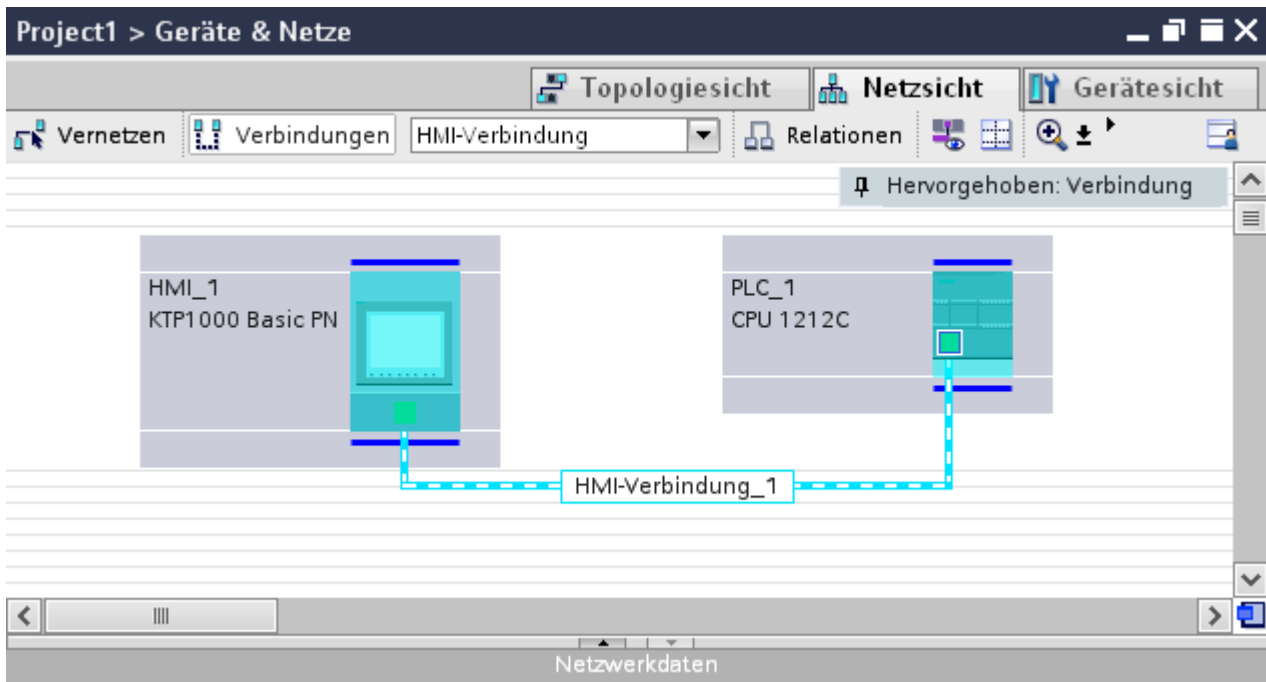
1. Fügen Sie in Ihr Projekt ein Bediengerät und eine SIMATIC S7 Steuerung ein.



2. Wechseln Sie in den Modus "Verbindungen".
3. Wählen Sie den Verbindungstyp "HMI-Verbindung" aus.



4. Verbinden Sie per Drag&Drop die beiden PROFINET-Schnittstellen miteinander.



5. Ändern Sie die Adressparameter IP-Adresse und Subnetzmaske nach den Anforderungen ihres Projekts.

## Besonderheiten des Editors "Geräte & Netze"

### Einleitung

Wenn Sie Netze oder HMI-Verbindungen projektieren oder bereits projektiert haben, unterstützt Sie der Editor "Geräte & Netze" mit folgenden Funktionen:

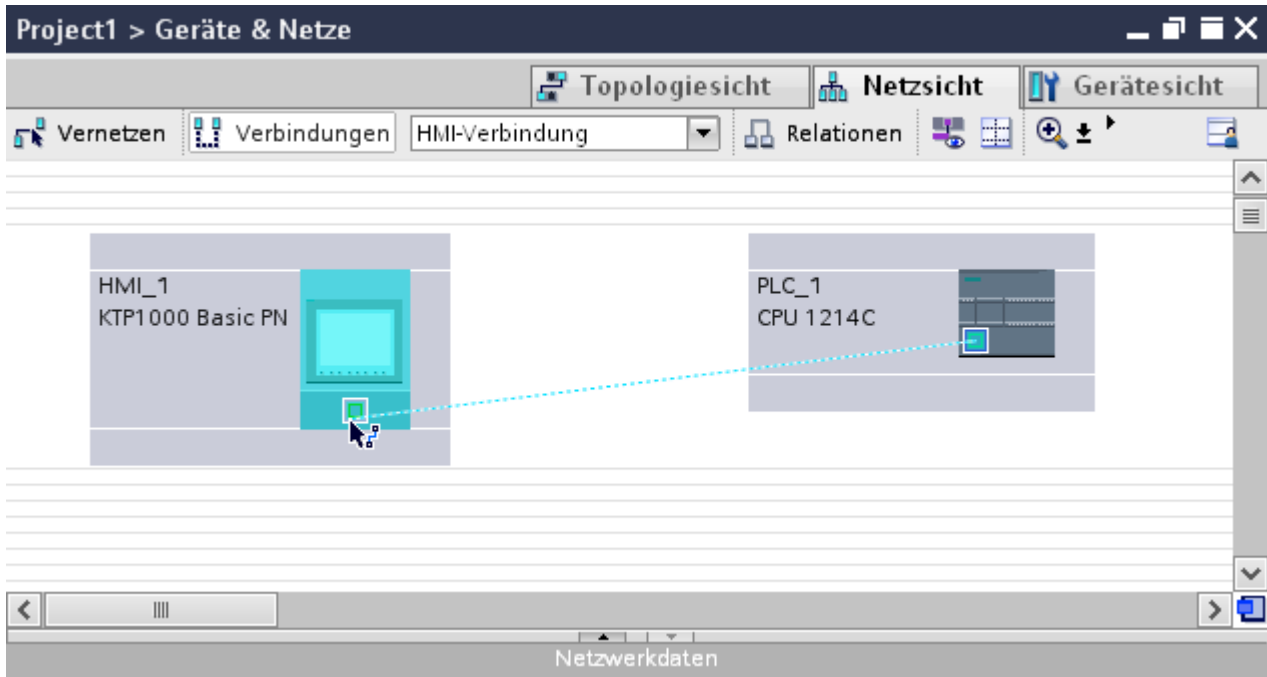
- Hervorhebung von Kommunikationspartnern
- Hervorhebung von HMI-Verbindungen
- Automatisches Anlegen von Subnetzen

### Hervorhebung von Kommunikationspartnern

Wenn Sie den Verbindungstyp "HMI-Verbindung" ausgewählt haben, dann werden alle Kommunikationspartner, bei denen eine HMI-Verbindung möglich ist türkisfarben hervorgehoben.

Ausgehend von der Schnittstelle eines Geräts legen Sie per Drag&Drop eine HMI-Verbindung zur Schnittstelle eines anderen Geräts an. Während des Drag&Drop werden alle potenziellen Kommunikationspartner türkisfarben hervorgehoben.

Das Verbinden von Schnittstellen per Drag&Drop unterbrechen Sie mit der ESC-Taste.



Folgende Symbole signalisieren beim Überfahren des Mauszeigers über die Schnittstelle eines Geräts, ob eine Verbindung möglich ist:



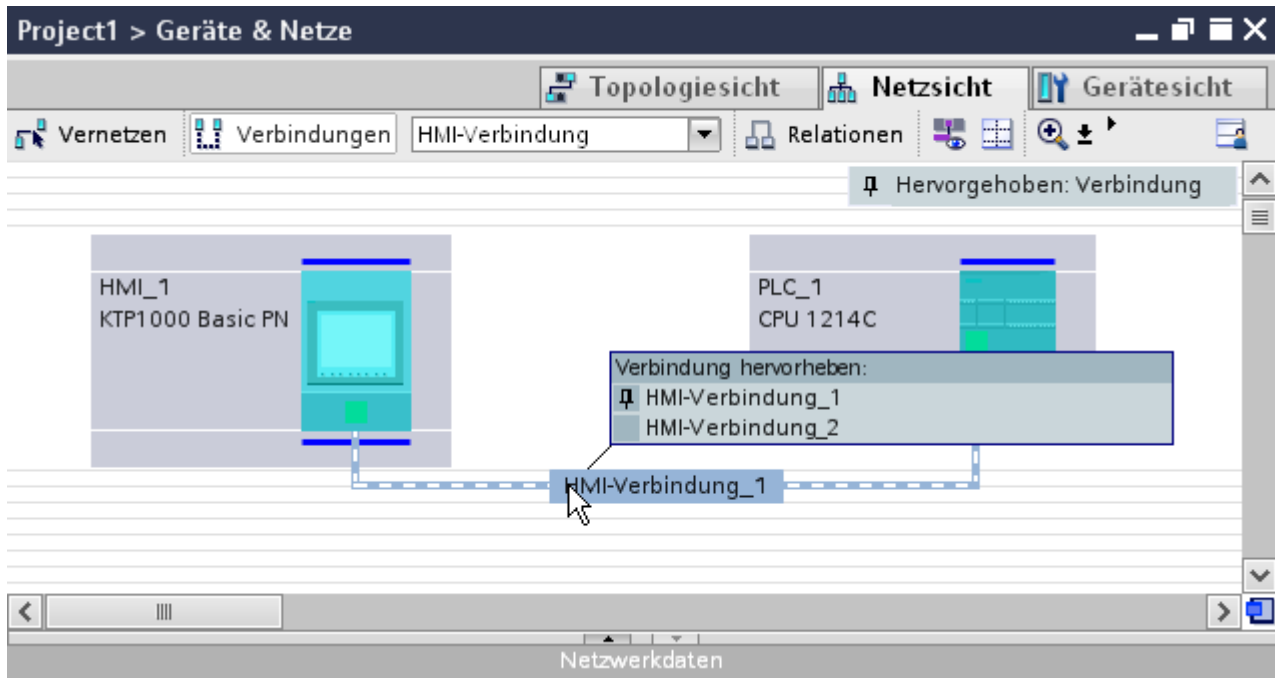
Eine Verbindung ist möglich.



Eine Verbindung ist nicht möglich.

## Hervorhebung von HMI-Verbindungen

Eine türkisfarbene Hervorhebung der Verbindung signalisiert, dass eine HMI-Verbindung angelegt wurde. Wenn mehrere HMI-Verbindungen angelegt sind, dann können Sie eine der bereits angelegten HMI-Verbindungen in einem Dialog auswählen.



Danach können Sie die Parameter der gewählten HMI-Verbindung und der Kommunikationspartner im Inspektorfenster projektieren.

## Subnetze

Subnetze werden bei folgenden Voraussetzungen automatisch angelegt oder verwendet:

- Wenn beide Kommunikationspartner nicht schon in unterschiedlichen Netzen bereits miteinander verbunden sind.
- Wenn beiden Kommunikationspartner eine freie Schnittstelle zur Verfügung steht.
- Wenn bereits ein Subnetz besteht, dann wird das bereits bestehende Subnetz für die HMI-Verbindung automatisch verwendet.

## Nicht integrierte Verbindung im Editor "Verbindungen" projektieren

### Einleitung

Eine Verbindung zwischen Bediengerät und einer Steuerung, die Sie nicht im Editor "Geräte & Netze" anlegen können, projektieren Sie im Editor "Verbindungen" des Bediengeräts.

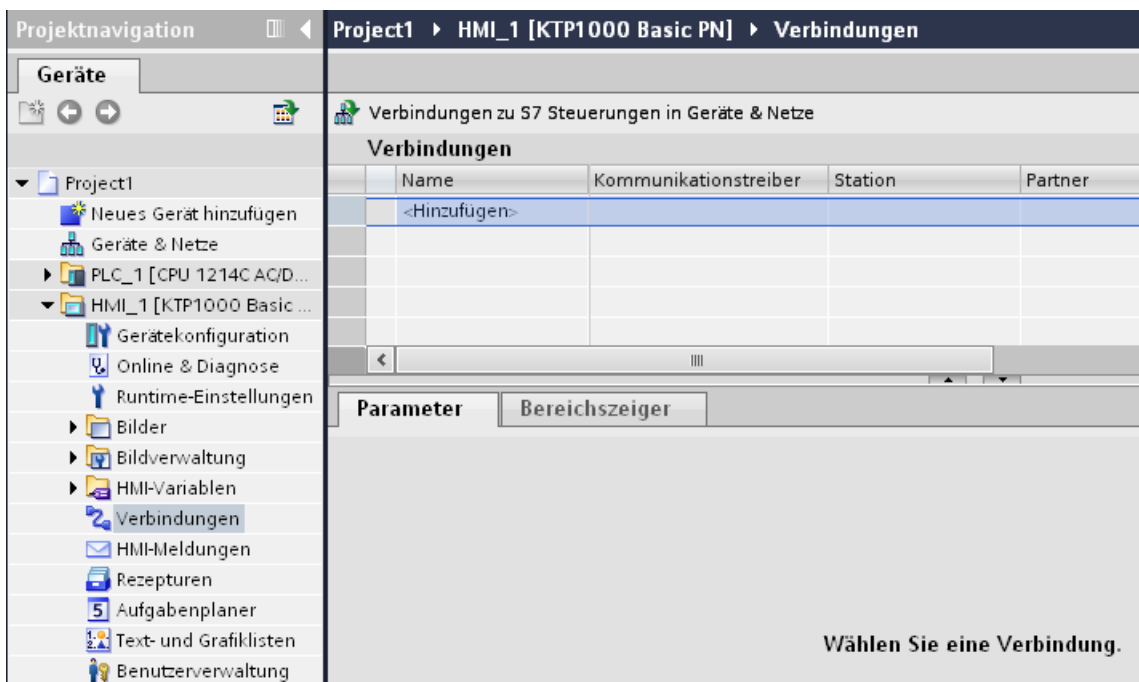
Diese Verbindungen werden als nicht integrierte Verbindungen bezeichnet.

### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

### Verbindung im Editor "Verbindungen" projektieren

1. Öffnen Sie den Editor "Verbindungen" des Bediengeräts.
2. Legen Sie eine neue Verbindung an.



3. Wählen Sie den "Kommunikationstreiber" aus.
4. Stellen Sie die Parameter der Verbindung ein.

The screenshot shows the 'Verbindungen' editor in WinCC Basic. The left pane displays the project structure, with 'HMI\_1 [KTP1000 Basic PN]' expanded. The main area features a table of connections and configuration panels for the selected connection.

| Name         | Kommunikationstreiber | Station | Partner | Knoten | Online |
|--------------|-----------------------|---------|---------|--------|--------|
| Verbindung_1 | SIMATIC S7 1200       |         |         |        |        |
| <Hinzufügen> |                       |         |         |        |        |

Configuration panels for 'KTP1000 Basic PN':

- Schnittstelle: PROFINET (X1)
- Bediengerät:
  - Adresse: 192 . 168 . 0 . 2
  - Zugangspunkt: S7ONLINE
- Steuerung:
  - Adresse: 192 . 168 . 0 . 2
  - Erweiterungssteckplat: 2
  - Baugruppenträger: 0

### Integrierte Verbindungen im Editor "Verbindungen"

Wenn Sie integrierte Verbindungen des Bediengeräts bereits im Editor "Geräte & Netze" projiziert haben, dann werden die integrierten Verbindungen ebenfalls im Editor "Verbindungen" angezeigt.

| Name             | Kommunikationstreiber | Station              | Partner | Knoten               | Online                              |
|------------------|-----------------------|----------------------|---------|----------------------|-------------------------------------|
| HMI-Verbindung_2 | SIMATIC S7 1200       | SIMATIC 1200-Station | PLC_1   | CPU 1214C AC/DC/R... | <input checked="" type="checkbox"/> |
| Verbindung_1     | SIMATIC S7 1200       |                      |         |                      | <input checked="" type="checkbox"/> |
| <Hinzufügen>     |                       |                      |         |                      |                                     |

Bedeutung der verwendeten Icons:



Integrierte Verbindung



Nicht integrierte Verbindung

## 10.8.4 Datenaustausch

### 10.8.4.1 Datenaustausch über Variablen

#### Grundlagen zu Variablen

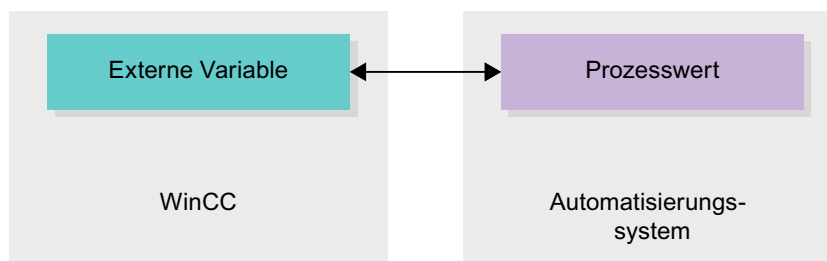
##### Einleitung

Prozesswerte werden in Runtime durch Variablen weitergegeben. Prozesswerte sind Daten, die im Speicher eines der angeschlossenen Automatisierungssysteme abgelegt werden. Sie stellen den Zustand einer Anlage dar, z. B. als Temperaturen, Füllstände oder Schaltzustände. Für die Verarbeitung der Prozesswerte in WinCC definieren Sie externe Variablen.

WinCC arbeitet mit zwei Arten von Variablen:

- Externe Variablen
- Interne Variablen

Die externen Variablen sind das Bindeglied zwischen WinCC und den Automatisierungssystemen. Die Werte von externen Variablen entsprechen den Prozesswerten aus dem Speicher eines Automatisierungssystems. Der Wert einer externen Variable wird ermittelt, indem der Prozesswert aus dem Speicher des Automatisierungssystems ausgelesen wird. Umgekehrt können Sie einen Prozesswert auch in den Speicher des Automatisierungssystems zurückschreiben.



Interne Variablen besitzen keine Prozessanbindung und transportieren nur Werte innerhalb von WinCC. Die Variablenwerte stehen nur so lange zur Verfügung wie Runtime läuft.

#### Variablen in WinCC

Bei externen Variablen wird in den Eigenschaften der jeweiligen Variable festgelegt, über welche Verbindung WinCC mit dem Automatisierungssystem kommuniziert und wie der Datenaustausch erfolgt.

Variablen ohne Werteverversorgung durch den Prozess, die so genannten internen Variablen, werden nicht mit dem Automatisierungssystem verbunden. In der Eigenschaft "Verbindung" der Variable erkennen Sie dies am Eintrag "Interne Variable".

Zur besseren Übersicht legen Sie die Variablen in verschiedenen Variablen Tabellen an. In der Projektnavigation greifen Sie dann im Knoten "HMI-Variablen" direkt auf die einzelnen Variablen Tabellen zu. Mithilfe der Tabelle "Zeige alle Variablen" lassen Sie sich die Variablen aus allen Variablen Tabellen anzeigen.

Mit Strukturen fassen Sie eine Anzahl verschiedener Variablen zusammen, die eine logische Einheit bilden. Strukturen sind projektzugehörige Daten und sind für alle HMI-Geräte des Projekts verfügbar. Für die Erstellung und Bearbeitung einer Struktur verwenden Sie den Editor "Typen" in der Projektbibliothek.

## Übersicht über HMI-Variablen tabellen

### Einleitung

HMI-Variablen tabellen enthalten die Definitionen der geräteweit gültigen HMI-Variablen. Für jedes im Projekt angelegte HMI-Gerät wird automatisch eine Variablen tabelle angelegt.

In der Projektnavigation gibt es für jedes HMI-Gerät einen Ordner "HMI-Variablen". Darin können folgende Tabellen enthalten sein:

- Standard-Variablen tabelle
- Anwenderdefinierte Variablen tabellen
- Tabelle aller Variablen

In der Projektnavigation können Sie im Ordner HMI-Variablen weitere Variablen tabellen anlegen, um Variablen und Konstanten darin zu sortieren und zu gruppieren. Die Variablen können Sie per Drag&Drop oder mithilfe des Felds "Variablen tabelle" in eine andere Variablen tabelle verschieben. Das Feld "Variablen tabelle" aktivieren Sie über das Kontextmenü der Spaltenüberschriften.

### Standard-Variablen tabelle

Für jedes HMI-Gerät des Projekts gibt es eine Standard-Variablen tabelle. Sie kann nicht gelöscht oder verschoben werden. Die Standard-Variablen tabelle enthält HMI-Variablen und abhängig vom HMI-Gerät auch Systemvariablen. Sie können alle HMI-Variablen in der Standard-Variablen tabelle deklarieren, oder nach Bedarf in weiteren, anwenderdefinierten Variablen tabellen.

### Anwenderdefinierte Variablen tabellen

Sie können mehrere anwenderdefinierte Variablen tabellen für jedes HMI-Gerät anlegen, um Variablen nach Ihren Bedürfnissen zu gruppieren. Anwenderdefinierte Variablen tabellen können Sie umbenennen, in Gruppen zusammenfassen oder löschen. Zur Gruppierung von Variablen tabellen legen Sie im Ordner HMI-Variablen weitere Unterordner an.

### Alle Variablen

Die Tabelle "Alle Variablen" zeigt eine Übersicht über alle HMI-Variablen und Systemvariablen des jeweiligen HMI-Geräts. Diese Tabelle kann nicht gelöscht, umbenannt oder verschoben werden. Diese Tabelle enthält zusätzlich die Spalte "Variablen tabelle", die anzeigt, in welcher Variablen tabelle eine Variable enthalten ist. Über das Feld "Variablen tabelle" können Sie die Zuordnung einer Variablen zu einer Variablen tabelle ändern.

Bei Geräten für Runtime Professional enthält die Tabelle "Alle Variablen" eine weitere Registerkarte "Systemvariablen". Die Systemvariablen werden vom System angelegt und für

die interne Verwaltung des Projekts benutzt. Die Namen der Systemvariablen beginnen mit dem Zeichen "@". Systemvariablen dürfen Sie weder löschen noch umbenennen. Den Wert einer Systemvariablen können Sie auswerten, jedoch nicht ändern.

## Zusätzliche Tabellen

In einer HMI-Variablen-Tabelle stehen Ihnen zusätzlich folgende Tabellen zur Verfügung:

- Bitmeldungen
- Analogmeldungen
- Archivvariablen

Mithilfe dieser Tabellen projektieren Sie Meldungen und Archivvariablen für die aktuell selektierte HMI-Variable.

## Tabelle Bitmeldungen

In der Tabelle "Bitmeldungen" projektieren Sie Bitmeldungen an die in der HMI-Variablen-Tabelle selektierte HMI-Variable. Wenn Sie eine Bitmeldung projektieren, ist eine Mehrfachselektion in der HMI-Variablen-Tabelle nicht möglich. Projektieren Sie die Bitmeldungen für jede HMI-Variable separat.

## Tabelle Analogmeldungen

In der Tabelle "Analogmeldungen" projektieren Sie Analogmeldungen an die in der HMI-Variablen-Tabelle selektierte HMI-Variable. Wenn Sie eine Analogmeldung projektieren, ist eine Mehrfachselektion in der HMI-Variablen-Tabelle nicht möglich. Projektieren Sie die Analogmeldungen für jede HMI-Variable separat.

## Tabelle Archivvariablen

In der Tabelle "Archivvariablen" projektieren Sie Archivvariablen an die in der HMI-Variablen-Tabelle selektierte HMI-Variable. Wenn Sie eine Archivvariable projektieren, ist eine Mehrfachselektion in der HMI-Variablen-Tabelle nicht möglich. Projektieren Sie die Archivvariablen für jede HMI-Variable separat. Die Tabelle "Archivvariablen" ist nur verfügbar, wenn das verwendete HMI-Gerät die Archivierung unterstützt.

Bei Verwendung von WinCC Runtime Professional können Sie einer Variablen auch mehrere Archivvariablen zuweisen. Bei den übrigen HMI-Geräten können Sie einer Variablen lediglich eine Archivvariable zuweisen.

## Externe Variablen

### Einleitung

Externe Variablen ermöglichen die Kommunikation, also den Datenaustausch, zwischen den Komponenten eines Automatisierungssystems, z. B. zwischen Bediengerät und Steuerung.



## Prinzip

Eine externe Variable ist das Abbild eines definierten Speicherplatzes in der Steuerung. Auf diesen Speicherplatz können Sie sowohl vom Bediengerät als auch von der Steuerung lesend und schreibend zugreifen.

Da externe Variablen das Abbild eines Speicherplatzes in der Steuerung sind, hängen die anwendbaren Datentypen von der Steuerung ab, mit der das Bediengerät verbunden ist.

Wenn Sie in STEP 7 ein PLC-Steuerungsprogramm schreiben, werden die im Steuerungsprogramm erstellten PLC-Variablen in die PLC-Variablen-Tabelle eingetragen. Wenn Sie eine externe Variable mit einer PLC-Variablen-Tabelle verbinden wollen, greifen Sie über die PLC-Variablen-Tabelle direkt auf die PLC-Variablen zu und verbinden sie mit der externen Variable.

## Datentypen

An einer externen Variable stehen Ihnen in WinCC alle Datentypen zur Verfügung, die an der verbundenen Steuerung verfügbar sind. Informationen über Datentypen, die bei einer Verbindung zu anderen Steuerungen zur Verfügung stehen, finden Sie in der Dokumentation zu den entsprechenden Kommunikationstreibern.

Nähere Hinweise hierzu finden Sie unter "Auto-Hotspot".

---

### Hinweis

Für die Kommunikation zwischen Bediengerät und Steuerung stehen Ihnen neben den externen Variablen auch Bereichszeiger zur Verfügung. Die Bereichszeiger können Sie im Editor "Verbindungen" einrichten und aktivieren.

---

## Aktualisierung von Variablenwerten

Bei externen Variablen werden in Runtime die aktuellen Variablenwerte über die Kommunikationsverbindung zwischen WinCC und den verbundenen Automatisierungssystemen übertragen und im Speicher der Runtime abgelegt. Danach wird der Variablenwert mit der eingestellten Zykluszeit aktualisiert. Für die Verwendung im Runtime-Projekt greift WinCC auf Variablenwerte im Speicher der Runtime zu, die zum vorherigen Zykluszeitpunkt aus der Steuerung gelesen wurden. Dadurch kann sich der Wert in der Steuerung bereits ändern, während der Wert aus dem Speicher der Runtime verarbeitet wird.


## Adressierung externer Variablen

### Einleitung

Die Möglichkeiten zur Adressierung externer Variablen sind abhängig von der Verbindungsart zwischen WinCC und der jeweiligen Steuerung. Folgende Verbindungsarten müssen Sie unterscheiden:

- **Integrierte Verbindung**  
Verbindungen von Geräten, die sich innerhalb eines Projekts befinden und mit dem Editor "Geräte & Netze" erstellt wurden, werden als integrierte Verbindungen bezeichnet.
- **Nicht integrierte Verbindung**  
Verbindungen von Geräten, die mit dem Editor "Verbindungen" erstellt wurden, werden als nicht integrierte Verbindungen bezeichnet. Die Geräte müssen sich nicht alle innerhalb eines Projekts befinden.

Die Art der Verbindung erkennen Sie auch an ihrem Symbol.

|   |                              |
|---|------------------------------|
|  | Integrierte Verbindung       |
|  | Nicht integrierte Verbindung |

Nähere Hinweise hierzu finden Sie im Kapitel Grundlagen zur Kommunikation (Seite 3441).

### Adressierung bei integrierter Verbindung

Eine integrierte Verbindung bietet den Vorteil, dass Sie eine Variable sowohl symbolisch als auch absolut adressieren können.

Bei der symbolischen Adressierung wählen Sie die PLC-Variable über ihren Namen aus und verbinden Sie mit der HMI-Variable. Der gültige Datentyp für die HMI-Variable wird vom System automatisch gewählt. Bei der Adressierung von Elementen in Datenbausteinen müssen Sie folgende Fälle unterscheiden:

#### **Symbolische Adressierung von Datenbausteinen mit optimiertem Zugriff und mit Standardzugriff:**

Bei der symbolischen Adressierung eines Datenbausteins mit optimiertem Zugriff und mit Standardzugriff wird die Adresse eines Elements im Datenbaustein dynamisch vergeben und bei einer Änderung automatisch in die HMI-Variable übernommen. Ein Übersetzen des verbundenen Datenbausteins oder des WinCC-Projekts ist dafür nicht notwendig.

Bei Datenbausteinen mit optimiertem Zugriff steht Ihnen nur die symbolische Adressierung zur Verfügung.

Bei der symbolischen Adressierung von Elementen eines Datenbausteins muss das WinCC-Projekt lediglich bei folgenden Änderungen neu übersetzt und geladen werden:

- Wenn sich der Name oder der Datentyp des verbundenen Datenbaustein-Elements oder der globalen PLC-Variable geändert hat.
- Wenn sich der Name oder der Datentyp eines im Datenbaustein-Element oder der globalen PLC-Variable übergeordneten Strukturknotens des verbundenen Elements geändert hat.
- Wenn sich die Nummer des verbundenen Datenbausteins geändert hat.

Die symbolische Adressierung steht Ihnen derzeit mit folgenden Steuerungen zur Verfügung:

- SIMATIC S7 1200
- SIMATIC S7 1500

Die symbolische Adressierung steht Ihnen außerdem bei einer integrierten Verbindung zur Verfügung.

---

**Hinweis**

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" verwenden, dann ist eine symbolische Adressierung nicht möglich.

---

Die absolute Adressierung können Sie auch bei einer integrierten Verbindung verwenden. PLC-Variablen aus einer SIMATIC S7 300/400 Steuerung können Sie nur absolut adressieren. Wenn Sie eine HMI-Variable mit einer PLC-Variable verbunden haben, und sich die Adresse der PLC-Variablen ändert, müssen Sie das Steuerungsprogramm neu übersetzen, damit die neue Adresse in WinCC aktualisiert wird. Übersetzen Sie anschließend auch das WinCC-Projekt neu und laden Sie es auf das Bediengerät.

In WinCC ist die symbolische Adressierung als Standard voreingestellt. Um die Standardeinstellung zu ändern, wählen Sie den Menübefehl "Extras > Einstellungen". Im Dialog "Einstellungen" wählen Sie "Visualisierung > Variablen". Deaktivieren Sie bei Bedarf die Option "Symbolischer Zugriff".

Die Verfügbarkeit einer integrierten Verbindung ist abhängig von der verwendeten Steuerung. Die folgende Tabelle zeigt die Verfügbarkeit:

| Steuerung  | Integrierte Verbindung | Bemerkungen   |
|------------|------------------------|---|
| S7 300/400 | Ja                     | Die Verknüpfung der Variablen wird in Runtime nicht geprüft. Wenn sich in der Steuerung die Variablenadresse ändert, und das HMI-Gerät nicht neu übersetzt und geladen wurde, wird die Änderung in Runtime nicht registriert.   |
| S7 1200    | Ja                     | Bei der symbolischen Adressierung erfolgt eine Gültigkeitsprüfung der Variablenverbindung in Runtime. Bei einer Adressänderung in der Steuerung wird die Änderung registriert und eine Fehlermeldung ausgegeben. Bei der absoluten Adressierung gilt das für die S7 300/400 beschriebene Verhalten. |
| S7 1500    | Ja                     | Bei der symbolischen Adressierung erfolgt eine Gültigkeitsprüfung der Variablenverbindung in Runtime. Bei einer Adressänderung in der Steuerung wird die Änderung registriert und eine Fehlermeldung ausgegeben. Bei der absoluten Adressierung gilt das für die S7 300/400 beschriebene Verhalten. |

Eine integrierte Verbindung erstellen Sie im Editor "Geräte & Netze". Wenn die Steuerung im Projekt enthalten ist und integrierte Verbindungen unterstützt, dann können Sie sich die Verbindung auch automatisch erstellen lassen. Wählen Sie dazu bei der Projektierung der HMI-Variable einfach eine bestehende PLC-Variable aus, mit der Sie die HMI-Variable verbinden wollen. Die integrierte Verbindung wird dann vom System automatisch erstellt.

### Adressierung bei nicht integrierter Verbindung

Bei einem Projekt mit nicht integrierter Verbindung projektieren Sie eine Variablenverbindung immer mit absoluter Adressierung. Den gültigen Datentyp müssen Sie selbst wählen. Wenn sich bei einem Projekt mit nicht integrierter Verbindung im Projektverlauf die Adresse einer PLC-Variable ändert, müssen Sie die Änderung in WinCC ebenfalls ändern. Die Variablenverbindung kann in Runtime nicht auf Gültigkeit geprüft werden, es erfolgt keine Fehlermeldung.

Eine nicht integrierte Verbindung ist für alle unterstützten Steuerungen verfügbar.

Die symbolische Adressierung ist bei nicht integrierter Verbindung nicht verfügbar.

Bei einer nicht integrierten Verbindung muss das Steuerungsprogramm nicht Bestandteil des WinCC Projekts sein. Die Projektierung der Steuerung und des WinCC-Projekts können Sie unabhängig voneinander durchführen. Für die Projektierung in WinCC müssen lediglich die verwendeten Adressen in der Steuerung und deren Funktion bekannt sein.

### Siehe auch

Grundlagen zur Kommunikation (Seite 3441)

### Interne Variablen

#### Einleitung

Interne Variablen haben keine Verbindung zur Steuerung. Interne Variablen transportieren Werte innerhalb des Bediengeräts. Die Variablenwerte stehen nur so lange zur Verfügung wie Runtime läuft.

#### Prinzip

Interne Variablen werden im Speicher des Bediengeräts abgelegt. Daher kann nur dieses Bediengerät lesend und schreibend auf die internen Variablen zugreifen. Interne Variablen legen Sie z. B. an, um lokale Berechnungen durchzuführen.

Für interne Variablen stehen Ihnen die HMI-Datentypen zur Verfügung. Die Verfügbarkeit ist abhängig vom verwendeten Bediengerät.

Folgende HMI-Datentypen sind verfügbar:

| HMI-Datentyp | Datenformat                      |
|--------------|----------------------------------|
| ARRAY        | eindimensionales Array           |
| BOOL         | Binäre Variable                  |
| DATETIME     | Datum/ Uhrzeit Format            |
| DINT         | Vorzeichenbehafteter 32-Bit Wert |
| INT          | Vorzeichenbehafteter 16-Bit Wert |
| LREAL        | Gleitkommazahl 64-Bit IEEE 754   |
| REAL         | Gleitkommazahl 32-Bit IEEE 754   |
| SINT         | Vorzeichenbehafteter 8-Bit Wert  |

| HMI-Datentyp | Datenformat                     |
|--------------|---------------------------------|
| UDINT        | Vorzeichenloser 32-Bit Wert     |
| UINT         | Vorzeichenloser 16-Bit Wert     |
| USINT        | Vorzeichenloser 8-Bit Wert      |
| WSTRING      | Textvariable 16-Bit Zeichensatz |

## 10.8.4.2 Datenaustausch über Bereichszeiger

### Grundlagen zu Bereichszeigern

#### Einleitung

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu. In diese Datenbereiche wird während der Kommunikation wechselseitig von der Steuerung und dem Bediengerät geschrieben und gelesen. Durch die Auswertung der abgelegten Daten lösen Steuerung und Bediengerät gegenseitig fest definierte Aktionen aus.

Bereichszeiger sind z. B. bei folgenden Daten erforderlich:

- Rezepturen
- Steuerungsaufträgen
- Lebenszeichenüberwachung

#### Bereichszeiger

Folgende Bereichszeiger werden unterstützt:

#### Bereichszeiger

Bereichszeiger sind für Verbindungen projektierbar.

- Datensatz
- Datum/Uhrzeit
- Koordinierung
- Steuerungsauftrag

#### Globale Bereichszeiger des Bediengeräts

Globale Bereichszeiger sind für getrennte Verbindung projektierbar.

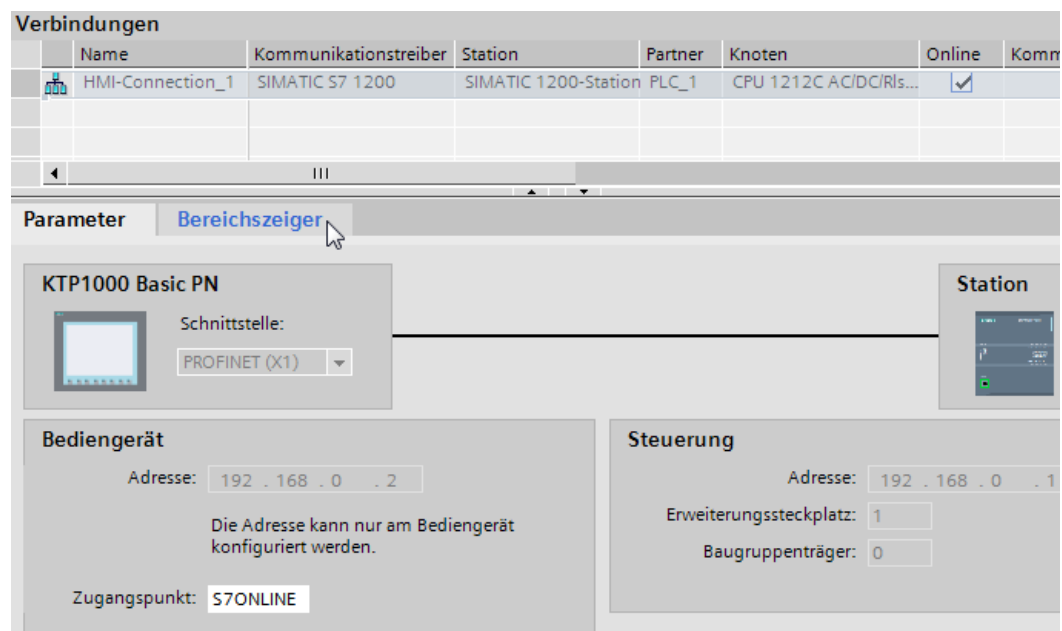
- Bildnummer
- Datum/Uhrzeit PLC
- Projektkennung

## Bereichszeiger für Verbindungen

### Einleitung

Im Editor "Verbindungen" projektieren Sie auf der Registerkarte "Bereichszeiger" die Verwendung der verfügbaren Bereichszeiger und deren Einstellungen.

Zur Projektierung der Bereichszeiger öffnen Sie den Editor "Verbindungen" und aktivieren die Registerkarte "Bereichszeiger".



### Aufbau

Die Registerkarte "Bereichszeiger" enthält zwei Tabellen mit Bereichszeigern. Die obere Tabelle enthält die Bereichszeiger, die für jede vorhandene Verbindung separat angelegt und aktiviert werden können.

Die Tabelle "Globale Bereichszeiger des Bediengeräts" enthält die Bereichszeiger, die im Projekt nur einmal angelegt und nur für eine Verbindung benutzt werden können.

| Parameter                |                   | Bereichszeiger |                      |         |       |                       |                  |         |  |
|--------------------------|-------------------|----------------|----------------------|---------|-------|-----------------------|------------------|---------|--|
| Aktiv                    | Anzeigename       | PLC-Variab     | Zugriffsart          | Adresse | Länge | Erfassungsart         | Erfassungszyklus | Komment |  |
| <input type="checkbox"/> | Koordinierung     | <Undefiniert>  | <Symbolischer Zug... |         | 1     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <input type="checkbox"/> | Datum/Uhrzeit     | <Undefiniert>  | <Symbolischer Zug... |         | 6     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <input type="checkbox"/> | Steuerungsauftrag | <Undefiniert>  | <Symbolischer Zug... |         | 4     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <input type="checkbox"/> | Datensatz         | <Undefiniert>  | <Symbolischer Zug... |         | 5     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |

| Globale Bereichszeiger des Bediengeräts |                   |               |                     |         |       |                       |                  |         |  |
|---|-------------------|---------------|---------------------|---------|-------|-----------------------|------------------|---------|--|
| Verbindung                              | Anzeigename       | PLC-Variab    | Zugriffsart         | Adresse | Länge | Erfassungsart         | Erfassungszyklus | Komment |  |
| <Undefiniert> ...                       | Projektkennung    | <Undefiniert> | <Symbolischer Zu... |         | 1     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <Undefiniert>                           | Bildnummer        | <Undefiniert> | <Symbolischer Zu... |         | 5     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <Undefiniert>                           | Datum/Uhrzeit PLC | <Undefiniert> | <Symbolischer Zu... |         | 6     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |

## Verwendung von Bereichszeigern

### Registerkarte "Bereichszeiger"

Bevor Sie den Bereichszeiger verwenden, aktivieren Sie die Bereichszeiger unter "Verbindungen > Bereichszeiger". Danach parametrieren Sie die Bereichszeiger.

In der Registerkarte "Bereichszeiger" parametrieren Sie Folgendes:

| Parameter                |                   | Bereichszeiger |                      |         |       |                       |                  |         |  |
|--------------------------|-------------------|----------------|----------------------|---------|-------|-----------------------|------------------|---------|--|
| Aktiv                    | Anzeigename       | PLC-Variab     | Zugriffsart          | Adresse | Länge | Erfassungsart         | Erfassungszyklus | Komment |  |
| <input type="checkbox"/> | Koordinierung     | <Undefiniert>  | <Symbolischer Zug... |         | 1     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <input type="checkbox"/> | Datum/Uhrzeit     | <Undefiniert>  | <Symbolischer Zug... |         | 6     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <input type="checkbox"/> | Steuerungsauftrag | <Undefiniert>  | <Symbolischer Zug... |         | 4     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <input type="checkbox"/> | Datensatz         | <Undefiniert>  | <Symbolischer Zug... |         | 5     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |

| Globale Bereichszeiger des Bediengeräts |                   |               |                     |         |       |                       |                  |         |  |
|---|-------------------|---------------|---------------------|---------|-------|-----------------------|------------------|---------|--|
| Verbindung                              | Anzeigename       | PLC-Variab    | Zugriffsart         | Adresse | Länge | Erfassungsart         | Erfassungszyklus | Komment |  |
| <Undefiniert> ...                       | Projektkennung    | <Undefiniert> | <Symbolischer Zu... |         | 1     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <Undefiniert>                           | Bildnummer        | <Undefiniert> | <Symbolischer Zu... |         | 5     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |
| <Undefiniert>                           | Datum/Uhrzeit PLC | <Undefiniert> | <Symbolischer Zu... |         | 6     | Zyklisch fortlaufe... | <Undefiniert>    |         |  |

- **Aktiv**  
Aktiviert den Bereichszeiger.
- **Anzeigename**  
Name des Bereichszeigers, der von WinCC vorgegeben wird.
- **PLC-Variab**  
Hier wählen Sie die PLC-Variab oder das Variablenarray, das Sie als Datenbereich für den Bereichszeiger projiziert haben.

- Zugriffsart  
Hier wählen Sie zwischen folgenden Zugriffsarten aus:
  - Symbolischer Zugriff
  - Absoluter Zugriff
- Adresse  
Wenn Sie "Symbolischer Zugriff" ausgewählt haben wird in diesem Feld keine Adresse ausgegeben.  
Wenn Sie "Absoluter Zugriff" ausgewählt haben, dann geben Sie im Feld "Adresse" die Adresse einer Variable ein.
- Länge  
Die Länge des Bereichszeigers wird von WinCC vorgegeben.
- Erfassungszyklus  
Für Bereichszeiger, die vom Bediengerät gelesen werden, legen Sie in diesem Feld den Erfassungszyklus fest. Berücksichtigen Sie, dass eine sehr kurze Erfassungszeit die Performance des Bediengeräts beeinflussen kann.
- Kommentar  
Hinterlegen Sie einen Kommentar, z. B. für die Verwendung des Bereichszeigers.

## Zugriff auf Datenbereiche

### Zugriff auf Datenbereiche

Die folgende Tabelle zeigt, wie Bediengerät und Steuerung auf einzelne Datenbereiche zugreifen, lesend (R) oder schreibend (W).

| Datenbereich      | Erforderlich für   | Bediengerät | Steuerung |
|-------------------|--|-------------|-----------|
| Bildnummer        | Auswertung von der Steuerung, welches Bild im Moment aufgeschlagen ist.                          | W           | R         |
| Datensatz         | Übertragung von Datensätzen mit Synchronisation  | R/W         | R/W       |
| Datum/Uhrzeit     | Übertragung von Datum und Uhrzeit vom Bediengerät zur Steuerung                                  | W           | R         |
| Datum/Uhrzeit PLC | Übertragung von Datum und Uhrzeit von der Steuerung zum Bediengerät                              | R           | W         |
| Koordinierung     | Status des Bediengeräts im Steuerungsprogramm abfragen   | W           | R         |
| Projektkennung    | Runtime überprüft, ob die WinCC Projektkennung und das Projekt in der Steuerung konsistent sind. | R           | W         |
| Steuerungsauftrag | Auslösen von Funktionen am Bediengerät durch das Steuerungsprogramm                              | R/W         | R/W       |



## Bereichszeiger projektieren

### Projektierung von Bereichszeigern

#### Einleitung

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu. Der Datenbereich wird in der Steuerung abgelegt.

#### Vor der Projektierung von Bereichszeigern

Bevor Sie den Bereichszeiger verwenden, muss dieser unter "Verbindungen > Bereichszeiger" aktiviert und parametrieren werden.

#### Globaler Datenbaustein

Um auf den Datenbereich in der Steuerung zuzugreifen, können Sie im Steuerungsprogramm einen globalen Datenbaustein anlegen. Das nachfolgende Beispiel zeigt die Verwendung eines Datenbausteins.

#### Länge von Bereichszeigern

Für Bereichszeiger mit einer Länge  $\geq 1$  legen Sie den Datenbereich als Variablenarray in einem globalen Datenbaustein oder einem Instanz-Datenbaustein an.

Für Bereichszeiger mit der Länge = 1 können Sie alternativ auch eine PLC-Variable verwenden.

Die Projektierung der Variablen in einem Datenbaustein ist abhängig von der Länge des Bereichszeigers, den Sie verwenden wollen. Die Einheit der Länge eines Bereichszeigers ist ein 16-Bit-Wort.

Wenn Sie z. B. einen Bereichszeiger mit der Länge "5" verwenden wollen, müssen Sie in dem Datenbaustein ein Array mit 5 Arrayelementen vom Datentyp UINT anlegen.

#### Alternative Vorgehensweise

Alternativ können Sie auch über die Absolute Zugriffsart verwenden, um auf Bereichszeiger zuzugreifen. Die absolute Zugriffsart funktioniert nur auf Standard-Datenbausteine der Steuerung.

### Globalen Datenbaustein parametrieren

#### Einleitung

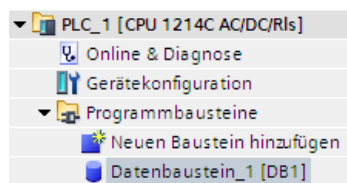
Um auf den Datenbereich in der Steuerung zuzugreifen, muss im Steuerungsprogramm ein globaler Datenbaustein für den Bereichszeiger parametrieren werden.

### Voraussetzungen

- Im Projekt ist eine Steuerung angelegt.
- Zwischen Steuerung und HMI-Gerät ist eine Verbindung projektiert.
- Das Steuerungsprogramm enthält einen globalen Datenbaustein.

### Vorgehensweise

1. Öffnen Sie in der Projektnavigation "PLC > Programmbausteine".
2. Doppelklicken Sie den bereits angelegten globalen Datenbaustein. Der Datenbaustein wird geöffnet.



3. Tragen Sie in der Spalte "Name" einen Variablennamen ein.
4. Wählen Sie in der Spalte "Datentyp" den Datentyp "Array[lo .. hi] of type".
5. Ersetzen Sie den Eintrag "lo" durch den unteren Wert für die Dimension des Arrays.
6. Ersetzen Sie den Eintrag "hi" durch den oberen Wert für die Dimension des Arrays. Beispiel: Wenn Sie einen Bereichszeiger mit der Länge "4" projektieren, geben Sie in der Klammer für "lo" den Wert "0" und für "hi" den Wert "3" ein.
7. Ersetzen Sie den Begriff "type" durch den Datentyp "word". Der vollständige Datentyp für ein Array von 4 Variablen lautet wie folgt: "Array[0 .. 3] of word". Nach Bestätigung der Eingabe wird das Variablenarray angelegt.
8. Klicken Sie auf "Übersetzen". Das Projekt wird kompiliert.



| Datenbaustein_1 |                      |                        |             |             |                          |           |
|-----------------|----------------------|------------------------|-------------|-------------|--------------------------|-----------|
|                 | Name                 | Datentyp               | Defaultwert | Anfangswert | Remanenz                 | Kommentar |
| 1               | Static               |                        |             |             | <input type="checkbox"/> |           |
| 2               | Steuerungsauftrag    | Array [0 .. 3] of word |             |             | <input type="checkbox"/> |           |
| 3               | Steuerungsauftrag[0] | Word                   | W#16#0000   | W#16#0000   | <input type="checkbox"/> |           |
| 4               | Steuerungsauftrag[1] | Word                   | W#16#0000   | W#16#0000   | <input type="checkbox"/> |           |
| 5               | Steuerungsauftrag[2] | Word                   | W#16#0000   | W#16#0000   | <input type="checkbox"/> |           |
| 6               | Steuerungsauftrag[3] | Word                   | W#16#0000   | W#16#0000   | <input type="checkbox"/> |           |

### Bereichszeiger für Verbindung projektieren

#### Einleitung

Nachdem Sie den globalen Datenbaustein parametrieren haben, legen Sie jetzt den Bereichszeiger für die Verbindung an.

## Voraussetzungen

- Der globale Datenbaustein wurde im Steuerungsprogramm parametrier.

## Vorgehensweise

1. Öffnen Sie in der Projektnavigation "HMI > Verbindungen".
2. Klicken Sie auf die Registerkarte "Bereichszeiger".
3. Aktivieren Sie den gewünschten Bereichszeiger.  
Einen globalen Bereichszeiger aktivieren Sie durch die Auswahl der Verbindung im Feld "Verbindung".
4. Klicken Sie im Feld "Steuerungsvariable" auf die Navigationsschaltfläche.  
Die Objektliste wird geöffnet.
5. Navigieren Sie in der Objektliste zu dem Datenbaustein und wählen Sie im rechten Fenster die Variable aus.  
Für die Projektierung eines Bereichszeigers der Länge "1" benötigen Sie keine Arrayvariable.

| Parameter                           |                   | Bereichszeiger                 |                   |       |                      |  |
|-------------------------------------|-------------------|--------------------------------|-------------------|-------|----------------------|--|
| Aktiv                               | Anzeigename       | PLC Variable                   | Adresse           | Länge | Erfassungsart        |  |
| <input type="checkbox"/>            | Datensatz         |                                |                   | 5     | Zyklisch fortlaufend |  |
| <input type="checkbox"/>            | Datum/Uhrzeit     |                                |                   | 6     | Zyklisch fortlaufend |  |
| <input type="checkbox"/>            | Koordinierung     |                                |                   | 1     | Zyklisch fortlaufend |  |
| <input checked="" type="checkbox"/> | Steuerungsauftrag | Datenbaustein_1.Steuerun...... | <symbolic access> | 4     | Zyklisch fortlaufend |  |

| Globale Bereichszeiger des Bediengeräts |                   |           |         |       |                      |                  |
|---|-------------------|-----------|---------|-------|----------------------|------------------|
| Verbindung                              | Anzeigename       | PLC Var.. | Adresse | Länge | Erfassungsart        | Erfassungszyklus |
| <Undefiniert>...                        | Bildnummer        |           |         | 5     | Zyklisch fortlaufend | <Undefiniert>    |
| <Undefiniert>                           | Datum/Uhrzeit PLC |           |         | 6     | Zyklisch fortlaufend | <Undefiniert>    |
| <Undefiniert>                           | Projektkennung    |           |         | 1     | Zyklisch fortlaufend | <Undefiniert>    |

6. Wählen Sie beim Anlegen der Variable im Datenbaustein den Datentyp "Word".  
Bei Bedarf ändern Sie während der Projektierung weitere Parameter, z. B. den Erfassungszyklus.

## Ergebnis

Der Bereichszeiger ist aktiviert und mit der Steuerungsvariable im globalen Datenbaustein verbunden.

### 10.8.5 Geräteabhängigkeit

#### 10.8.5.1 Basic Panel

#### Kommunikationstreiber für Basic Panels

#### Geräteabhängigkeit der Basic Panels

Die nachfolgende Tabelle zeigt, welche Kommunikationstreiber Sie mit den verschiedenen Basic Panels projektieren können.

#### Kommunikationstreiber

| Bedien-<br>gerät           | SIMA<br>TIC<br>S7<br>1500 | SIMA<br>TIC<br>S7<br>1200 | SIMA<br>TIC<br>S7<br>300/4<br>00 | SIMA<br>TIC<br>S7<br>200 | SIMA<br>TIC<br>LOG<br>OI | SIMA<br>TIC<br>HTTP<br>Proto-<br>col | OPC  | Allen-<br>Bradl-<br>ey<br>Ether-<br>Net/IP | Allen-<br>Bradl-<br>ey<br>DF1 | Mitsu-<br>bishi<br>MC<br>TCP/<br>IP | Mitsu-<br>bishi<br>FX | Modic-<br>on<br>Modb-<br>us<br>TCP/<br>IP | Modic-<br>on<br>Modb-<br>us<br>RTU | Omro-<br>n<br>Host<br>Link |
|----------------------------|---------------------------|---------------------------|----------------------------------|--------------------------|--------------------------|--------------------------------------|------|--|-------------------------------|-------------------------------------|-----------------------|---|------------------------------------|----------------------------|
| KP30<br>0<br>Basic         | ja                        | ja                        | ja                               | ja                       | ja                       | nein                                 | nein | ja   | nein                          | ja                                  | nein                  | ja  | nein                               | nein                       |
| KP40<br>0<br>Basic         | ja                        | ja                        | ja                               | ja                       | ja                       | nein                                 | nein | ja   | nein                          | ja                                  | nein                  | ja  | nein                               | nein                       |
| KTP4<br>00<br>Basic<br>PN  | ja                        | ja                        | ja                               | ja                       | ja                       | nein                                 | nein | ja   | nein                          | ja                                  | nein                  | ja  | nein                               | nein                       |
| KTP6<br>00<br>Basic<br>DP  | ja                        | ja                        | ja                               | ja                       | ja                       | nein                                 | nein | nein                                       | ja <sup>2)</sup>              | nein                                | ja                    | nein                                      | ja <sup>1)</sup>                   | ja                         |
| KTP6<br>00<br>Basic<br>PN  | ja                        | ja                        | ja                               | ja                       | ja                       | nein                                 | nein | ja   | nein                          | ja                                  | nein                  | ja  | nein                               | nein                       |
| KTP1<br>000<br>Basic<br>DP | ja                        | ja                        | ja                               | ja                       | ja                       | nein                                 | nein | nein                                       | ja <sup>2)</sup>              | nein                                | ja                    | nein                                      | ja <sup>1)</sup>                   | ja                         |

| Bediengerät      | SIMA TIC S7 1500 | SIMA TIC S7 1200 | SIMA TIC S7 300/400 | SIMA TIC S7 200 | SIMA TIC LOG O! | SIMA TIC HTTP Protocol | OPC  | Allen-Bradley EtherNet/IP | Allen-Bradley DF1 | Mitsubishi MC TCP/IP | Mitsubishi FX | Modicon Modbus TCP/IP | Modicon Modbus RTU | Omron Host Link |
|------------------|------------------|------------------|---------------------|-----------------|-----------------|------------------------|------|---------------------------|-------------------|----------------------|---------------|-----------------------|--------------------|-----------------|
| KTP1000 Basic PN | ja               | ja               | ja                  | ja              | ja              | nein                   | nein | ja                        | nein              | ja                   | nein          | ja                    | nein               | nein            |
| TP1500 Basic PN  | ja               | ja               | ja                  | ja              | ja              | nein                   | nein | ja                        | nein              | ja                   | nein          | ja                    | nein               | nein            |

- 1) nur mit Konverter RS 422-RS232  
Bestellnummer: 6AV6 671-8XE00-0AX0
- 2) Direkte Kommunikation mit PLC 5 oder KF2-Modul, ansonsten nur mit Konverter RS422-RS232 (Option) freigegeben.  
Bestellnummer: 6AV6 671-8XE00-0AX0

### Schnittstellen der Basic Panels

### Geräteabhängigkeit der Basic Panels

Die nachfolgende Tabelle zeigt, welche Schnittstellen am Bediengerät für die Protokolle der Kommunikationstreiber zur Verfügung stehen.

Tabelle 10-12 Basic Panels

|                                | KP300 Basic<br>KP400 Basic<br>KTP400 Basic PN<br>KTP600 Basic PN<br>KTP1000 Basic PN<br>TP1500 Basic PN | KTP600 Basic DP<br>KTP1000 Basic DP |
|--------------------------------|---|-------------------------------------|
| SIMATIC S7 - PPI <sup>1)</sup> | —   | MPI/DP (X2)                         |
| SIMATIC S7 - MPI               | —   | MPI/DP (X2)                         |
| SIMATIC S7 - PROFIBUS          | —   | MPI/DP (X2)                         |
| SIMATIC S7 - PROFINET          | PROFINET (X1)   | —                                   |
| SIMATIC HMI HTTP Protokoll     | —   | —                                   |
| OPC                            | —   | —                                   |
| Allen-Bradley EtherNet/IP      | PROFINET (X1)   | —                                   |
| Allen-Bradley DF1              | —   | MPI/DP (X2) <sup>2)</sup>           |
| Mitsubishi TCP/IP              | PROFINET (X1)   | —                                   |

|                    | KP300 Basic<br>KP400 Basic<br>KTP400 Basic PN<br>KTP600 Basic PN<br>KTP1000 Basic PN<br>TP1500 Basic PN | KTP600 Basic DP<br>KTP1000 Basic DP |
|--------------------|---|-------------------------------------|
| Mitsubishi FX      | —   | MPI/DP (X2) (RS422)                 |
| Modicon Modbus TCP | PROFINET (X1)   | —                                   |
| Modicon Modbus RTU | —   | MPI/DP (X2) <sup>3)</sup>           |
| Omron Host Link    | —   | MPI/DP (X2) (RS422)                 |

- 1) nur an SIMATIC S7-200
- 2) Direkte Kommunikation mit PLC5 oder KF2-Modul, ansonsten nur mit Konverter RS422-RS232(Optional) freigegeben.  
Bestellnummer: 6AV6 671-8XE00-0AX0
- 3) nur mit Konverter RS 422-RS232 freigegeben  
Bestellnummer: 6AV6 671-8XE00-0AX0

## Bereichszeiger für Basic Panels

### Einleitung

Bereichszeiger sind Parameterfelder, aus denen das Bediengerät die Informationen über die Lage und Größe von Datenbereichen in der Steuerung erhält. In diese Datenbereiche wird während der Kommunikation wechselseitig von der Steuerung und dem Bediengerät geschrieben und gelesen. Durch Auswertung der in den Datenbereichen abgelegten Daten lösen Steuerung und Bediengerät gegenseitig fest definierte Aktionen aus.

WinCC verwendet folgende Bereichszeiger:

- Steuerungsauftrag
- Projektkennung
- Bildnummer
- Datensatz
- Datum/Uhrzeit
- Datum/Uhrzeit PLC
- Koordinierung

### Verfügbarkeit der Bereichszeiger

Die nachfolgende Tabelle zeigt die Verfügbarkeit der Bereichszeiger auf den Bediengeräten. Beachten Sie, dass die Bereichszeiger nur bei verfügbaren Kommunikationstreibern eingesetzt werden können.

## Bereichszeiger

|                      | KP300 Basic | KTP400<br>Basic PN | KTP600<br>Basic PN | KTP600<br>Basic DP | KTP1000<br>Basic PN | KTP1000<br>Basic DP | TP1500<br>Basic PN |
|----------------------|-------------|--------------------|--------------------|--------------------|---------------------|---------------------|--------------------|
| Bildnummer           | ja          | ja                 | ja                 | ja                 | ja                  | ja                  | ja                 |
| Datensatz            | ja          | ja                 | ja                 | ja                 | ja                  | ja                  | ja                 |
| Datum/Uhrzeit        | ja          | ja                 | ja                 | ja                 | ja                  | ja                  | ja                 |
| Datum/Uhrzeit<br>PLC | ja          | ja                 | ja                 | ja                 | ja                  | ja                  | ja                 |
| Koordinierung        | ja          | ja                 | ja                 | ja                 | ja                  | ja                  | ja                 |
| Projektkennung       | ja          | ja                 | ja                 | ja                 | ja                  | ja                  | ja                 |
| Steuerungsauftrag    | ja          | ja                 | ja                 | ja                 | ja                  | ja                  | ja                 |

## 10.8.6 Mit SIMATIC S7 1500 kommunizieren

### 10.8.6.1 Kommunikation mit SIMATIC S7 1500

#### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und der Steuerung SIMATIC S7 1500 beschrieben.

Folgende Kommunikationskanäle können Sie für die Steuerung SIMATIC S7 1500 projektieren:

- PROFINET
- PROFIBUS

#### HMI-Verbindung für die Kommunikation

Verbindungen von Bediengerät und SIMATIC S7 1500 projektieren Sie im Editor "Geräte & Netze".

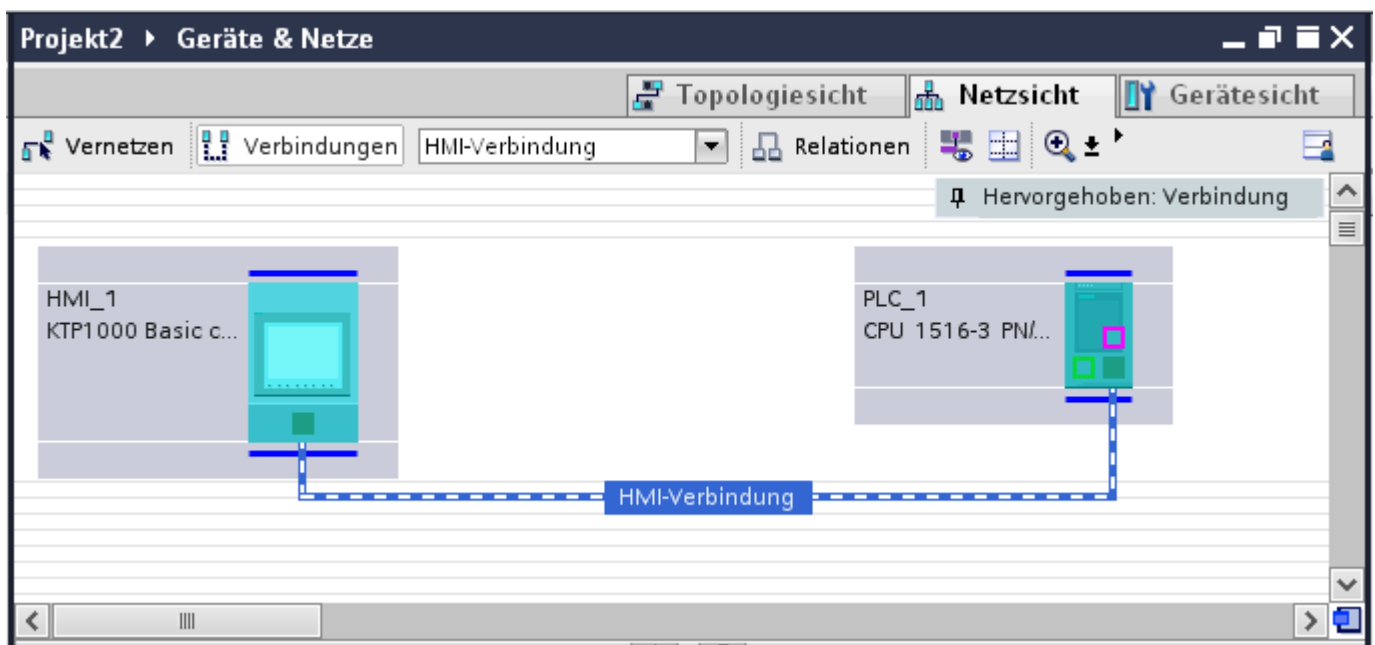
## 10.8.6.2 Kommunikation über PROFINET

### HMI-Verbindung projektieren

#### Kommunikation über PROFINET

#### HMI-Verbindungen über PROFINET

Wenn Sie ein Bediengerät und eine SIMATIC S7 1500 in das Projekt eingefügt haben, verbinden Sie die beiden PROFINET-Schnittstellen im Editor "Geräte & Netze" miteinander.



Sie können auch mehrere Bediengeräte an eine SIMATIC S7 1500 anschließen und mehrere SIMATIC S7 1500 an ein Bediengerät anschließen.

Die maximale Anzahl der Kommunikationspartner, die Sie an ein Bediengerät anschließen können, ist abhängig vom verwendeten Bediengerät.

Weitere Informationen dazu finden Sie in der Dokumentation zum jeweiligen Bediengerät.

#### HMI-Verbindung im Editor "Geräte & Netze"

Die HMI-Verbindung über PROFINET zwischen der Steuerung mit dem Bediengerät projektieren Sie im Editor "Geräte & Netze".

#### Verbindung im Editor "Verbindungen"


Alternativ projektieren Sie die Verbindung über PROFINET zwischen Steuerung und Bediengerät im Editor "Verbindungen" des Bediengeräts.



## HMI-Verbindung über PROFINET projektieren

### Einleitung

Eine HMI-Verbindung über PROFINET bzw. Ethernet zwischen Bediengeräten und einer SIMATIC S7 1500 projektieren Sie im Editor "Geräte & Netze".

|   |
|---|
|  <b>VORSICHT</b>                     |
| <b>Kommunikation über Ethernet</b>  |
| Bei Ethernet-basierter Kommunikation ist der Endanwender für die Sicherheit seines Datennetzes selbst verantwortlich. |
| Gezielte Angriffe können zur Überlast des Geräts führen und die Funktionsfähigkeit beeinträchtigen.                   |

### Voraussetzungen

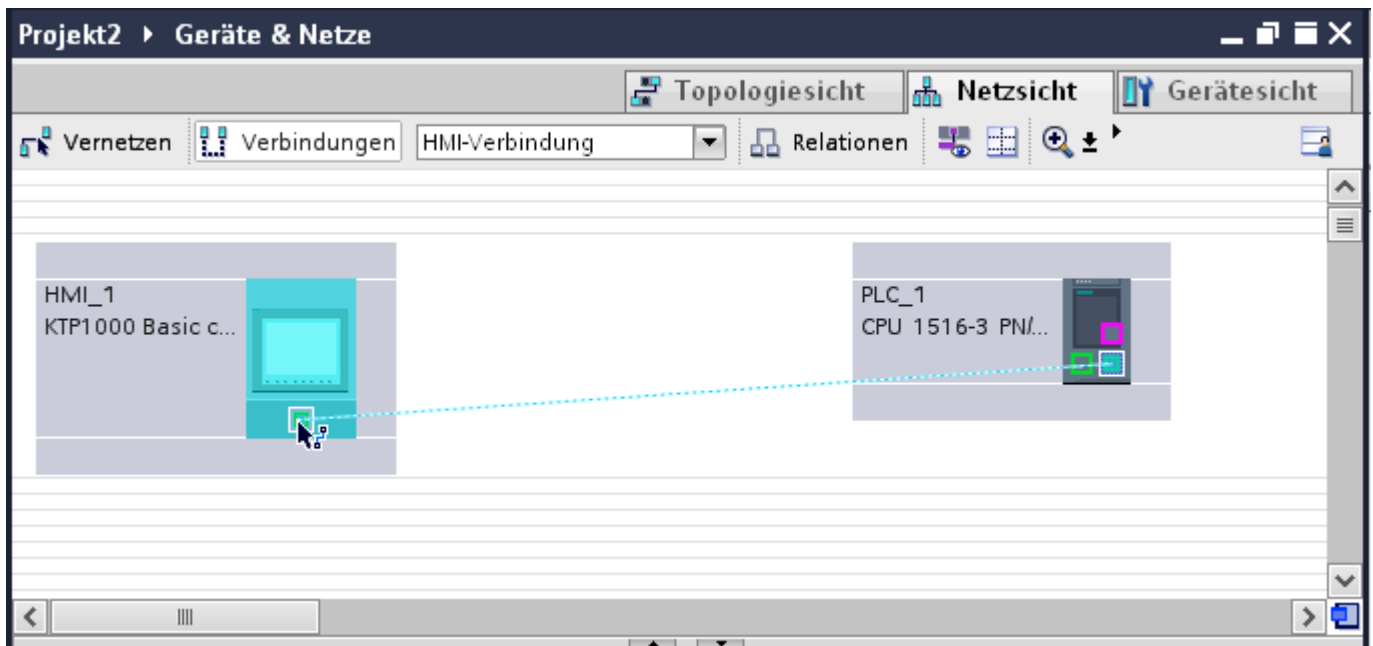
Folgende Kommunikationspartner sind im Editor "Geräte & Netze" angelegt:

- Bediengerät mit PROFINET- bzw. Ethernet-Schnittstelle
- SIMATIC S7 1500 mit PROFINET-Schnittstelle.

### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Geräte & Netze".  
In der Netzsicht werden die im Projekt vorhandenen Kommunikationspartner grafisch dargestellt.
2. Klicken Sie auf die Schaltfläche "Verbindungen" und wählen Sie den Verbindungstyp "HMI-Verbindung".  
Die für eine Verbindung verfügbaren Geräte werden farbig markiert dargestellt.

3. Klicken Sie in die PROFINET-Schnittstelle der Steuerung und ziehen Sie per Drag&Drop eine Verbindung zur PROFINET- bzw. Ethernet-Schnittstelle des Bediengeräts.



4. Klicken Sie auf die Verbindungslinie.
5. Klicken Sie auf "HMI-Verbindung Hervorheben" und wählen Sie die HMI-Verbindung aus. Im Inspektorfenster wird die Verbindung grafisch dargestellt.
6. Klicken Sie auf die Kommunikationspartner in der "Netzschritt" und ändern Sie im Inspektorfenster die PROFINET-Parameter nach den Anforderungen ihres Projekts. Nähere Hinweise hierzu finden Sie im Kapitel "Auto-Hotspot".

---

#### Hinweis

Die erstellte HMI-Verbindung wird auch im tabellarischen Bereich des Editors auf der Registerkarte "Verbindungen" angezeigt. In der Tabelle kontrollieren Sie die Verbindungsparameter.

Den lokalen Verbindungsnamen der Verbindung können Sie nur in der Tabelle ändern.

---

## Ergebnis

Sie haben eine Verbindung zwischen einem Bediengerät und einer SIMATIC S7 1500 angelegt. Die Verbindungsparameter IP-Adresse und Subnetzmaske sind projektiert.

## PROFINET-Parameter

### PROFINET-Parameter für die HMI-Verbindung

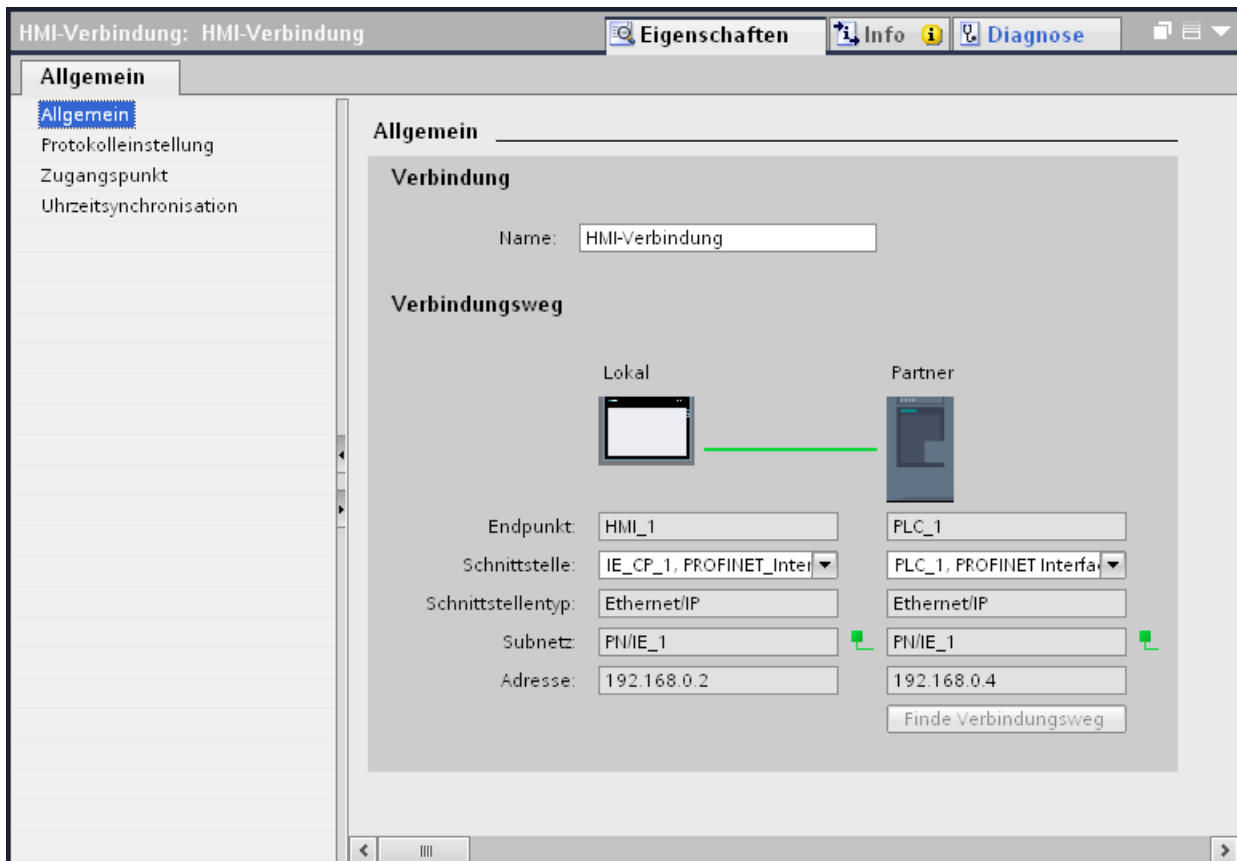
### PROFINET-Parameter für die HMI-Verbindung

Eine Übersicht über die projektierten Parameter einer HMI-Verbindung finden Sie in den Eigenschaften der HMI-Verbindung.

Änderungen sind in diesem Inspektorfenster nur eingeschränkt möglich.

### Parameter der HMI-Verbindung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die HMI-Verbindung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der HMI-Verbindung.



### "Verbindung"

Zeigt den Namen der HMI-Verbindung an.

## "Verbindungsweg"

Im Bereich "Verbindungsweg" werden die Kommunikationspartner der gewählten HMI-Verbindung und deren PROFINET-Parameter angezeigt. Einige der gezeigten Bereiche sind in diesem Dialog nicht editierbar.

- "Endpunkt"  
Zeigt den Gerätenamen an. Dieser Bereich ist nicht editierbar.
- "Schnittstelle"  
Zeigt die gewählte Schnittstelle des Geräts an. Abhängig vom Gerät können Sie zwischen mehreren Schnittstellen wählen.
- "Schnittstellen-Typ"  
Zeigt den gewählten Schnittstellen-Typ an. Dieser Bereich ist nicht editierbar.
- "Subnetz"  
Zeigt das gewählte Subnetz an. Dieser Bereich ist nicht editierbar.
- "Adresse"  
Zeigt die gewählte IP-Adresse des Geräts an. Dieser Bereich ist nicht editierbar.
- Schaltfläche "Finde Verbindungsweg"  
Ermöglicht das nachträgliche Spezifizieren von Verbindungen.

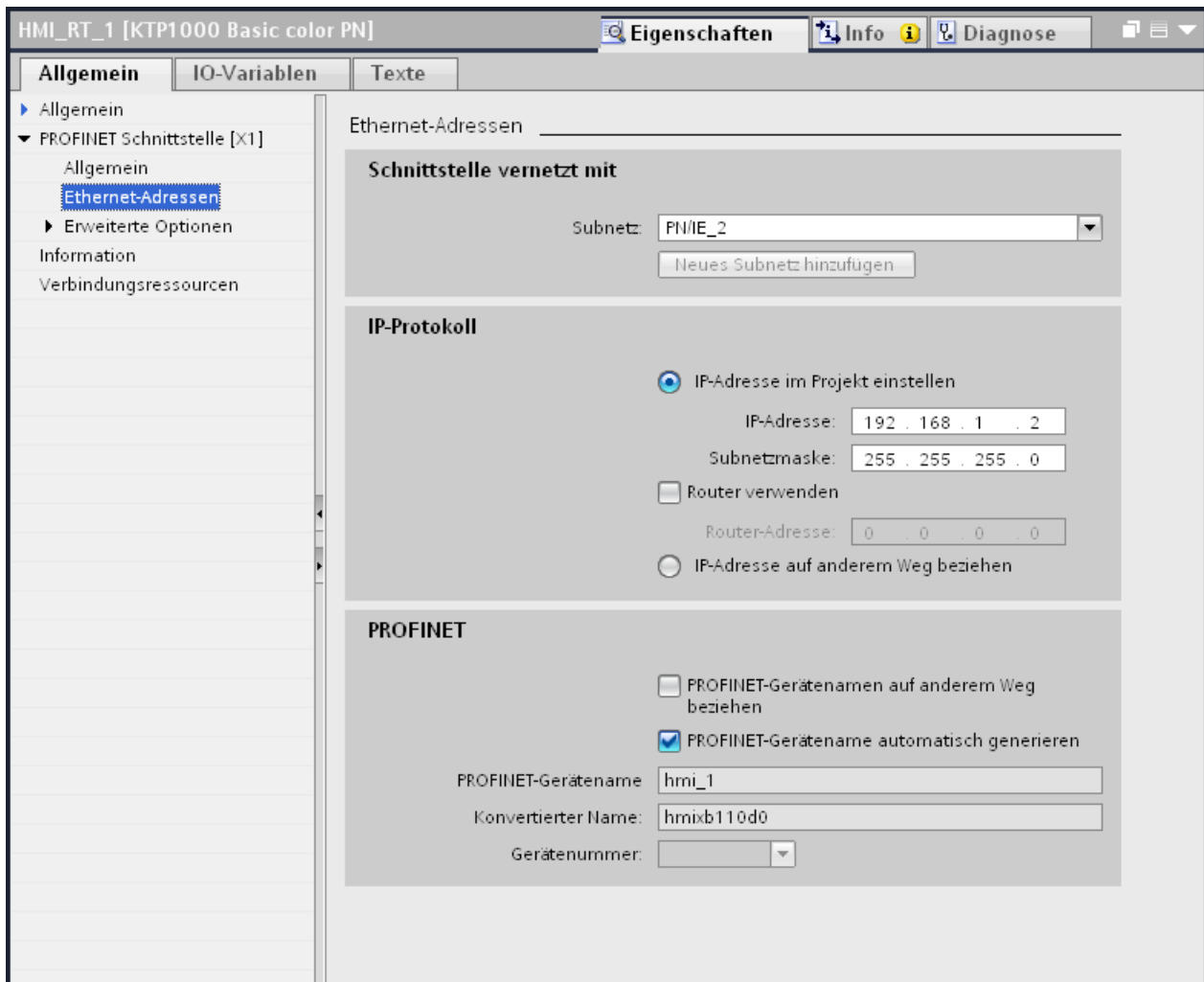
## PROFINET-Parameter für das Bediengerät

### PROFINET Parameter für das Bediengerät

Eine Übersicht über die projektierten Parameter des Bediengeräts finden Sie in den Eigenschaften des Bediengeräts.

## PROFINET-Parameter des Bediengeräts anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf das Bediengerät.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein" die Parameter des Bediengeräts.



### "Schnittstelle vernetzt mit"

Im Bereich "Schnittstelle vernetzt mit" wählen Sie das Subnetz der HMI-Verbindung aus, über die das Bediengerät an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

## "IP-Protokoll"

- "IP-Adresse im Projekt einstellen"  
Wenn Sie das WinCC-Projekt auf das Bediengerät transferieren, dann wird diese IP-Adresse direkt im Bediengerät eingerichtet.

---

### Hinweis

Bei Bediengeräten mit dem Betriebssystem Windows CE 3.0 erfolgt automatisch ein Neustart.

Bediengeräte mit Windows CE 3.0:

- OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"
- 
- "Subnetzmaske"  
Im Bereich "Subnetzmaske" vergeben Sie Daten der Subnetzmaske.
  - "IP-Router verwenden"  
Wenn Sie einen IP-Router verwenden, aktivieren Sie "IP-Router verwenden" und geben Sie die Routeradresse im Feld "Routeradresse" ein.
  - "IP-Adresse auf anderem Weg beziehen"  
Wenn die Funktion "IP-Adresse auf anderem Weg beziehen" aktiviert ist, dann wird die IP-Adresse nicht aus dem Projekt übernommen. Sie müssen die IP-Adresse direkt im Control Panel des Bediengeräts eingeben.

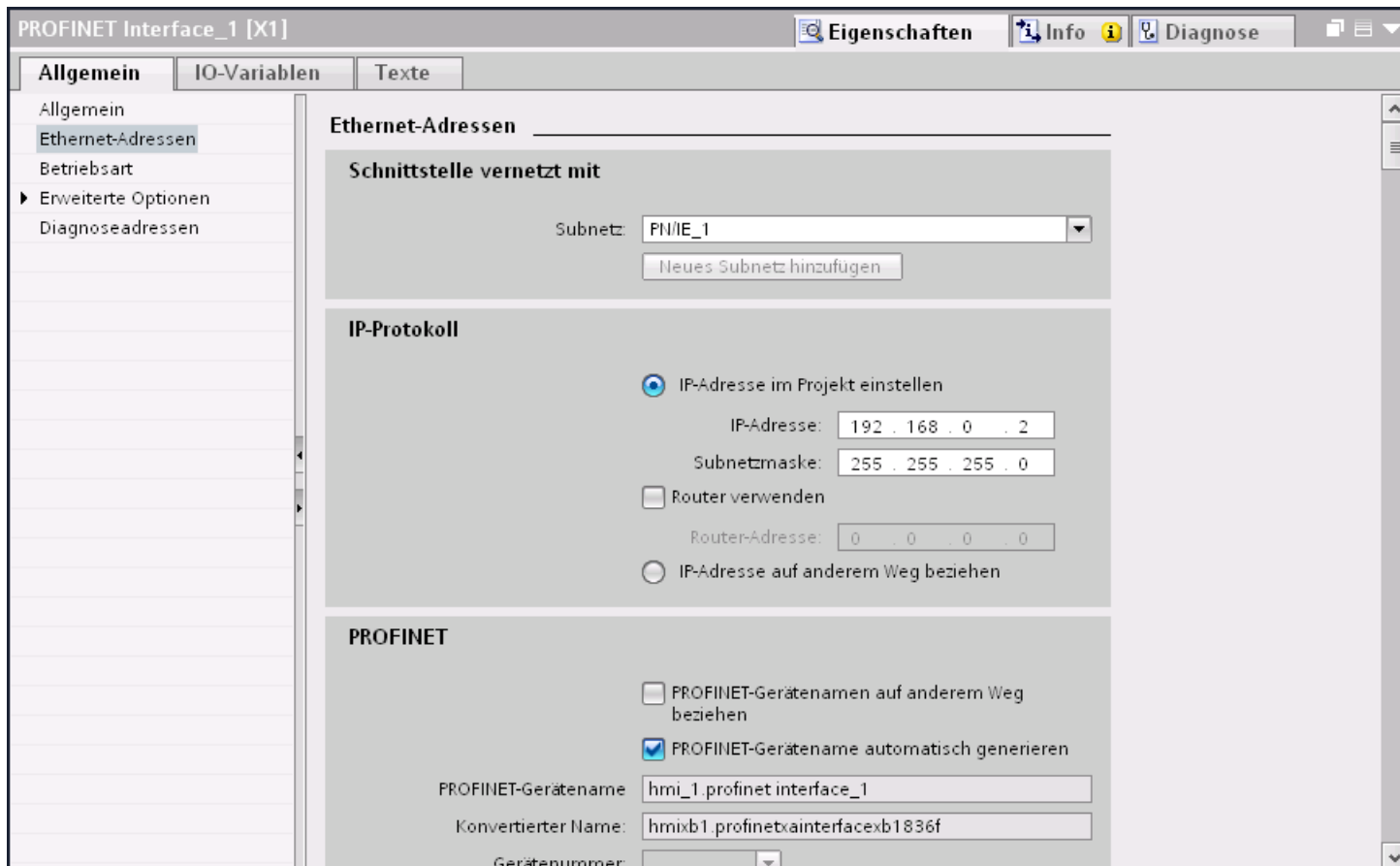
## PROFINET-Parameter für die Steuerung

### PROFINET-Parameter für die Steuerung

Eine Übersicht über die projektierten Parameter finden Sie in den Eigenschaften der Steuerung.

## PROFINET-Parameter der Steuerung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die Steuerung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der Steuerung.



### "Schnittstelle vernetzt mit"

Im Bereich "Subnetz" wählen Sie das Subnetz der HMI-Verbindung aus, über die die Steuerung an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "IP-Protokoll"

- "Schnittstellen-Typ"  
Abhängig vom Bediengerätetyp stehen verschiedene Schnittstellen zur Auswahl.
- "IP-Adresse"  
Im Bereich "IP-Adresse" vergeben Sie die IP-Adresse des Bediengeräts.
- "Subnetzmaske"  
Im Bereich "Subnetzmaske" vergeben Sie Daten der Subnetzmaske.  
Wenn Sie einen IP-Router verwenden, aktivieren Sie "IP-Router verwenden" und geben Sie die Routeradresse im Feld darunter ein.

## Schutz der Kommunikation

### Schutzstufen

Wenn Sie die Kommunikation von Steuerung und Bediengerät schützen wollen, dann können Sie Schutzstufen für die Kommunikation vergeben.

Für eine S7-1500-CPU können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in einer Tabelle eingegeben, so dass jedem Passwort genau eine Schutzstufe zugeordnet ist.

Wie das Passwort wirkt, steht in der Spalte "Schutz".

### Beispiel

Sie wählen die bei der Projektierung der Steuerung die Schutzstufe "Kompletter Schutz" für eine Standard-CPU (d. h. keine F-CPU) und geben für jede in der Tabelle darüber liegenden Schutzstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist die CPU komplett geschützt. Nicht einmal HMI-Zugriffe sind möglich.

Für Nutzer, die eines der parametrisierten Passwörter kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile 1 (Kein Schutz) wirkt, als wäre die CPU ungeschützt. Nutzer, die dieses Passwort kennen, haben uneingeschränkten Zugriff auf die CPU.
- Das Passwort in Zeile 2 (Schreibschutz) wirkt, als wäre die CPU schreibgeschützt. Trotz Passwort-Kenntnis haben Nutzer, die dieses Passwort kennen, nur lesenden Zugriff auf die CPU.
- Das Passwort in Zeile 3 (Schreib- und Leseschutz) wirkt, als wäre die CPU schreib- und lesegeschützt, so dass für die Nutzer, die dieses Passwort kennen, nur HMI-Zugriffe möglich sind.

## Zugangspasswort für die HMI-Verbindung

### Einleitung

Den Zugriff auf eine Steuerung sichern Sie durch die Vergabe eines Passworts ab.

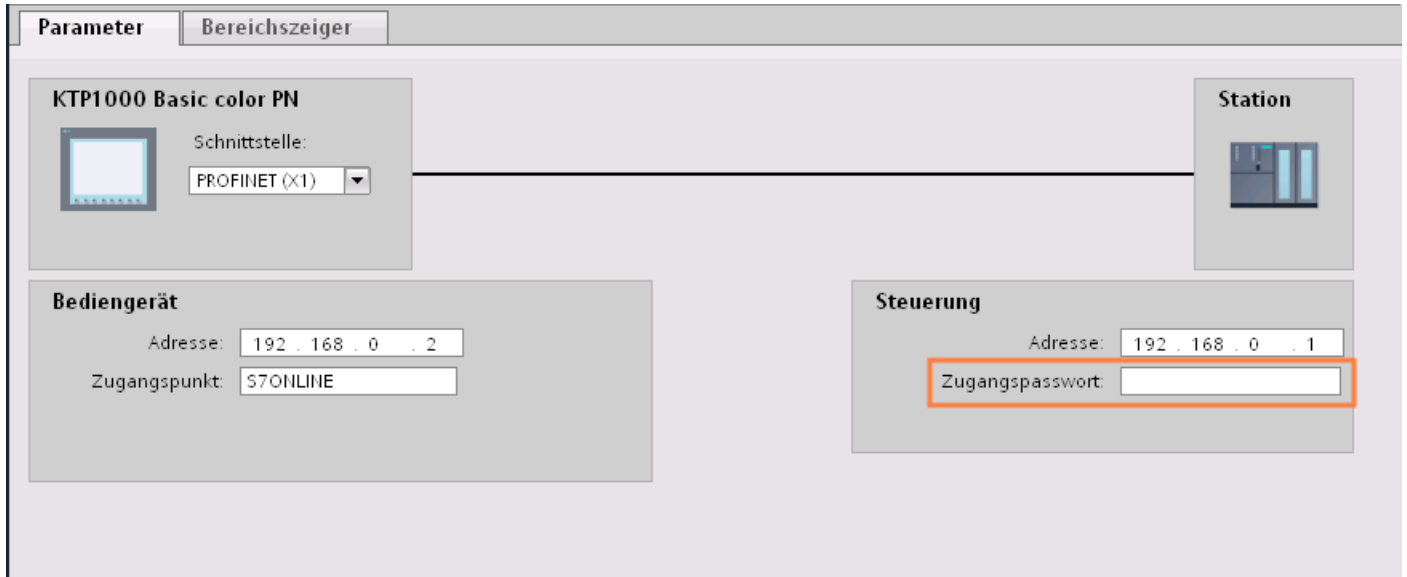
Bei der Projektierung der Passwort vergeben Sie das Passwort.

Ab Schutzstufe "Kompletter Schutz" ist es zwingend notwendig das Passwort einzugeben. Wenn Sie das Passwort nicht oder falsch eingeben wird keine Kommunikation zur Steuerung aufgebaut.



## Zugangspasswort eingeben

Im Editor "Verbindungen" geben Sie das Zugangspasswort für die Steuerung ein.



## Port-Optionen festlegen

### Festlegen der Port-Optionen

### Verbindungseinstellungen für den PROFINET-IO-Port ändern

Bei Bedarf können Sie die Netzwerkeinstellungen für den PROFINET-IO-Port ändern. Standardmäßig werden die Einstellungen automatisch festgelegt, was im Normalfall eine problemlose Kommunikation gewährleistet.

## Einstellmöglichkeiten Übertragungsrate/Duplex

Abhängig von dem gewählten Gerät können Sie folgende Einstellungen für "Übertragungsrate/Duplex" vornehmen:

- Automatische Einstellung  
Empfohlene Default-Einstellung des Ports. Die Übertragungseinstellungen mit dem Partner-Port werden automatisch "ausgehandelt". In der Voreinstellung ist automatisch auch die Option "Autonegotiation aktivieren" aktiviert, d. h. Sie können Crosskabel oder Patchkabel für den Anschluss verwenden.
- TP/ITP mit x Mbit/s. Vollduplex (Halbduplex)  
Einstellung des Übertragungsrate und des Vollduplex/Halbduplex-Betriebs. Die Wirkung ist abhängig von der eingestellten Option "Autonegotiation aktivieren":
  - Autonegotiation aktiviert  
Sie können sowohl Crosskabel als auch Patchkabel verwenden.
  - Autonegotiation deaktiviert  
Achten Sie darauf, dass Sie das korrekte Kabel verwenden (Patch- oder Crosskabel)! Auch bei dieser Einstellung wird der Port überwacht.
- Deaktiviert  
Abhängig vom Baugruppentyp kann in der Klappliste die Option "deaktiviert" angeboten werden. Sie haben damit beispielsweise die Möglichkeit, den Zugang zu einem unbenutzten Port aus Sicherheitsgründen zu unterbinden. Bei dieser Einstellung werden keine Diagnoseereignisse erzeugt.

### Option "Überwachen"

Mit dieser Option wird die Portdiagnose aktiviert bzw. deaktiviert. Beispiele für Portdiagnose: Der Link-Status wird überwacht, d. h. Diagnose bei Link-Down erzeugt und bei Fiber Optic Ports wird die Systemreserve überwacht.

### Option "Autonegotiation aktivieren"

Die Einstellung für Autonegotiation ist nur änderbar, wenn ein konkretes Medium (z. B. TP 100 Mbit/s vollduplex) ausgewählt ist. Ob ein konkretes Medium eingestellt werden kann, hängt von den Eigenschaften des Moduls ab.

Ist die Autonegotiation deaktiviert, dann wird die feste Einstellung des Ports erzwungen, ähnlich wie das z. B. bei einem priorisierten Hochlaufs des IO-Devices erforderlich ist.

Sie müssen für gleiche Einstellungen beim Partner-Port sorgen, da bei dieser Option die Betriebsparameter des angeschlossenen Netzes nicht erkannt werden und entsprechend die Datenübertragungsrate und Übertragungsmodus nicht optimal eingestellt werden können.

---

#### Hinweis

STEP 7 übernimmt bei einem verschalteten lokalen Port die Einstellung für den Partner Port, falls der Partner Port die Einstellung unterstützt. Falls der Partner Port die Einstellung nicht unterstützt, wird eine Fehlermeldung erzeugt.

---

## Verkabelungsregeln bei deaktivierter Autonegotiation

### Voraussetzung

Sie haben z. B. zur Beschleunigung der Hochlaufzeit des IO-Devices für den betreffenden Port folgende Einstellungen vorgenommen:

- Feste Übertragungsrate
- Autonegotiation inkl. Autocrossing deaktiviert

Die Zeit für das Aushandeln der Übertragungsrate beim Hochlauf wird somit eingespart.

Wenn Sie Autonegotiation deaktiviert haben, müssen Sie die Verkabelungsregeln beachten.

## Verkabelungsregeln bei deaktivierter Autonegotiation

PROFINET-Geräte besitzen folgende zwei Arten von Ports:

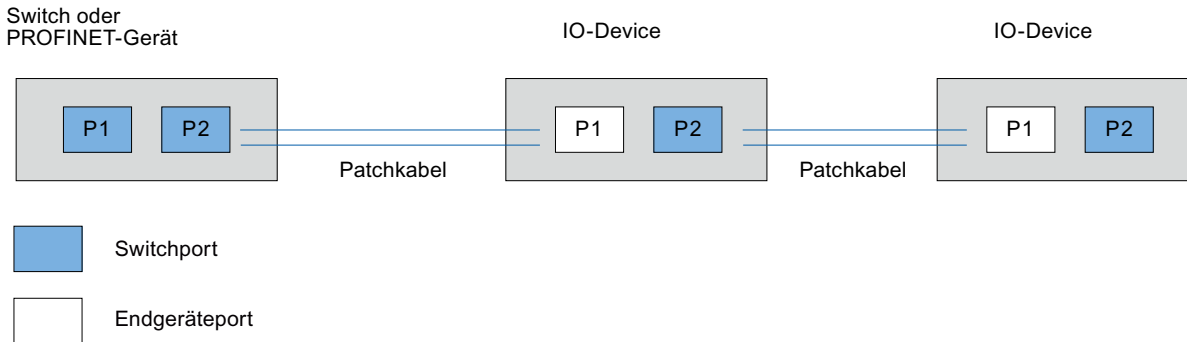
| Art des Ports                                  | PROFINET-Geräte   | Bemerkung  |
|--|---|--|
| Switchport mit gekreuzter Pinbelegung          | Bei IO-Devices: Port 2<br>Bei S7-CPU's mit 2 Ports: Port 1 und Port 2 | Gekreuzte Pinbelegung bedeutet, dass die Pinbelegung der Ports für Senden und Empfangen zwischen den betreffenden PROFINET-Geräten intern vertauscht werden. |
| Endgeräteport mit nicht gekreuzter Pinbelegung | Bei IO-Devices: Port 1<br>Bei S7-CPU's mit einem Port: Port 1         | -  |

### Gültigkeit der Verkabelungsregeln

Die im folgenden Absatz beschriebenen Verkabelungsregeln gelten ausschließlich für den Fall, dass Sie eine feste Porteinstellung vorgenommen haben.

## Regeln für die Verkabelung

Mehrere IO-Devices können Sie mit einem Kabeltyp (Patch-Kabel) in Linie verbinden. Dazu verbinden Sie Port 2 des IO-Devices (Dezentrale Peripherie) mit dem Port 1 des nächsten IO-Devices. In der folgenden Grafik ist ein Beispiel mit zwei IO-Devices dargestellt.



## Begrenzungen am Port

### Voraussetzung

Um mit Boundaries ("Begrenzungen") arbeiten zu können, muss das jeweilige Gerät mehr als einen Port besitzen. Wenn das PROFINET keine Boundaries-Einstellungen unterstützt, werden sie auch nicht angezeigt.

### Boundaries aktivieren

Unter "Boundaries" versteht man Grenzen für die Übertragung bestimmter Ethernet-Frames. Es können folgende Begrenzungen an einem Port gesetzt werden:

- "Ende der Erfassung erreichbarer Teilnehmer"  
DCP-Frames zur Erfassung erreichbarer Teilnehmer werden nicht weitergeleitet. Hinter diesem Port liegende Teilnehmer werden nicht mehr unter "erreichbare Teilnehmer" in der Projektnavigation angezeigt. Hinter diesem Port liegende Teilnehmer können von der CPU nicht mehr erreicht werden.
- "Ende der Topologieerkennung"  
LLDP-Frames (Link Layer Discovery Protocol) zur Topologieerkennung werden nicht weitergeleitet.
- "Ende der Sync-Domain"  
Sync-Frames, die zur Synchronisierung von Teilnehmern innerhalb einer Sync-Domain übertragen werden, werden nicht weitergeleitet.  
Wenn Sie z. B. ein PROFINET-Gerät mit mehr als zwei Ports in einem Ring betreiben, dann sollten Sie das Einspeisen von Sync-Frames in den Ring durch Setzen einer Sync-Boundary (an den Ports, die nicht im Ring sind) verhindern.  
Weiteres Beispiel: Möchten Sie mehrere Sync-Domains nutzen, dann projektieren Sie eine Sync-Domain-Boundary für den Port, der mit einem PROFINET-Gerät der jeweils anderen Sync-Domain verbunden ist.

## **Einschränkungen**

Folgende Einschränkungen sind zu beachten:

- Nur wenn der Port das jeweilige Feature unterstützt sind die einzelnen Optionskästchen bedienbar.
- Wenn für den Port ein Partnerport bestimmt wurde sind folgende Optionskästchen nicht bedienbar:
  - "Ende der Erfassung erreichbarer Teilnehmer"
  - "Ende der Topologieerkennung"
- Wenn Autonegotiation deaktiviert ist, dann sind alle Optionskästchen nicht bedienbar.

### **10.8.6.3 Kommunikation über PROFIBUS**

#### **HMI-Verbindung projektieren**

#### **HMI-Verbindung über PROFIBUS projektieren**

#### **Einleitung**

Eine HMI-Verbindung über PROFIBUS zwischen Bediengeräten und einer SIMATIC S7 1500 projektieren Sie im Editor "Geräte & Netze".

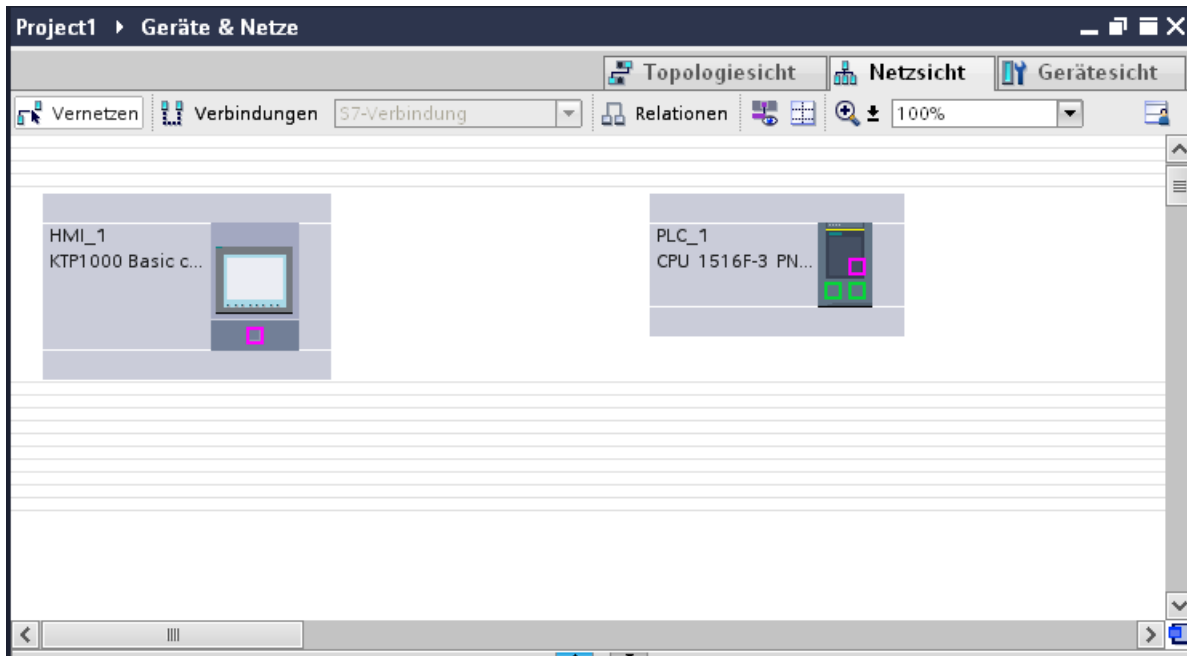
#### **Voraussetzungen**

Folgende Kommunikationspartner sind im Editor "Geräte & Netze" angelegt:

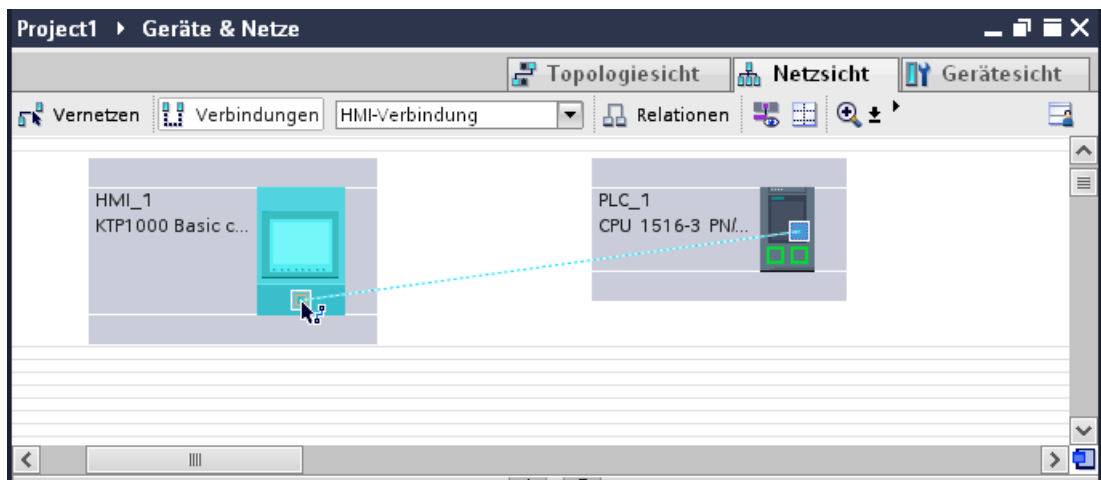
- Bediengerät mit MPI/DP-Schnittstelle
- SIMATIC SIMATIC S7 1500 mit PROFIBUS-Schnittstelle

### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Geräte & Netze".  
In der Netzsicht werden die im Projekt vorhandenen Kommunikationspartner grafisch dargestellt.



2. Klicken Sie auf die Schaltfläche "Verbindungen".  
Die für eine Verbindung verfügbaren Geräte werden farbig markiert dargestellt.
3. Klicken Sie auf die Schnittstelle des Bediengeräts.
4. Wählen Sie im Inspektorfenster "Eigenschaften > Allgemein > HMI MPIDP > Parameter" den Schnittstellen-Typ "PROFIBUS" aus.
5. Klicken Sie in die Schnittstelle der Steuerung und ziehen Sie per Drag&Drop eine Verbindung zum Bediengerät.



6. Klicken Sie auf die Verbindungslinie.

7. Klicken Sie auf "HMI-Verbindung Hervorheben" und wählen Sie die HMI-Verbindung aus. Im Inspektorfenster wird die Verbindung grafisch dargestellt.
8. Klicken Sie auf die Kommunikationspartner in der "Netzansicht" und ändern Sie im Inspektorfenster die PROFINET-Parameter nach den Anforderungen ihres Projekts. Nähere Hinweise hierzu finden Sie im Kapitel "Auto-Hotspot".

#### Hinweis

Die erstellte HMI-Verbindung wird auch im tabellarischen Bereich des Editors auf der Registerkarte "Verbindungen" angezeigt. In der Tabelle kontrollieren Sie die Verbindungsparameter.

Den lokalen Verbindungsnamen der Verbindung können Sie nur in der Tabelle ändern.

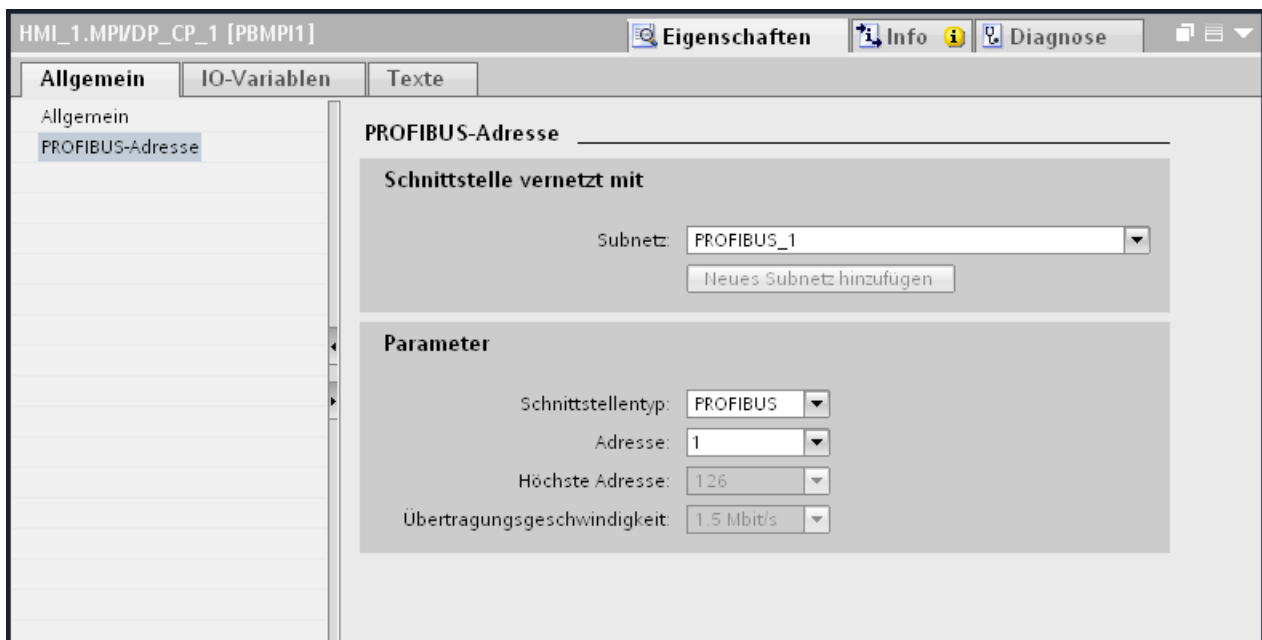
## Ergebnis

Sie haben eine HMI-Verbindung zwischen einem Bediengerät und einer SIMATIC S7 1500 über PROFIBUS angelegt.

## Kommunikation über PROFIBUS

### HMI-Verbindungen über PROFIBUS

Wenn Sie ein Bediengerät und eine SIMATIC S7 1500 in das Projekt eingefügt haben, verbinden Sie die beiden PROFIBUS-Schnittstellen im Editor "Geräte & Netze" miteinander.



### **HMI-Verbindung im Editor "Geräte & Netze"**

Die HMI-Verbindung über PROFIBUS zwischen der Steuerung mit dem Bediengerät projektieren Sie im Editor "Geräte & Netze".

### **Verbindung im Editor "Verbindungen"**

Alternativ projektieren Sie die Verbindung zwischen Steuerung und Bediengerät im Editor "Verbindungen" des Bediengeräts.

### **PROFIBUS-Parameter**

#### **PROFIBUS-Parameter für die HMI-Verbindung**

#### **PROFIBUS-Parameter für die HMI-Verbindung**

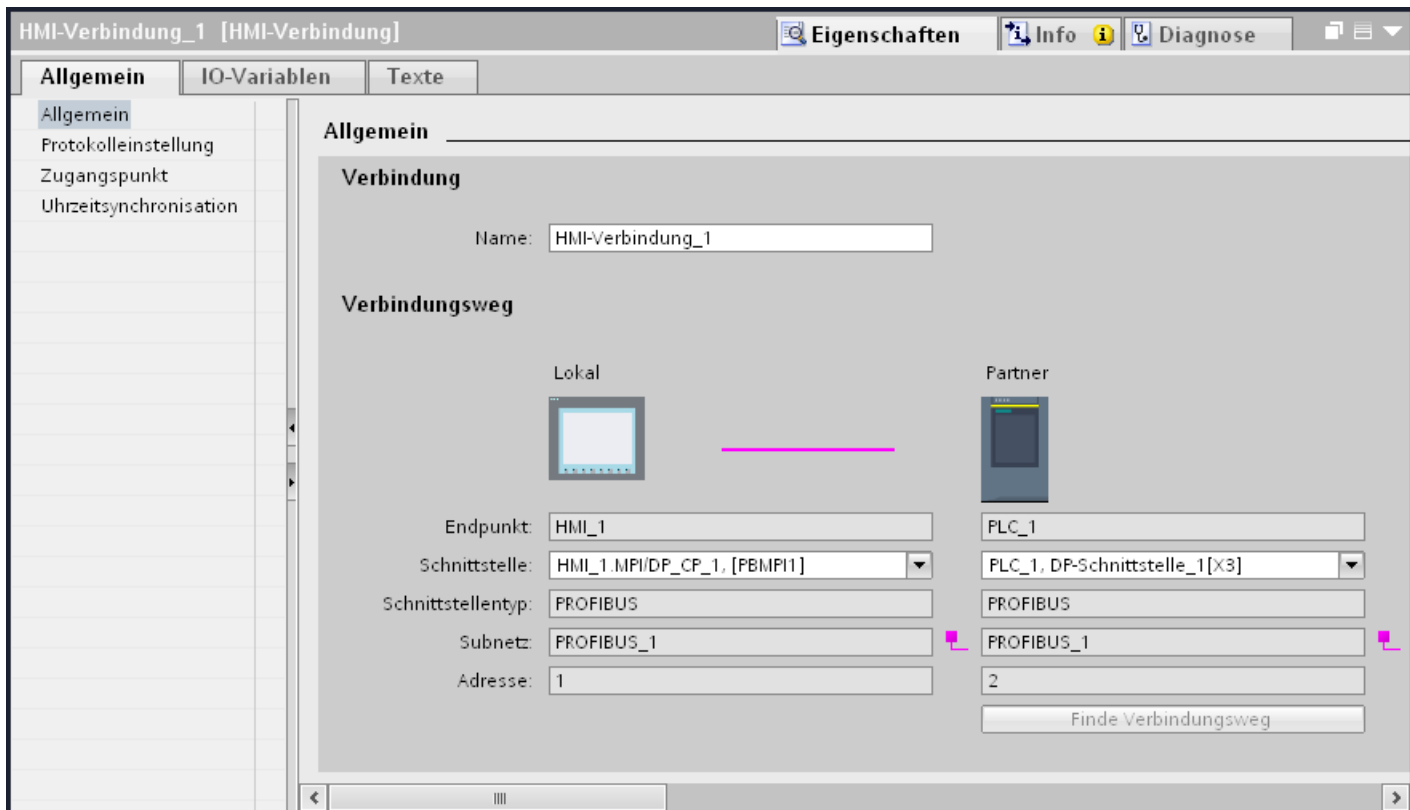
Eine Übersicht über die projizierten Parameter einer HMI-Verbindung finden Sie in den Eigenschaften der HMI-Verbindung.

Änderungen sind in diesem Inspektorfenster nur eingeschränkt möglich.





## Parameter der HMI-Verbindung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die HMI-Verbindung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der HMI-Verbindung.



### "Verbindung"

Zeigt an, ob die Geräte bereits miteinander vernetzt sind.

-  - Wird angezeigt, wenn die Geräte miteinander vernetzt sind.
-  - Wird angezeigt, wenn die Geräte nicht miteinander vernetzt sind.

### "Verbindungsweg"

Im Bereich "Verbindungsweg" werden die Kommunikationspartner der gewählten HMI-Verbindung und deren PROFIBUS-Parameter angezeigt. Einige der gezeigten Bereiche sind in diesem Dialog nicht editierbar.

- "Endpunkt"  
Zeigt den Gerätenamen an. Dieser Bereich ist nicht editierbar.
- "Schnittstelle"  
Zeigt die gewählte Schnittstelle des Geräts an. Abhängig vom Gerät können Sie zwischen mehreren Schnittstellen wählen.

- "Schnittstellen-Typ"  
Zeigt den gewählten Schnittstellen-Typ an. Dieser Bereich ist nicht editierbar.
- "Subnetz"  
Zeigt das gewählte Subnetz an. Dieser Bereich ist nicht editierbar.
- "Adresse"  
Zeigt die PROFIBUS-Adresse des Geräts an. Dieser Bereich ist nicht editierbar.
- Schaltfläche "Finde Verbindungsweg"  
Ermöglicht das nachträgliche Spezifizieren von Verbindungen.

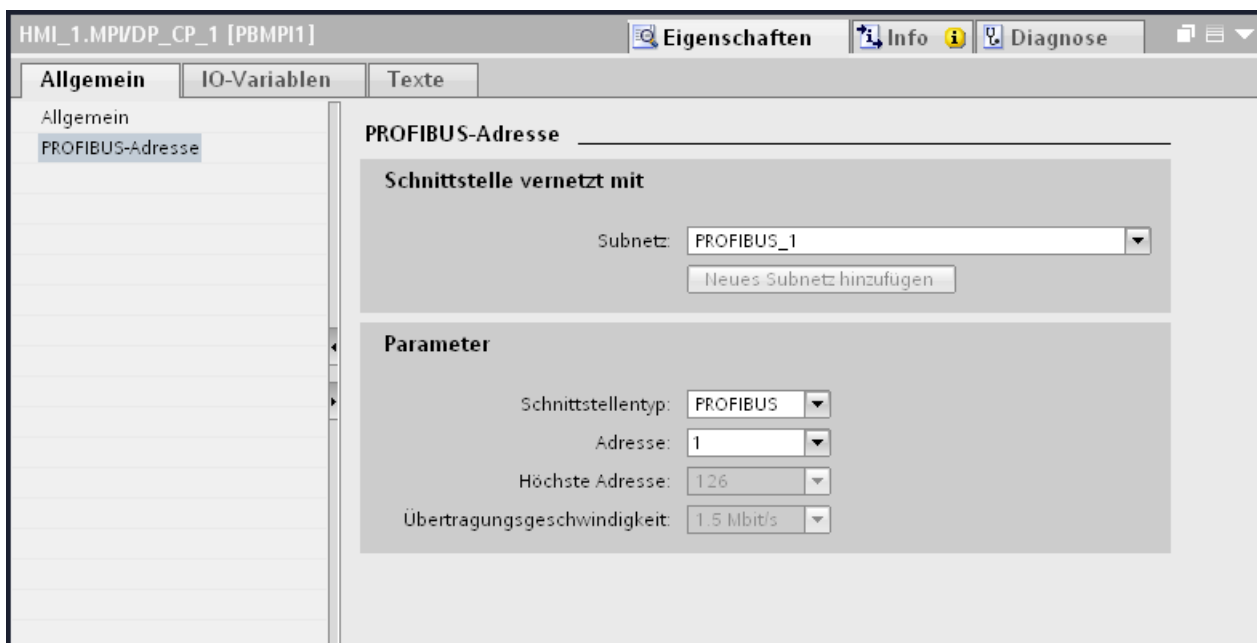
## PROFIBUS-Parameter für das Bediengerät

### PROFIBUS-Parameter für das Bediengerät

Eine Übersicht über die projektierten Parameter des Bediengeräts finden Sie in den Eigenschaften des Bediengeräts.

### PROFINET-Parameter des Bediengeräts anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf das Bediengerät.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter des Bediengeräts.



### "Schnittstelle vernetzt mit"

Im Bereich "Schnittstelle vernetzt mit" wählen Sie das Subnetz der HMI-Verbindung aus, über die das Bediengerät an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

## "Parameter"

- "Schnittstellen-Typ"  
Im Bereich "Schnittstellentyp" parametrieren Sie den Schnittstellentyp. Abhängig vom Bediengerätetyp stehen hier verschiedenen Schnittstellen zur Auswahl.
- "Adresse"  
Im Bereich "Adresse" parametrieren Sie die PROFIBUS-Adresse des Bediengeräts. Die PROFIBUS-Adresse muss im PROFIBUS-Netzwerk eindeutig sein
- "Höchste Adresse"  
Im Bereich "Höchste Adresse" wird die höchste Adresse des PROFIBUS-Netzwerks angezeigt.
- "Übertragungsgeschwindigkeit"  
Die "Übertragungsgeschwindigkeit" wird durch das langsamste ans Netzwerk angeschlossene Gerät festgelegt. Die Einstellung ist im gesamten Netzwerk gleich.

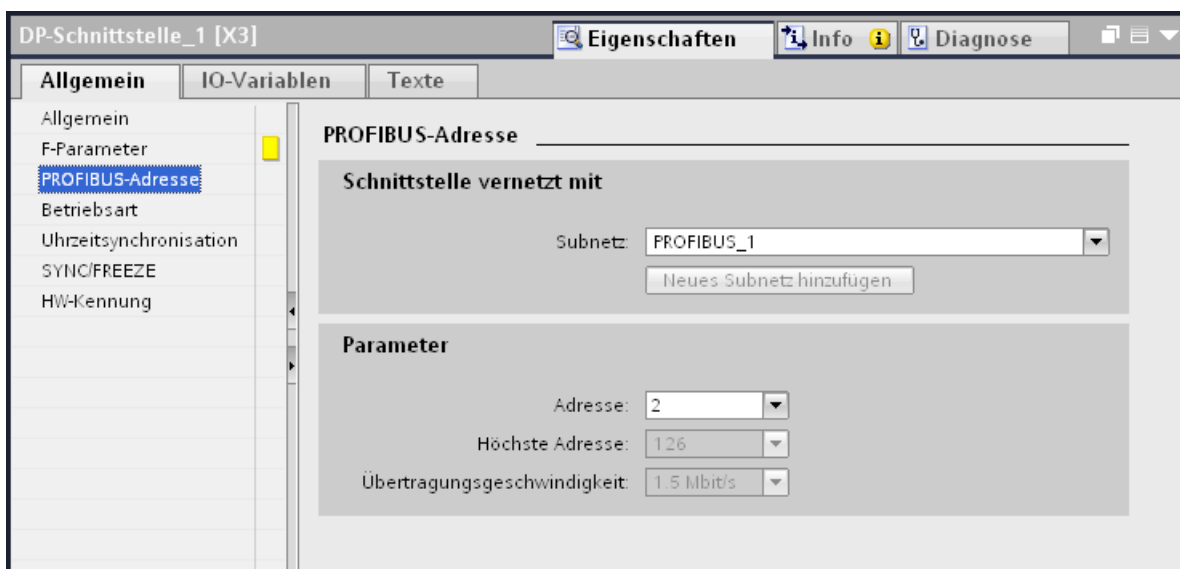
## PROFIBUS-Parameter für die Steuerung

### PROFIBUS-Parameter für Steuerung

Eine Übersicht über die projizierten Parameter finden Sie in den Eigenschaften der Steuerung.

### PROFIBUS-Parameter der Steuerung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die Steuerung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der Steuerung.



### "Schnittstelle vernetzt mit"

Im Bereich "Subnetz" wählen Sie das Subnetz der HMI-Verbindung aus, über die die Steuerung an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "Parameter"

- "Schnittstellen-Typ"  
Abhängig vom Bediengerätetyp stehen verschiedene Schnittstellen zur Auswahl.
- "Adresse"  
Im Bereich "Adresse" parametrieren Sie die PROFIBUS-Adresse des Bediengeräts. Die PROFIBUS-Adresse muss im PROFIBUS-Netzwerk eindeutig sein
- "Höchste Adresse"  
Im Bereich "Höchste Adresse" wird die höchste Adresse des PROFIBUS-Netzwerks angezeigt.
- "Übertragungsgeschwindigkeit"  
Die "Übertragungsgeschwindigkeit" wird durch das langsamste ans Netzwerk angeschlossene Gerät festgelegt. Die Einstellung ist im gesamten Netzwerk gleich.

### Schutz der Kommunikation

#### Zugangspasswort für die HMI-Verbindung

#### Einleitung

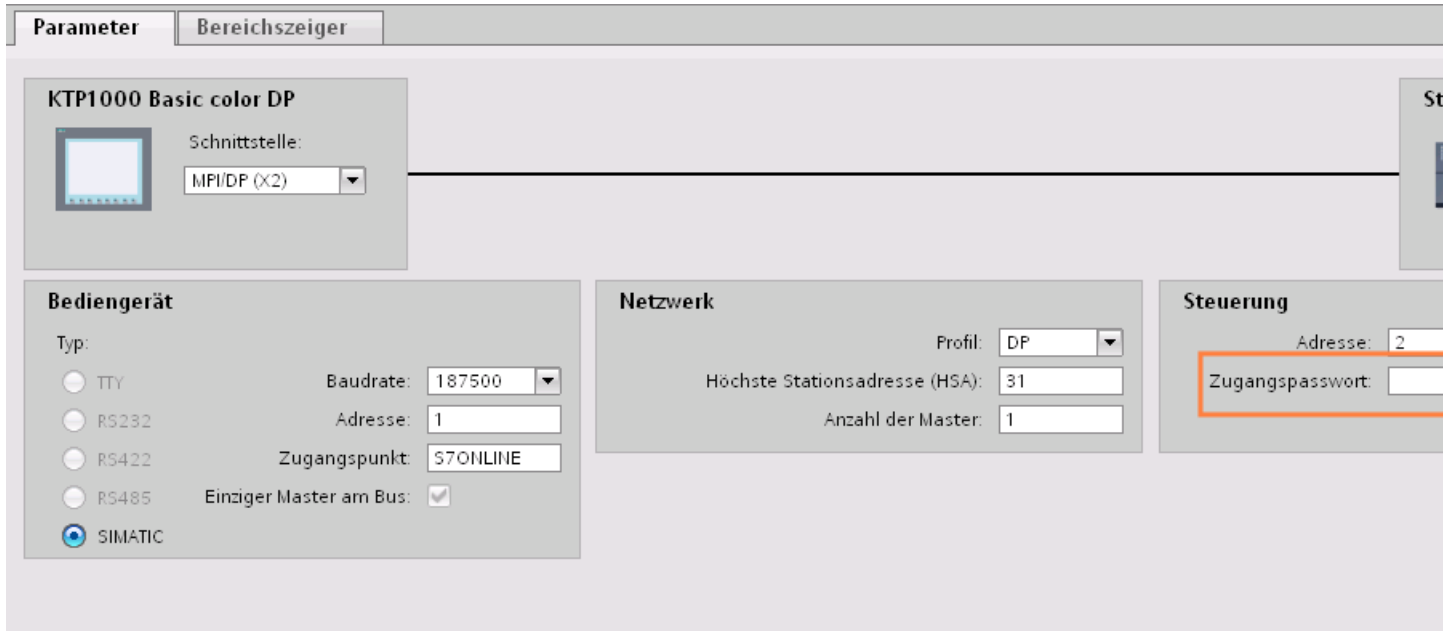
Den Zugriff auf eine Steuerung sichern Sie durch die Vergabe eines Passworts ab.

Bei der Projektierung der Passwort vergeben Sie das Passwort.

Ab Schutzstufe "Kompletter Schutz" ist es zwingend notwendig das Passwort einzugeben. Wenn Sie das Passwort nicht oder falsch eingeben wird keine Kommunikation zur Steuerung aufgebaut.

## Kennwort vergeben

Im Editor "Verbindungen" geben Sie das Zugangspasswort für die Steuerung ein.



## Schutzstufen

Wenn Sie die Kommunikation von Steuerung und Bediengerät schützen wollen, dann können Sie Schutzstufen für die Kommunikation vergeben.

Für eine S7-1500-CPU können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in einer Tabelle eingegeben, so dass jedem Passwort genau eine Schutzstufe zugeordnet ist.

Wie das Passwort wirkt, steht in der Spalte "Schutz".

## Beispiel

Sie wählen die bei der Projektierung der Steuerung die Schutzstufe "Kompletter Schutz" für eine Standard-CPU (d. h. keine F-CPU) und geben für jede in der Tabelle darüber liegenden Schutzstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist die CPU komplett geschützt. Nicht einmal HMI-Zugriffe sind möglich.

Für Nutzer, die eines der parametrisierten Passwörter kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile 1 (Kein Schutz) wirkt, als wäre die CPU ungeschützt. Nutzer, die dieses Passwort kennen, haben uneingeschränkten Zugriff auf die CPU.
- Das Passwort in Zeile 2 (Schreibschutz) wirkt, als wäre die CPU schreibgeschützt. Trotz Passwort-Kennntnis haben Nutzer, die dieses Passwort kennen, nur lesenden Zugriff auf die CPU.
- Das Passwort in Zeile 3 (Schreib- und Leseschutz) wirkt, als wäre die CPU schreib- und lesegeschützt, so dass für die Nutzer, die dieses Passwort kennen, nur HMI-Zugriffe möglich sind.

#### 10.8.6.4 Datenaustausch

### Bereichszeiger

#### Allgemeines zu Bereichszeigern

##### Einleitung

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu. In diese Datenbereiche wird während der Kommunikation wechselseitig von der Steuerung und dem Bediengerät geschrieben und gelesen.

Durch die Auswertung der abgelegten Daten lösen Steuerung und Bediengerät gegenseitig fest definierte Aktionen aus.

#### Projektierung von Bereichszeigern

Bevor Sie den Bereichszeiger verwenden, aktivieren Sie die Bereichszeiger unter "Verbindungen > Bereichszeiger". Danach parametrieren Sie die Bereichszeiger.

Nähere Hinweise zum Projektieren von Bereichszeigern finden Sie unter:

Datenaustausch über Bereichszeiger (Seite 3487)

#### Einschränkungen

Beim Datenaustausch über Bereichszeiger sind bei der Kommunikation mit SIMATIC S7 1500 nur folgende Datentypen projektierbar:

- UInt und Array von UInt
- Word und Array von Word
- Int und Array von Int
- "Array[0..15] of Bool" beim Bereichszeiger "Koordination"
- Date\_And\_Time
- DTL und LDT

## Bereichszeiger "Bildnummer"

### Funktion

Die Bediengeräte legen im Bereichszeiger "Bildnummer" Informationen über das am Bediengerät aufgerufene Bild ab.

Dadurch ist es möglich, Informationen zum aktuellen Bildinhalt des Bediengeräts zur Steuerung zu übertragen. In der Steuerung können bestimmte Reaktionen getriggert werden, z. B. den Aufruf eines anderen Bildes.

### Verwendung

Bevor der Bereichszeiger "Bildnummer" verwendet werden kann, muss dieser unter "Kommunikation > Verbindungen" eingerichtet und aktiviert werden. Der Bereichszeiger "Bildnummer" kann nur in **einer** Steuerung und in dieser Steuerung nur **einmal** angelegt werden.

Die Bildnummer wird immer zur Steuerung übertragen, wenn ein neues Bild aktiviert wird oder der Fokus innerhalb eines Bildes von einem Bildobjekt zu einem anderen wechselt.

### Aufbau

Der Bereichszeiger ist ein Datenbereich im Speicher der Steuerung mit einer festen Länge von 5 Worten.

|        | 15                  | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1.Wort | Aktueller Bildtyp   |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 2.Wort | Aktuelle Bildnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 3.Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 4.Wort | Aktuelle Feldnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 5.Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

- Aktueller Bildtyp  
"1" für Grundbild oder  
"4" für Permanentfenster
- Aktuelle Bildnummer  
1 bis 32767
- Aktuelle Feldnummer  
1 bis 32767

## Bereichszeiger "Datum/Uhrzeit"

### Funktion

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von dem Bediengerät zur Steuerung verwendet.

Die Steuerung schreibt den Steuerauftrag "41" oder "40" in das Auftragsfach.

Mit der Auswertung des Steuerauftrags schreibt das Bediengerät sein aktuelles Datum und die Uhrzeit in den im Bereichszeiger "Datum/Uhrzeit" projektierten Datenbereich. Alle Angaben sind BCD-codiert.

---

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" projektiert haben, dann können Sie den Bereichszeiger "Datum/Uhrzeit PLC" nicht verwenden.

---

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" verwenden, dann ist eine symbolische Adressierung nicht möglich.

Wenn in einem Projekt mehrere Verbindungen projektiert sind, und in einer der Verbindungen der Bereichszeiger "Datum / Uhrzeit" verwendet werden soll, so muss der Bereichszeiger für jede projektierte Verbindung aktiviert werden.

Der Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

| Datenwort | Höherwertiges Byte |  |  |  |  |   | Niederwertiges Byte   |  |  |  |  |   |         |
|-----------|--------------------|--|--|--|--|---|-----------------------|--|--|--|--|---|---------|
|           | 7                  |  |  |  |  | 0 | 7                     |  |  |  |  | 0 |         |
| n+0       | Reserviert         |  |  |  |  |   | Stunde (0-23)         |  |  |  |  |   | Uhrzeit |
| n+1       | Minute (0-59)      |  |  |  |  |   | Sekunde (0-59)        |  |  |  |  |   |         |
| n+2       | Reserviert         |  |  |  |  |   | Reserviert            |  |  |  |  |   |         |
| n+3       | Reserviert         |  |  |  |  |   | Wochentag (1-7, 1=So) |  |  |  |  |   | Datum   |
| n+4       | Tag (1-31)         |  |  |  |  |   | Monat (1-12)          |  |  |  |  |   |         |
| n+5       | Jahr (80-99/0-29)  |  |  |  |  |   | Reserviert            |  |  |  |  |   |         |

---

#### Hinweis

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

### Verwendung von Datentypen

Die Datentypen "Date\_And\_Time, DTL" und "LDT" können nur bei den Bereichszeigern "Datum/Uhrzeit" und "Datum/Uhrzeit PLC" verwendet werden.



Das benutzte Datenformat des Bereichszeigers "Datum/Uhrzeit" ist vom verwendeten Steuerungsauftrag 40/41 abhängig.

Falls an dem Bereichszeiger keine Steuerungsvariable verbunden ist oder eine Steuerungsvariable mit dem Datentyp "Array[0..5] of UInt/Word/Int" verbunden ist, gilt folgendes:

Der dargestellte Aufbau des Bereichszeigers "Datum/Uhrzeit" wird nur beim Steuerungsauftrag 41 verwendet.

Falls der Steuerungsauftrag 40 verwendet wird, wird das weiter unten dargestellt Datenformat "DATE\_AND\_TIME (BCD-codiert)" verwendet.

Falls an den Bereichszeigern "Datum/Uhrzeit" und "Datum/Uhrzeit PLC" eine Steuerungsvariable mit dem Datentyp "DATE\_AND\_TIME", "DTL" oder "LDT" verbunden ist, wird in dem entsprechenden Bereichszeiger genau das zugehörige Datenformat verwendet.

## **Bereichszeiger "Datum/Uhrzeit PLC"**

### **Funktion**

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von der Steuerung zum Bediengerät verwendet. Sie setzen diesen Bereichszeiger ein, wenn die Steuerung Master für die Zeit ist.

Die Steuerung lädt den Datenbereich des Bereichszeigers. Alle Angaben sind BCD-codiert.

Das Bediengerät liest zyklisch die Daten über den projektierten Erfassungszyklus und synchronisiert sich.

---

#### **Hinweis**

Wählen Sie in der Projektierung den Erfassungszyklus für den Bereichszeiger Datum/Uhrzeit nicht zu klein, da dies die Performance des Bediengeräts beeinflusst.  
Empfehlung: Erfassungszyklus 1 Minute, wenn dies Ihr Prozess erlaubt.

---

"Datum/Uhrzeit PLC" ist ein globaler Bereichszeiger und kann in einem Projekt nur ein Mal projektiert werden.

---

#### **Hinweis**

Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" projektiert haben, können Sie den Bereichszeiger "Datum/Uhrzeit" nicht verwenden.

---

Der Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

**Format DATE\_AND\_TIME (BCD-codiert)**

| Datenwort         | Höherwertiges Byte |       |   | Niederwertiges Byte |                       |   |
|-------------------|--------------------|-------|---|---------------------|-----------------------|---|
|                   | 7                  | ..... | 0 | 7                   | .....                 | 0 |
| n+0               | Jahr (80-99/0-29)  |       |   | Monat (1-12)        |                       |   |
| n+1               | Tag (1-31)         |       |   | Stunde (0-23)       |                       |   |
| n+2               | Minute (0-59)      |       |   | Sekunde (0-59)      |                       |   |
| n+3               | Reserviert         |       |   | Reserviert          | Wochentag (1-7, 1=So) |   |
| n+4 <sup>1)</sup> | Reserviert         |       |   | Reserviert          |                       |   |
| n+5 <sup>1)</sup> | Reserviert         |       |   | Reserviert          |                       |   |

- 1) Die beiden Datenwörter müssen im Datenbereich vorhanden sein, um eine Übereinstimmung des Datenformats mit WinCC flexible sicherzustellen und das Lesen falscher Informationen zu vermeiden.

**Hinweis**

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

**Verwendung von Datentypen**

Die Datentypen "Date\_And\_Time, DTL" und "LDT" können nur bei den Bereichszeigern "Datum/Uhrzeit" und "Datum/Uhrzeit PLC" verwendet werden.

Das benutzte Datenformat des Bereichszeigers "Datum/Uhrzeit" ist vom verwendeten Steuerungsauftrag 40/41 abhängig.

Falls an dem Bereichszeiger keine Steuerungsvariable verbunden ist oder eine Steuerungsvariable mit dem Datentyp "Array[0..5] of UInt/Word/Int" verbunden ist, gilt folgendes:

Der dargestellte Aufbau des Bereichszeigers "Datum/Uhrzeit" wird nur beim Steuerungsauftrag 41 verwendet.

Falls der Steuerungsauftrag 40 verwendet wird, wird das weiter unten dargestellt Datenformat "DATE\_AND\_TIME (BCD-codiert)" verwendet.

Falls an den Bereichszeigern "Datum/Uhrzeit" und "Datum/Uhrzeit PLC" eine Steuerungsvariable mit dem Datentyp "DATE\_AND\_TIME", "DTL" oder "LDT" verbunden ist, wird in dem entsprechenden Bereichszeiger genau das zugehörige Datenformat verwendet.

## Bereichszeiger "Koordinierung"

### Funktion

Der Bereichszeiger "Koordinierung" dient zur Realisierung der folgenden Funktionen:

- Anlauf des Bediengeräts im Steuerungsprogramm erkennen
- Aktuelle Betriebsart des Bediengeräts im Steuerungsprogramm erkennen
- Kommunikationsbereitschaft des Bediengeräts im Steuerungsprogramm erkennen

Der Bereichszeiger "Koordinierung" standardmäßig hat eine Länge von einem Wort und kann nicht verändert werden.

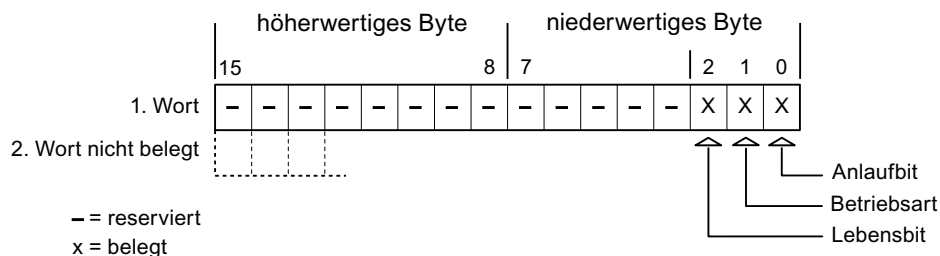
### Verwendung

#### Hinweis

Bei jeder Aktualisierung des Bereichszeigers durch das Bediengerät wird immer der komplette Koordinierungsbereich geschrieben.

Das SPS-Programm darf deshalb im Koordinierungsbereich keine Änderungen vornehmen.

### Belegung der Bits im Bereichszeiger "Koordinierung"



### Anlaufbit

Das Anlaufbit wird durch das Bediengerät während des Anlaufvorgangs kurzfristig auf "0" gesetzt. Nach dem Anlaufvorgang steht das Bit dauerhaft auf "1".

### Betriebsart

Sobald das Bediengerät durch den Benutzer offline geschaltet wird, wird das Betriebsartenbit auf 1 gesetzt. Im Normalbetrieb des Bediengeräts ist der Zustand des Betriebsartenbits "0". Im Steuerungsprogramm können Sie durch Abfrage dieses Bits die aktuelle Betriebsart des Bediengeräts ermitteln.

## Lebensbit

Das Lebensbit wird durch das Bediengerät im zeitlichen Abstand von ca. einer Sekunde invertiert. Im Steuerungsprogramm können Sie durch Abfrage dieses Bits überprüfen, ob die Verbindung zum Bediengerät noch besteht.

## Bereichszeiger "Projektkennung"

### Funktion

Beim Start der Runtime kann überprüft werden, ob das Bediengerät an die richtige Steuerung angeschlossen ist. Diese Überprüfung ist beim Einsatz mehrerer Bediengeräte wichtig.

Dazu vergleicht das Bediengerät einen in der Steuerung hinterlegten Wert mit dem in der Projektierung angegebenen Wert. Damit wird die Kompatibilität der Projektierungsdaten mit dem Steuerungsprogramm sichergestellt. Eine fehlende Übereinstimmung führt zur Anzeige einer Systemmeldung am Bediengerät und zum Stopp der Runtime.

### Verwendung

Um diesen Bereichszeiger zu verwenden, richten Sie bei der Projektierung Folgendes ein:

- Angabe der Version, welche die Projektierung hat. Möglicher Wert zwischen 1 und 255. Sie geben die Version ein im Editor "Runtime-Einstellungen > Allgemein" im Bereich "Identifizierung".
- Datenadresse des Werts für die Version, die in der Steuerung hinterlegt ist: Sie geben die Datenadresse ein im Editor "Kommunikation > Verbindungen" unter "Adresse".

### Ausfall einer Verbindung

Ein Verbindungsausfall zu einem Gerät, auf dem der Bereichszeiger "Projektkennung" projiziert ist, hat zur Folge, dass auch alle anderen Verbindungen des Geräts "Offline" geschaltet werden.

Dieses Verhalten hat folgende Voraussetzungen:

- Sie haben in einem Projekt mehrere Verbindungen projiziert.
- Sie verwenden in mindestens einer Verbindung den Bereichszeiger "Projektkennung".

Folgende Ursachen können Verbindungen in den Zustand "Offline" setzen:

- Die Steuerung ist nicht erreichbar.
- Die Verbindung wurde im Engineering System offline geschaltet.

## Bereichszeiger "Steuerungsauftrag"

### Funktion

Über das Steuerungsauftrags-Fach können dem Bediengerät Steuerungsaufträge gegeben und damit Aktionen am Bediengerät getriggert werden. Zu diesen Funktionen gehören z. B.:

- Bild anzeigen
- Datum und Uhrzeit stellen

### Datenstruktur

Im ersten Wort des Steuerungsauftrags-Fachs steht die Auftragsnummer. Je nach Steuerungsauftrag können dann bis zu drei Parameter übergeben werden.

| Wort | Höherwertiges Byte | Niederwertiges Byte |
|------|--------------------|---------------------|
| n+0  | 0                  | Auftragsnummer      |
| n+1  |                    | Parameter 1         |
| n+2  |                    | Parameter 2         |
| n+3  |                    | Parameter 3         |

Wenn das erste Wort des Steuerungsauftrags-Fachs ungleich 0 ist, wertet das Bediengerät den Steuerungsauftrag aus. Aus diesem Grund müssen zuerst die Parameter in das Steuerungsauftrags-Fach eingetragen werden und dann erst die Auftragsnummer.

Wenn das Bediengerät den Steuerungsauftrag angenommen hat, wird das erste Wort wieder auf 0 gesetzt. Die Ausführung des Steuerungsauftrags ist zu diesem Zeitpunkt im Allgemeinen noch nicht abgeschlossen.

### Steuerungsaufträge

Nachfolgend sind alle Steuerungsaufträge und deren Parameter aufgelistet. Die Spalte "Nr." enthält die Auftragsnummer des Steuerungsauftrags. Generell können Steuerungsaufträge nur dann von der Steuerung getriggert werden, wenn das Bediengerät im Online-Betrieb ist.

| Nr.       | Funktion  |   |
|-----------|---|---|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>                |   |
|           | Parameter 1   | Linkes Byte: -<br>Rechtes Byte: Stunden (0-23)                      |
|           | Parameter 2   | Linkes Byte: Minuten (0-59)<br>Rechtes Byte: Sekunden (0-59)        |
|           | Parameter 3   | -   |
| <b>15</b> | <b>Datum stellen (BCD-codiert)</b> <sup>3) 4)</sup> |   |
|           | Parameter 1   | Linkes Byte: -<br>Rechtes Byte: Wochentag<br>(1-7: Sonntag-Samstag) |
|           | Parameter 2   | Linkes Byte: Tag (1-31)<br>Rechtes Byte: Monat (1-12)               |

| Nr.       | Funktion   |  |
|-----------|--|--|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>   |  |
|           | Parameter 3  | Linkes Byte: Jahr  |
| <b>23</b> | <b>Benutzer anmelden</b>   |  |
|           | Meldet den Benutzer "PLC User" mit der im Parameter 1 übergebenen Gruppennummer am Bediengerät an.<br>Voraussetzung für die Anmeldung ist, dass die übergebene Gruppennummer im Projekt vorhanden ist.   |  |
|           | Parameter 1  | Gruppennummer 1 - 255  |
|           | Parameter 2, 3   | -  |
| <b>24</b> | <b>Benutzer abmelden</b>   |  |
|           | Meldet den aktuell angemeldeten Benutzer ab.<br>(Funktion entspricht der Systemfunktion "Abmelden")  |  |
|           | Parameter 1, 2, 3  | -  |
| <b>40</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |  |
|           | (Im S7-Format DATE_AND_TIME)<br>Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.   |  |
|           | Parameter 1, 2, 3  | -  |
| <b>41</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |  |
|           | (Im Format OP/MP)<br>Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.  |  |
|           | Parameter 1, 2, 3  | -  |
| <b>46</b> | <b>Variable aktualisieren</b>  |  |
|           | Veranlasst das Bediengerät den aktuellen Wert der Variablen aus der Steuerung zu lesen, deren Aktualisierungskennung mit dem im Parameter 1 übergebenen Wert übereinstimmt.<br>(Funktion entspricht der Systemfunktion "AktualisiereVariable") |  |
|           | Parameter 1  | 1 - 100  |
| <b>49</b> | <b>Meldepuffer löschen</b>   |  |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Warnings" aus dem meldepuffer.  |  |
|           | Parameter 1, 2, 3  | -  |
| <b>50</b> | <b>Meldepuffer löschen</b>   |  |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Errors" aus dem Meldepuffer   |  |
|           | Parameter 1, 2, 3  | -  |
| <b>51</b> | <b>Bildanwahl <sup>2)</sup></b>  |  |
|           | Parameter 1  | Bildnummer   |
|           | Parameter 2  | -  |
|           | Parameter 3  | Feldnummer   |
| <b>69</b> | <b>Datensatz aus Steuerung lesen <sup>1)</sup></b>   |  |
|           | Parameter 1  | Rezepturnummer (1-999)   |
|           | Parameter 2  | Datensatznummer (1-65535)  |
|           | Parameter 3  | 0: Vorhandenen Datensatz nicht überschreiben<br>1: Vorhandenen Datensatz überschreiben |

| Nr. | Funktion                                       |                           |
|-----|--|---------------------------|
| 14  | Uhrzeit stellen (BCD-codiert)                  |                           |
| 70  | Datensatz in Steuerung schreiben <sup>1)</sup> |                           |
|     | Parameter 1                                    | Rezepturnummer (1-999)    |
|     | Parameter 2                                    | Datensatznummer (1-65535) |
|     | Parameter 3                                    | -                         |

|    |   |
|----|---|
| 1) | Nur bei Geräten, die Rezepturen unterstützen.   |
| 2) | Auf den Bediengeräten OP 73, OP 77A und TP 177A wird der Steuerungsauftrag "Bildanwahl" auch dann ausgeführt, wenn die Bildschirmstatur geöffnet ist. |
| 3) | Beim Bediengerät KTP 600 BASIC PN wird der Wochentag ignoriert.   |
| 4) | Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" verwenden, wird der Wochentag ignoriert.  |

## Bereichszeiger "Datensatz"

## Bereichszeiger "Datensatz"

## Funktion

Bei der Übertragung von Datensätzen zwischen Bediengerät und Steuerung greifen beide Kommunikationspartner wechselseitig auf gemeinsame Kommunikationsbereiche in der Steuerung zu.

## Übertragungsarten

Für die Übertragung von Datensätzen zwischen Bediengerät und Steuerung gibt es zwei Möglichkeiten:

- Übertragung ohne Synchronisation
- Übertragung mit Synchronisation über das Datenfach

Datensätze werden immer direkt übertragen. D. h., die Variablenwerte werden direkt, ohne den Umweg über eine Zwischenablage, aus der Adresse gelesen oder in die Adresse geschrieben, die für die Variable projiziert ist.

## Übertragung von Datensätzen anstoßen

Für den Anstoß der Übertragung gibt es drei Möglichkeiten:

- Bedienung in der Rezepturanzeige
- Steuerungsaufträge  
Die Übertragung der Datensätze kann auch durch die Steuerung getriggert werden.
- Auslösen projektierter Funktionen

Wenn die Übertragung von Datensätzen durch eine projizierte Funktion oder einen Steuerungsauftrag getriggert wird, ist die Rezepturanzeige am Bediengerät weiterhin ungehindert bedienbar. Die Datensätze werden im Hintergrund übertragen.

Das gleichzeitige Abarbeiten mehrerer Übertragungsanforderungen ist jedoch nicht möglich. In diesem Fall lehnt das Bediengerät eine weitere Übertragung mit einer Systemmeldung ab.

### Übertragung ohne Synchronisation

Bei der asynchronen Übertragung von Datensätzen zwischen Bediengerät und Steuerung findet keine Koordination über gemeinsam benutzte Datenbereiche statt. Die Einrichtung eines Datenbereichs beim Projektieren ist daher nicht nötig.

Die asynchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Ein unkontrolliertes Überschreiben der Daten durch die Kommunikationspartner kann systembedingt ausgeschlossen werden.
- Die Steuerung braucht keine Informationen über die Rezepturnummer und die Datensatznummer.
- Die Übertragung von Datensätzen wird durch Bedienung am Bediengerät getriggert.

### Werte lesen

Beim Anstoß der Übertragung zum Lesen werden die Werte aus den Steuerungsadressen gelesen und zum Bediengerät übertragen.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die Werte werden in das Bediengerät geladen. Im Bediengerät können Sie diese weiterverarbeiten, z. B. Werte ändern, speichern etc.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte werden sofort auf dem Datenträger gespeichert.

### Werte schreiben

Beim Anstoß der Übertragung zum Schreiben werden die Werte in die Steuerungsadressen geschrieben.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die aktuellen Werte werden in die Steuerung geschrieben.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte vom Datenträger werden in die Steuerung geschrieben.

### Übertragung mit Synchronisation

Bei der synchronen Übertragung setzen beide Kommunikationspartner Status-Bits im gemeinsam benutzten Datenbereich. Dadurch können Sie in Ihrem Steuerungsprogramm ein unkontrolliertes gegenseitiges Überschreiben der Daten verhindern.



## Anwendung

Die synchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Die Steuerung ist der "aktive Partner" bei der Übertragung von Datensätzen.
- In der Steuerung werden Informationen über die Rezepturnummer und die Datensatznummer ausgewertet.
- Die Übertragung von Datensätzen wird per Steuerungsauftrag getriggert.

## Voraussetzungen

Damit Datensätze synchronisiert zwischen Bediengerät und Steuerung übertragen werden, müssen beim Projektieren folgende Voraussetzungen erfüllt sein:

- Ein Bereichszeiger ist eingerichtet: Editor "Kommunikation > Verbindungen" unter "Bereichszeiger".
- In der Rezeptur ist die Steuerung angegeben, mit der das Bediengerät die Übertragung der Datensätze synchronisiert:  
Editor "Rezepturen" im Inspektorfenster unter "Allgemein > Synchronisation > Einstellungen" die Auswahl "Koordinierte Übertragung der Datensätze"

## Aufbau des Datenbereichs

Der Datenbereich hat eine feste Länge von 5 Worten. Der Datenbereich ist wie folgt aufgebaut:

|         | 15                                    | 0 |
|---------|---------------------------------------|---|
| 1. Wort | Aktuelle Rezepturnummer (1 - 999)     |   |
| 2. Wort | Aktuelle Datensatznummer (0 - 65.535) |   |
| 3. Wort | Reserviert                            |   |
| 4. Wort | Status (0, 2, 4, 12)                  |   |
| 5. Wort | Reserviert                            |   |

- Status  
Das Statuswort (Wort 4) kann folgende Werte annehmen:

| Wert    |           | Bedeutung                            |
|---------|-----------|--------------------------------------|
| Dezimal | Binär     |                                      |
| 0       | 0000 0000 | Übertragung zulässig, Datenfach frei |
| 2       | 0000 0010 | Übertragung läuft.                   |
| 4       | 0000 0100 | Übertragung fehlerfrei beendet       |
| 12      | 0000 1100 | Übertragung mit Fehler beendet       |

## Ablauf der Übertragung bei Steuerungsauftrag

Die Übertragung von Datensätzen zwischen Bediengerät und Steuerung kann vom Bediengerät oder von der Steuerung initiiert werden.

Für diese Art der Übertragung stehen die beiden Steuerungsaufträge Nr. 69 und Nr. 70 zur Verfügung.

**Nr. 69: Datensatz aus Steuerung lesen ("SPS → DAT")**

Der Steuerungsauftrag Nr. 69 überträgt Datensätze von der Steuerung zum Bediengerät. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Linkes Byte (LB)   | Rechtes Byte (RB) |
|--------|--|-------------------|
| Wort 1 | 0  | 69                |
| Wort 2 | Rezepturnummer (1-999)   |                   |
| Wort 3 | Datensatznummer (1-65.535)   |                   |
| Wort 4 | Vorhandenen Datensatz nicht überschreiben: 0<br>Vorhandenen Datensatz überschreiben: 1 |                   |

**Nr. 70: Datensatz in Steuerung schreiben ("DAT → SPS")**

Der Steuerungsauftrag Nr. 70 überträgt Datensätze vom Bediengerät zur Steuerung. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Linkes Byte (LB)           | Rechtes Byte (RB) |
|--------|----------------------------|-------------------|
| Wort 1 | 0                          | 70                |
| Wort 2 | Rezepturnummer (1-999)     |                   |
| Wort 3 | Datensatznummer (1-65.535) |                   |
| Wort 4 | —                          |                   |

**Ablauf bei Lesen aus der Steuerung durch Steuerungsauftrag "SPS → DAT" (Nr. 69)**

| Schritt | Aktion  |                           |
|---------|---|---------------------------|
| 1       | Prüfung: Statuswort = 0?  |                           |
|         | Ja  | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese Werte in dem Datensatz, der im Steuerungsauftrag angegeben ist.   |                           |
| 4       | <ul style="list-style-type: none"> <li>Falls im Auftrag "Überschreiben" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben. Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>Falls im Auftrag "Nicht überschreiben" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                           |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                           |

**Ablauf bei Schreiben in die Steuerung durch Steuerungsauftrag "DAT → SPS" (Nr. 70)**

| Schritt | Aktion   |                           |
|---------|--|---------------------------|
| 1       | Prüfung: Statuswort = 0?   |                           |
|         | Ja   | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät holt die Werte des im Auftrag angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.   |                           |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                           |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                           |

**Ablauf der Übertragung bei Anstoß durch projizierte Funktion**

**Lesen aus der Steuerung durch projizierte Funktion**

| Schritt | Aktion  |                            |
|---------|---|----------------------------|
| 1       | Prüfung: Statuswort = 0?  |                            |
|         | Ja  | Nein                       |
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese in dem Datensatz ab, der in der Funktion angegeben ist.   |                            |
| 4       | <ul style="list-style-type: none"> <li>Falls in der Funktion bei "Überschreiben" "Ja" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben.<br/>Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>Falls in der Funktion bei "Überschreiben" "Nein" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                            |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                            |

**Schreiben in die Steuerung durch projizierte Funktion**

| Schritt | Aktion                   |      |
|---------|--------------------------|------|
| 1       | Prüfung: Statuswort = 0? |      |
|         | Ja                       | Nein |

| Schritt | Aktion   |                            |
|---------|--|----------------------------|
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät holt die Werte des in der Funktion angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.  |                            |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                            |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                            |

### Mögliche Fehlerursachen bei der Übertragung von Datensätzen

#### Mögliche Fehlerursachen

Falls die Übertragung von Datensätzen mit Fehler beendet wird, so kann dies u. a. folgende Ursachen haben:

- Variablen-Adresse in der Steuerung nicht eingerichtet
- Überschreiben von Datensätzen nicht möglich
- Rezepturnummer nicht vorhanden
- Datensatznummer nicht vorhanden

---

#### Hinweis

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

---

#### Hinweis

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach wurde auf "Übertragung beendet" gesetzt.
  - Der Status im Datenfach wurde auf "Übertragung mit Fehler beendet" gesetzt.
- 

### Reaktion auf fehlerbedingten Abbruch

Das Bediengerät reagiert auf einen fehlerbedingten Abbruch der Übertragung von Datensätzen wie folgt:

- Anstoß über Bedienung in der Rezepturanzeige  
Hinweise in der Statuszeile der Rezepturanzeige und Ausgabe von Systemmeldungen
- Anstoß über Funktion  
Ausgabe von Systemmeldungen
- Anstoß über Steuerungsauftrag  
Keine Rückmeldung am Bediengerät

Unabhängig davon können Sie den Status der Übertragung durch Abfragen des Statusworts im Datenfach auswerten.

## Kurven

### Allgemeines zu Kurven

#### Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt bei den Basic Panels zeitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Auto-Hotspot

#### Zeitgetriggerte Kurven

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein.

Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

### Kurvenanforderung und Kurvenübertragung

#### Kurvenanforderungsbereich

Wenn am Bediengerät ein Bild mit einer oder mehreren Kurven aufgeschlagen wird, setzt das Bediengerät die zugehörigen Bits im Kurvenanforderungsbereich. Nach Abwahl des Bildes setzt das Bediengerät die entsprechenden Bits im Kurvenanforderungsbereich zurück.

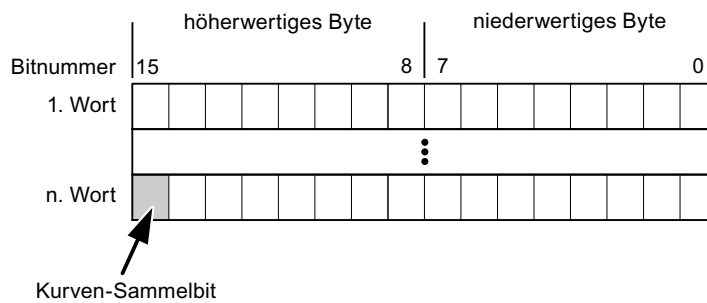
Über den Kurvenanforderungsbereich kann in der Steuerung ausgewertet werden, welche Kurve am Bediengerät gerade dargestellt wird. Kurven können auch ohne Auswertung des Kurvenanforderungsbereichs getriggert werden.

#### Kurvenübertragungsbereich 1

Dieser Bereich dient zur Triggerung von Kurven. Sie müssen in Ihrem Steuerungsprogramm das der Kurve zugeordnete Bit im Kurvenübertragungsbereich sowie das Kurven-Sammelbit setzen. Das Kurven-Sammelbit ist das letzte Bit im Kurvenübertragungsbereich.

Das Bediengerät erkennt die Triggerung. Das Bediengerät liest entweder einen Wert oder den gesamten Puffer aus der Steuerung. Danach setzt es das Kurvenbit und das Kurven-Sammelbit zurück.

Das folgende Bild zeigt den Aufbau eines Kurvenübertragungsbereichs.



Solange das Kurven-Sammelbit nicht zurückgesetzt wurde, darf der Kurvenübertragungsbereich nicht durch das Steuerungsprogramm verändert werden.

## Kurvenübertragungsbereich 2

Der Kurvenübertragungsbereich 2 ist für Kurven erforderlich, die mit Wechselpuffer projiziert werden. Der Kurvenübertragungsbereich 2 ist genauso aufgebaut wie der Kurvenübertragungsbereich 1.

## Wechselpuffer

Der Wechselpuffer ist ein zweiter Puffer für dieselbe Kurve, der bei der Projektierung eingerichtet werden kann.

Während das Bediengerät die Werte aus dem Puffer 1 liest, schreibt die Steuerung in den Puffer 2. Während das Bediengerät den Puffer 2 liest, schreibt die Steuerung in den Puffer 1. Dadurch wird verhindert, dass während des Auslesens der Kurve durch das Bediengerät die Kurvenwerte von der Steuerung überschrieben werden.

## Zulässige Datentypen für Kurven

### Für SIMATIC S7

In der Projektierung ordnen Sie jeder Kurve ein Bit zu. Zulässig sind Variablen vom Datentyp "Word" oder "Int" und Arrayvariablen vom Datentyp "Word" oder "Int".

## Meldungen

## Allgemeines zu Kurven

## Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt je nach Projektierung zeit- oder bitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Auto-Hotspot

---

#### **Hinweis**

Bei Basic Panels erfolgt das Auslesen des Werts zeitgetriggert.

---

### **Zeitgetriggerte Kurven**

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein. Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

### **Bitgetriggerte Kurven**

Durch Setzen eines Triggerbits in der Variablen Kurvenanforderung liest das Bediengerät entweder einen Kurvenwert oder einen gesamten Kurvenpuffer ein. Diese Einstellung wird in der Projektierung festgelegt. Bitgetriggerte Kurven werden in der Regel zur Darstellung sich schnell ändernder Werte verwendet. Ein Beispiel hierfür ist der Einspritzdruck bei der Fertigung von Kunststoffteilen.

Zum Auslösen bitgetriggelter Kurven müssen bei der Projektierung entsprechende externe Variablen im Editor "HMI-Variablen" angelegt und mit Kurvenbereichen verbunden werden. Über diese Kurvenbereiche kommunizieren dann Bediengerät und Steuerung miteinander.

Folgende Bereiche stehen für Kurven zur Verfügung:

- Kurvenanforderungsbereich
- Kurvenübertragungsbereich 1
- Kurvenübertragungsbereich 2 (nur bei Wechselpuffer erforderlich)

### **Quittierung von Meldungen**

#### **Vorgehensweise**

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

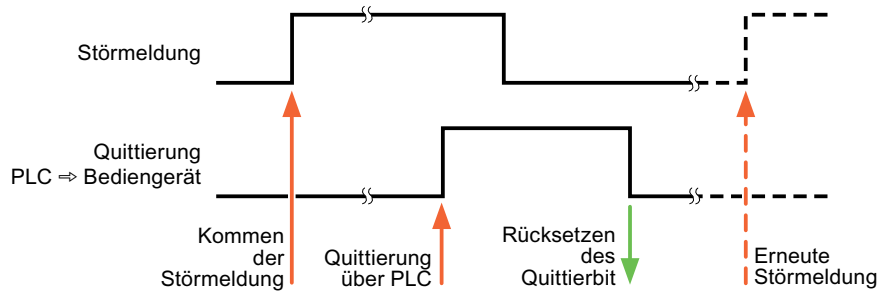
#### **Quittieren von der Steuerung**

Unter "PLC-Quittiervariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulsdiagramm.



### Quittierung am Bediengerät

Unter "HMI-Quittiervariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittiervariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittiervariable in die Steuerung eine gewisse Zeitspanne.

Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittiervariablen gesetzt. Die gesamte Quittiervariable wird dann vom Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

---

#### Hinweis

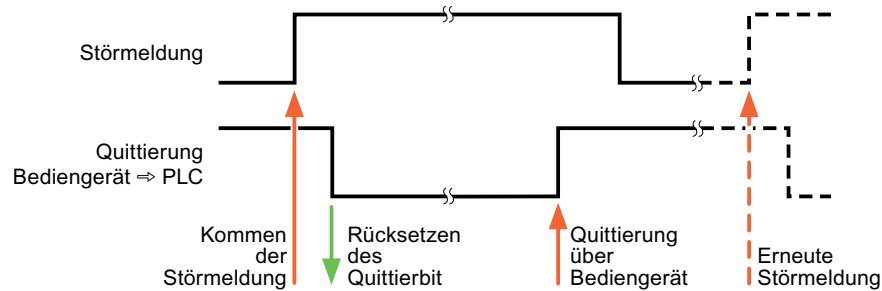
Alle seit dem letzten Runtime-Start quitierten Meldebites bleiben in der Quittiervariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittiervariablen der gesamte Bereich vom Bediengerät überschrieben wird.

---

Das nachfolgende Bild zeigt das Impulsdiagramm.





## LED-Abbild

### Funktion

Tastengeräte besitzen Leuchtdioden (LED) in den Funktionstasten. Diese Leuchtdioden können von der Steuerung aus angesteuert werden. Damit ist es z. B. möglich, dem Bediener situationsabhängig durch eine leuchtende LED zu signalisieren, welche Taste er drücken soll.

#### Hinweis

Die LED-Funktion ist Basic Panels nicht projektierbar.

### Voraussetzungen

Damit Sie eine Leuchtdiode ansteuern können, muss in der Steuerung eine LED-Variable oder eine Arrayvariable eingerichtet und bei der Projektierung als LED-Variable angegeben werden.

### LED-Zuordnung

Die Zuordnung der einzelnen Leuchtdioden zu den Bits in der LED-Variablen wird beim Projektieren der Funktionstasten festgelegt. Dabei wird für jede Funktionstaste im Eigenschaftsfenster, Gruppe "Allgemein" die "LED-Variable" und das zugeordnete "Bit" angegeben.

Die Bitnummer "Bit" bezeichnet das Erste von zwei aufeinander folgenden Bits, welche die folgenden LED-Zustände steuern:

| Bit n+ 1 | Bit n | LED-Funktion   |            |
|----------|-------|--|------------|
|          |       | alle Mobile Panels, alle Operator Panels,<br>alle Multi Panels | Panel PCs  |
| 0        | 0     | Aus  | Aus        |
| 0        | 1     | Schnelles Blinken  | Blinken    |
| 1        | 0     | Langsames Blinken  | Blinken    |
| 1        | 1     | Dauerlicht   | Dauerlicht |

### 10.8.6.5 Leistungsmerkmale der Kommunikation

#### Geräteabhängigkeit S7 1500

##### Geräteabhängigkeit

Wenn Sie mit dem TIA Portal V 12 SP1 Geräte aus einer früheren Version vom TIA Portal verwenden kann die Projektierung von Verbindungen zu bestimmten Bediengeräten nicht möglich sein.

Die Steuerung SIMATIC S7 1500 mit der Version V1.0 können Sie in folgenden Szenarien projektieren:

| TIA Portal | Basic Panels |
|------------|--------------|
| V11        | nein         |
| V12        | ja           |

#### Zulässige Datentypen für SIMATIC S7 1500

#### Zulässige Datentypen für Verbindungen mit SIMATIC S7 1500

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp      | Länge                     |
|---------------|---------------------------|
| BOOL          | 1 Bit                     |
| BYTE          | 1 Byte                    |
| WORD          | 2 Byte                    |
| DWORD         | 4 Byte                    |
| CHAR          | 1 Byte                    |
| INT           | 2 Byte                    |
| DINT          | 4 Byte                    |
| REAL          | 4 Byte                    |
| TIME          | 4 Byte                    |
| DATE          | 2 Byte                    |
| TIME_OF_DAY   | 4 Byte                    |
| S5TIME        | 2 Byte                    |
| COUNTER       | 2 Byte                    |
| TIMER         | 2 Byte                    |
| DATE_AND_TIME | 8 Byte                    |
| STRING        | (2+n) Byte, n = 0 bis 254 |
| DTL           | 12 Byte                   |
| LDT           | 8 Byte                    |

| Datentyp     | Länge  |
|--------------|--------|
| LINT         | 8 Byte |
| LREAL        | 8 Byte |
| LTIME        | 8 Byte |
| LTIME_OF_DAY | 8 Byte |
| SINT         | 1 Byte |
| UDINT        | 4 Byte |
| UINT         | 2 Byte |
| ULINT        | 8 Byte |
| USINT        | 1 Byte |

#### 10.8.6.6 Verbindungen im Editor "Verbindungen" projektieren

##### Parameter für die Verbindung

##### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät", "Netzwerk" und "Steuerung".

Prozesse visualisieren (Basic)  
10.8 Mit Steuerungen kommunizieren

The screenshot displays the 'Verbindungen' (Connections) window in WinCC Basic. At the top, a table lists the connection details:

| Name         | Kommunikationstreiber | Modus HMI-Zeitsynchronisation | Station | Partner | Knoten | Online                              | Kommentar |
|--------------|-----------------------|-------------------------------|---------|---------|--------|-------------------------------------|-----------|
| Connection_1 | SIMATIC S7 1500       | None                          |         |         |        | <input checked="" type="checkbox"/> |           |
| <Hinzufügen> |                       |                               |         |         |        |                                     |           |

Below the table, the 'Parameter' tab is active, showing configuration for 'TP1200 Comfort'. The 'Schnittstelle' (Interface) is set to 'IF1 B'. A visual diagram shows a connection line between the 'TP1200 Comfort' HMI and a 'Station' (Control System). The configuration is divided into three main sections:

- Bediengerät (Operator Station):**
  - Typ:  SIMATIC
  - Baudrate: 187500
  - Adresse: 1
  - Zugangspunkt: S7ONLINE
  - Einziger Master am Bus:
- Netzwerk (Network):**
  - Profil: DP
  - Höchste Stationsadresse (HSA): 31
  - Anzahl der Master: 1
- Steuerung (Control):**
  - Adresse: 2
  - Zugangspasswort: [Empty field]

## Ethernet-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie die Parameter für das Bediengerät im Netzwerk ein. Die Änderungen werden nicht automatisch auf das Bediengerät übertragen. Sie müssen die Einstellungen in der Systemsteuerung des Bediengeräts ändern.

- "Schnittstelle"  
Wenn Sie beim Projektieren direkt mit dem Bediengerät verbunden sind, können Sie die IP-Adresse des Bediengeräts in WinCC einrichten.

---

#### Hinweis

Wenn Sie die IP-Adresse bereits im Control-Panel des Bediengeräts eingerichtet haben, wird die IP-Adresse im Control Panel beim nächsten Laden überschrieben.

Wenn Sie "IP-Adresse auf anderem Weg beziehen" aktivieren, bleibt die bereits eingerichtete IP-Adresse im Control Panel beim nächsten Laden erhalten.

---

Beim Projekttransfer wird die IP-Adresse auf das Bediengerät übertragen. Die IP-Adresse des Bediengeräts richten Sie folgendermaßen ein:

- Klicken Sie auf das Bediengerät.
- Öffnen Sie den Editor "Gerätekonfiguration"
- Klicken Sie auf die Ethernet-Schnittstelle.
- Vergeben Sie im Inspektorfenster die IP-Adresse unter:  
"Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"
- "Adresse"  
Im Bereich "Adresse" vergeben Sie die IP-Adresse des Bediengeräts. Wenn Sie das WinCC-Projekt auf das Bediengerät transferieren, dann wird diese IP-Adresse direkt im Bediengerät eingerichtet.
- "Zugangspunkt"  
Der Zugangspunkt legt den Zugangspunkt für die PG/PC-Schnittstelle fest, über den der Kommunikationspartner erreicht werden kann.

### Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die IP-Adresse der S7-Baugruppe ein, an die das Bediengerät angeschlossen ist.
- "Zugangspasswort"  
Unter "Zugangspasswort" geben Sie ein Passwort ein. Das Passwort muss mit dem Passwort übereinstimmen, das Sie in der Steuerung hinterlegt haben.

---

#### Hinweis

Ein Passwort ist nur notwendig, wenn Sie in der Steuerung die Schutzstufe "Kompletter Schutz" hinterlegt haben.

Wenn die Schutzstufe "Kompletter Schutz" in der Steuerung hinterlegt ist und Sie kein Passwort eingeben, wird keine Verbindung zur Steuerung aufgebaut.

---

### PROFIBUS-Parameter

#### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie einmal die Parameter für das Bediengerät im Netzwerk ein. Die Änderung gilt für jeden Kommunikationspartner.

- "Typ"  
Legt den benutzten physikalischen Anschluss fest.
- "Schnittstelle"  
Unter "Schnittstelle" wählen Sie die Schnittstelle des Bediengeräts aus, über die das Bediengerät am PROFIBUS-Netz angeschlossen ist.
- "Baudrate"  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit der Daten im Netzwerk ein. Die Baudrate wird durch das langsamste ans Netzwerk angeschlossene Bediengerät festgelegt. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

#### Hinweis

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5 Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Adresse"  
Unter "Adresse" stellen Sie die PROFIBUS-Adresse des Bediengeräts ein. Die PROFIBUS-Adresse muss im PROFIBUS-Netz eindeutig sein.

- "Einzigster Master am Bus"  
Deaktiviert eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des Bediengeräts an das Netz. Eine passive Station (Slave) kann nur dann Daten senden, wenn Sie von einer aktiven Station (Master) dazu aufgefordert wird.  
Bei S7-200 müssen Sie ein Bediengerät als Master einstellen.
- "Zugangspunkt"  
Der Zugangspunkt legt einen logischen Gerätenamen fest, über den der Kommunikationspartner erreicht wird.

### Parameter für das Netzwerk

Unter "Netzwerk" stellen Sie die Parameter für das PROFIBUS-Netz ein, an dem das Bediengerät eingebunden ist.

- "Profil"  
Unter "Profil" wählen Sie das Netzprofil aus, das im Netzwerk verwendet wird. Stellen Sie unter "Profil" "DP", "Universal" oder "Standard" ein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Höchste Stationsadresse"  
Unter "Höchste Stationsadresse" stellen Sie die höchste Stationsadresse ein. Die höchste Stationsadresse muss größer gleich der größten tatsächlichen PROFIBUS-Adresse sein. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

#### Hinweis

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Anzahl der Master"  
Unter "Anzahl der Master" stellen Sie die Anzahl der Master im PROFIBUS-Netz ein. Diese Angabe ist erforderlich, damit die Busparameter korrekt berechnet werden.

### Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die PROFIBUS-Adresse der S7-Baugruppe (CPU, FM oder CP) ein, an die das Bediengerät angeschlossen ist.
- "Zugangspasswort"  
Unter "Zugangspasswort" geben Sie ein Passwort ein. Das Passwort muss mit dem Passwort übereinstimmen, das Sie in der Steuerung hinterlegt haben.

---

#### Hinweis

Ein Passwort ist nur notwendig, wenn Sie in der Steuerung die Schutzstufe "Kompletter Schutz" hinterlegt haben.

Wenn die Schutzstufe "Kompletter Schutz" in der Steuerung hinterlegt ist und Sie kein Passwort eingeben, wird keine Verbindung zur Steuerung aufgebaut.

---

## 10.8.7 Mit SIMATIC S7 1200 kommunizieren

### 10.8.7.1 Kommunikation mit SIMATIC S7 1200

#### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und der Steuerung SIMATIC S7 1200 beschrieben.

Folgende Kommunikationskanäle können Sie für die Steuerung SIMATIC S7 1200 projektieren:

- PROFINET
- PROFIBUS

#### HMI-Verbindung für die Kommunikation

Verbindungen von Bediengerät und SIMATIC S7 1200 projektieren Sie im Editor "Geräte & Netze". Wenn Sie ein Bediengerät mit serielltem Anschluss projektieren, müssen Sie ein PROFIBUS-fähiges Kommunikationsmodul an die SIMATIC S7 1200 projektieren.

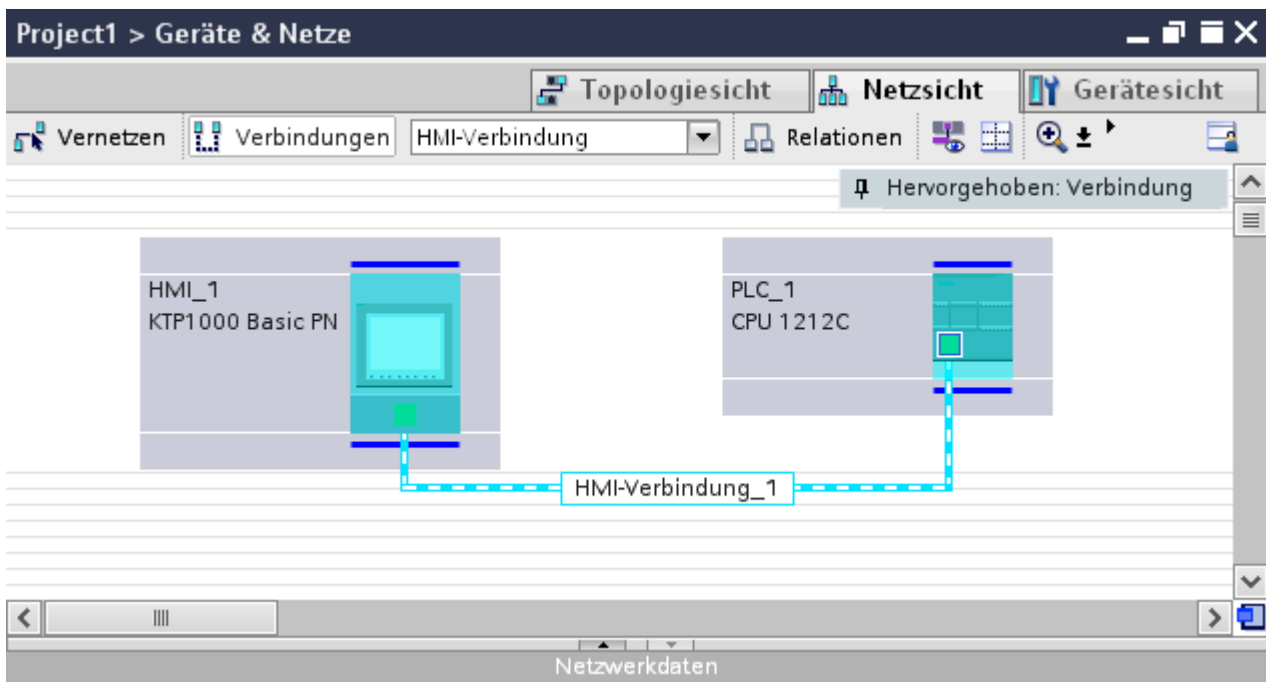


## 10.8.7.2 Kommunikation über PROFINET

### Kommunikation über PROFINET

#### HMI-Verbindungen über PROFINET

Wenn Sie ein Bediengerät und eine SIMATIC S7 1200 in das Projekt eingefügt haben, verbinden Sie die beiden PROFINET-Schnittstellen im Editor "Geräte & Netze" miteinander.



Sie können auch mehrere Bediengeräte an eine SIMATIC S7 1200 anschließen und mehrere SIMATIC S7 1200 an ein Bediengerät anschließen. Die maximale Anzahl der Kommunikationspartner, die Sie an ein Bediengerät anschließen können, ist abhängig vom verwendeten Bediengerät.

Weitere Informationen dazu finden Sie in der Dokumentation zum jeweiligen Bediengerät.

#### HMI-Verbindung im Editor "Geräte & Netze"

Die HMI-Verbindung über PROFINET zwischen der Steuerung mit dem Bediengerät projektieren Sie im Editor "Geräte & Netze".

#### Verbindung im Editor "Verbindungen"

Alternativ projektieren Sie die Verbindung über PROFINET zwischen Steuerung und Bediengerät im Editor "Verbindungen" des Bediengeräts.

## HMI-Verbindung über PROFINET projektieren

### Einleitung

Eine HMI-Verbindung über PROFINET bzw. Ethernet zwischen Bediengeräten und einer SIMATIC S7 1200 projektieren Sie im Editor "Geräte & Netze".



#### **VORSICHT**

##### **Kommunikation über Ethernet**

Bei Ethernet-basierter Kommunikation ist der Endanwender für die Sicherheit seines Datennetzes selbst verantwortlich.

Gezielte Angriffe können zur Überlast des Geräts führen und die Funktionsfähigkeit beeinträchtigen.

### Voraussetzungen

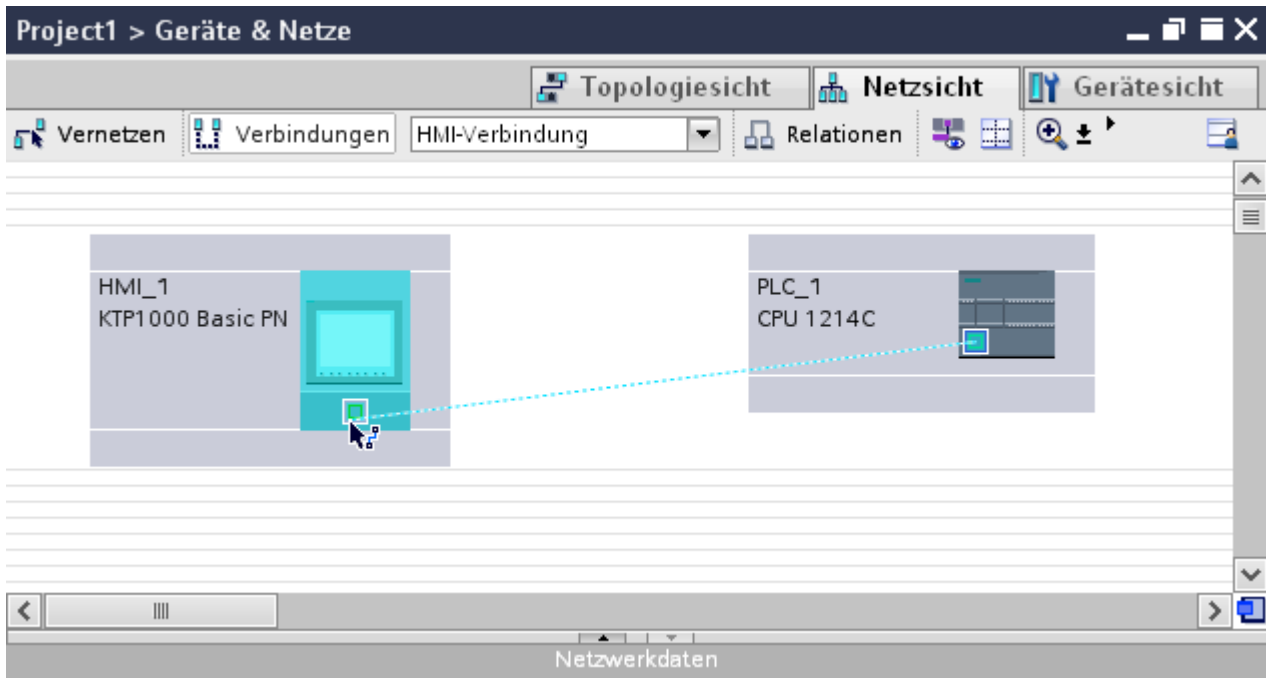
Folgende Kommunikationspartner sind im Editor "Geräte & Netze" angelegt:

- SIMATIC S7 1200
- Bediengerät mit PROFINET- bzw. Ethernet-Schnittstelle

### Vorgehensweise

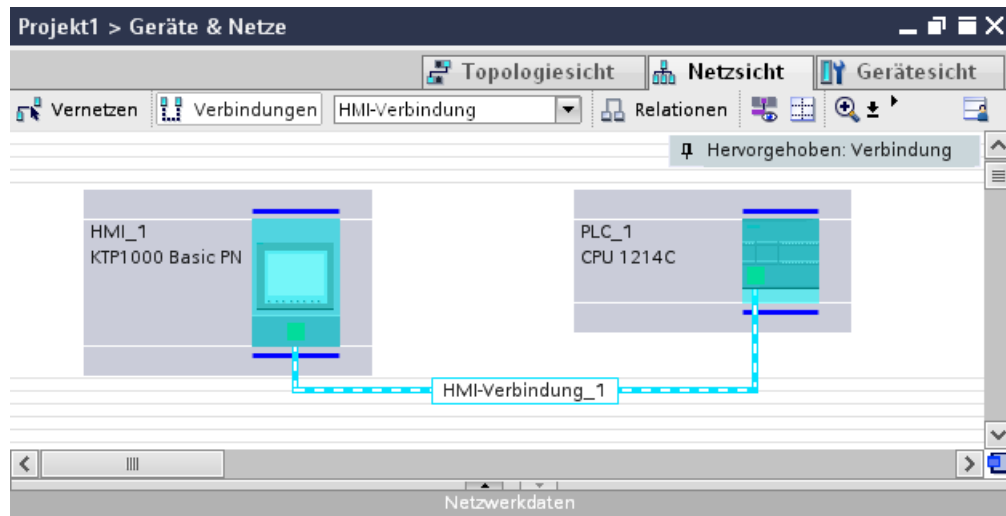
1. Doppelklicken Sie in der Projektnavigation den Eintrag "Geräte & Netze".  
In der Netzsicht werden die im Projekt vorhandenen Kommunikationspartner grafisch dargestellt.
2. Klicken Sie auf die Schaltfläche "Verbindungen" und wählen Sie den Verbindungstyp "HMI-Verbindung".  
Die für eine Verbindung verfügbaren Geräte werden farbig markiert dargestellt.

3. Klicken Sie in die PROFINET-Schnittstelle der Steuerung und ziehen Sie per Drag&Drop eine Verbindung zur PROFINET- bzw. Ethernet-Schnittstelle des Bediengeräts.



4. Klicken Sie auf die Verbindungslinie.

5. Klicken Sie auf "HMI-Verbindung Hervorheben" und wählen Sie die HMI-Verbindung aus.



Im Inspektorfenster wird die Verbindung grafisch dargestellt.

6. Klicken Sie auf die Kommunikationspartner in der "Netzsicht" und ändern Sie im Inspektorfenster die PROFINET-Parameter nach den Anforderungen ihres Projekts. Nähere Hinweise hierzu finden Sie im Kapitel "PROFINET-Parameter (Seite 3550)".

---

#### Hinweis

Die erstellte HMI-Verbindung wird auch im tabellarischen Bereich des Editors auf der Registerkarte "Verbindungen" angezeigt. In der Tabelle kontrollieren Sie die Verbindungsparameter.

Den lokalen Verbindungsnamen der Verbindung können Sie nur in der Tabelle ändern.

---

## Ergebnis

Sie haben eine Verbindung zwischen einem Bediengerät und einer SIMATIC S7 1200 angelegt. Die Verbindungsparameter IP-Adresse und Subnetzmaske sind projektiert.

## PROFINET-Parameter

### PROFINET-Parameter für die HMI-Verbindung

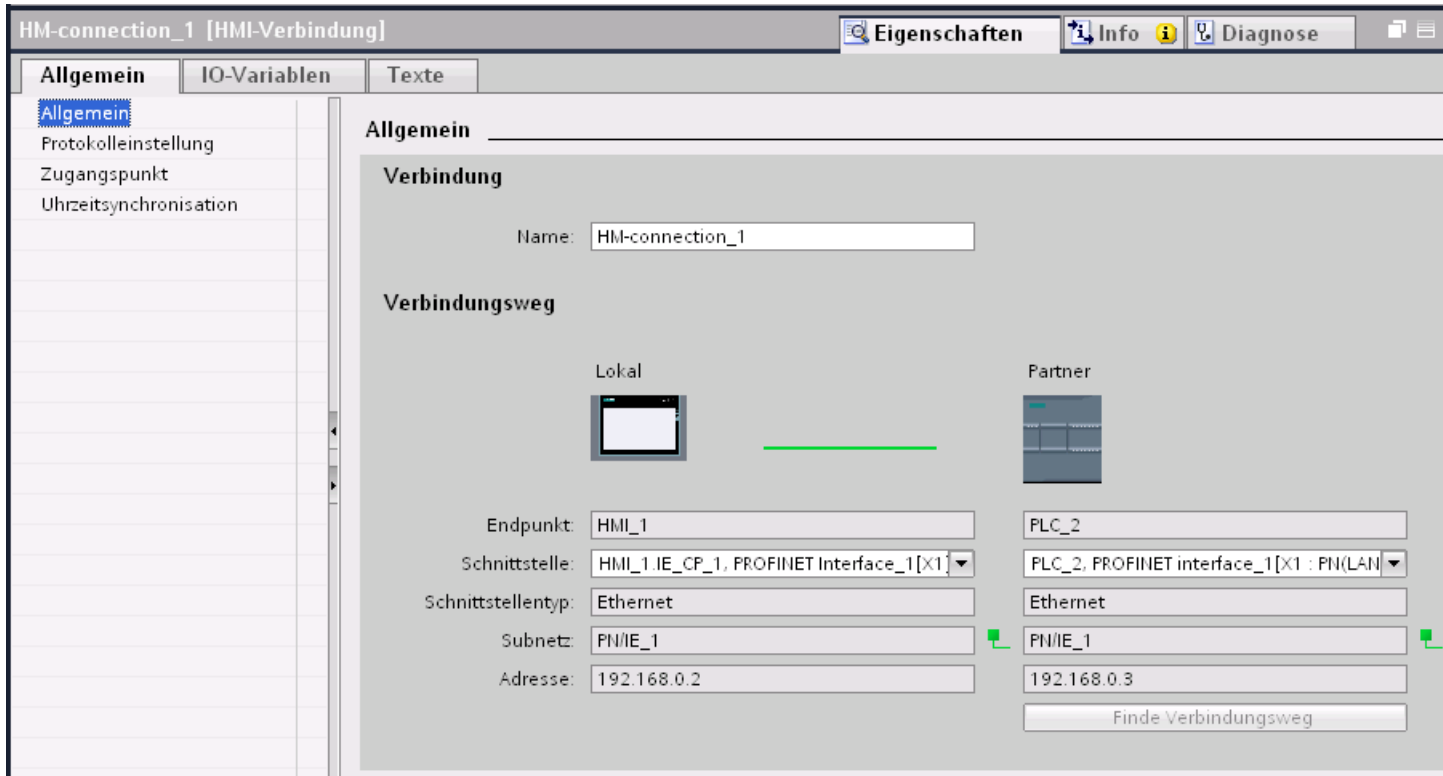
### PROFINET-Parameter für die HMI-Verbindung

Eine Übersicht über die projektierten Parameter einer HMI-Verbindung finden Sie in den Eigenschaften der HMI-Verbindung.

Änderungen sind in diesem Inspektorfenster nur eingeschränkt möglich.

## Parameter der HMI-Verbindung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die HMI-Verbindung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der HMI-Verbindung.



## Verbindung

Im Bereich "Verbindung" wird die HMI-Verbindung angezeigt, die zwischen den Geräten angelegt wurde.

Der Name der HMI-Verbindung ist in diesem Bereich editierbar.

## "Verbindungsweg"

Im Bereich "Verbindungsweg" werden die Kommunikationspartner der gewählten HMI-Verbindung und deren PROFINET-Parameter angezeigt. Einige der gezeigten Bereiche sind in diesem Dialog nicht editierbar.

- "Endpunkt"  
Zeigt den Gerätenamen an. Dieser Bereich ist nicht editierbar.
- "Schnittstelle"  
Zeigt die gewählte Schnittstelle des Geräts an. Abhängig vom Gerät können Sie zwischen mehreren Schnittstellen wählen.
- "Schnittstellen-Typ"  
Zeigt den gewählten Schnittstellen-Typ an. Dieser Bereich ist nicht editierbar.

- "Subnetz"  
Zeigt das gewählte Subnetz an. Dieser Bereich ist nicht editierbar.
- "Adresse"  
Zeigt die gewählte IP-Adresse des Geräts an. Dieser Bereich ist nicht editierbar.
- Schaltfläche "Finde Verbindungsweg"  
Ermöglicht das nachträgliche Spezifizieren von Verbindungen.

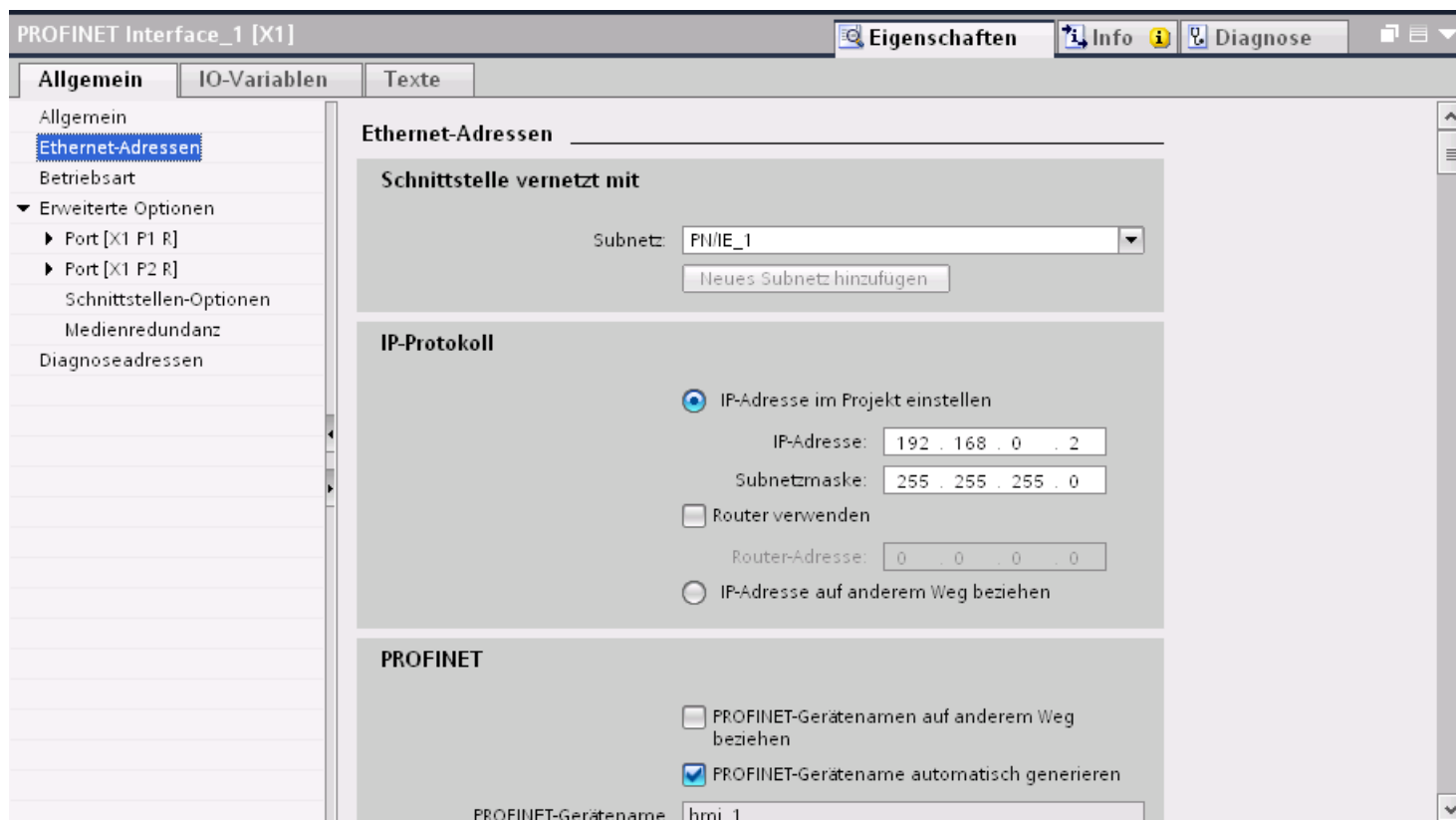
## PROFINET-Parameter für das Bediengerät

## PROFINET-Parameter für das Bediengerät

Eine Übersicht über die projektierten Parameter des Bediengeräts finden Sie in den Eigenschaften des Bediengeräts.

## PROFINET-Parameter des Bediengeräts anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf das Bediengerät.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter des Bediengeräts.



### "Schnittstelle vernetzt mit"

Im Bereich "Schnittstelle vernetzt mit" wählen Sie das Subnetz der HMI-Verbindung aus, über die das Bediengerät an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "IP-Protokoll"

- "IP-Adresse im Projekt einstellen"  
Wenn Sie das WinCC-Projekt auf das Bediengerät transferieren, dann wird diese IP-Adresse direkt im Bediengerät eingerichtet.

---

#### Hinweis

Bei Bediengeräten mit dem Betriebssystem Windows CE 3.0 erfolgt automatisch ein Neustart.

Bediengeräte mit Windows CE 3.0:

- OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"
- 
- "Subnetzmaske"  
Im Bereich "Subnetzmaske" vergeben Sie Daten der Subnetzmaske.
  - "IP-Router verwenden"  
Wenn Sie einen IP-Router verwenden, aktivieren Sie "IP-Router verwenden" und geben Sie die Routeradresse im Feld "Routeradresse" ein.
  - "IP-Adresse auf anderem Weg beziehen"  
Wenn die Funktion "IP-Adresse auf anderem Weg beziehen" aktiviert ist, dann wird die IP-Adresse nicht aus dem Projekt übernommen. Sie müssen die IP-Adresse direkt im Control Panel des Bediengeräts eingeben.

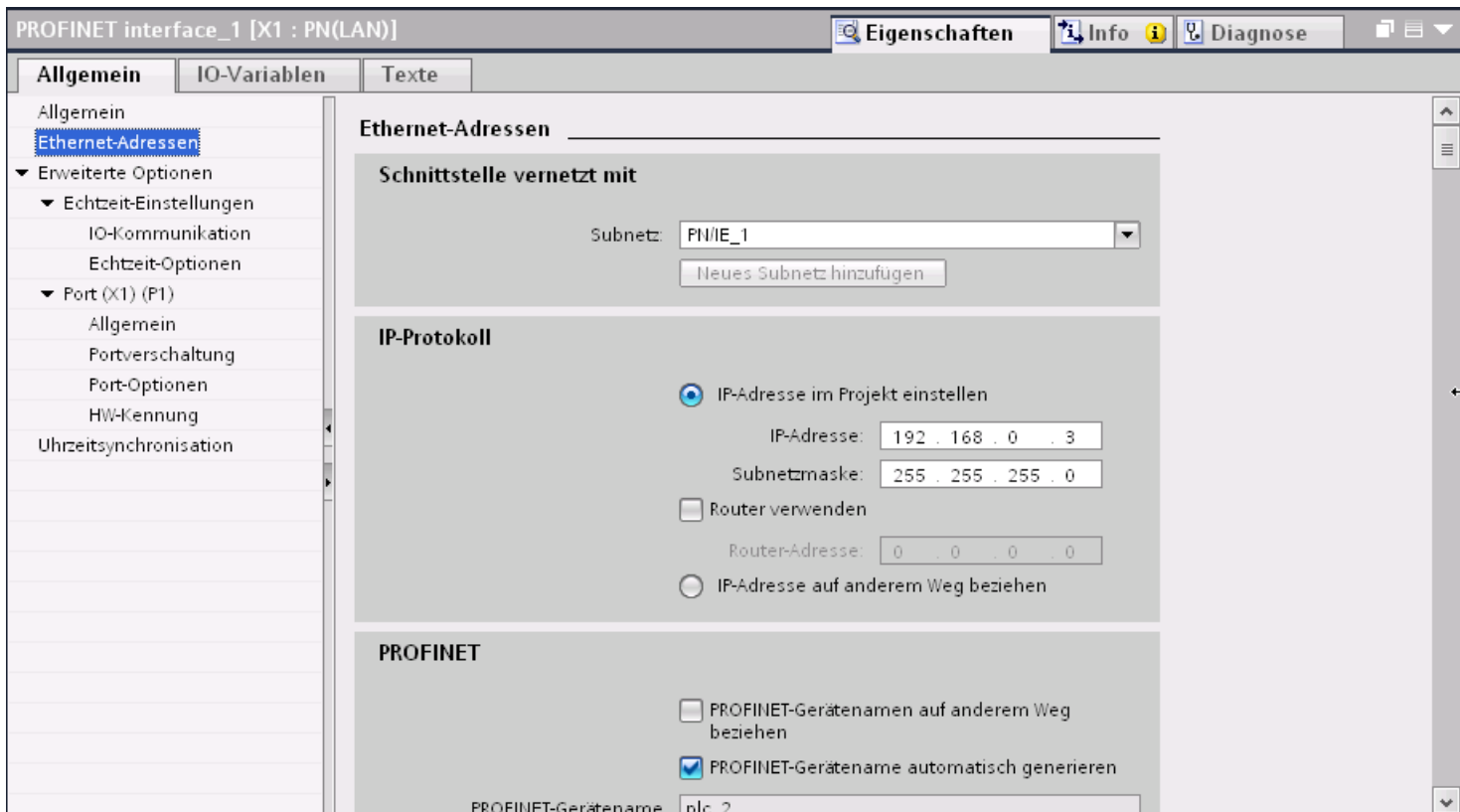
### PROFINET-Parameter für die Steuerung

### PROFINET-Parameter für die Steuerung

Eine Übersicht über die projektierten Parameter finden Sie in den Eigenschaften der Steuerung.

## PROFINET-Parameter der Steuerung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die Steuerung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der Steuerung.



### "Schnittstelle vernetzt mit"

Im Bereich "Subnetz" wählen Sie das Subnetz der HMI-Verbindung aus, über die die Steuerung an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "IP-Protokoll"

- "Schnittstellen-Typ"  
Abhängig vom Bediengerätetyp stehen verschiedene Schnittstellen zur Auswahl.
- "IP-Adresse"  
Im Bereich "IP-Adresse" vergeben Sie die IP-Adresse des Bediengeräts.
- "Subnetzmaske"  
Im Bereich "Subnetzmaske" vergeben Sie Daten der Subnetzmaske.  
Wenn Sie einen IP-Router verwenden, aktivieren Sie "IP-Router verwenden" und geben Sie die Routeradresse im Feld darunter ein.



## Netzkonfiguration Industrial Ethernet

### Regeln für die Netzkonfiguration

Die Ethernet-Schnittstellen der Geräte besitzen als IP-Adresse eine Default-Adresse, die von Ihnen geändert werden kann.

### IP-Adresse

Die IP-Parameter sind sichtbar, wenn die kommunikationsfähigen Geräte das TCP/IP-Protokoll unterstützen.

Die IP-Adresse besteht aus 4 Dezimalzahlen mit dem Wertebereich 0 bis 255. Die Dezimalzahlen sind durch einen Punkt voneinander getrennt.

Beispiel: 140.80.0.2

Die IP-Adresse setzt sich folgendermaßen zusammen:

- Der Adresse des (Sub-) Netzes
- Der Adresse des Teilnehmers (im allgemeinen auch Host oder Netzknoten genannt)

### Subnetzmaske

Die Subnetzmaske trennt diese beiden Adressen. Sie bestimmt, welcher Teil der IP-Adresse das Netz adressiert und welcher Teil der IP-Adresse den Teilnehmer adressiert.

Die gesetzten Bits der Subnetzmaske bestimmen den Netzteil der IP-Adresse.

Beispiel:

Subnetzmaske: 255.255.0.0 = 11111111.11111111.00000000.00000000

Im Beispiel für die oben genannte IP-Adresse hat die hier gezeigte Subnetzmaske folgende Bedeutung:

Die ersten 2 Bytes der IP-Adresse bestimmen das Subnetz - also 140.80. Die letzten beiden Bytes adressieren den Teilnehmer - also 0.2.

Allgemein gilt:

- Die Netzadresse ergibt sich aus der UND-Verknüpfung von IP-Adresse und Subnetzmaske.
- Die Teilnehmeradresse ergibt sich aus der UND-NICHT-Verknüpfung von IP-Adresse und Subnetzmaske.

### Zusammenhang IP-Adresse und Default-Subnetzmaske

Es gibt eine Vereinbarung hinsichtlich der Zuordnung von IP-Adressbereichen und so genannten "Default-Subnetzmasken". Die erste Dezimalzahl der IP-Adresse (von links)

bestimmt den Aufbau der Default-Subnetzmaske hinsichtlich der Anzahl der Werte "1" (binär) wie folgt:

| IP-Adresse (dez.) | IP-Adresse (bin.)     | Adressklasse | Default-Subnetzmaske |
|-------------------|-----------------------|--------------|----------------------|
| 0 bis 126         | 0xxxxxxx.xxxxxxxx.... | A            | 255.0.0.0            |
| 128 bis 191       | 10xxxxxx.xxxxxxxx...  | B            | 255.255.0.0          |
| 192 bis 223       | 110xxxxx.xxxxxxxx...  | C            | 255.255.255.0        |

**Hinweis**

**Wertebereich für erste Dezimalstelle**

Für die erste Dezimalzahl der IP-Adresse ist auch ein Wert zwischen 224 und 255 möglich (Adressklasse D etc). Dies ist jedoch nicht empfehlenswert, da für diese Werte keine Adressprüfung erfolgt.

**Weitere Subnetze maskieren**

Über die Subnetzmaske können Sie ein Subnetz, das einer der Adressklassen A, B oder C zugeordnet ist, weiter strukturieren und "private" Subnetze bilden, indem Sie weitere niederwertige Stellen der Subnetzmaske auf "1" setzen. Pro jedem auf "1" gesetztem Bit verdoppelt sich die Anzahl der "privaten" Netze und halbiert sich die Anzahl der darin enthaltenen Teilnehmer. Nach außen wirkt das Netzwerk nach wie vor wie ein einzelnes Netzwerk.

Beispiel:

Sie ändern bei einem Subnetz der Adressklasse B (z. B. IP-Adresse 129.80.xxx.xxx) die Default-Subnetzmaske wie folgt:

| Masken               | Dezimal       | Binär                               |
|----------------------|---------------|-------------------------------------|
| Default-Subnetzmaske | 255.255.0.0   | 11111111.11111111.00000000.00000000 |
| Subnetzmaske         | 255.255.128.0 | 11111111.11111111.10000000.00000000 |

Ergebnis:

Alle Teilnehmer mit Adressen von 129.80.001.xxx bis 129.80.127.xxx befinden sich auf einem Subnetz, alle Teilnehmer mit Adressen von 129.80.128.xxx bis 129.80.255.xxx auf einem anderen Subnetz.

**Netzübergang (Router)**

Die Netzübergänge (Router) haben die Aufgabe, die Subnetze zu verbinden. Wenn ein IP-Datagramm an ein anderes Netzwerk geschickt werden soll, dann muss es zunächst an einen Router vermittelt werden. Damit das möglich ist, müssen Sie in diesem Fall für jeden Teilnehmer des Subnetzes die Adresse des Routers eingeben.

Die IP-Adresse eines Teilnehmers im Subnetz und die Adresse des Netzübergangs (Router) dürfen nur an den Stellen unterschiedlich sein, an denen in der Subnetzmaske "0" steht.

## Schutz der Kommunikation

### Schutzstufen

Wenn Sie die Kommunikation von Steuerung und Bediengerät schützen wollen, dann können Sie Schutzstufen für die Kommunikation vergeben.

Für eine S7-1500-CPU können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in einer Tabelle eingegeben, so dass jedem Passwort genau eine Schutzstufe zugeordnet ist.

Wie das Passwort wirkt, steht in der Spalte "Schutz".

### Beispiel

Sie wählen die bei der Projektierung der Steuerung die Schutzstufe "Kompletter Schutz" für eine Standard-CPU (d. h. keine F-CPU) und geben für jede in der Tabelle darüber liegenden Schutzstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist die CPU komplett geschützt. Nicht einmal HMI-Zugriffe sind möglich.

Für Nutzer, die eines der parametrisierten Passworte kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile 1 (Kein Schutz) wirkt, als wäre die CPU ungeschützt. Nutzer, die dieses Passwort kennen, haben uneingeschränkten Zugriff auf die CPU.
- Das Passwort in Zeile 2 (Schreibschutz) wirkt, als wäre die CPU schreibgeschützt. Trotz Passwort-Kennntnis haben Nutzer, die dieses Passwort kennen, nur lesenden Zugriff auf die CPU.
- Das Passwort in Zeile 3 (Schreib- und Leseschutz) wirkt, als wäre die CPU schreib- und lesegeschützt, so dass für die Nutzer, die dieses Passwort kennen, nur HMI-Zugriffe möglich sind.

## Zugangspasswort für die HMI-Verbindung

### Einleitung

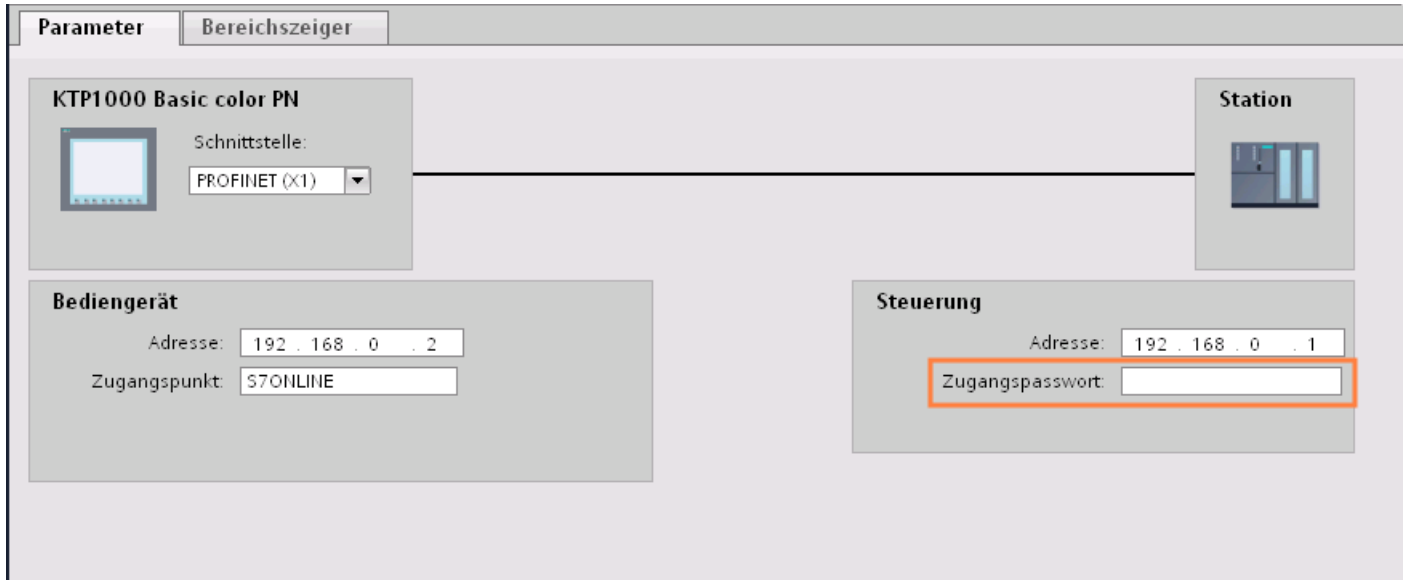
Den Zugriff auf eine Steuerung sichern Sie durch die Vergabe eines Passworts ab.

Bei der Projektierung der Passwort vergeben Sie das Passwort.

Ab Schutzstufe "Kompletter Schutz" ist es zwingend notwendig das Passwort einzugeben. Wenn Sie das Passwort nicht oder falsch eingeben wird keine Kommunikation zur Steuerung aufgebaut.

### Zugangspasswort eingeben

Im Editor "Verbindungen" geben Sie das Zugangspasswort für die Steuerung ein.



### Port-Optionen festlegen

### Festlegen der Port-Optionen

### Verbindungseinstellungen für den PROFINET-IO-Port ändern

Bei Bedarf können Sie die Netzwerkeinstellungen für den PROFINET-IO-Port ändern. Standardmäßig werden die Einstellungen automatisch festgelegt, was im Normalfall eine problemlose Kommunikation gewährleistet.

## Einstellmöglichkeiten Übertragungsrate/Duplex

Abhängig von dem gewählten Gerät können Sie folgende Einstellungen für "Übertragungsrate/Duplex" vornehmen:

- Automatische Einstellung  
Empfohlene Default-Einstellung des Ports. Die Übertragungseinstellungen mit dem Partner-Port werden automatisch "ausgehandelt". In der Voreinstellung ist automatisch auch die Option "Autonegotiation aktivieren" aktiviert, d. h. Sie können Crosskabel oder Patchkabel für den Anschluss verwenden.
- TP/ITP mit x Mbit/s. Vollduplex (Halbduplex)  
Einstellung des Übertragungsrate und des Vollduplex/Halbduplex-Betriebs. Die Wirkung ist abhängig von der eingestellten Option "Autonegotiation aktivieren":
  - Autonegotiation aktiviert  
Sie können sowohl Crosskabel als auch Patchkabel verwenden.
  - Autonegotiation deaktiviert  
Achten Sie darauf, dass Sie das korrekte Kabel verwenden (Patch- oder Crosskabel)! Auch bei dieser Einstellung wird der Port überwacht.
- Deaktiviert  
Abhängig vom Baugruppentyp kann in der Klappliste die Option "deaktiviert" angeboten werden. Sie haben damit beispielsweise die Möglichkeit, den Zugang zu einem unbenutzten Port aus Sicherheitsgründen zu unterbinden. Bei dieser Einstellung werden keine Diagnoseereignisse erzeugt.

## Option "Überwachen"

Mit dieser Option wird die Portdiagnose aktiviert bzw. deaktiviert. Beispiele für Portdiagnose: Der Link-Status wird überwacht, d. h. Diagnose bei Link-Down erzeugt und bei Fiber Optic Ports wird die Systemreserve überwacht.

## Option "Autonegotiation aktivieren"

Die Einstellung für Autonegotiation ist nur änderbar, wenn ein konkretes Medium (z. B. TP 100 Mbit/s vollduplex) ausgewählt ist. Ob ein konkretes Medium eingestellt werden kann, hängt von den Eigenschaften des Moduls ab.

Ist die Autonegotiation deaktiviert, dann wird die feste Einstellung des Ports erzwungen, ähnlich wie das z. B. bei einem priorisierten Hochlaufs des IO-Devices erforderlich ist.

Sie müssen für gleiche Einstellungen beim Partner-Port sorgen, da bei dieser Option die Betriebsparameter des angeschlossenen Netzes nicht erkannt werden und entsprechend die Datenübertragungsrate und Übertragungsmodus nicht optimal eingestellt werden können.

---

### Hinweis

STEP 7 übernimmt bei einem verschalteten lokalen Port die Einstellung für den Partner Port, falls der Partner Port die Einstellung unterstützt. Falls der Partner Port die Einstellung nicht unterstützt, wird eine Fehlermeldung erzeugt.

---

## Verkabelungsregeln bei deaktivierter Autonegotiation

### Voraussetzung

Sie haben z. B. zur Beschleunigung der Hochlaufzeit des IO-Devices für den betreffenden Port folgende Einstellungen vorgenommen:

- Feste Übertragungsrate
- Autonegotiation inkl. Autocrossing deaktiviert

Die Zeit für das Aushandeln der Übertragungsrate beim Hochlauf wird somit eingespart.

Wenn Sie Autonegotiation deaktiviert haben, müssen Sie die Verkabelungsregeln beachten.

## Verkabelungsregeln bei deaktivierter Autonegotiation

PROFINET-Geräte besitzen folgende zwei Arten von Ports:

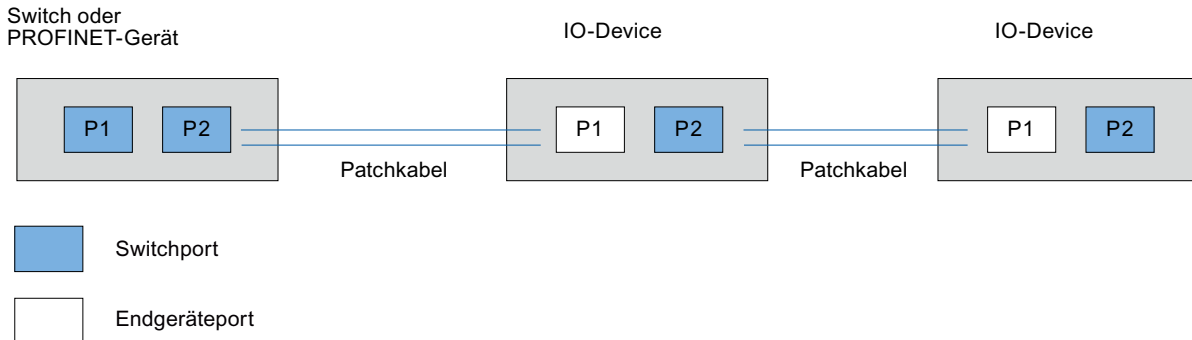
| Art des Ports                                  | PROFINET-Geräte   | Bemerkung  |
|--|---|--|
| Switchport mit gekreuzter Pinbelegung          | Bei IO-Devices: Port 2<br>Bei S7-CPU's mit 2 Ports: Port 1 und Port 2 | Gekreuzte Pinbelegung bedeutet, dass die Pinbelegung der Ports für Senden und Empfangen zwischen den betreffenden PROFINET-Geräten intern vertauscht werden. |
| Endgeräteport mit nicht gekreuzter Pinbelegung | Bei IO-Devices: Port 1<br>Bei S7-CPU's mit einem Port: Port 1         | -  |

### Gültigkeit der Verkabelungsregeln

Die im folgenden Absatz beschriebenen Verkabelungsregeln gelten ausschließlich für den Fall, dass Sie eine feste Porteinstellung vorgenommen haben.

## Regeln für die Verkabelung

Mehrere IO-Devices können Sie mit einem Kabeltyp (Patch-Kabel) in Linie verbinden. Dazu verbinden Sie Port 2 des IO-Devices (Dezentrale Peripherie) mit dem Port 1 des nächsten IO-Devices. In der folgenden Grafik ist ein Beispiel mit zwei IO-Devices dargestellt.



## Begrenzungen am Port

### Voraussetzung

Um mit Boundaries ("Begrenzungen") arbeiten zu können, muss das jeweilige Gerät mehr als einen Port besitzen. Wenn das PROFINET keine Boundaries-Einstellungen unterstützt, werden sie auch nicht angezeigt.

### Boundaries aktivieren

Unter "Boundaries" versteht man Grenzen für die Übertragung bestimmter Ethernet-Frames. Es können folgende Begrenzungen an einem Port gesetzt werden:

- "Ende der Erfassung erreichbarer Teilnehmer"  
DCP-Frames zur Erfassung erreichbarer Teilnehmer werden nicht weitergeleitet. Hinter diesem Port liegende Teilnehmer werden nicht mehr unter "erreichbare Teilnehmer" in der Projektnavigation angezeigt. Hinter diesem Port liegende Teilnehmer können von der CPU nicht mehr erreicht werden.
- "Ende der Topologieerkennung"  
LLDP-Frames (Link Layer Discovery Protocol) zur Topologieerkennung werden nicht weitergeleitet.
- "Ende der Sync-Domain"  
Sync-Frames, die zur Synchronisierung von Teilnehmern innerhalb einer Sync-Domain übertragen werden, werden nicht weitergeleitet.  
Wenn Sie z. B. ein PROFINET-Gerät mit mehr als zwei Ports in einem Ring betreiben, dann sollten Sie das Einspeisen von Sync-Frames in den Ring durch Setzen einer Sync-Boundary (an den Ports, die nicht im Ring sind) verhindern.  
Weiteres Beispiel: Möchten Sie mehrere Sync-Domains nutzen, dann projektieren Sie eine Sync-Domain-Boundary für den Port, der mit einem PROFINET-Gerät der jeweils anderen Sync-Domain verbunden ist.

## Einschränkungen

Folgende Einschränkungen sind zu beachten:

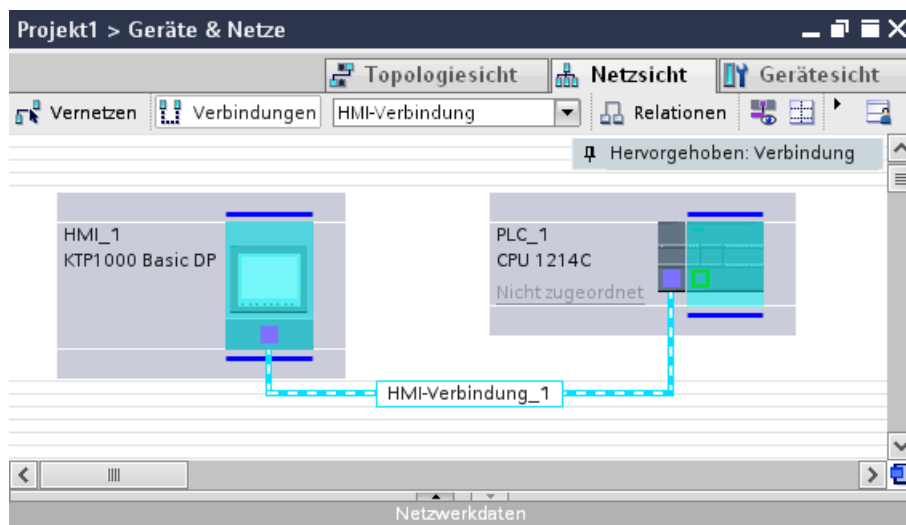
- Nur wenn der Port das jeweilige Feature unterstützt sind die einzelnen Optionskästchen bedienbar.
- Wenn für den Port ein Partnerport bestimmt wurde sind folgende Optionskästchen nicht bedienbar:
  - "Ende der Erfassung erreichbarer Teilnehmer"
  - "Ende der Topologieerkennung"
- Wenn Autonegotiation deaktiviert ist, dann sind alle Optionskästchen nicht bedienbar.

### 10.8.7.3 Kommunikation über PROFIBUS

#### Kommunikation über PROFIBUS

#### HMI-Verbindungen über PROFIBUS

Wenn Sie eine SIMATIC S7 1200 über PROFIBUS mit einem Bediengerät verbinden wollen, dann müssen Sie vorher ein PROFIBUS-fähiges Kommunikationsmodul auf einen Steckplatz der Steuerung projektieren.



#### HMI-Verbindung im Editor "Geräte & Netze"

Die HMI-Verbindung über PROFIBUS zwischen der Steuerung mit dem Bediengerät projektieren Sie im Editor "Geräte & Netze".



## Verbindung im Editor "Verbindungen"

Alternativ projektieren Sie die Verbindung zwischen Steuerung und Bediengerät im Editor "Verbindungen" des Bediengeräts.

## HMI-Verbindung über PROFIBUS projektieren

### Einleitung

Eine HMI-Verbindung über PROFIBUS zwischen Bediengeräten und einer SIMATIC S7 1200 projektieren Sie im Editor "Geräte & Netze".

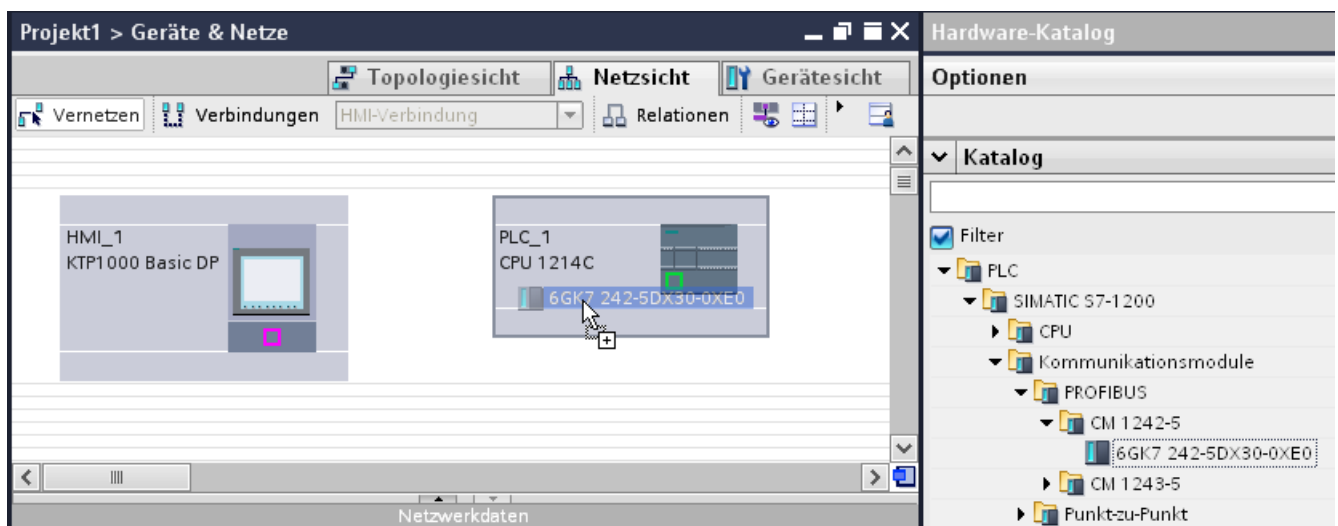
### Voraussetzungen

Folgende Kommunikationspartner sind im Editor "Geräte & Netze" angelegt:

- Bediengerät mit MPI/DP-Schnittstelle
- SIMATIC S7 1200

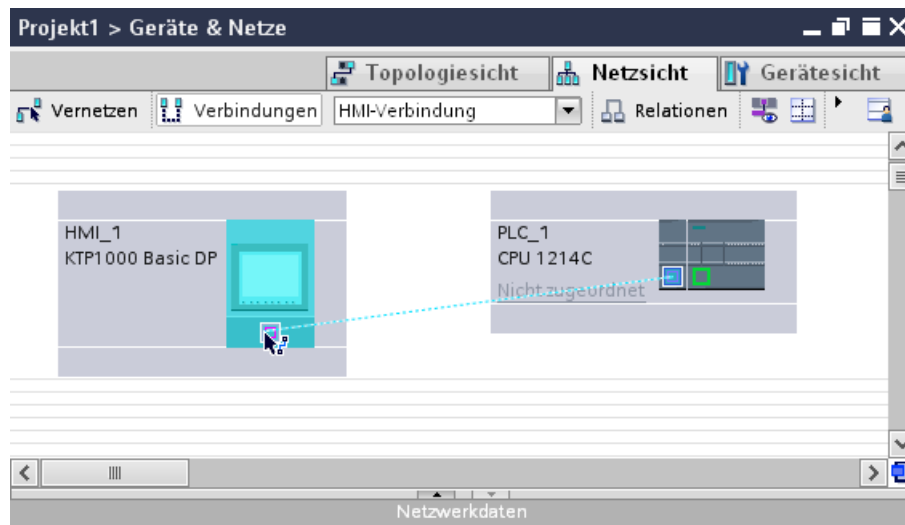
### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Geräte & Netze". In der Netzansicht werden die im Projekt vorhandenen Kommunikationspartner grafisch dargestellt.
2. Klicken Sie auf die Schaltfläche "Verbindungen". Die für eine Verbindung verfügbaren Geräte werden farblich markiert dargestellt.
3. Ziehen Sie per Drag&Drop ein PROFIBUS-fähiges Kommunikationsmodul aus dem Hardware-Katalog in die Steuerung.



4. Klicken Sie auf die Schnittstelle des Bediengeräts.
5. Wählen Sie im Inspektorfenster "Eigenschaften > Allgemein > PROFIBUS-Adresse/ MPI-Adresse > Parameter" den Schnittstellen-Typ "PROFIBUS" aus.

6. Klicken Sie in die Schnittstelle des Kommunikationsmoduls und ziehen Sie per Drag&Drop eine Verbindung zum Bediengerät.



7. Klicken Sie auf den Namen der Verbindung.  
Im Inspektorfenster wird die Verbindung grafisch dargestellt.
8. Klicken Sie auf "HMI-Verbindung Hervorheben" und wählen Sie die HMI-Verbindung aus.
9. Klicken Sie auf die Kommunikationspartner in der "Netzsicht" und ändern Sie im Inspektorfenster die PROFINET-Parameter nach den Anforderungen ihres Projekts. Nähere Hinweise hierzu finden Sie im Kapitel "PROFIBUS-Parameter (Seite 3564)".

#### Hinweis

Die erstellte HMI-Verbindung wird auch im tabellarischen Bereich des Editors auf der Registerkarte "Verbindungen" angezeigt. In der Tabelle kontrollieren Sie die Verbindungsparameter.

Den lokalen Verbindungsnamen der Verbindung können Sie nur in der Tabelle ändern.

## Ergebnis

Sie haben eine HMI-Verbindung zwischen einem Bediengerät und einer SIMATIC S7 1200 über PROFIBUS angelegt.

## PROFIBUS-Parameter

### PROFIBUS-Parameter für die HMI-Verbindung

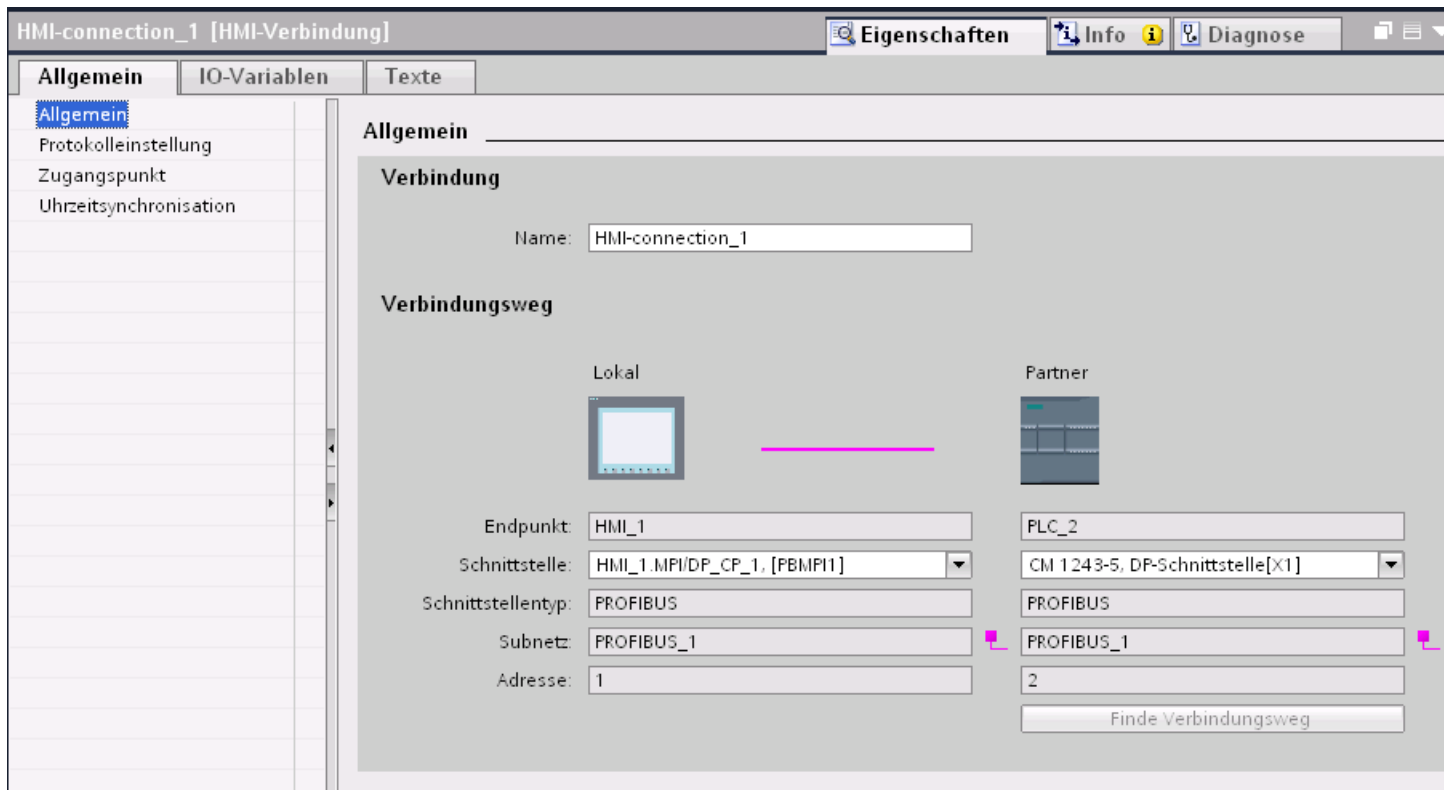
### PROFIBUS-Parameter für die HMI-Verbindung

Eine Übersicht über die projektierten Parameter einer HMI-Verbindung finden Sie in den Eigenschaften der HMI-Verbindung.

Änderungen sind in diesem Inspektorfenster nur eingeschränkt möglich.

## Parameter der HMI-Verbindung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die HMI-Verbindung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der HMI-Verbindung.



### "Verbindung"

Im Bereich "Verbindung" wird die HMI-Verbindung angezeigt, die zwischen den Geräten angelegt wurde.

Der Name der HMI-Verbindung ist in diesem Bereich editierbar.

### "Verbindungsweg"

Im Bereich "Verbindungsweg" werden die Kommunikationspartner der gewählten HMI-Verbindung und deren PROFIBUS-Parameter angezeigt. Einige der gezeigten Bereiche sind in diesem Dialog nicht editierbar.

- "Endpunkt"  
Zeigt den Gerätenamen an. Dieser Bereich ist nicht editierbar.
- "Schnittstelle"  
Zeigt die gewählte Schnittstelle des Geräts an. Abhängig vom Gerät können Sie zwischen mehreren Schnittstellen wählen.
- "Schnittstellen-Typ"  
Zeigt den gewählten Schnittstellen-Typ an. Dieser Bereich ist nicht editierbar.

- "Subnetz"  
Zeigt das gewählte Subnetz an. Dieser Bereich ist nicht editierbar.
- "Adresse"  
Zeigt die PROFIBUS-Adresse des Geräts an. Dieser Bereich ist nicht editierbar.
- Schaltfläche "Finde Verbindungsweg"  
Ermöglicht das nachträgliche Spezifizieren von Verbindungen.

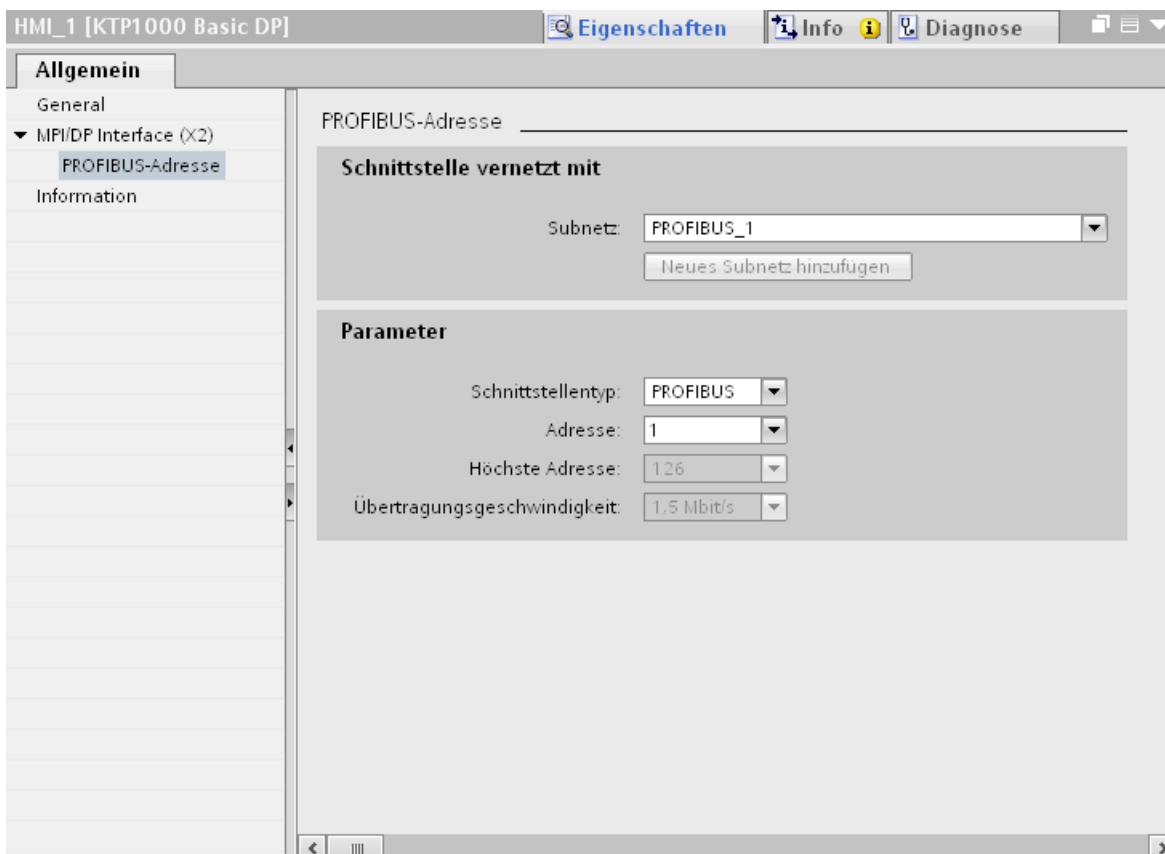
## PROFIBUS-Parameter für das Bediengerät

## PROFIBUS-Parameter für das Bediengerät

Eine Übersicht über die projektierten Parameter des Bediengeräts finden Sie in den Eigenschaften des Bediengeräts.

## PROFIBUS-Parameter des Bediengeräts anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf das Bediengerät.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter des Bediengeräts.



### **"Schnittstelle vernetzt mit"**

Im Bereich "Schnittstelle vernetzt mit" wählen Sie das Subnetz der HMI-Verbindung aus, über die das Bediengerät an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### **"Parameter"**

- "Schnittstellen-Typ"  
Abhängig vom Bediengerätetyp stehen verschiedene Schnittstellen zur Auswahl.
- "Adresse"  
Im Bereich "Adresse" parametrieren Sie die PROFIBUS-Adresse des Bediengeräts. Die PROFIBUS-Adresse muss im PROFIBUS-Netzwerk eindeutig sein
- "Höchste Adresse"  
Im Bereich "Höchste Adresse" wird die höchste Adresse des PROFIBUS-Netzwerks angezeigt.
- "Übertragungsgeschwindigkeit"  
Die "Übertragungsgeschwindigkeit" wird durch das langsamste ans Netzwerk angeschlossene Gerät festgelegt. Die Einstellung ist im gesamten Netzwerk gleich.

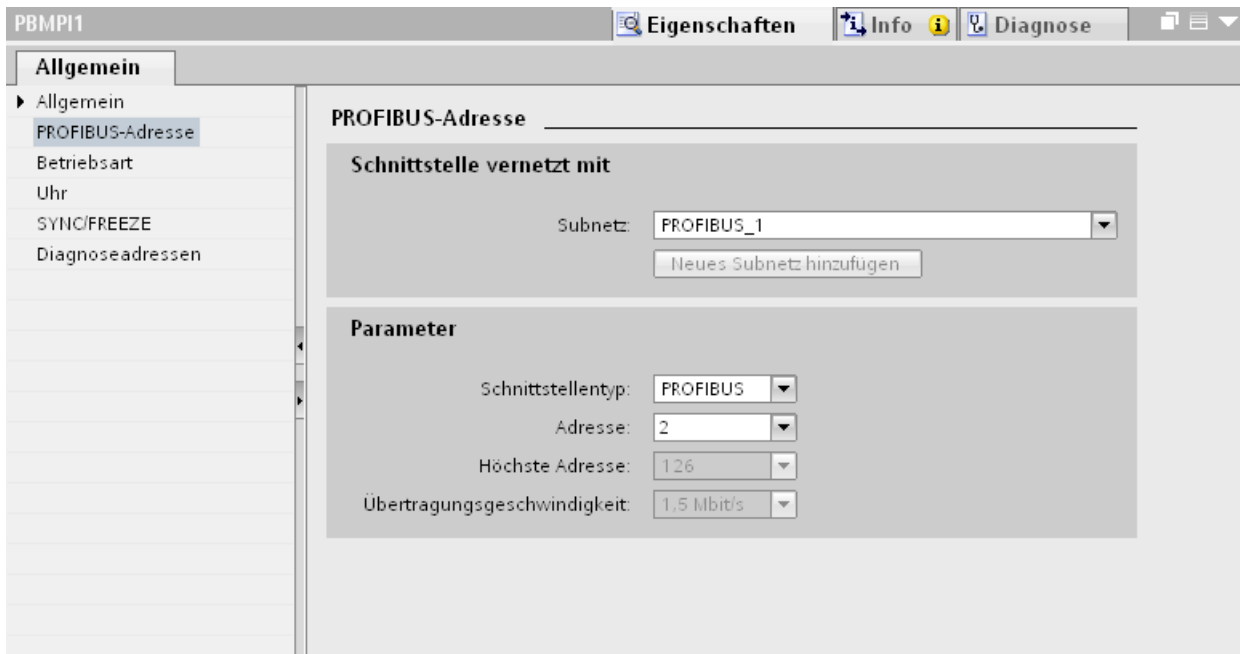
### **PROFIBUS-Parameter für die Steuerung**

### **PROFIBUS-Parameter für die Steuerung**

Eine Übersicht über die projektierten Parameter finden Sie in den Eigenschaften der Steuerung.

## PROFIBUS-Parameter der Steuerung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die Steuerung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der Steuerung.



### "Schnittstelle vernetzt mit"

Im Bereich "Subnetz" wählen Sie das Subnetz der HMI-Verbindung aus, über die die Steuerung an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "Parameter"

- "Schnittstellen-Typ"  
Abhängig vom Bediengerätetyp stehen verschiedene Schnittstellen zur Auswahl.
- "Adresse"  
Im Bereich "Adresse" parametrieren Sie die PROFIBUS-Adresse des Bediengeräts. Die PROFIBUS-Adresse muss im PROFIBUS-Netzwerk eindeutig sein
- "Höchste Adresse"  
Im Bereich "Höchste Adresse" wird die höchste Adresse des PROFIBUS-Netzwerks angezeigt.
- "Übertragungsgeschwindigkeit"  
Die "Übertragungsgeschwindigkeit" wird durch das langsamste ans Netzwerk angeschlossene Gerät festgelegt. Die Einstellung ist im gesamten Netzwerk gleich.

## Busprofile bei PROFIBUS

### Einführung

Je nach angeschlossenen Gerätetypen und verwendeten Protokollen am PROFIBUS stehen unterschiedliche Profile zur Verfügung. Die Profile unterscheiden sich hinsichtlich der Einstellmöglichkeiten und hinsichtlich der Berechnung der Busparameter. Im Folgenden werden die Profile erläutert.

### Teilnehmer mit unterschiedlichen Profilen am selben PROFIBUS-Subnetz

Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Busparameter aller Teilnehmer dieselben Werte haben.

### Profile und Übertragungsgeschwindigkeiten

| Profile    | Unterstützte Übertragungsgeschwindigkeiten in kbit/s |
|------------|--|
| DP         | 9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000  |
| Standard   | 9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000  |
| Universell | 9,6 19,2 93,75 187,5 500 1500                        |

### Bedeutung der Profile

| Profil     | Bedeutung  |
|------------|--|
| DP         | <p>Wählen Sie das Busprofil "DP", wenn nur Geräte an das PROFIBUS-Subnetz angeschlossen sind, welche die Anforderungen der Norm EN 50170 Volume 2/3, Part 8-2 PROFIBUS erfüllen. Die Einstellung der Busparameter ist auf diese Geräte optimiert.</p> <p>Hierzu zählen Geräte mit DP-Master- und DP-Slave-Schnittstellen der SIMATIC S7 sowie Dezentrale Peripheriegeräte anderer Hersteller.</p>  |
| Standard   | <p>Gegenüber dem Profil "DP" bietet das Profil "Standard" zusätzlich die Möglichkeit, Teilnehmer eines anderen Projektes oder Teilnehmer, die hier nicht projiziert wurden, bei der Berechnung der Busparameter zu berücksichtigen. Die Busparameter werden dann nach einem einfachen, nicht optimierten Algorithmus berechnet.</p>  |
| Universell | <p>Wählen Sie das Busprofil "Universell", wenn einzelne Teilnehmer am PROFIBUS-Subnetz den Dienst PROFIBUS-FMS nutzen.</p> <p>Hierzu gehören z. B. folgende Geräte:</p> <ul style="list-style-type: none"> <li>• CP 343-5</li> <li>• PROFIBUS-FMS-Geräte anderer Hersteller</li> </ul> <p>Auch hier besteht wie im Profil "Standard" die Möglichkeit, zusätzliche Teilnehmer bei der Berechnung der Busparameter zu berücksichtigen.</p> |

## Schutz der Kommunikation

### Schutzstufen

Wenn Sie die Kommunikation von Steuerung und Bediengerät schützen wollen, dann können Sie Schutzstufen für die Kommunikation vergeben.

Für eine S7-1500-CPU können Sie mehrere Passwörter eingeben und damit unterschiedliche Zugriffsrechte für verschiedene Nutzergruppen einrichten.

Die Passwörter werden in einer Tabelle eingegeben, so dass jedem Passwort genau eine Schutzstufe zugeordnet ist.

Wie das Passwort wirkt, steht in der Spalte "Schutz".

### Beispiel

Sie wählen die bei der Projektierung der Steuerung die Schutzstufe "Kompletter Schutz" für eine Standard-CPU (d. h. keine F-CPU) und geben für jede in der Tabelle darüber liegenden Schutzstufen ein eigenes Passwort ein.

Für Nutzer, die keines der Passwörter kennen, ist die CPU komplett geschützt. Nicht einmal HMI-Zugriffe sind möglich.

Für Nutzer, die eines der parametrisierten Passwörter kennen, hängt die Wirkung ab von der Tabellenzeile, in der das Passwort steht:

- Das Passwort in Zeile 1 (Kein Schutz) wirkt, als wäre die CPU ungeschützt. Nutzer, die dieses Passwort kennen, haben uneingeschränkten Zugriff auf die CPU.
- Das Passwort in Zeile 2 (Schreibschutz) wirkt, als wäre die CPU schreibgeschützt. Trotz Passwort-Kenntnis haben Nutzer, die dieses Passwort kennen, nur lesenden Zugriff auf die CPU.
- Das Passwort in Zeile 3 (Schreib- und Leseschutz) wirkt, als wäre die CPU schreib- und lesegeschützt, so dass für die Nutzer, die dieses Passwort kennen, nur HMI-Zugriffe möglich sind.

## Zugangspasswort für die HMI-Verbindung

### Einleitung

Den Zugriff auf eine Steuerung sichern Sie durch die Vergabe eines Passworts ab.

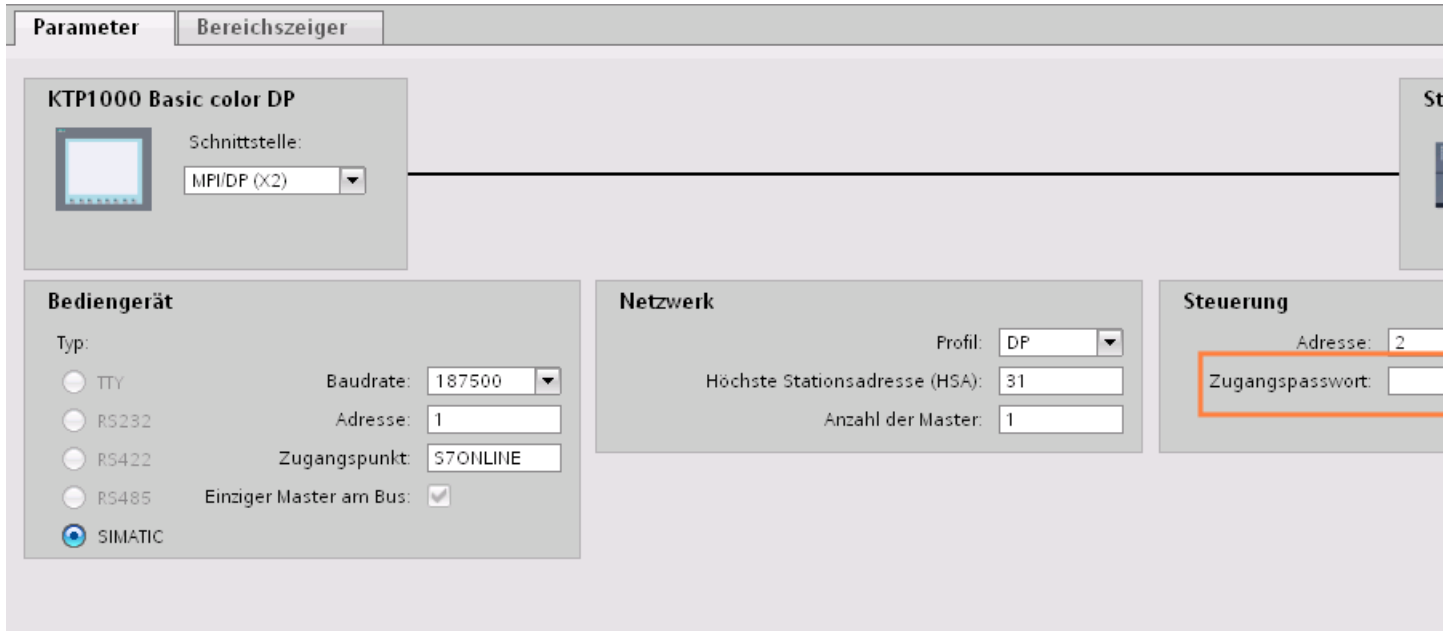
Bei der Projektierung der Passwort vergeben Sie das Passwort.

Ab Schutzstufe "Kompletter Schutz" ist es zwingend notwendig das Passwort einzugeben. Wenn Sie das Passwort nicht oder falsch eingeben wird keine Kommunikation zur Steuerung aufgebaut.



## Kennwort vergeben

Im Editor "Verbindungen" geben Sie das Zugangspasswort für die Steuerung ein.



### 10.8.7.4 Datenaustausch

#### Datenaustausch über Bereichszeiger

#### Allgemeines zu Bereichszeigern

#### Einleitung

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu. In diese Datenbereiche wird während der Kommunikation wechselseitig von der Steuerung und dem Bediengerät geschrieben und gelesen.

Durch die Auswertung der abgelegten Daten lösen Steuerung und Bediengerät gegenseitig fest definierte Aktionen aus.

#### Projektierung von Bereichszeigern

Bevor Sie den Bereichszeiger verwenden, aktivieren Sie die Bereichszeiger unter "Verbindungen > Bereichszeiger". Danach parametrieren Sie die Bereichszeiger.

Nähere Hinweise zum Projektieren von Bereichszeigern finden Sie unter:

Projektierung von Bereichszeigern

## Bereichszeiger "Datum/Uhrzeit"

### Funktion

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von dem Bediengerät zur Steuerung verwendet.

Die Steuerung schreibt den Steuerauftrag "41" in das Auftragsfach.

Mit der Auswertung des Steuerauftrags schreibt das Bediengerät sein aktuelles Datum und die Uhrzeit in den im Bereichszeiger "Datum/Uhrzeit" projektierten Datenbereich.

---

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" projektiert haben, dann können Sie den Bereichszeiger "Datum/Uhrzeit PLC" nicht verwenden.

---

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" verwenden, dann ist eine symbolische Adressierung nicht möglich.

Wenn in einem Projekt mehrere Verbindungen projektiert sind, und in einer der Verbindungen der Bereichszeiger "Datum / Uhrzeit" verwendet werden soll, so muss der Bereichszeiger für jede projektierte Verbindung aktiviert werden.

Wenn in einem Projekt mehrere Verbindungen projektiert sind, und in einer der Verbindungen der Bereichszeiger "Datum / Uhrzeit" verwendet werden soll, so muss der Bereichszeiger für jede projektierte Verbindung aktiviert werden.

Der Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

| Datenwort | Höherwertiges Byte |  |  |  |  |  |   | Niederwertiges Byte   |  |  |  |  |  |   |         |
|-----------|--------------------|--|--|--|--|--|---|-----------------------|--|--|--|--|--|---|---------|
|           | 7                  |  |  |  |  |  | 0 | 7                     |  |  |  |  |  | 0 |         |
| n+0       | Reserviert         |  |  |  |  |  |   | Stunde (0-23)         |  |  |  |  |  |   | Uhrzeit |
| n+1       | Minute (0-59)      |  |  |  |  |  |   | Sekunde (0-59)        |  |  |  |  |  |   |         |
| n+2       | Reserviert         |  |  |  |  |  |   | Reserviert            |  |  |  |  |  |   |         |
| n+3       | Reserviert         |  |  |  |  |  |   | Wochentag (1-7, 1=So) |  |  |  |  |  |   | Datum   |
| n+4       | Tag (1-31)         |  |  |  |  |  |   | Monat (1-12)          |  |  |  |  |  |   |         |
| n+5       | Jahr (80-99/0-29)  |  |  |  |  |  |   | Reserviert            |  |  |  |  |  |   |         |

---

#### Hinweis

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

## Erlaubte Datentypen

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" projektieren, können Sie folgende Datentypen verwenden:

- Word
- UInt
- DTL

## Verwendung des Datentyps "DTL"

Bei Einsatz des Kommunikationstreibers S7 1200 verwenden Sie den Datentyp "DTL". Eine Variable vom Datentyp "DTL" hat eine Länge von 12 Byte und speichert Angaben zu Datum und Uhrzeit in einer vordefinierten Struktur.

Der Datentyp "DTL" hat folgende Struktur:

| Byte | Komponente   | Datentyp | Wertebereich  |
|------|--------------|----------|---|
| 0    | Jahr         | UINT     | 1970 bis 2554   |
| 1    |              |          |   |
| 2    | Monat        | USINT    | 0 bis 12  |
| 3    | Tag          | USINT    | 1 bis 31  |
| 4    | Wochentag    | USINT    | 1(Sonntag) bis 7(Samstag)<br>Der Wochentag wird bei der Werteingabe nicht berücksichtigt. |
| 5    | Stunde       | USINT    | 0 bis 23  |
| 6    | Minute       | USINT    | 0 bis 59  |
| 7    | Sekunde      | USINT    | 0 bis 59  |
| 8    | Nanosekunden | UDINT    | 0 bis 999 999 999   |
| 9    |              |          |   |
| 10   |              |          |   |
| 11   |              |          |   |

## Bereichszeiger "Datum/Uhrzeit PLC"

### Funktion

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von der Steuerung zum Bediengerät verwendet. Sie setzen diesen Bereichszeiger ein, wenn die Steuerung Master für die Zeit ist.

Die Steuerung lädt den Datenbereich des Bereichszeigers.

Das Bediengerät liest zyklisch die Daten über den projektierten Erfassungszyklus und synchronisiert sich.

**Hinweis**

Wählen Sie in der Projektierung den Erfassungszyklus für den Bereichszeiger Datum/Uhrzeit PLC nicht zu klein, da dies die Performance des Bediengeräts beeinflusst.  
Empfehlung: Erfassungszyklus 1 Minute, wenn dies Ihr Prozess erlaubt.

"Datum/Uhrzeit PLC" ist ein globaler Bereichszeiger und kann in einem Projekt nur ein Mal projektiert werden.

**Hinweis**

Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" projektiert haben, können Sie den Bereichszeiger "Datum/Uhrzeit" nicht verwenden.

Der Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

**Format DATE\_AND\_TIME (BCD-codiert)**

| Datenwort         | Höherwertiges Byte |       |   | Niederwertiges Byte |                       |   |
|-------------------|--------------------|-------|---|---------------------|-----------------------|---|
|                   | 7                  | ..... | 0 | 7                   | .....                 | 0 |
| n+0               | Jahr (80-99/0-29)  |       |   | Monat (1-12)        |                       |   |
| n+1               | Tag (1-31)         |       |   | Stunde (0-23)       |                       |   |
| n+2               | Minute (0-59)      |       |   | Sekunde (0-59)      |                       |   |
| n+3               | Reserviert         |       |   | Reserviert          | Wochentag (1-7, 1=So) |   |
| n+4 <sup>1)</sup> | Reserviert         |       |   | Reserviert          |                       |   |
| n+5 <sup>1)</sup> | Reserviert         |       |   | Reserviert          |                       |   |

- 1) Die beiden Datenwörter müssen im Datenbereich vorhanden sein, um eine Übereinstimmung des Datenformats mit WinCC flexible sicherzustellen und das Lesen falscher Informationen zu vermeiden.

**Hinweis**

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

## Erlaubte Datentypen

Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" projektieren, können Sie folgende Datentypen verwenden:

- Word
- UInt
- DTL

## Verwendung des Datentyps "DTL"

Bei Einsatz des Kommunikationstreibers S7 1200 verwenden Sie den Datentyp "DTL". Eine Variable vom Datentyp "DTL" hat eine Länge von 12 Byte und speichert Angaben zu Datum und Uhrzeit in einer vordefinierten Struktur.

Der Datentyp "DTL" hat folgende Struktur:

| Byte | Komponente   | Datentyp | Wertebereich  |
|------|--------------|----------|---|
| 0    | Jahr         | UINT     | 1970 bis 2554   |
| 1    |              |          |   |
| 2    | Monat        | USINT    | 0 bis 12  |
| 3    | Tag          | USINT    | 1 bis 31  |
| 4    | Wochentag    | USINT    | 1(Sonntag) bis 7(Samstag)<br>Der Wochentag wird bei der Werteingabe nicht berücksichtigt. |
| 5    | Stunde       | USINT    | 0 bis 23  |
| 6    | Minute       | USINT    | 0 bis 59  |
| 7    | Sekunde      | USINT    | 0 bis 59  |
| 8    | Nanosekunden | UDINT    | 0 bis 999 999 999   |
| 9    |              |          |   |
| 10   |              |          |   |
| 11   |              |          |   |

Die Bediengeräte unterstützen nicht die Verwendung von Nanosekunden. Bei der Verarbeitung in Runtime werden Werte im Bereich von Nanosekunden ignoriert.

## Bereichszeiger "Koordinierung"

### Funktion

Der Bereichszeiger "Koordinierung" dient zur Realisierung der folgenden Funktionen:

- Anlauf des Bediengeräts im Steuerungsprogramm erkennen
- Aktuelle Betriebsart des Bediengeräts im Steuerungsprogramm erkennen
- Kommunikationsbereitschaft des Bediengeräts im Steuerungsprogramm erkennen

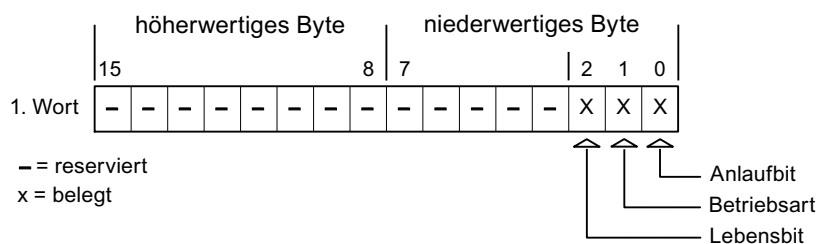
Der Bereichszeiger "Koordinierung" standardmäßig hat eine Länge von einem Wort und kann nicht verändert werden.

## Verwendung

### Hinweis

Bei jeder Aktualisierung des Bereichszeigers durch das Bediengerät wird immer der komplette Koordinierungsbereich geschrieben.  
Das SPS-Programm darf deshalb im Koordinierungsbereich keine Änderungen vornehmen.

## Belegung der Bits im Bereichszeiger "Koordinierung"



### Anlaufbit

Das Anlaufbit wird durch das Bediengerät während des Anlaufvorgangs kurzfristig auf "0" gesetzt. Nach dem Anlaufvorgang steht das Bit dauerhaft auf "1".

### Betriebsart

Sobald das Bediengerät durch den Benutzer offline geschaltet wird, wird das Betriebsartenbit auf 1 gesetzt. Im Normalbetrieb des Bediengeräts ist der Zustand des Betriebsartenbits "0". Im Steuerungsprogramm können Sie durch Abfrage dieses Bits die aktuelle Betriebsart des Bediengeräts ermitteln.

### Lebensbit

Das Lebensbit wird durch das Bediengerät im zeitlichen Abstand von ca. einer Sekunde invertiert. Im Steuerungsprogramm können Sie durch Abfrage dieses Bits überprüfen, ob die Verbindung zum Bediengerät noch besteht.

## Verarbeitung in der Steuerung

Für eine einfachere Auswertbarkeit im Steuerungsprogramm verwenden Sie beim Einsatz des Kommunikationstreibers SIMATIC S7 1200 für diesen Bereichszeiger ein Bool-Array. Sie müssen das vollständige 16 Bit-Wort des Bereichszeigers abbilden. Projektieren Sie deshalb im Steuerungsprogramm eine Variable vom Datentyp "Array [0 .. 15] of bool".

## Erlaubte Datentypen

Wenn Sie den Bereichszeiger "Koordinierung" projektieren, können Sie folgende Datentypen verwenden:

- Word
- UInt
- Bool

## Bereichszeiger "Bildnummer"

### Funktion

Die Bediengeräte legen im Bereichszeiger "Bildnummer" Informationen über das am Bediengerät aufgerufene Bild ab.

Dadurch ist es möglich, Informationen zum aktuellen Bildinhalt des Bediengeräts zur Steuerung zu übertragen. In der Steuerung können bestimmte Reaktionen getriggert werden, z. B. der Aufruf eines anderen Bildes.

### Verwendung

Bevor der Bereichszeiger "Bildnummer" verwendet werden kann, muss dieser unter "Verbindungen ▶ Bereichszeiger" eingerichtet und aktiviert werden. Der Bereichszeiger "Bildnummer" kann nur in **einer** Steuerung und in dieser Steuerung nur **einmal** angelegt werden.

Die Bildnummer wird immer zur Steuerung übertragen, wenn ein neues Bild aktiviert wird oder der Fokus innerhalb eines Bildes von einem Bildobjekt zu einem anderen wechselt.

### Aufbau

Der Bereichszeiger ist ein Datenbereich im Speicher der Steuerung mit einer festen Länge von 5 Worten.

|         | 15                  | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------|---------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1. Wort | Aktueller Bildtyp   |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 2. Wort | Aktuelle Bildnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 3. Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 4. Wort | Aktuelle Feldnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 5. Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

- Aktueller Bildtyp  
"1" für Grundbild oder  
"4" für Permanentfenster
  - Aktuelle Bildnummer  
1 bis 32767
  - Aktuelle Feldnummer  
1 bis 32767
- 

**Hinweis**

**Geräteabhängigkeit**

Permanentfenster stehen auf Basic Panels nicht zur Verfügung.

---

### Erlaubte Datentypen

Wenn Sie den Bereichszeiger "Bildnummer" projektieren, können Sie folgende Datentypen verwenden:

- Word
- UInt

### Bereichszeiger "Projektkennung"

#### Funktion

Beim Start der Runtime kann überprüft werden, ob das Bediengerät an die richtige Steuerung angeschlossen ist. Diese Überprüfung ist beim Einsatz mehrerer Bediengeräte wichtig.

Dazu vergleicht das Bediengerät einen in der Steuerung hinterlegten Wert mit dem in der Projektierung angegebenen Wert. Damit wird die Kompatibilität der Projektierungsdaten mit dem Steuerungsprogramm sichergestellt.

Eine fehlende Übereinstimmung führt zur Ausgabe einer entsprechenden Meldung und die Runtime wird nicht gestartet.

#### Verwendung

Um diesen Bereichszeiger zu verwenden, richten Sie bei der Projektierung Folgendes ein:

- Angabe der Version, welche die Projektierung hat. Möglicher Wert zwischen 1 und 255. Sie geben die Version ein im Editor "Runtime-Einstellungen > Allgemein" im Bereich "Identifizierung".
- Die Steuerungsvariable oder das Variablenarray, das Sie als Datenbereich für den Bereichszeiger projiziert haben.



## Ausfall einer Verbindung

Ein Verbindungsausfall zu einem Gerät, auf dem der Bereichszeiger "Projektkennung" projiziert ist, hat zur Folge, dass auch alle anderen Verbindungen des Geräts "Offline" geschaltet werden.

Dieses Verhalten hat folgende Voraussetzungen:

- Sie haben in einem Projekt mehrere Verbindungen projiziert.
- Sie verwenden in mindestens einer Verbindung den Bereichszeiger "Projektkennung".

Folgende Ursachen können Verbindungen in den Zustand "Offline" setzen:

- Die Steuerung ist nicht erreichbar.
- Die Verbindung wurde im Engineering System offline geschaltet.

## Erlaubte Datentypen

Wenn Sie den Bereichszeiger "Projektkennung" projizieren, können Sie folgende Datentypen verwenden:

- Word
- UInt

## Bereichszeiger "Steuerungsauftrag"

### Funktion

Über das Steuerungsauftrags-Fach können dem Bediengerät Steuerungsaufträge gegeben und damit Aktionen am Bediengerät getriggert werden. Zu diesen Funktionen gehören z. B.:

- Bild anzeigen
- Datum und Uhrzeit stellen

### Datenstruktur

Im ersten Wort des Steuerungsauftrags-Fachs steht die Auftragsnummer. Je nach Steuerungsauftrag können dann bis zu drei Parameter übergeben werden.

| Wort | Höherwertiges Byte | Niederwertiges Byte |
|------|--------------------|---------------------|
| n+0  | 0                  | Auftragsnummer      |
| n+1  | Parameter 1        |                     |
| n+2  | Parameter 2        |                     |
| n+3  | Parameter 3        |                     |

Wenn das erste Wort des Steuerungsauftrags-Fachs ungleich 0 ist, wertet das Bediengerät den Steuerungsauftrag aus. Aus diesem Grund müssen zuerst die Parameter in das Steuerungsauftrags-Fach eingetragen werden und dann erst die Auftragsnummer.

Wenn das Bediengerät den Steuerungsauftrag angenommen hat, wird das erste Wort wieder auf 0 gesetzt. Die Ausführung des Steuerungsauftrags ist zu diesem Zeitpunkt im Allgemeinen noch nicht abgeschlossen.

### Steuerungsaufträge

Nachfolgend sind alle Steuerungsaufträge und deren Parameter aufgelistet. Die Spalte "Nr." enthält die Auftragsnummer des Steuerungsauftrags. Generell können Steuerungsaufträge nur dann von der Steuerung getriggert werden, wenn das Bediengerät im Online-Betrieb ist.

| Nr.       | Funktion   |   |
|-----------|--|---|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>   |   |
|           | Parameter 1  | Linkes Byte: -<br>Rechtes Byte: Stunden (0-23)                      |
|           | Parameter 2  | Linkes Byte: Minuten (0-59)<br>Rechtes Byte: Sekunden (0-59)        |
|           | Parameter 3  | -   |
| <b>15</b> | <b>Datum stellen (BCD-codiert)</b>   |   |
|           | Parameter 1  | Linkes Byte: -<br>Rechtes Byte: Wochentag<br>(1-7: Sonntag-Samstag) |
|           | Parameter 2  | Linkes Byte: Tag (1-31)<br>Rechtes Byte: Monat (1-12)               |
|           | Parameter 3  | Linkes Byte: Jahr   |
| <b>23</b> | <b>Benutzer anmelden</b>   |   |
|           | Meldet den Benutzer "PLC User" mit der im Parameter 1 übergebenen Gruppennummer am Bediengerät an.<br>Voraussetzung für die Anmeldung ist, dass die übergebene Gruppennummer im Projekt vorhanden ist.   |   |
|           | Parameter 1  | Gruppennummer 1 - 255   |
|           | Parameter 2, 3   | -   |
| <b>24</b> | <b>Benutzer abmelden</b>   |   |
|           | Meldet den aktuell angemeldeten Benutzer ab.<br>(Funktion entspricht der Systemfunktion "Abmelden")  |   |
|           | Parameter 1, 2, 3  | -   |
| <b>40</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |   |
|           | Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.   |   |
|           | Parameter 1, 2, 3  | -   |
| <b>41</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |   |
|           | Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.   |   |
|           | Parameter 1, 2, 3  | -   |
| <b>46</b> | <b>Variable aktualisieren</b>  |   |
|           | Veranlasst das Bediengerät den aktuellen Wert der Variablen aus der Steuerung zu lesen, deren Aktualisierungskennung mit dem im Parameter 1 übergebenen Wert übereinstimmt.<br>(Funktion entspricht der Systemfunktion "AktualisiereVariable") |   |

| Nr.       | Funktion  |  |
|-----------|---|--|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>  |  |
|           | Parameter 1   | 1 - 100  |
| <b>49</b> | <b>Betriebsmeldepuffer löschen</b>  |  |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Warnings" aus dem Meldepuffer. |  |
|           | Parameter 1, 2, 3   | -  |
| <b>50</b> | <b>Störmeldepuffer löschen</b>  |  |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Errors" aus dem Meldepuffer.   |  |
|           | Parameter 1, 2, 3   | -  |
| <b>51</b> | <b>Bildanwahl</b>   |  |
|           | Parameter 1   | Bildnummer   |
|           | Parameter 2   | -  |
|           | Parameter 3   | Feldnummer   |
| <b>69</b> | <b>Datensatz aus Steuerung lesen <sup>1)</sup></b>  |  |
|           | Parameter 1   | Rezepturnummer (1-999)   |
|           | Parameter 2   | Datensatznummer (1-65535)  |
|           | Parameter 3   | 0: Vorhandenen Datensatz nicht überschreiben<br>1: Vorhandenen Datensatz überschreiben |
| <b>70</b> | <b>Datensatz in Steuerung schreiben <sup>1)</sup></b>   |  |
|           | Parameter 1   | Rezepturnummer (1-999)   |
|           | Parameter 2   | Datensatznummer (1-65535)  |
|           | Parameter 3   | -  |

|               |   |
|---------------|---|
| <sup>1)</sup> | Nur bei Geräten, die Rezepturen unterstützen  |
| <sup>2)</sup> | Auf den Bediengeräten OP 73, OP 77A und TP 177A wird der Steuerungsauftrag "Bildanwahl" auch dann ausgeführt, wenn die Bildschirmtastatur geöffnet ist. |
| <sup>3)</sup> | Beim Bediengerät KTP 600 BASIC PN wird der Wochentag ignoriert.   |

## Erlaubte Datentypen

Wenn Sie den Bereichszeiger "Bildnummer" projektieren, können Sie folgende Datentypen verwenden:

- Word
- UInt

## Bereichszeiger "Datensatz"

## Bereichszeiger "Datensatz"

### Funktion

Bei der Übertragung von Datensätzen zwischen Bediengerät und Steuerung greifen beide Kommunikationspartner wechselseitig auf gemeinsame Kommunikationsbereiche in der Steuerung zu.

### Übertragungsarten

Für die Übertragung von Datensätzen zwischen Bediengerät und Steuerung gibt es zwei Möglichkeiten:

- Übertragung ohne Synchronisation
- Übertragung mit Synchronisation über das Datenfach

Datensätze werden immer direkt übertragen. D. h., die Variablenwerte werden direkt, ohne den Umweg über eine Zwischenablage, aus der Adresse gelesen oder in die Adresse geschrieben, die für die Variable projektiert ist.

### Übertragung von Datensätzen anstoßen

Für den Anstoß der Übertragung gibt es drei Möglichkeiten:

- Bedienung in der Rezepturanzeige
- Steuerungsaufträge  
Die Übertragung der Datensätze kann auch durch die Steuerung getriggert werden.
- Auslösen projektiierter Funktionen

Wenn die Übertragung von Datensätzen durch einen Steuerungsauftrag getriggert wird, werden auch die Daten in der Rezepturanzeige aktualisiert. Vermeiden Sie es daher gleichzeitig die Rezepturanzeige zu bedienen während Steuerungsaufträge zur Übertragung von Datensätzen getriggert werden. Haben Sie bereits mit dem Editieren eines Datensatzes begonnen und es wird ein Steuerungsauftrag zur Übertragung von Datensätzen angestossen, so wird dieser Steuerungsauftrag verworfen.

### Erlaubte Datentypen

Wenn Sie den Bereichszeiger "Datensatz" projektieren, können Sie folgende Datentypen verwenden:

- Word
- UInt

## Übertragung ohne Synchronisation

Bei der asynchronen Übertragung von Datensätzen zwischen Bediengerät und Steuerung findet keine Koordination über gemeinsam benutzte Datenbereiche statt. Die Einrichtung eines Datenbereichs beim Projektieren ist daher nicht nötig.

Die asynchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Ein unkontrolliertes Überschreiben der Daten durch die Kommunikationspartner kann systembedingt ausgeschlossen werden.
- Die Steuerung braucht keine Informationen über die Rezepturnummer und die Datensatznummer.
- Die Übertragung von Datensätzen wird durch Bedienung am Bediengerät getriggert.

## Werte lesen

Beim Anstoß der Übertragung zum Lesen werden die Werte aus den Steuerungsadressen gelesen und zum Bediengerät übertragen.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die Werte werden in das Bediengerät geladen. Im Bediengerät können Sie diese weiterverarbeiten, z. B. Werte ändern, speichern etc.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte werden sofort auf dem Datenträger gespeichert.

## Werte schreiben

Beim Anstoß der Übertragung zum Schreiben werden die Werte in die Steuerungsadressen geschrieben.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die aktuellen Werte werden in die Steuerung geschrieben.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte vom Datenträger werden in die Steuerung geschrieben.

## Ablauf der Übertragung bei Bedienung in der Rezepturanzeige

### Lesen aus der Steuerung durch Bedienung in der Rezepturanzeige

| Schritt | Aktion   |                            |
|---------|--|----------------------------|
| 1       | Prüfung: Statuswort = 0?   |                            |
|         | Ja   | Nein                       |
| 2       | Das Bediengerät trägt die zu lesende Rezepturnummer sowie den Status "Übertragung läuft" im Datenfach ein und setzt die Datensatznummer auf 0. | Abbruch mit Systemmeldung. |

| Schritt | Aktion  |
|---------|---|
| 3       | Das Bediengerät liest die Werte aus der Steuerung und zeigt diese in der Rezepturanzeige an.<br>Bei Rezepturen mit synchronisierten Variablen werden die Werte aus der Steuerung auch in die Variablen geschrieben. |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".   |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |

### Schreiben in die Steuerung durch Bedienung in der Rezepturanzeige

| Schritt | Aktion  |
|---------|---|
|         | Prüfung: Statuswort = 0?  |
| 1       | Ja  |
|         | Nein  |
|         | Das Bediengerät trägt die zu schreibende Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   |
| 2       | Das Bediengerät schreibt die aktuellen Werte in die Steuerung.<br>Bei Rezepturen mit synchronisierten Variablen werden die geänderten Werte zwischen Rezepturanzeige und Variablen abgeglichen und dann in die Steuerung geschrieben. |
| 3       | Das Bediengerät setzt den Status "Übertragung beendet".   |
| 4       | Das Steuerungsprogramm kann jetzt ggf. die übertragenen Daten auswerten.  |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |

#### Hinweis

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

#### Hinweis

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach ist auf "Übertragung beendet" gesetzt.
- Der Status im Datenfach ist auf "Übertragung mit Fehler beendet" gesetzt.

### Ablauf der Übertragung bei Steuerungsauftrag

Die Übertragung von Datensätzen zwischen Bediengerät und Steuerung kann vom Bediengerät oder von der Steuerung initiiert werden.

Für diese Art der Übertragung stehen die beiden Steuerungsaufträge Nr. 69 und Nr. 70 zur Verfügung.

### Nr. 69: Datensatz aus Steuerung lesen ("SPS → DAT")

Der Steuerungsauftrag Nr. 69 überträgt Datensätze von der Steuerung zum Bediengerät. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Höherwertiges Byte   | Niederwertiges Byte |
|--------|--|---------------------|
| Wort 1 | 0  | 69                  |
| Wort 2 | Rezepturnummer (1-999)   |                     |
| Wort 3 | Datensatznummer (1-65.535)   |                     |
| Wort 4 | Vorhandenen Datensatz nicht überschreiben: 0<br>Vorhandenen Datensatz überschreiben: 1 |                     |

### Nr. 70: Datensatz in Steuerung schreiben ("DAT → SPS")

Der Steuerungsauftrag Nr. 70 überträgt Datensätze vom Bediengerät zur Steuerung. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Höherwertiges Byte         | Niederwertiges Byte |
|--------|----------------------------|---------------------|
| Wort 1 | 0                          | 70                  |
| Wort 2 | Rezepturnummer (1-999)     |                     |
| Wort 3 | Datensatznummer (1-65.535) |                     |
| Wort 4 | —                          |                     |

### Ablauf bei Lesen aus der Steuerung durch Steuerungsauftrag "SPS → DAT" (Nr. 69)

| Schritt | Aktion  |                           |
|---------|---|---------------------------|
| 1       | Prüfung: Statuswort = 0?  |                           |
|         | Ja  | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese Werte im Steuerungsauftrag angegebenen Datensatz.   |                           |
| 4       | <ul style="list-style-type: none"> <li>• Falls im Auftrag "Überschreiben" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben. Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>• Falls im Auftrag "Nicht überschreiben" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                           |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                           |

**Ablauf bei Schreiben in die Steuerung durch Steuerungsauftrag "DAT → SPS" (Nr. 70)**

| Schritt | Aktion   |                           |
|---------|--|---------------------------|
| 1       | Prüfung: Statuswort = 0?   |                           |
|         | Ja   | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät holt die Werte des im Auftrag angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.   |                           |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                           |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                           |

**Ablauf der Übertragung bei Anstoß durch projektierte Funktion**

**Lesen aus der Steuerung durch projektierte Funktion**

| Schritt | Aktion  |                            |
|---------|---|----------------------------|
| 1       | Prüfung: Statuswort = 0?  |                            |
|         | Ja  | Nein                       |
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese in dem Datensatz ab, der in der Funktion angegeben ist.   |                            |
| 4       | <ul style="list-style-type: none"> <li>Falls in der Funktion bei "Überschreiben" "Ja" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben.<br/>Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>Falls in der Funktion bei "Überschreiben" "Nein" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                            |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                            |

**Schreiben in die Steuerung durch projektierte Funktion**

| Schritt | Aktion                   |      |
|---------|--------------------------|------|
| 1       | Prüfung: Statuswort = 0? |      |
|         | Ja                       | Nein |



| Schritt | Aktion   |                            |
|---------|--|----------------------------|
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät holt die Werte des in der Funktion angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.  |                            |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                            |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                            |

## Mögliche Fehlerursachen bei der Übertragung von Datensätzen

### Mögliche Fehlerursachen

Falls die Übertragung von Datensätzen mit Fehler beendet wird, so kann dies u. a. folgende Ursachen haben:

- Variablen-Adresse in der Steuerung nicht eingerichtet
- Überschreiben von Datensätzen nicht möglich
- Rezepturnummer nicht vorhanden
- Datensatznummer nicht vorhanden

---

#### Hinweis

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

---

#### Hinweis

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach wurde auf "Übertragung beendet" gesetzt.
  - Der Status im Datenfach wurde auf "Übertragung mit Fehler beendet" gesetzt.
- 

## Reaktion auf fehlerbedingten Abbruch

Das Bediengerät reagiert auf einen fehlerbedingten Abbruch der Übertragung von Datensätzen wie folgt:

- Anstoß über Bedienung in der Rezepturanzeige  
Hinweise in der Statuszeile der Rezepturanzeige und Ausgabe von Systemmeldungen
- Anstoß über Funktion  
Ausgabe von Systemmeldungen
- Anstoß über Steuerungsauftrag  
Keine Rückmeldung am Bediengerät

Unabhängig davon können Sie den Status der Übertragung durch Abfragen des Statusworts im Datenfach auswerten.

---

**Hinweis**

**Geräteabhängigkeit**

Hinweise in der Statuszeile der Rezepturanzeige stehen auf Basic Panels nicht zur Verfügung.

---

## Kurven

### Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt bei den Basic Panels zeitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 3201)

### Zeitgetriggerte Kurven

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein.

Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

## Meldungen

### Meldungen projektieren

### Meldungen projektieren

Um Meldungen wie Betriebs-, Störmeldungen und Quittierung zu projektieren sind mehrere Schritte notwendig.

- Schritt 1: Variablen anlegen
- Schritt 2: Meldungen projektieren
- Schritt 3: Quittierung projektieren

Nähere Hinweise hierzu finden Sie im Kapitel:

Arbeiten mit Meldungen (Seite 3213)

## Besonderheiten bei der Projektierung von Meldungen

Wenn Sie Verbindungen von Bediengeräten mit Steuerungen anderer Hersteller projektieren, beachten Sie folgende Besonderheiten bei der Projektierung:

- Datentypen der Variablen
- Adressierung der Variablen
- Zählweise der Bitstellen

## Datentypen

Für Verbindungen mit SIMATIC Kommunikationstreiber, werden folgende Datentypen unterstützt:

| Steuerung              | Zulässige Datentypen |   |
|------------------------|----------------------|---|
|                        | Bitmeldungen         | Analogmeldungen                                 |
| SIMATIC S7-Steuerungen | WORD, INT            | BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER |

## Zählweise der Bitstellen

Für Verbindungen mit SIMATIC Kommunikationstreiber, gilt folgende Zählweise:

| Zählweise der Bitstellen   | Byte 0             |  |  |  |  |  |   | Byte 1              |  |  |  |  |  |   |
|----------------------------|--------------------|--|--|--|--|--|---|---------------------|--|--|--|--|--|---|
|                            | Höherwertiges Byte |  |  |  |  |  |   | Niederwertiges Byte |  |  |  |  |  |   |
| In SIMATIC S7-Steuerungen  | 7                  |  |  |  |  |  | 0 | 7                   |  |  |  |  |  | 0 |
| Im WinCC projektieren Sie: | 15                 |  |  |  |  |  | 8 | 7                   |  |  |  |  |  | 0 |

## Quittierung von Meldungen

### Vorgehensweise

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

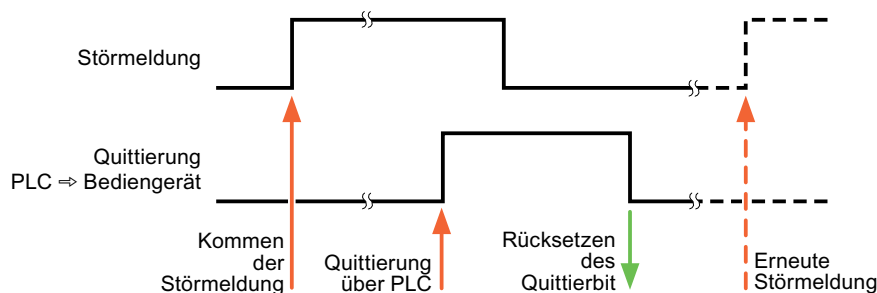
### Quittieren von der Steuerung

Unter "PLC-Quittiervariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulsdiagramm.



### Quittierung am Bediengerät

Unter "HMI-Quittiervariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittiervariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittiervariable in die Steuerung eine gewisse Zeitspanne.

Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittiervariablen gesetzt. Die gesamte Quittiervariable wird dann vom Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

---

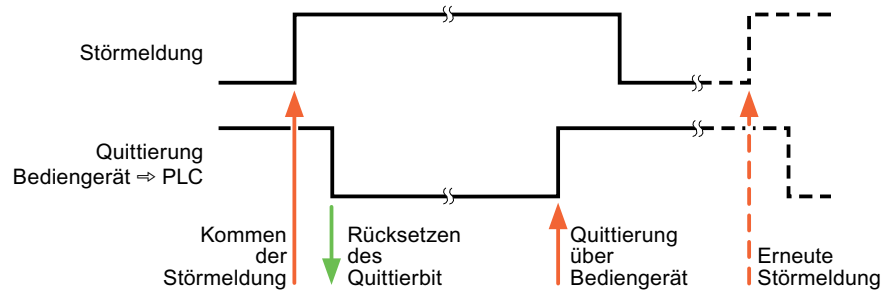
#### Hinweis

Alle seit dem letzten Runtime-Start quitierten Meldebites bleiben in der Quittiervariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittiervariablen der gesamte Bereich vom Bediengerät überschrieben wird.

---

Das nachfolgende Bild zeigt das Impulsdiagramm.



### 10.8.7.5 Leistungsmerkmale der Kommunikation

#### Geräteabhängigkeit S7 1200

#### Kommunikation mit der Steuerung SIMATIC S7 1200

Wenn Sie mit dem TIA Portal V 12 SP1 Geräte aus einer früheren Version vom TIA Portal verwenden kann die Projektierung von integrierten Verbindungen zu bestimmten Bediengeräten nicht möglich sein.

#### Version V1.0

Die Steuerung SIMATIC S7 1200 mit der Version V1.0 können Sie in folgenden Szenarien projektieren:

| TIA Portal | Basic Panels |
|------------|--------------|
| V11        | ja           |
| V12        | nein         |

#### Version V2.0

Die Steuerung SIMATIC S7 1200 mit der Version V2.0 können Sie in folgenden Szenarien projektieren:

| TIA Portal | Basic Panels |
|------------|--------------|
| V11        | ja           |
| V12        | ja           |

### Version V3.0

Die Steuerung SIMATIC S7 1200 mit der Version V3.0 können Sie in folgenden Szenarien projektieren:

| TIA Portal | Basic Panels |
|------------|--------------|
| V11        | ja           |
| V12        | ja           |

### Version V4.0

Die Steuerung SIMATIC S7 1200 mit der Version V4.0 können Sie in folgenden Szenarien projektieren:

| TIA Portal | Basic Panels |
|------------|--------------|
| V11        | nein         |
| V12        | ja           |

### Zulässige Datentypen für SIMATIC S7 1200 - V2

#### Zulässige Datentypen für Verbindungen mit SIMATIC S7 1200 (V2)

V2: Firmware Version V2.0

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeichern verwendet werden können.

| Datentyp         | Länge                      |
|------------------|----------------------------|
| BOOL             | 1 Bit                      |
| SINT             | 1 Byte                     |
| INT              | 2 Byte                     |
| DINT             | 4 Byte                     |
| USINT            | 1 Byte                     |
| UINT             | 2 Byte                     |
| UDINT            | 4 Byte                     |
| REAL             | 4 Byte                     |
| LREAL            | 8 Byte                     |
| TIME             | 4 Byte                     |
| DATE             | 2 Byte                     |
| DTL              | 12 Byte                    |
| TIME_OF_DAY, TOD | 4 Byte                     |
| STRING           | (2+n) Bytes, n = 0 bis 254 |
| CHAR             | 1 Byte                     |
| Array of CHAR    | --                         |

| Datentyp | Länge  |
|----------|--------|
| BYTE     | 1 Byte |
| WORD     | 2 Byte |
| DWORD    | 4 Byte |

### 10.8.7.6 Verbindungen im Editor "Verbindungen" anlegen

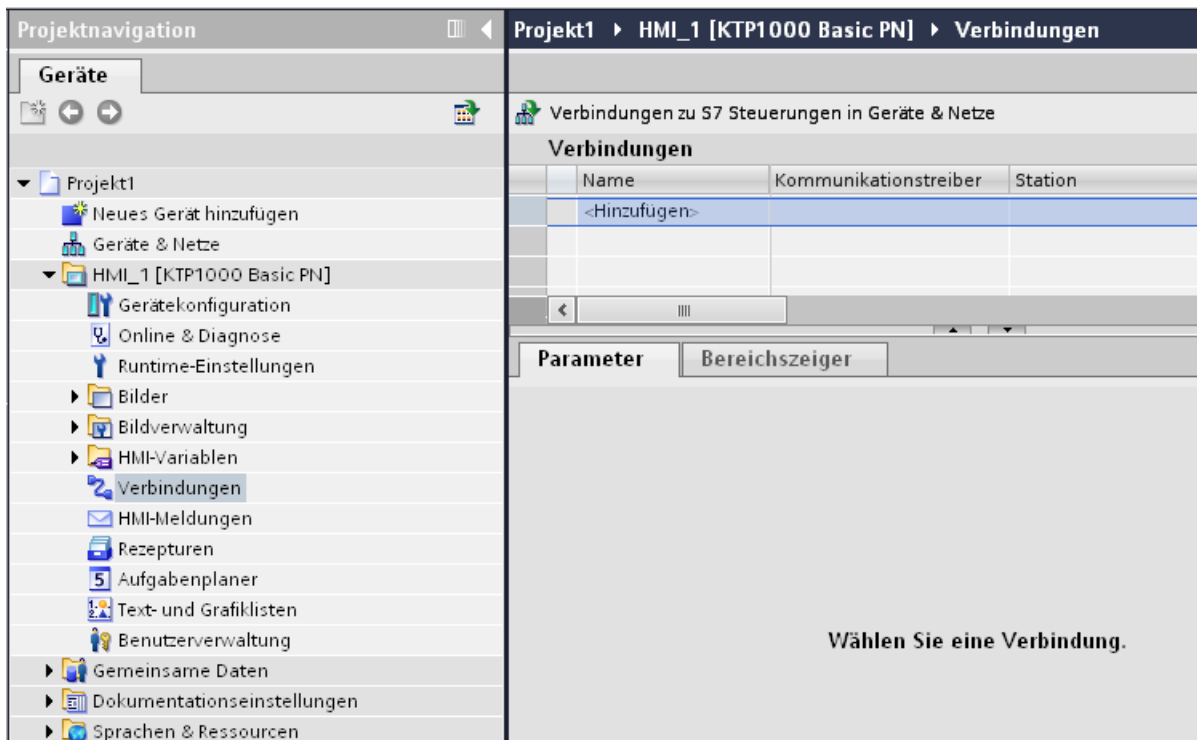
#### PROFINET-Verbindung anlegen

##### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät mit PROFINET-Schnittstelle ist angelegt.

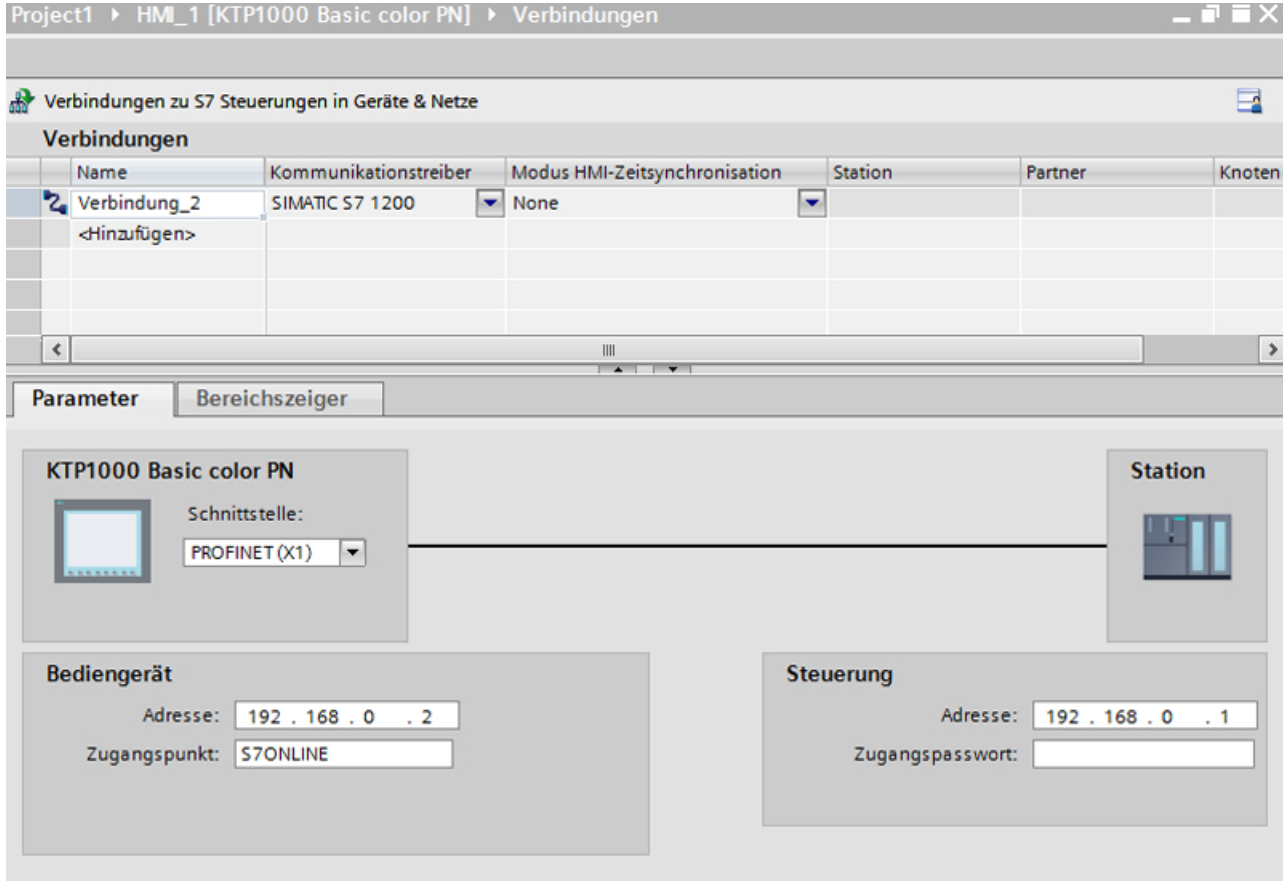
##### Vorgehensweise

1. Öffnen Sie den Editor "Verbindungen" des Bediengeräts.
2. Doppelklicken Sie auf "<Hinzufügen>".



3. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "SIMATIC S7 1200" aus.

4. Klicken Sie auf den Namen der Verbindung.
5. Wählen Sie im Inspektorfenster unter "Parameter > Schnittstelle" eine PROFINET-Schnittstelle des Bediengeräts aus.



6. Stellen Sie im Inspektorfenster die IP-Adressen der Kommunikationspartner ein:
  - Bediengerät: "Parameter > Bediengerät > Adresse"
  - Steuerung: "Parameter > Steuerung > Adresse"

## PROFIBUS DP-Verbindung anlegen

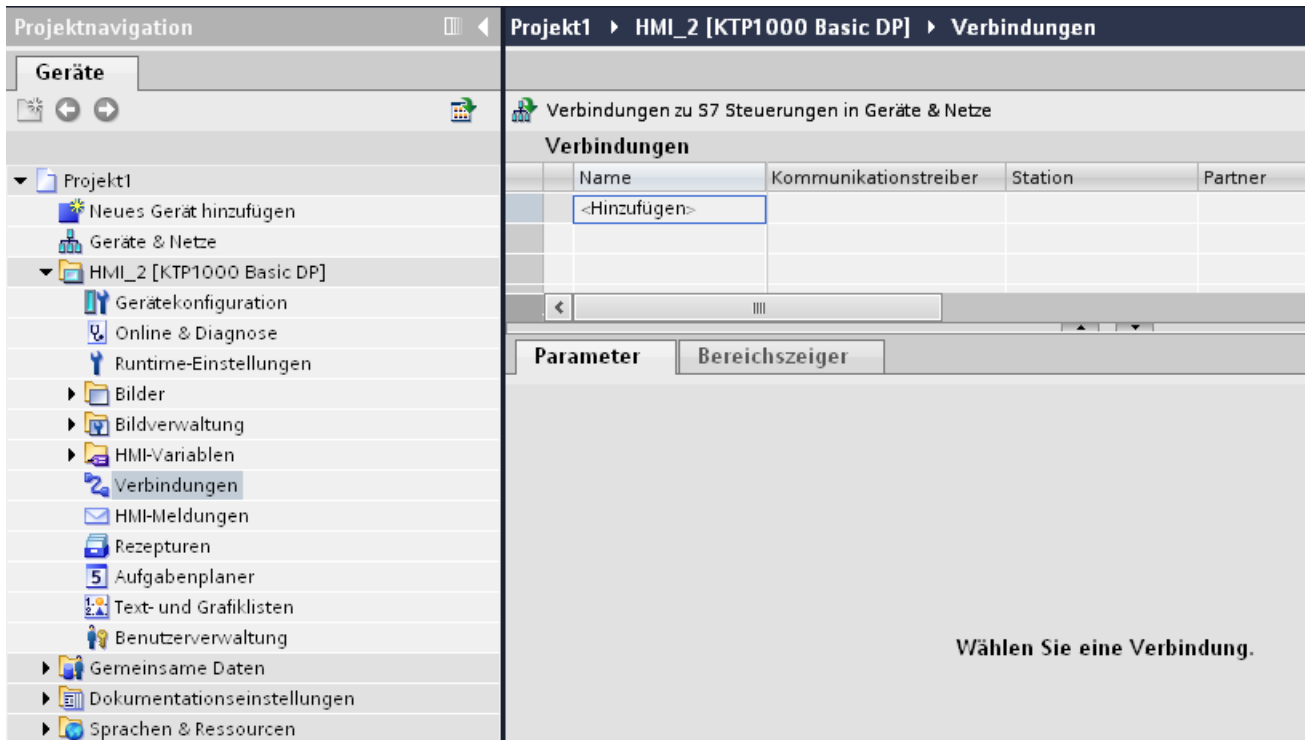
### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät mit PROFIBUS-Schnittstelle ist angelegt.



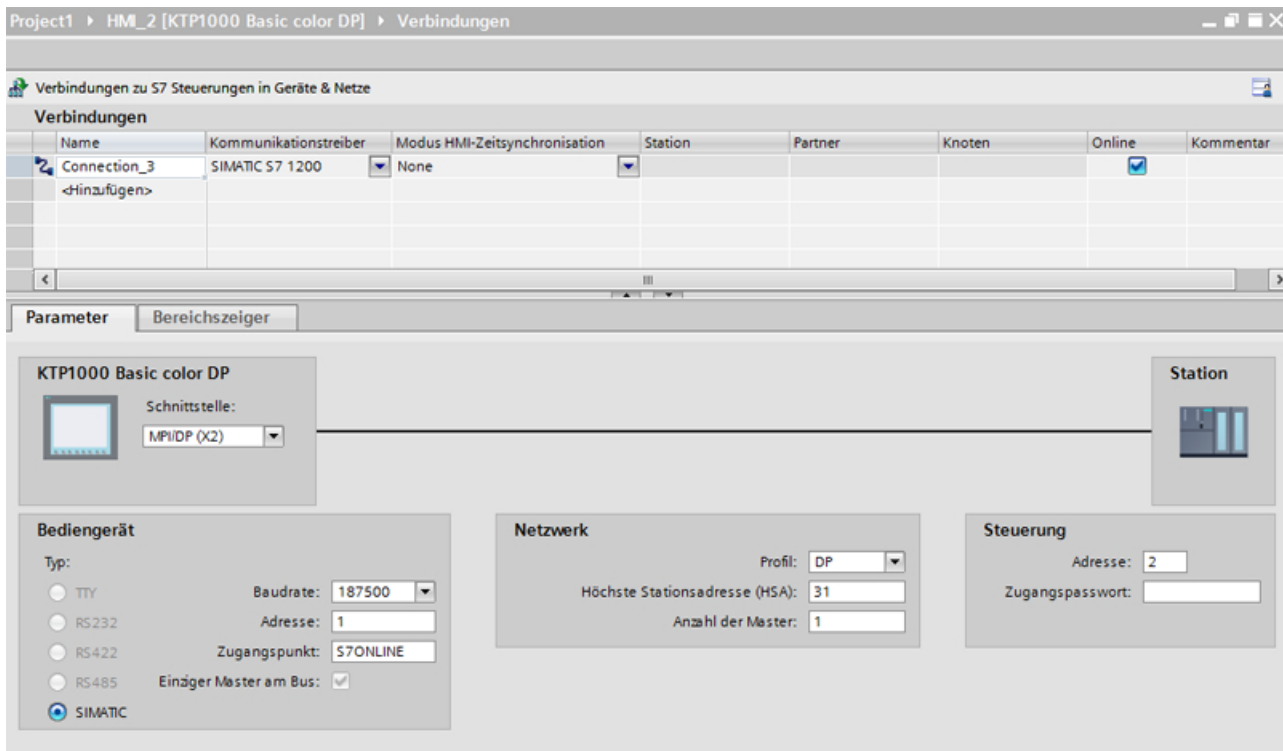
## Vorgehensweise

1. Öffnen Sie den Editor "Verbindungen" des Bediengeräts.
2. Doppelklicken Sie auf "<Hinzufügen>".



3. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "SIMATIC S7 1200" aus.
4. Klicken Sie auf den Namen der Verbindung.
5. Wählen Sie im Inspektorfenster unter "Parameter" die Schnittstelle "MPI/DP" aus.

6. Wählen Sie im Inspektorfenster unter "Parameter > Netzwerk" das Profil "DP" aus.



7. Stellen Sie im Inspektorfenster die Adressen der Kommunikationspartner ein:

- Bediengerät: "Parameter > Bediengerät > Adresse"
- Steuerung: "Parameter > Steuerung > Adresse"

## Parameter für die Verbindung

### Parameter für die Verbindung (SIMATIC S7 1200)

#### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät", "Netzwerk" und "Steuerung".

Project1 ▶ HM\_1 [KTP1000 Basic color PN] ▶ Verbindungen


Verbindungen zu S7 Steuerungen in Geräte & Netze

**Verbindungen**

| Name         | Kommunikationstreiber | Modus HMI-Zeitsynchronisation | Station | Partner | Knoten |
|--------------|-----------------------|-------------------------------|---------|---------|--------|
| Verbindung_2 | SIMATIC S7 1200       | None                          |         |         |        |
| <Hinzufügen> |                       |                               |         |         |        |


Parameter Bereichszeiger

**KTP1000 Basic color PN**



Schnittstelle: PROFINET (X1)

**Station**



---

**Bediengerät**

Adresse: 192 . 168 . 0 . 2

Zugangspunkt: S7ONLINE

**Steuerung**

Adresse: 192 . 168 . 0 . 1

Zugangspasswort:

## Ethernet-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie die Parameter für das Bediengerät im Netzwerk ein. Die Änderungen werden nicht automatisch auf das Bediengerät übertragen. Sie müssen die Einstellungen in der Systemsteuerung des Bediengeräts ändern.

- "Schnittstelle"  
Wenn Sie beim Projektieren direkt mit dem Bediengerät verbunden sind, können Sie die IP-Adresse des Bediengeräts in WinCC einrichten.

---

#### Hinweis

Wenn Sie die IP-Adresse bereits im Control-Panel des Bediengeräts eingerichtet haben, wird die IP-Adresse im Control Panel beim nächsten Laden überschrieben.

Wenn Sie "IP-Adresse auf anderem Weg beziehen" aktivieren, bleibt die bereits eingerichtete IP-Adresse im Control Panel beim nächsten Laden erhalten.

---

Beim Projekttransfer wird die IP-Adresse auf das Bediengerät übertragen. Die IP-Adresse des Bediengeräts richten Sie folgendermaßen ein:

- Klicken Sie auf das Bediengerät.
- Öffnen Sie den Editor "Gerätekonfiguration"
- Klicken Sie auf die Ethernet-Schnittstelle.
- Vergeben Sie im Inspektorfenster die IP-Adresse unter:  
"Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"
- "Adresse"  
Im Bereich "Adresse" vergeben Sie die IP-Adresse des Bediengeräts. Wenn Sie das WinCC-Projekt auf das Bediengerät transferieren, dann wird diese IP-Adresse direkt im Bediengerät eingerichtet.
- "Zugangspunkt"  
Der Zugangspunkt legt den Zugangspunkt für die PG/PC-Schnittstelle fest, über den der Kommunikationspartner erreicht werden kann.

## Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die IP-Adresse der S7-Baugruppe ein, an die das Bediengerät angeschlossen ist.
- "Zugangspasswort"  
Unter "Zugangspasswort" geben Sie ein Passwort ein. Das Passwort muss mit dem Passwort übereinstimmen, das Sie in der Steuerung hinterlegt haben.

---

### Hinweis

Ein Passwort ist nur notwendig, wenn Sie in der Steuerung die Schutzstufe "Kompletter Schutz" hinterlegt haben.

Wenn die Schutzstufe "Kompletter Schutz" in der Steuerung hinterlegt ist und Sie kein Passwort eingeben, wird keine Verbindung zur Steuerung aufgebaut.

---

## PROFIBUS-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie einmal die Parameter für das Bediengerät im Netzwerk ein. Die Änderung gilt für jeden Kommunikationspartner.

- "Typ"  
Legt den benutzten physikalischen Anschluss fest.
- "Schnittstelle"  
Unter "Schnittstelle" wählen Sie die Schnittstelle des Bediengeräts aus, über die das Bediengerät am PROFIBUS-Netz angeschlossen ist.
- "Baudrate"  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit der Daten im Netzwerk ein. Die Baudrate wird durch das langsamste ans Netzwerk angeschlossene Bediengerät festgelegt. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

### Hinweis

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5 Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Adresse"  
Unter "Adresse" stellen Sie die PROFIBUS-Adresse des Bediengeräts ein. Die PROFIBUS-Adresse muss im PROFIBUS-Netz eindeutig sein.

- "Einzigster Master am Bus"  
Deaktiviert eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des Bediengeräts an das Netz. Eine passive Station (Slave) kann nur dann Daten senden, wenn Sie von einer aktiven Station (Master) dazu aufgefordert wird.  
Bei S7-200 müssen Sie ein Bediengerät als Master einstellen.
- "Zugangspunkt"  
Der Zugangspunkt legt einen logischen Gerätenamen fest, über den der Kommunikationspartner erreicht wird.

### Parameter für das Netzwerk

Unter "Netzwerk" stellen Sie die Parameter für das PROFIBUS-Netz ein, an dem das Bediengerät eingebunden ist.

- "Profil"  
Unter "Profil" wählen Sie das Netzprofil aus, das im Netzwerk verwendet wird. Stellen Sie unter "Profil" "DP", "Universal" oder "Standard" ein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Höchste Stationsadresse"  
Unter "Höchste Stationsadresse" stellen Sie die höchste Stationsadresse ein. Die höchste Stationsadresse muss größer gleich der größten tatsächlichen PROFIBUS-Adresse sein. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

#### Hinweis

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Anzahl der Master"  
Unter "Anzahl der Master" stellen Sie die Anzahl der Master im PROFIBUS-Netz ein. Diese Angabe ist erforderlich, damit die Busparameter korrekt berechnet werden.

## Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die PROFIBUS-Adresse der S7-Baugruppe (CPU, FM oder CP) ein, an die das Bediengerät angeschlossen ist.
- "Zugangspasswort"  
Unter "Zugangspasswort" geben Sie ein Passwort ein. Das Passwort muss mit dem Passwort übereinstimmen, das Sie in der Steuerung hinterlegt haben.

---

### Hinweis

Ein Passwort ist nur notwendig, wenn Sie in der Steuerung die Schutzstufe "Kompletter Schutz" hinterlegt haben.

Wenn die Schutzstufe "Kompletter Schutz" in der Steuerung hinterlegt ist und Sie kein Passwort eingeben, wird keine Verbindung zur Steuerung aufgebaut.

---

## 10.8.8 Mit SIMATIC S7 300/400 kommunizieren

### 10.8.8.1 Kommunikation mit SIMATIC S7 300/400

#### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und den Steuerungen S7 300 und S7 400 beschrieben. Die beiden Steuerungen werden zusammengefasst als SIMATIC S7 300/400 benannt.

Folgende Kommunikationskanäle können Sie für die Steuerung SIMATIC S7 300/400 projektieren:

- PROFINET
- PROFIBUS
- MPI

#### HMI-Verbindung für die Kommunikation

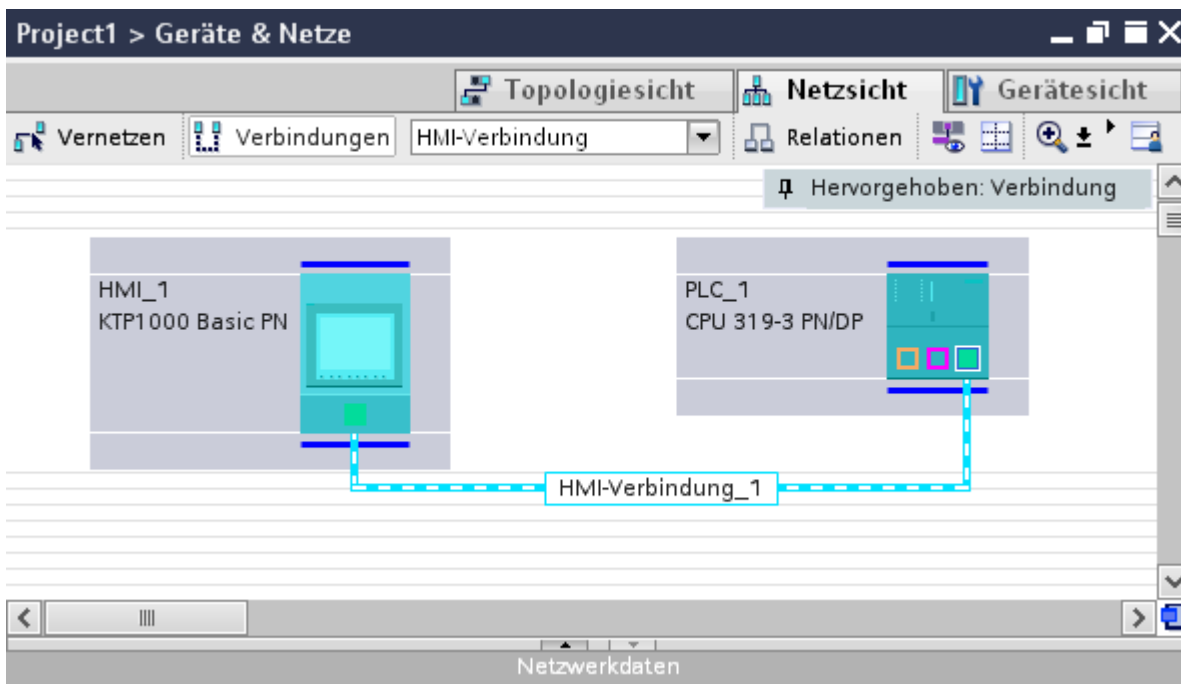
HMI-Verbindungen von Bediengerät und SIMATIC S7 300/400 projektieren Sie im Editor "Geräte & Netze".

## 10.8.8.2 Kommunikation über PROFINET

### Kommunikation über PROFINET

#### HMI-Verbindungen über PROFINET

Wenn Sie ein Bediengerät und eine SIMATIC S7 300/400 in das Projekt eingefügt haben, verbinden Sie die beiden PROFINET-Schnittstellen im Editor "Geräte & Netze" miteinander.



Sie können auch mehrere Bediengeräte an eine SIMATIC S7 300/400 anschließen und mehrere SIMATIC S7 300/400 an ein Bediengerät anschließen. Die maximale Anzahl der Kommunikationspartner, die Sie an ein Bediengerät anschließen können, ist abhängig vom verwendeten Bediengerät.

Weitere Informationen dazu finden Sie in der Dokumentation zum jeweiligen Bediengerät.

#### HMI-Verbindung im Editor "Geräte & Netze"

Die HMI-Verbindung über PROFINET zwischen der Steuerung mit dem Bediengerät projektieren Sie im Editor "Geräte & Netze".

#### Verbindung im Editor "Verbindungen"


Alternativ projektieren Sie die Verbindung über PROFINET zwischen Steuerung und Bediengerät im Editor "Verbindungen" des Bediengeräts.



## HMI-Verbindung über PROFINET projektieren

### Einleitung

Eine HMI-Verbindung über PROFINET bzw. Ethernet zwischen Bediengeräten und einer SIMATIC S7 300/400 projektieren Sie im Editor "Geräte & Netze".

|   |
|---|
|  <b>VORSICHT</b>                     |
| <b>Kommunikation über Ethernet</b>  |
| Bei Ethernet-basierter Kommunikation ist der Endanwender für die Sicherheit seines Datennetzes selbst verantwortlich. |
| Gezielte Angriffe können zur Überlast des Geräts führen und die Funktionsfähigkeit beeinträchtigen.                   |

### Voraussetzungen

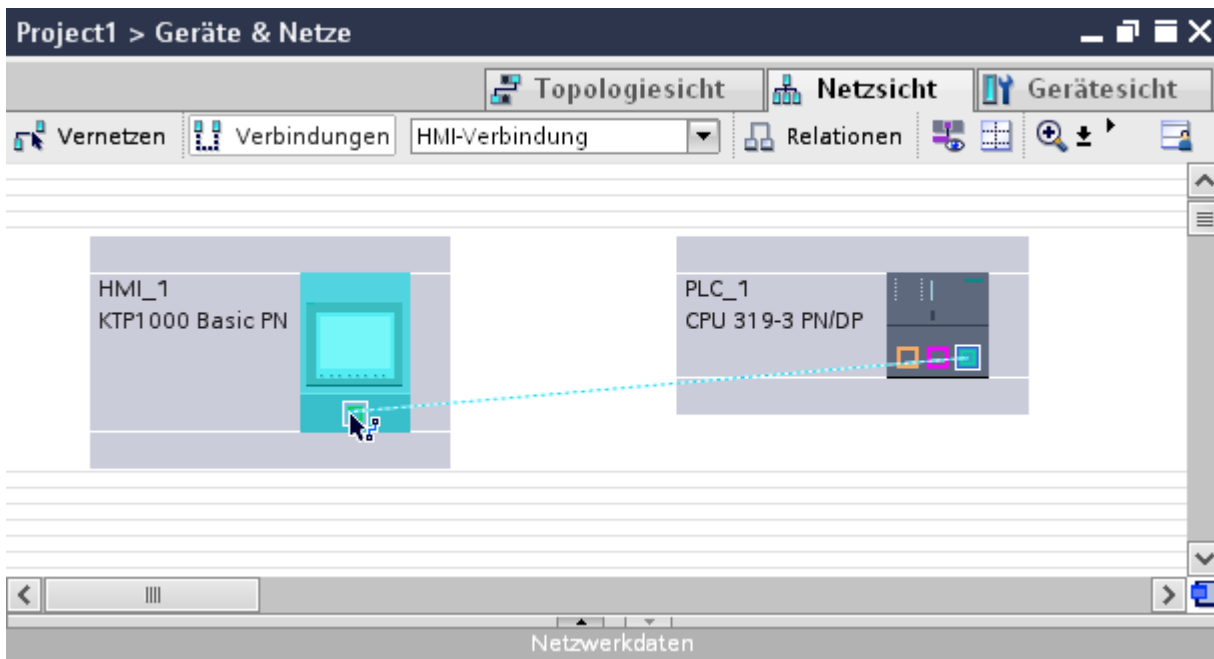
Folgende Kommunikationspartner sind im Editor "Geräte & Netze" angelegt:

- Bediengerät mit PROFINET- bzw. Ethernet-Schnittstelle
- SIMATIC S7 300/400 mit PROFINET-Schnittstelle.

### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Geräte & Netze".  
In der Netzsicht werden die im Projekt vorhandenen Kommunikationspartner grafisch dargestellt.
2. Klicken Sie auf die Schaltfläche "Verbindungen" und wählen Sie den Verbindungstyp "HMI-Verbindung".  
Die für eine Verbindung verfügbaren Geräte werden farbig markiert dargestellt.

3. Klicken Sie in die PROFINET-Schnittstelle der Steuerung und ziehen Sie per Drag&Drop eine Verbindung zur PROFINET- bzw. Ethernet-Schnittstelle des Bediengeräts.



4. Klicken Sie auf die Verbindungslinie.
5. Klicken Sie auf "HMI-Verbindung Hervorheben" und wählen Sie die HMI-Verbindung aus. Im Inspektorfenster wird die Verbindung grafisch dargestellt.
6. Klicken Sie auf die Kommunikationspartner in der "Netzsicht" und ändern Sie im Inspektorfenster die PROFINET-Parameter nach den Anforderungen ihres Projekts. Nähere Hinweise hierzu finden Sie im Kapitel "PROFINET-Parameter (Seite 3605)".

#### Hinweis

Die erstellte HMI-Verbindung wird auch im tabellarischen Bereich des Editors auf der Registerkarte "Verbindungen" angezeigt. In der Tabelle kontrollieren Sie die Verbindungsparameter.

Den lokalen Verbindungsnamen der Verbindung können Sie nur in der Tabelle ändern.

## Ergebnis

Sie haben eine Verbindung zwischen einem Bediengerät und einer SIMATIC S7 300/400 angelegt. Die Verbindungsparameter IP-Adresse und Subnetzmaske sind projiziert.

## PROFINET-Parameter

### PROFINET-Parameter für die HMI-Verbindung

### PROFINET-Parameter für die HMI-Verbindung

Eine Übersicht über die projektierten Parameter einer HMI-Verbindung finden Sie in den Eigenschaften der HMI-Verbindung.

Änderungen sind in diesem Inspektorfenster nur eingeschränkt möglich.

### Parameter der HMI-Verbindung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die HMI-Verbindung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der HMI-Verbindung.

The screenshot shows the 'HMI Verbindung: HMI-Verbindung\_1' properties window. The 'Allgemein' tab is active, showing the 'Verbindung' section with 'Offline Status' checked. The 'Verbindungsweg' section displays a diagram of the connection between a local HMI device and a partner PLC device. Below the diagram, the connection parameters are listed in a table:

| Parameter          | Lokal                  | Partner                 |
|--------------------|------------------------|-------------------------|
| Endpunkt:          | HMI_1                  | PLC_1                   |
| Schnittstelle:     | IE_CP_1, PROFINET_Schr | PLC_1, PROFINET-Schnitt |
| Schnittstellentyp: | Ethernet/IP            | Ethernet/IP             |
| Subnetz:           | PN/IE_1                | PN/IE_1                 |
| Adresse:           | 192.168.0.2            | 192.168.0.1             |

A 'Finde Verbindungsweg' button is located at the bottom right of the parameter table.

## "Verbindung"

Zeigt an, ob die Geräte bereits miteinander vernetzt sind.



- Wird angezeigt, wenn die Geräte miteinander vernetzt sind.



- Wird angezeigt, wenn die Geräte nicht miteinander vernetzt sind.

## "Verbindungsweg"

Im Bereich "Verbindungsweg" werden die Kommunikationspartner der gewählten HMI-Verbindung und deren PROFINET-Parameter angezeigt. Einige der gezeigten Bereiche sind in diesem Dialog nicht editierbar.

- "Endpunkt"  
Zeigt den Gerätenamen an. Dieser Bereich ist nicht editierbar.
- "Schnittstelle"  
Zeigt die gewählte Schnittstelle des Geräts an. Abhängig vom Gerät können Sie zwischen mehreren Schnittstellen wählen.
- "Schnittstellen-Typ"  
Zeigt den gewählten Schnittstellen-Typ an. Dieser Bereich ist nicht editierbar.
- "Subnetz"  
Zeigt das gewählte Subnetz an. Dieser Bereich ist nicht editierbar.
- "Adresse"  
Zeigt die gewählte IP-Adresse des Geräts an. Dieser Bereich ist nicht editierbar.
- Schaltfläche "Finde Verbindungsweg"  
Ermöglicht das nachträgliche Spezifizieren von Verbindungen.

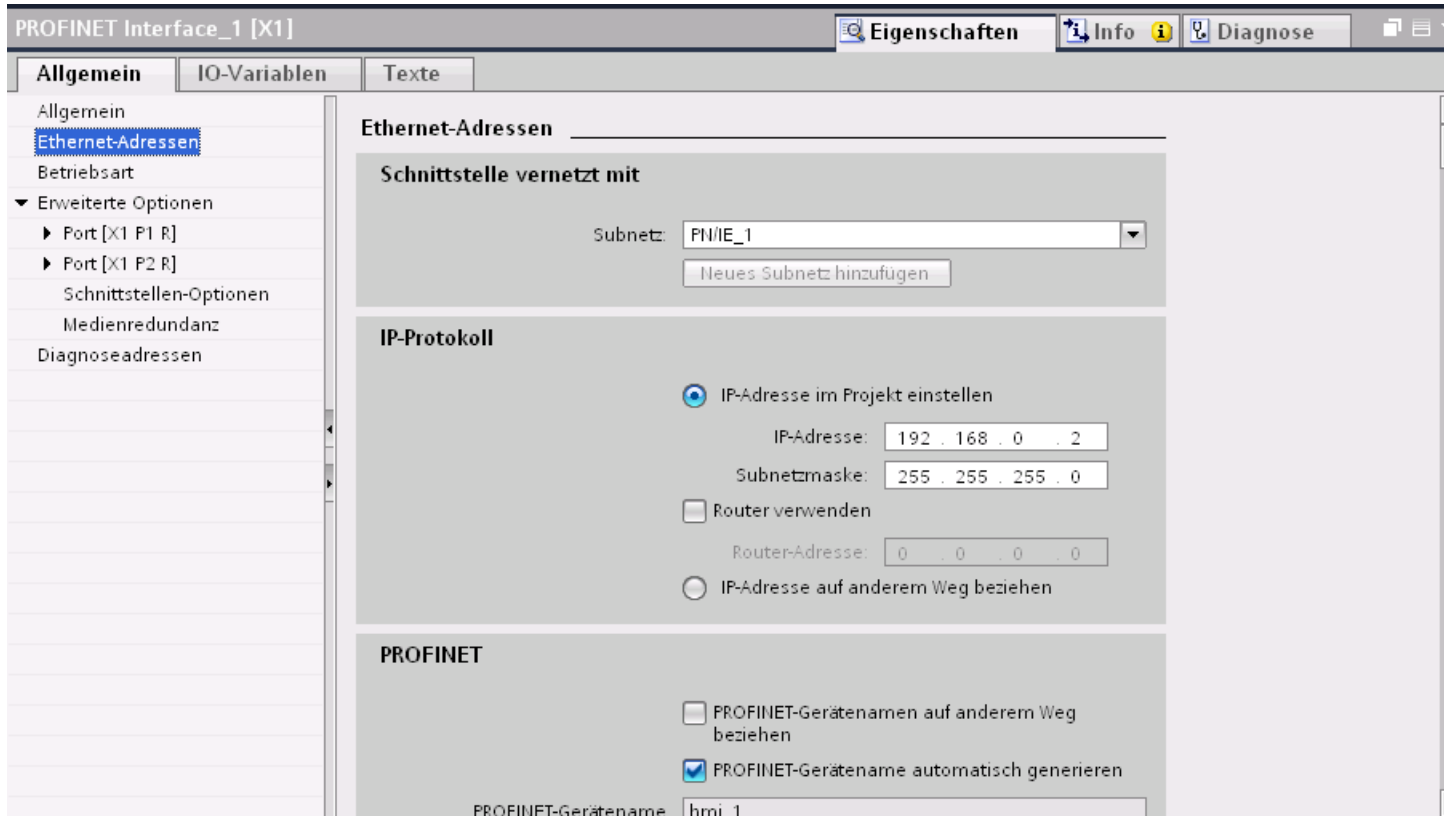
## PROFINET-Parameter für das Bediengerät

### PROFINET Parameter für das Bediengerät

Eine Übersicht über die projektierten Parameter des Bediengeräts finden Sie in den Eigenschaften des Bediengeräts.

## PROFINET-Parameter des Bediengeräts anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf das Bediengerät.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein" die Parameter des Bediengeräts.



### "Schnittstelle vernetzt mit"

Im Bereich "Schnittstelle vernetzt mit" wählen Sie das Subnetz der HMI-Verbindung aus, über die das Bediengerät an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

## "IP-Protokoll"

- "IP-Adresse im Projekt einstellen"  
Wenn Sie das WinCC-Projekt auf das Bediengerät transferieren, dann wird diese IP-Adresse direkt im Bediengerät eingerichtet.

---

### Hinweis

Bei Bediengeräten mit dem Betriebssystem Windows CE 3.0 erfolgt automatisch ein Neustart.

Bediengeräte mit Windows CE 3.0:

- OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"
- 
- "Subnetzmaske"  
Im Bereich "Subnetzmaske" vergeben Sie Daten der Subnetzmaske.
  - "IP-Router verwenden"  
Wenn Sie einen IP-Router verwenden, aktivieren Sie "IP-Router verwenden" und geben Sie die Routeradresse im Feld "Routeradresse" ein.
  - "IP-Adresse auf anderem Weg beziehen"  
Wenn die Funktion "IP-Adresse auf anderem Weg beziehen" aktiviert ist, dann wird die IP-Adresse nicht aus dem Projekt übernommen. Sie müssen die IP-Adresse direkt im Control Panel des Bediengeräts eingeben.

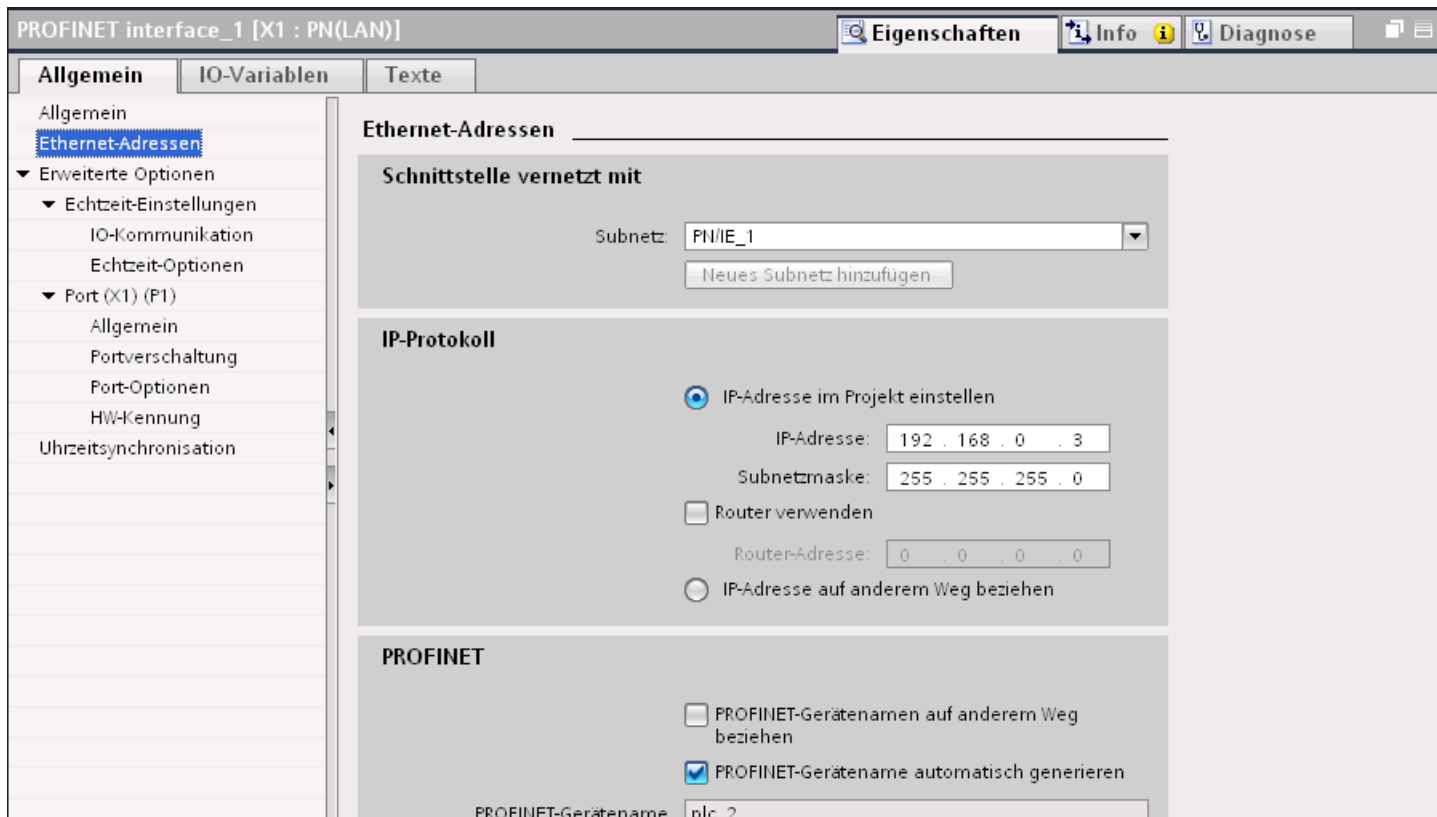
## PROFINET-Parameter für die Steuerung

### PROFINET-Parameter für die Steuerung

Eine Übersicht über die projektierten Parameter finden Sie in den Eigenschaften der Steuerung.

## PROFINET-Parameter der Steuerung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die Steuerung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der Steuerung.



### "Schnittstelle vernetzt mit"

Im Bereich "Subnetz" wählen Sie das Subnetz der HMI-Verbindung aus, über die die Steuerung an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "IP-Protokoll"

- "Schnittstellen-Typ"  
Abhängig vom Bediengerätetyp stehen verschiedene Schnittstellen zur Auswahl.
- "IP-Adresse"  
Im Bereich "IP-Adresse" vergeben Sie die IP-Adresse des Bediengeräts.
- "Subnetzmaske"  
Im Bereich "Subnetzmaske" vergeben Sie Daten der Subnetzmaske.  
Wenn Sie einen IP-Router verwenden, aktivieren Sie "IP-Router verwenden" und geben Sie die Routeradresse im Feld darunter ein.

## Netzkonfiguration Industrial Ethernet

### Regeln für die Netzkonfiguration

Die Ethernet-Schnittstellen der Geräte besitzen als IP-Adresse eine Default-Adresse, die von Ihnen geändert werden kann.

### IP-Adresse

Die IP-Parameter sind sichtbar, wenn die kommunikationsfähigen Geräte das TCP/IP-Protokoll unterstützen.

Die IP-Adresse besteht aus 4 Dezimalzahlen mit dem Wertebereich 0 bis 255. Die Dezimalzahlen sind durch einen Punkt voneinander getrennt.

Beispiel: 140.80.0.2

Die IP-Adresse setzt sich folgendermaßen zusammen:

- Der Adresse des (Sub-) Netzes
- Der Adresse des Teilnehmers (im allgemeinen auch Host oder Netzknoten genannt)

### Subnetzmaske

Die Subnetzmaske trennt diese beiden Adressen. Sie bestimmt, welcher Teil der IP-Adresse das Netz adressiert und welcher Teil der IP-Adresse den Teilnehmer adressiert.

Die gesetzten Bits der Subnetzmaske bestimmen den Netzteil der IP-Adresse.

Beispiel:

Subnetzmaske: 255.255.0.0 = 11111111.11111111.00000000.00000000

Im Beispiel für die oben genannte IP-Adresse hat die hier gezeigte Subnetzmaske folgende Bedeutung:

Die ersten 2 Bytes der IP-Adresse bestimmen das Subnetz - also 140.80. Die letzten beiden Bytes adressieren den Teilnehmer - also 0.2.

Allgemein gilt:

- Die Netzadresse ergibt sich aus der UND-Verknüpfung von IP-Adresse und Subnetzmaske.
- Die Teilnehmeradresse ergibt sich aus der UND-NICHT-Verknüpfung von IP-Adresse und Subnetzmaske.

### Zusammenhang IP-Adresse und Default-Subnetzmaske

Es gibt eine Vereinbarung hinsichtlich der Zuordnung von IP-Adressbereichen und so genannten "Default-Subnetzmasken". Die erste Dezimalzahl der IP-Adresse (von links)



bestimmt den Aufbau der Default-Subnetzmaske hinsichtlich der Anzahl der Werte "1" (binär) wie folgt:

| IP-Adresse (dez.) | IP-Adresse (bin.)     | Adressklasse | Default-Subnetzmaske |
|-------------------|-----------------------|--------------|----------------------|
| 0 bis 126         | 0xxxxxxx.xxxxxxxx.... | A            | 255.0.0.0            |
| 128 bis 191       | 10xxxxxx.xxxxxxxx...  | B            | 255.255.0.0          |
| 192 bis 223       | 110xxxxx.xxxxxxxx...  | C            | 255.255.255.0        |

#### Hinweis

##### Wertebereich für erste Dezimalstelle

Für die erste Dezimalzahl der IP-Adresse ist auch ein Wert zwischen 224 und 255 möglich (Adressklasse D etc). Dies ist jedoch nicht empfehlenswert, da für diese Werte keine Adressprüfung erfolgt.

### Weitere Subnetze maskieren

Über die Subnetzmaske können Sie ein Subnetz, das einer der Adressklassen A, B oder C zugeordnet ist, weiter strukturieren und "private" Subnetze bilden, indem Sie weitere niederwertige Stellen der Subnetzmaske auf "1" setzen. Pro jedem auf "1" gesetztem Bit verdoppelt sich die Anzahl der "privaten" Netze und halbiert sich die Anzahl der darin enthaltenen Teilnehmer. Nach außen wirkt das Netzwerk nach wie vor wie ein einzelnes Netzwerk.

Beispiel:

Sie ändern bei einem Subnetz der Adressklasse B (z. B. IP-Adresse 129.80.xxx.xxx) die Default-Subnetzmaske wie folgt:

| Masken               | Dezimal       | Binär                               |
|----------------------|---------------|-------------------------------------|
| Default-Subnetzmaske | 255.255.0.0   | 11111111.11111111.00000000.00000000 |
| Subnetzmaske         | 255.255.128.0 | 11111111.11111111.10000000.00000000 |

Ergebnis:

Alle Teilnehmer mit Adressen von 129.80.001.xxx bis 129.80.127.xxx befinden sich auf einem Subnetz, alle Teilnehmer mit Adressen von 129.80.128.xxx bis 129.80.255.xxx auf einem anderen Subnetz.

### Netzübergang (Router)

Die Netzübergänge (Router) haben die Aufgabe, die Subnetze zu verbinden. Wenn ein IP-Datagramm an ein anderes Netzwerk geschickt werden soll, dann muss es zunächst an einen Router vermittelt werden. Damit das möglich ist, müssen Sie in diesem Fall für jeden Teilnehmer des Subnetzes die Adresse des Routers eingeben.

Die IP-Adresse eines Teilnehmers im Subnetz und die Adresse des Netzübergangs (Router) dürfen nur an den Stellen unterschiedlich sein, an denen in der Subnetzmaske "0" steht.

## Port-Optionen festlegen

### Festlegen der Port-Optionen

#### Verbindungseinstellungen für den PROFINET-IO-Port ändern

Bei Bedarf können Sie die Netzwerkeinstellungen für den PROFINET-IO-Port ändern. Standardmäßig werden die Einstellungen automatisch festgelegt, was im Normalfall eine problemlose Kommunikation gewährleistet.

#### Einstellmöglichkeiten Übertragungsrate/Duplex

Abhängig von dem gewählten Gerät können Sie folgende Einstellungen für "Übertragungsrate/Duplex" vornehmen:

- Automatische Einstellung  
Empfohlene Default-Einstellung des Ports. Die Übertragungseinstellungen mit dem Partner-Port werden automatisch "ausgehandelt". In der Voreinstellung ist automatisch auch die Option "Autonegotiation aktivieren" aktiviert, d. h. Sie können Crosskabel oder Patchkabel für den Anschluss verwenden.
- TP/ITP mit x Mbit/s. Vollduplex (Halbduplex)  
Einstellung des Übertragungsrate und des Vollduplex/Halbduplex-Betriebs. Die Wirkung ist abhängig von der eingestellten Option "Autonegotiation aktivieren":
  - Autonegotiation aktiviert  
Sie können sowohl Crosskabel als auch Patchkabel verwenden.
  - Autonegotiation deaktiviert  
Achten Sie darauf, dass Sie das korrekte Kabel verwenden (Patch- oder Crosskabel)! Auch bei dieser Einstellung wird der Port überwacht.
- Deaktiviert  
Abhängig vom Baugruppentyp kann in der Klappliste die Option "deaktiviert" angeboten werden. Sie haben damit beispielsweise die Möglichkeit, den Zugang zu einem unbenutzten Port aus Sicherheitsgründen zu unterbinden. Bei dieser Einstellung werden keine Diagnoseereignisse erzeugt.

#### Option "Überwachen"

Mit dieser Option wird die Portdiagnose aktiviert bzw. deaktiviert. Beispiele für Portdiagnose: Der Link-Status wird überwacht, d. h. Diagnose bei Link-Down erzeugt und bei Fiber Optic Ports wird die Systemreserve überwacht.

#### Option "Autonegotiation aktivieren"

Die Einstellung für Autonegotiation ist nur änderbar, wenn ein konkretes Medium (z. B. TP 100 Mbit/s vollduplex) ausgewählt ist. Ob ein konkretes Medium eingestellt werden kann, hängt von den Eigenschaften des Moduls ab.

Ist die Autonegotiation deaktiviert, dann wird die feste Einstellung des Ports erzwungen, ähnlich wie das z. B. bei einem priorisierten Hochlaufs des IO-Devices erforderlich ist.

Sie müssen für gleiche Einstellungen beim Partner-Port sorgen, da bei dieser Option die Betriebsparameter des angeschlossenen Netzes nicht erkannt werden und entsprechend die Datenübertragungsrate und Übertragungsmodus nicht optimal eingestellt werden können.

---

**Hinweis**

STEP 7 übernimmt bei einem verschalteten lokalen Port die Einstellung für den Partner Port, falls der Partner Port die Einstellung unterstützt. Falls der Partner Port die Einstellung nicht unterstützt, wird eine Fehlermeldung erzeugt.

---

**Verkabelungsregeln bei deaktivierter Autonegotiation**

**Voraussetzung**

Sie haben z. B. zur Beschleunigung der Hochlaufzeit des IO-Devices für den betreffenden Port folgende Einstellungen vorgenommen:

- Feste Übertragungsrate
- Autonegotiation inkl. Autocrossing deaktiviert

Die Zeit für das Aushandeln der Übertragungsrate beim Hochlauf wird somit eingespart.

Wenn Sie Autonegotiation deaktiviert haben, müssen Sie die Verkabelungsregeln beachten.

**Verkabelungsregeln bei deaktivierter Autonegotiation**

PROFINET-Geräte besitzen folgende zwei Arten von Ports:

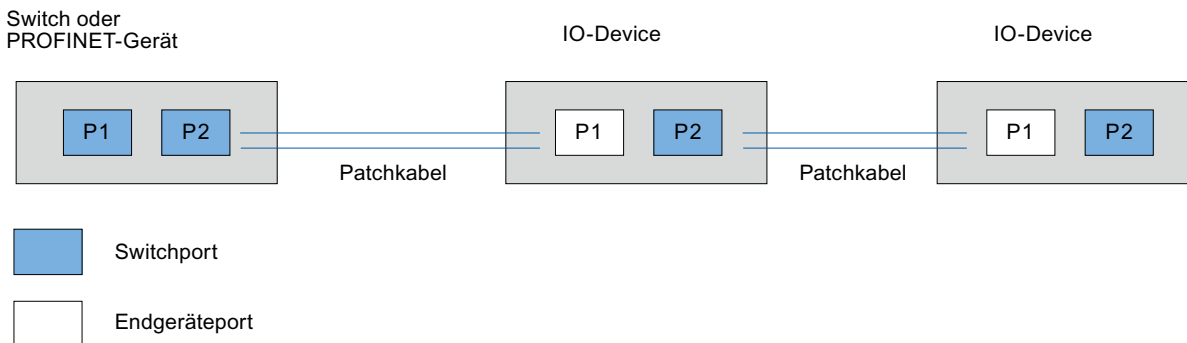
| Art des Ports                                  | PROFINET-Geräte   | Bemerkung  |
|--|---|--|
| Switchport mit gekreuzter Pinbelegung          | Bei IO-Devices: Port 2<br>Bei S7-CPU's mit 2 Ports: Port 1 und Port 2 | Gekreuzte Pinbelegung bedeutet, dass die Pinbelegung der Ports für Senden und Empfangen zwischen den betreffenden PROFINET-Geräten intern vertauscht werden. |
| Endgeräteport mit nicht gekreuzter Pinbelegung | Bei IO-Devices: Port 1<br>Bei S7-CPU's mit einem Port: Port 1         | -  |

**Gültigkeit der Verkabelungsregeln**

Die im folgenden Absatz beschriebenen Verkabelungsregeln gelten ausschließlich für den Fall, dass Sie eine feste Porteingstellung vorgenommen haben.

## Regeln für die Verkabelung

Mehrere IO-Devices können Sie mit einem Kabeltyp (Patch-Kabel) in Linie verbinden. Dazu verbinden Sie Port 2 des IO-Devices (Dezentrale Peripherie) mit dem Port 1 des nächsten IO-Devices. In der folgenden Grafik ist ein Beispiel mit zwei IO-Devices dargestellt.



## Begrenzungen am Port

### Voraussetzung

Um mit Boundaries ("Begrenzungen") arbeiten zu können, muss das jeweilige Gerät mehr als einen Port besitzen. Wenn das PROFINET keine Boundaries-Einstellungen unterstützt, werden sie auch nicht angezeigt.

### Boundaries aktivieren

Unter "Boundaries" versteht man Grenzen für die Übertragung bestimmter Ethernet-Frames. Es können folgende Begrenzungen an einem Port gesetzt werden:

- "Ende der Erfassung erreichbarer Teilnehmer"  
DCP-Frames zur Erfassung erreichbarer Teilnehmer werden nicht weitergeleitet. Hinter diesem Port liegende Teilnehmer werden nicht mehr unter "erreichbare Teilnehmer" in der Projektnavigation angezeigt. Hinter diesem Port liegende Teilnehmer können von der CPU nicht mehr erreicht werden.
- "Ende der Topologieerkennung"  
LLDP-Frames (Link Layer Discovery Protocol) zur Topologieerkennung werden nicht weitergeleitet.
- "Ende der Sync-Domain"  
Sync-Frames, die zur Synchronisierung von Teilnehmern innerhalb einer Sync-Domain übertragen werden, werden nicht weitergeleitet.  
Wenn Sie z. B. ein PROFINET-Gerät mit mehr als zwei Ports in einem Ring betreiben, dann sollten Sie das Einspeisen von Sync-Frames in den Ring durch Setzen einer Sync-Boundary (an den Ports, die nicht im Ring sind) verhindern.  
Weiteres Beispiel: Möchten Sie mehrere Sync-Domains nutzen, dann projektieren Sie eine Sync-Domain-Boundary für den Port, der mit einem PROFINET-Gerät der jeweils anderen Sync-Domain verbunden ist.

## Einschränkungen

Folgende Einschränkungen sind zu beachten:

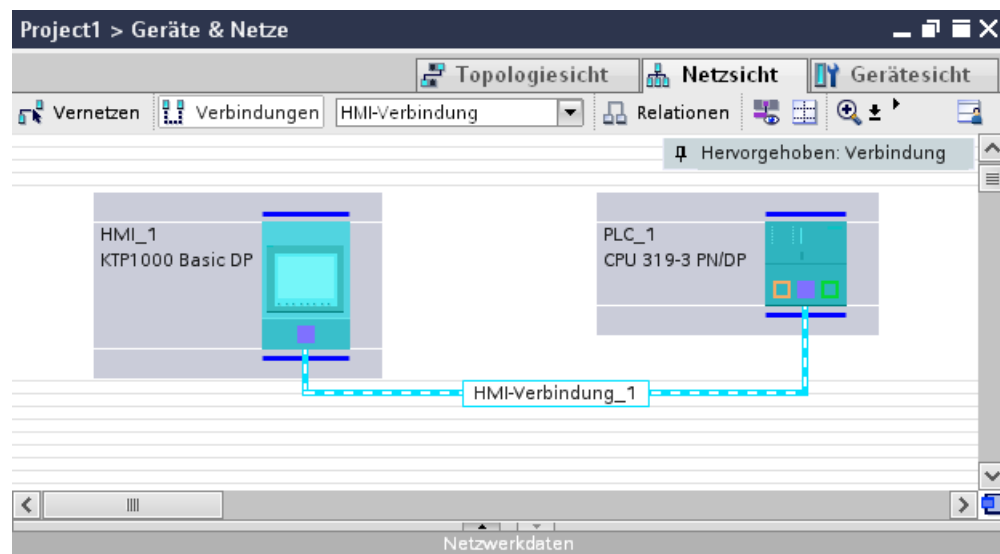
- Nur wenn der Port das jeweilige Feature unterstützt sind die einzelnen Optionskästchen bedienbar.
- Wenn für den Port ein Partnerport bestimmt wurde sind folgende Optionskästchen nicht bedienbar:
  - "Ende der Erfassung erreichbarer Teilnehmer"
  - "Ende der Topologieerkennung"
- Wenn Autonegotiation deaktiviert ist, dann sind alle Optionskästchen nicht bedienbar.

### 10.8.8.3 Kommunikation über PROFIBUS

## Kommunikation über PROFIBUS

### HMI-Verbindungen über PROFIBUS

Wenn Sie ein Bediengerät und eine SIMATIC S7 300/400 in das Projekt eingefügt haben, verbinden Sie die beiden PROFIBUS-Schnittstellen im Editor "Geräte & Netze" miteinander.



### HMI-Verbindung im Editor "Geräte & Netze"

Die HMI-Verbindung über PROFIBUS zwischen der Steuerung mit dem Bediengerät projektieren Sie im Editor "Geräte & Netze".

## Verbindung im Editor "Verbindungen"

Alternativ projektieren Sie die Verbindung zwischen Steuerung und Bediengerät im Editor "Verbindungen" des Bediengeräts.

## HMI-Verbindung über PROFIBUS projektieren

### Einleitung

Eine HMI-Verbindung über PROFIBUS zwischen Bediengeräten und einer SIMATIC S7 300/400 projektieren Sie im Editor "Geräte & Netze".

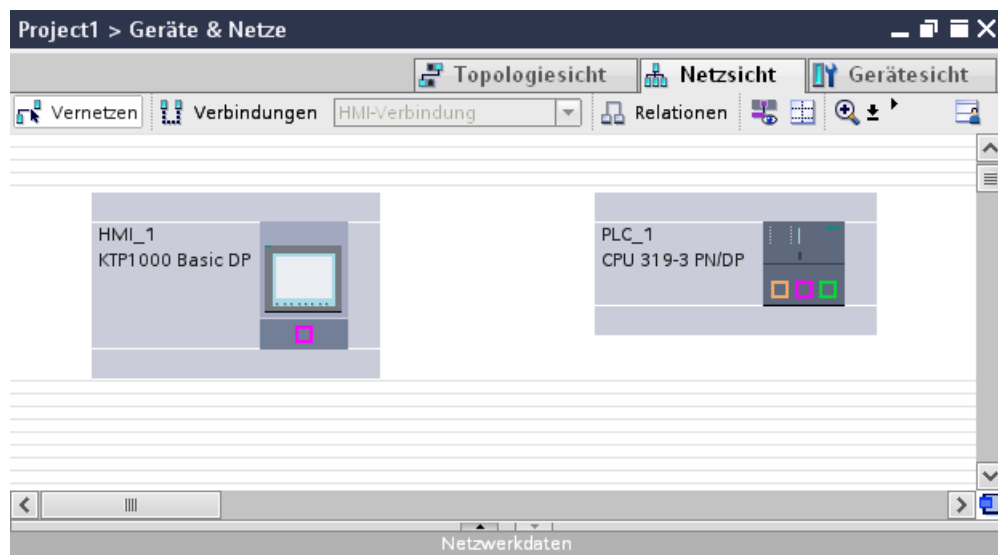
### Voraussetzungen

Folgende Kommunikationspartner sind im Editor "Geräte & Netze" angelegt:

- Bediengerät mit MPI/DP-Schnittstelle
- SIMATIC S7 300/400 mit PROFIBUS-Schnittstelle

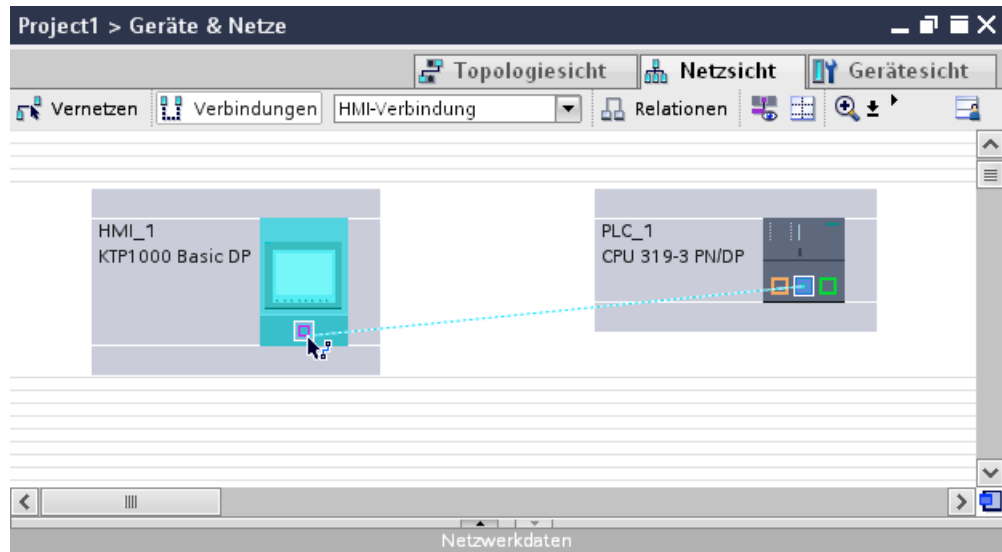
### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Geräte & Netze". In der Netzsicht werden die im Projekt vorhandenen Kommunikationspartner grafisch dargestellt.



2. Klicken Sie auf die Schaltfläche "Verbindungen". Die für eine Verbindung verfügbaren Geräte werden farbig markiert dargestellt.
3. Klicken Sie auf die Schnittstelle des Bediengeräts.
4. Wählen Sie im Inspektorfenster "Eigenschaften > Allgemein > HMI MPIDP > Parameter" den Schnittstellen-Typ "PROFIBUS" aus.

5. Klicken Sie in die Schnittstelle der Steuerung und ziehen Sie per Drag&Drop eine Verbindung zum Bediengerät.



6. Klicken Sie auf die Verbindungslinie.
7. Klicken Sie auf "HMI-Verbindung Hervorheben" und wählen Sie die HMI-Verbindung aus. Im Inspektorfenster wird die Verbindung grafisch dargestellt.
8. Klicken Sie auf die Kommunikationspartner in der "Netzansicht" und ändern Sie im Inspektorfenster die PROFINET-Parameter nach den Anforderungen ihres Projekts. Nähere Hinweise hierzu finden Sie im Kapitel "PROFIBUS-Parameter (Seite 3617)".

---

#### Hinweis

Die erstellte HMI-Verbindung wird auch im tabellarischen Bereich des Editors auf der Registerkarte "Verbindungen" angezeigt. In der Tabelle kontrollieren Sie die Verbindungsparameter.

Den lokalen Verbindungsnamen der Verbindung können Sie nur in der Tabelle ändern.

---

## Ergebnis

Sie haben eine HMI-Verbindung zwischen einem Bediengerät und einer SIMATIC S7 300/400 über PROFIBUS angelegt.

## PROFIBUS-Parameter

### PROFIBUS-Parameter für die HMI-Verbindung

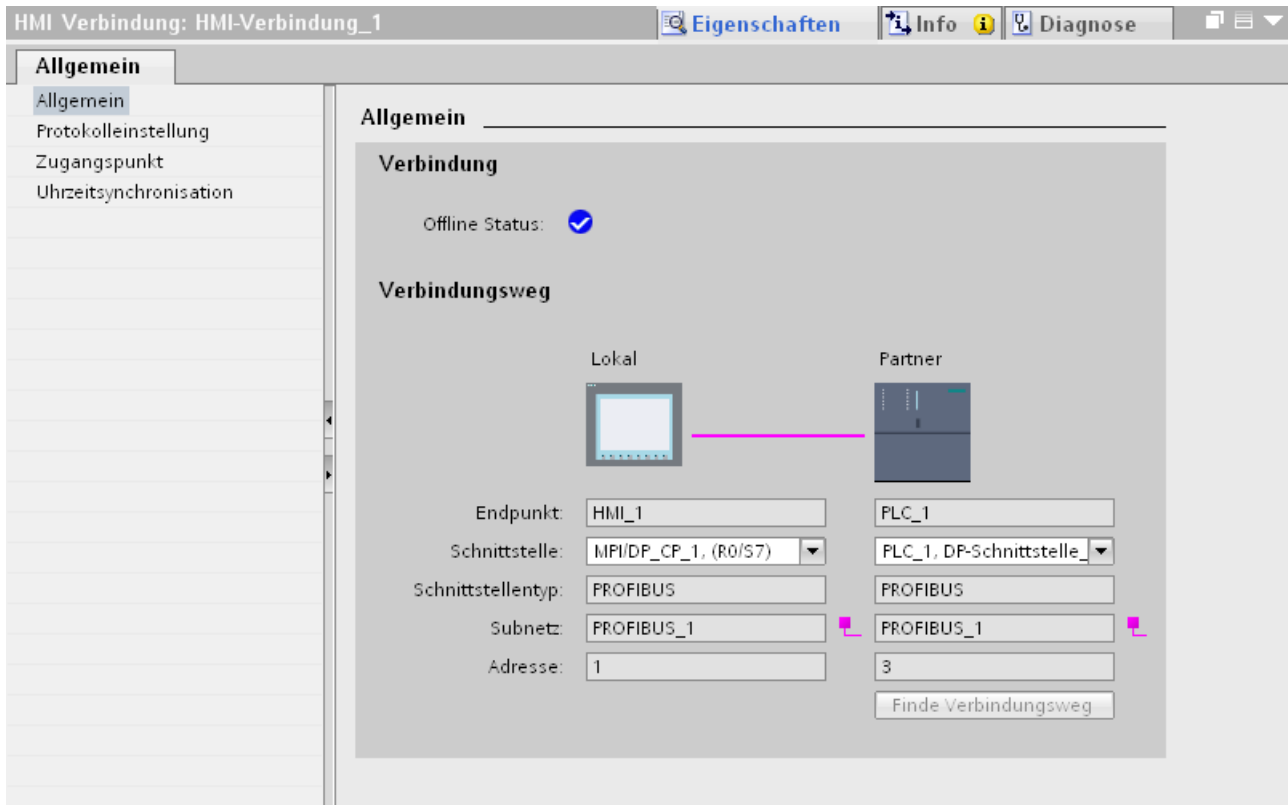
### PROFIBUS-Parameter für die HMI-Verbindung

Eine Übersicht über die projektierten Parameter einer HMI-Verbindung finden Sie in den Eigenschaften der HMI-Verbindung.

Änderungen sind in diesem Inspektorfenster nur eingeschränkt möglich.



### Parameter der HMI-Verbindung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die HMI-Verbindung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der HMI-Verbindung.



### "Verbindung"

Zeigt an, ob die Geräte bereits miteinander vernetzt sind.

-  - Wird angezeigt, wenn die Geräte miteinander vernetzt sind.
-  - Wird angezeigt, wenn die Geräte nicht miteinander vernetzt sind.



## "Verbindungsweg"

Im Bereich "Verbindungsweg" werden die Kommunikationspartner der gewählten HMI-Verbindung und deren PROFIBUS-Parameter angezeigt. Einige der gezeigten Bereiche sind in diesem Dialog nicht editierbar.

- "Endpunkt"  
Zeigt den Gerätenamen an. Dieser Bereich ist nicht editierbar.
- "Schnittstelle"  
Zeigt die gewählte Schnittstelle des Geräts an. Abhängig vom Gerät können Sie zwischen mehreren Schnittstellen wählen.
- "Schnittstellen-Typ"  
Zeigt den gewählten Schnittstellen-Typ an. Dieser Bereich ist nicht editierbar.
- "Subnetz"  
Zeigt das gewählte Subnetz an. Dieser Bereich ist nicht editierbar.
- "Adresse"  
Zeigt die PROFIBUS-Adresse des Geräts an. Dieser Bereich ist nicht editierbar.
- Schaltfläche "Finde Verbindungsweg"  
Ermöglicht das nachträgliche Spezifizieren von Verbindungen.

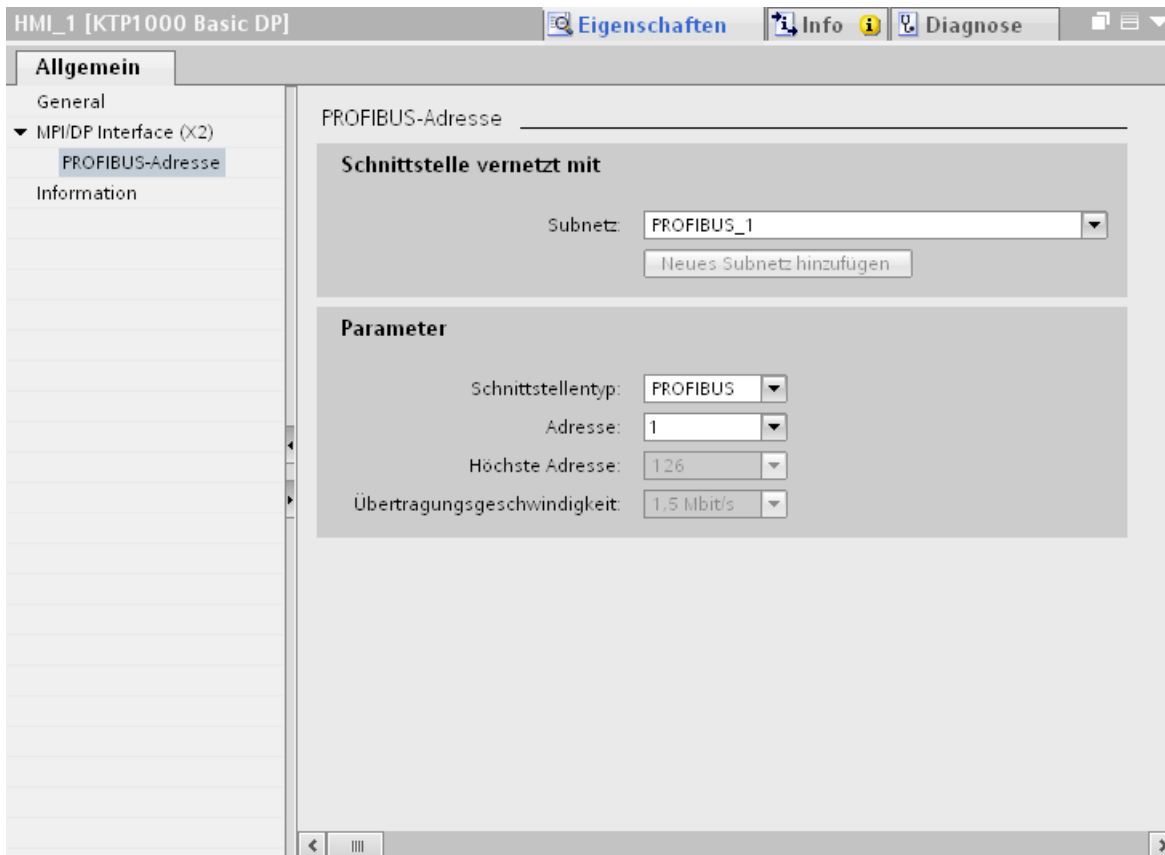
## PROFIBUS-Parameter für das Bediengerät

### PROFIBUS-Parameter für das Bediengerät

Eine Übersicht über die projektierten Parameter des Bediengeräts finden Sie in den Eigenschaften des Bediengeräts.

## PROFINET-Parameter des Bediengeräts anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf das Bediengerät.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter des Bediengeräts.



### "Schnittstelle vernetzt mit"

Im Bereich "Schnittstelle vernetzt mit" wählen Sie das Subnetz der HMI-Verbindung aus, über die das Bediengerät an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "Parameter"

- "Schnittstellen-Typ"  
Im Bereich "Schnittstellentyp" parametrieren Sie den Schnittstellentyp. Abhängig vom Bediengerätetyp stehen hier verschiedenen Schnittstellen zur Auswahl.
- "Adresse"  
Im Bereich "Adresse" parametrieren Sie die PROFIBUS-Adresse des Bediengeräts. Die PROFIBUS-Adresse muss im PROFIBUS-Netzwerk eindeutig sein

- "Höchste Adresse"  
Im Bereich "Höchste Adresse" wird die höchste Adresse des PROFIBUS-Netzwerks angezeigt.
- "Übertragungsgeschwindigkeit"  
Die "Übertragungsgeschwindigkeit" wird durch das langsamste ans Netzwerk angeschlossene Gerät festgelegt. Die Einstellung ist im gesamten Netzwerk gleich.

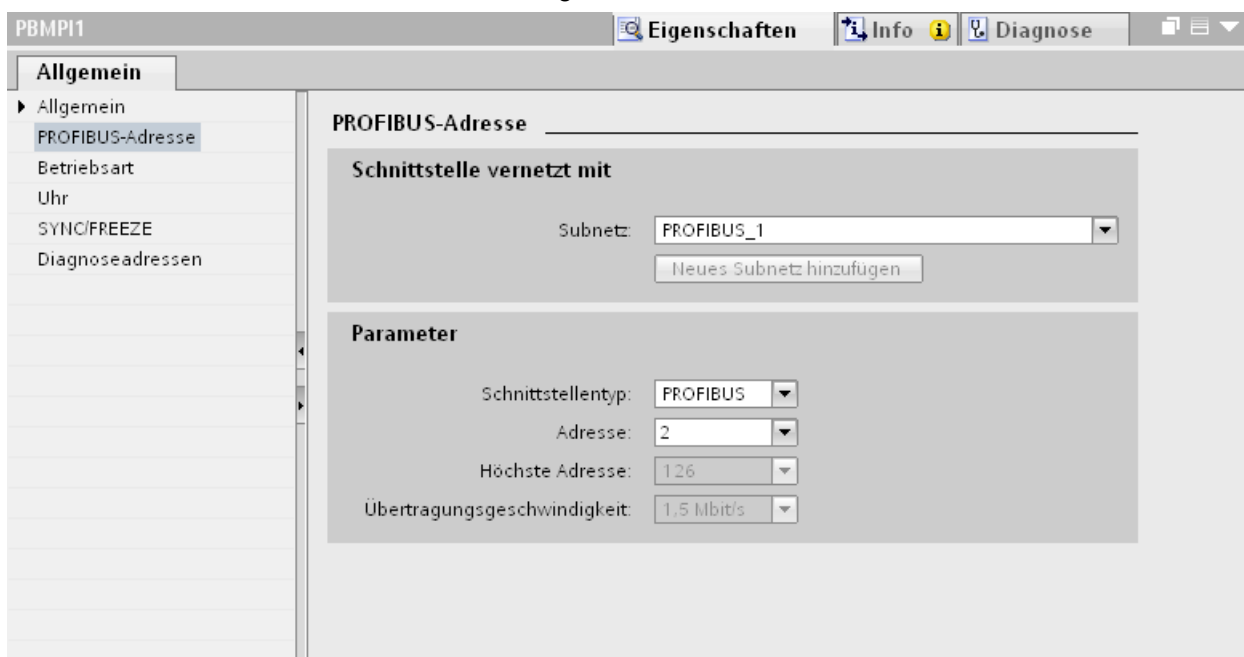
## PROFIBUS-Parameter für die Steuerung

### PROFIBUS-Parameter für Steuerung

Eine Übersicht über die projektierten Parameter finden Sie in den Eigenschaften der Steuerung.

### PROFIBUS-Parameter der Steuerung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die Steuerung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der Steuerung.



### "Schnittstelle vernetzt mit"

Im Bereich "Subnetz" wählen Sie das Subnetz der HMI-Verbindung aus, über die die Steuerung an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "Parameter"

- "Schnittstellen-Typ"  
Abhängig vom Bediengerätetyp stehen verschiedene Schnittstellen zur Auswahl.
- "Adresse"  
Im Bereich "Adresse" parametrieren Sie die PROFIBUS-Adresse des Bediengeräts. Die PROFIBUS-Adresse muss im PROFIBUS-Netzwerk eindeutig sein
- "Höchste Adresse"  
Im Bereich "Höchste Adresse" wird die höchste Adresse des PROFIBUS-Netzwerks angezeigt.
- "Übertragungsgeschwindigkeit"  
Die "Übertragungsgeschwindigkeit" wird durch das langsamste ans Netzwerk angeschlossene Gerät festgelegt. Die Einstellung ist im gesamten Netzwerk gleich.

### Busprofile bei PROFIBUS

#### Einführung

Je nach angeschlossenen Gerätetypen und verwendeten Protokollen am PROFIBUS stehen unterschiedliche Profile zur Verfügung. Die Profile unterscheiden sich hinsichtlich der Einstellmöglichkeiten und hinsichtlich der Berechnung der Busparameter. Im Folgenden werden die Profile erläutert.

#### Teilnehmer mit unterschiedlichen Profilen am selben PROFIBUS-Subnetz

Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Busparameter aller Teilnehmer dieselben Werte haben.

#### Profile und Übertragungsgeschwindigkeiten

| Profile    | Unterstützte Übertragungsgeschwindigkeiten in kbit/s |
|------------|--|
| DP         | 9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000  |
| Standard   | 9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000  |
| Universell | 9,6 19,2 93,75 187,5 500 1500                        |

## Bedeutung der Profile

| Profil     | Bedeutung   |
|------------|---|
| DP         | <p>Wählen Sie das Busprofil "DP", wenn nur Geräte an das PROFIBUS-Subnetz angeschlossen sind, welche die Anforderungen der Norm EN 50170 Volume 2/3, Part 8-2 PROFIBUS erfüllen. Die Einstellung der Busparameter ist auf diese Geräte optimiert.</p> <p>Hierzu zählen Geräte mit DP-Master- und DP-Slave-Schnittstellen der SIMATIC S7 sowie Dezentrale Peripheriegeräte anderer Hersteller.</p>                                     |
| Standard   | <p>Gegenüber dem Profil "DP" bietet das Profil "Standard" zusätzlich die Möglichkeit, Teilnehmer eines anderen Projektes oder Teilnehmer, die hier nicht projektiert wurden, bei der Berechnung der Busparameter zu berücksichtigen. Die Busparameter werden dann nach einem einfachen, nicht optimierten Algorithmus berechnet.</p>  |
| Universell | <p>Wählen Sie das Busprofil "Universell", wenn einzelne Teilnehmer am PROFIBUS-Subnetz den Dienst PROFIBUS-FMS nutzen.</p> <p>Hierzu gehören z. B. folgende Geräte:</p> <ul style="list-style-type: none"><li>• CP 343-5</li><li>• PROFIBUS-FMS-Geräte anderer Hersteller</li></ul> <p>Auch hier besteht wie im Profil "Standard" die Möglichkeit, zusätzliche Teilnehmer bei der Berechnung der Busparameter zu berücksichtigen.</p> |

### 10.8.8.4 Kommunikaiton über MPI

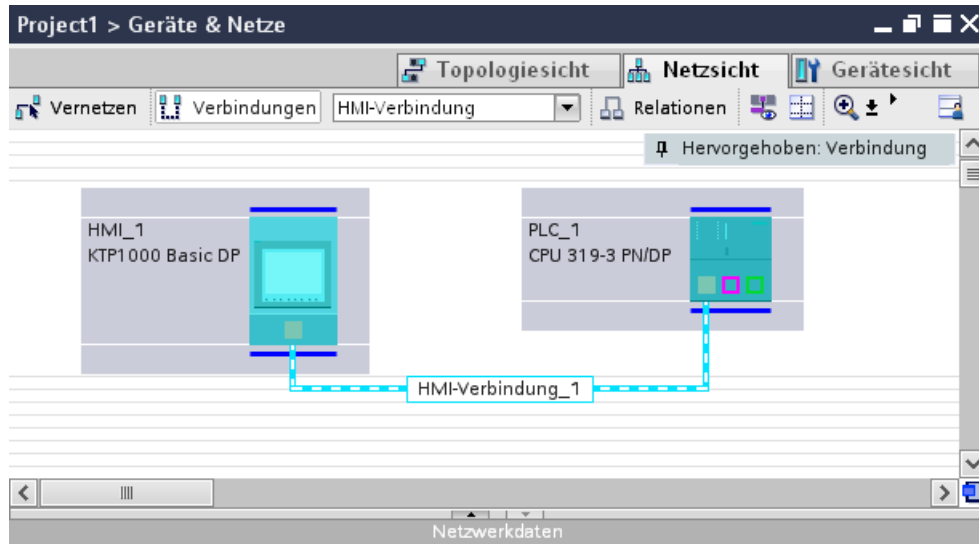
#### Kommunikation über MPI

#### HMI-Verbindungen über MPI

Wenn Sie ein Bediengerät und eine SIMATIC S7 300/400 in das Projekt eingefügt haben, verbinden Sie die beiden MPI-Schnittstellen im Editor "Geräte & Netze" miteinander.

## HMI-Verbindung im Editor "Geräte & Netze"

Die HMI-Verbindung über PROFIBUS zwischen der Steuerung mit dem Bediengerät projektieren Sie im Editor "Geräte & Netze".



## Verbindung im Editor "Verbindungen"

Alternativ projektieren Sie die Verbindung zwischen Steuerung und Bediengerät im Editor "Verbindungen" des Bediengeräts.

## HMI-Verbindung über MPI projektieren

### Einleitung

Eine HMI-Verbindung über MPI zwischen Bediengeräten und einer SIMATIC S7 300/400 projektieren Sie im Editor "Geräte & Netze".

### Voraussetzungen

Folgende Kommunikationspartner sind im Editor "Geräte & Netze" angelegt:

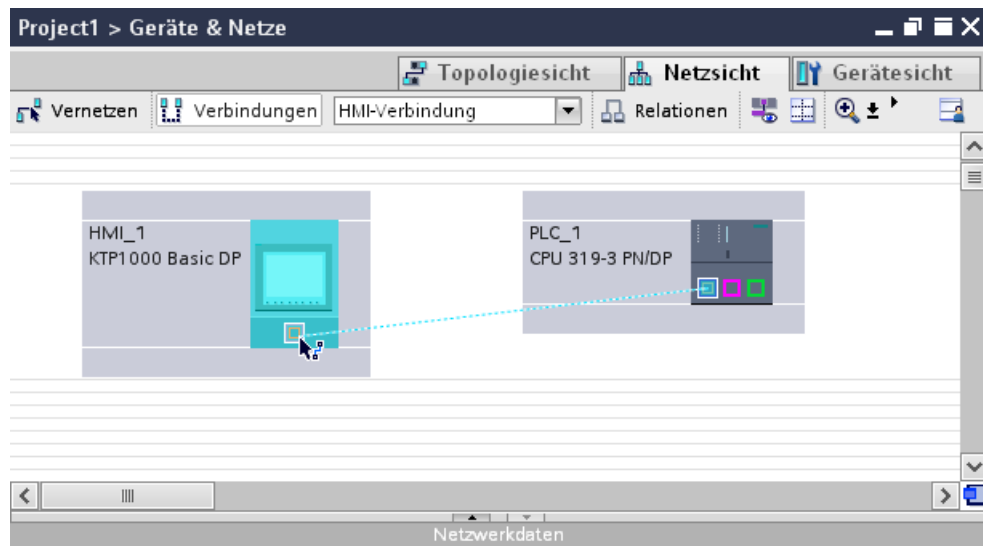
- Bediengerät mit MPI/DP-Schnittstelle
- SIMATIC S7 300/400 mit MPI/DP-Schnittstelle

## Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation den Eintrag "Geräte & Netze".  
In der Netzsicht werden die im Projekt vorhandenen Kommunikationspartner grafisch dargestellt.



2. Klicken Sie auf die Schaltfläche "Verbindungen".  
Die für eine Verbindung verfügbaren Geräte werden farbig markiert dargestellt.
3. Klicken Sie in die Schnittstelle der Steuerung und ziehen Sie per Drag&Drop eine Verbindung zum Bediengerät.



4. Klicken Sie auf die Verbindungslinie.  
Im Inspektorfenster wird die Verbindung grafisch dargestellt.

5. Klicken Sie auf "HMI-Verbindung Hervorheben" und wählen Sie die HMI-Verbindung aus.
  6. Klicken Sie auf die Kommunikationspartner in der "Netzansicht" und ändern Sie im Inspektorfenster die MPI-Parameter nach den Anforderungen ihres Projekts. Nähere Hinweise hierzu finden Sie im Kapitel "MPI-Parameter (Seite 3626)".
- 

#### **Hinweis**

Die erstellte HMI-Verbindung wird auch im tabellarischen Bereich des Editors auf der Registerkarte "Verbindungen" angezeigt. In der Tabelle kontrollieren die Verbindungsparameter, hier können Sie auch den Verbindungspartner ändern. Den lokalen Verbindungsnamen der Verbindung können Sie nur in der Tabelle ändern.

---

### **Ergebnis**

Sie haben eine HMI-Verbindung zwischen einem Bediengerät und einer SIMATIC S7 300/400 über MPI angelegt.

### **MPI-Parameter**

#### **MPI-Parameter für die HMI-Verbindung**

#### **MPI-Parameter für die HMI-Verbindung**

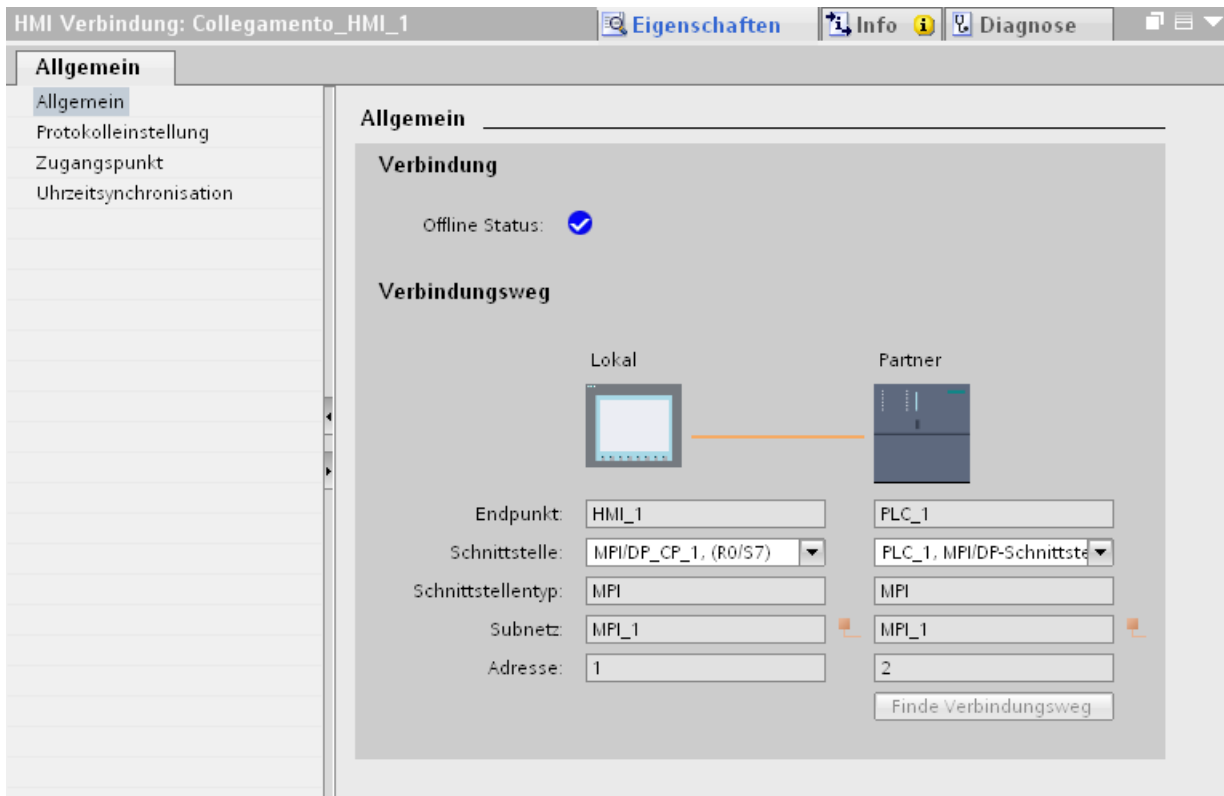
Eine Übersicht über die projizierten Parameter einer HMI-Verbindung finden Sie in den Eigenschaften der HMI-Verbindung.

Änderungen sind in diesem Inspektorfenster nur eingeschränkt möglich.



## Parameter der HMI-Verbindung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die HMI-Verbindung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der HMI-Verbindung.



### "Verbindung"

Zeigt an, ob die Geräte bereits miteinander vernetzt sind.

- Wird angezeigt, wenn die Geräte miteinander vernetzt sind.
- Wird angezeigt, wenn die Geräte nicht miteinander vernetzt sind.

### "Verbindungsweg"

Im Bereich "Verbindungsweg" werden die Kommunikationspartner der gewählten HMI-Verbindung und deren MPI-Parameter angezeigt. Einige der gezeigten Bereiche sind in diesem Dialog nicht editierbar.

- "Endpunkt"  
Zeigt den Namen des Geräts an. Dieser Bereich ist nicht editierbar.
- "Schnittstelle"  
Zeigt die gewählte Schnittstelle des Geräts an. Abhängig vom Gerät können Sie zwischen mehreren Schnittstellen wählen.

- "Schnittstellen-Typ"  
Zeigt den gewählten Schnittstellen-Typ an. Dieser Bereich ist nicht editierbar.
- "Subnetz"  
Zeigt das gewählte Subnetz an. Dieser Bereich ist nicht editierbar.
- "Adresse"  
Zeigt die MPI-Adresse des Geräts an. Dieser Bereich ist nicht editierbar.
- Schaltfläche "Finde Verbindungsweg"  
Ermöglicht das nachträgliche Spezifizieren von Verbindungen.

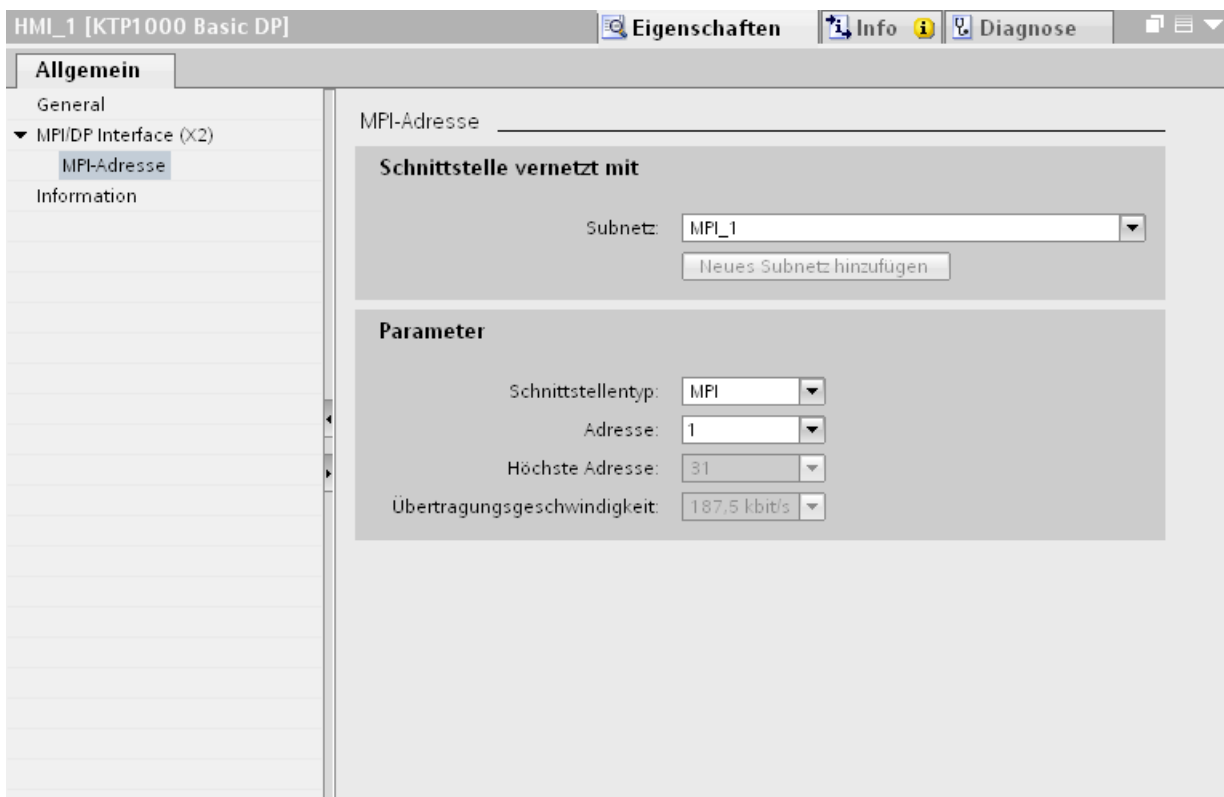
### MPI-Parameter für das Bediengerät

### MPI-Parameter für das Bediengerät

Eine Übersicht über die projektierten Parameter des Bediengeräts finden Sie in den Eigenschaften des Bediengeräts.

### MPI-Parameter des Bediengeräts anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf das Bediengerät.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter des Bediengeräts.



### **"Schnittstelle vernetzt mit"**

Im Bereich "Schnittstelle vernetzt mit" wählen Sie das Subnetz der HMI-Verbindung aus, über die das Bediengerät an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### **"Parameter"**

- "Schnittstellen-Typ"  
Im Bereich "Schnittstellentyp" parametrieren Sie den Schnittstellentyp. Abhängig vom Bediengerätentyp stehen hier verschiedenen Schnittstellen zur Auswahl.
- "Adresse"  
Im Bereich "Adresse" parametrieren Sie die MPI-Adresse des Bediengeräts. Die MPI-Adresse muss im MPI-Netzwerk eindeutig sein
- "Höchste Adresse"  
Im Bereich "Höchste Adresse" wird die höchste Adresse des MPI-Netzwerks angezeigt.
- "Übertragungsgeschwindigkeit"  
Die "Übertragungsgeschwindigkeit" wird durch das langsamste ans Netzwerk angeschlossene Gerät festgelegt. Die Einstellung ist im gesamten Netzwerk gleich.

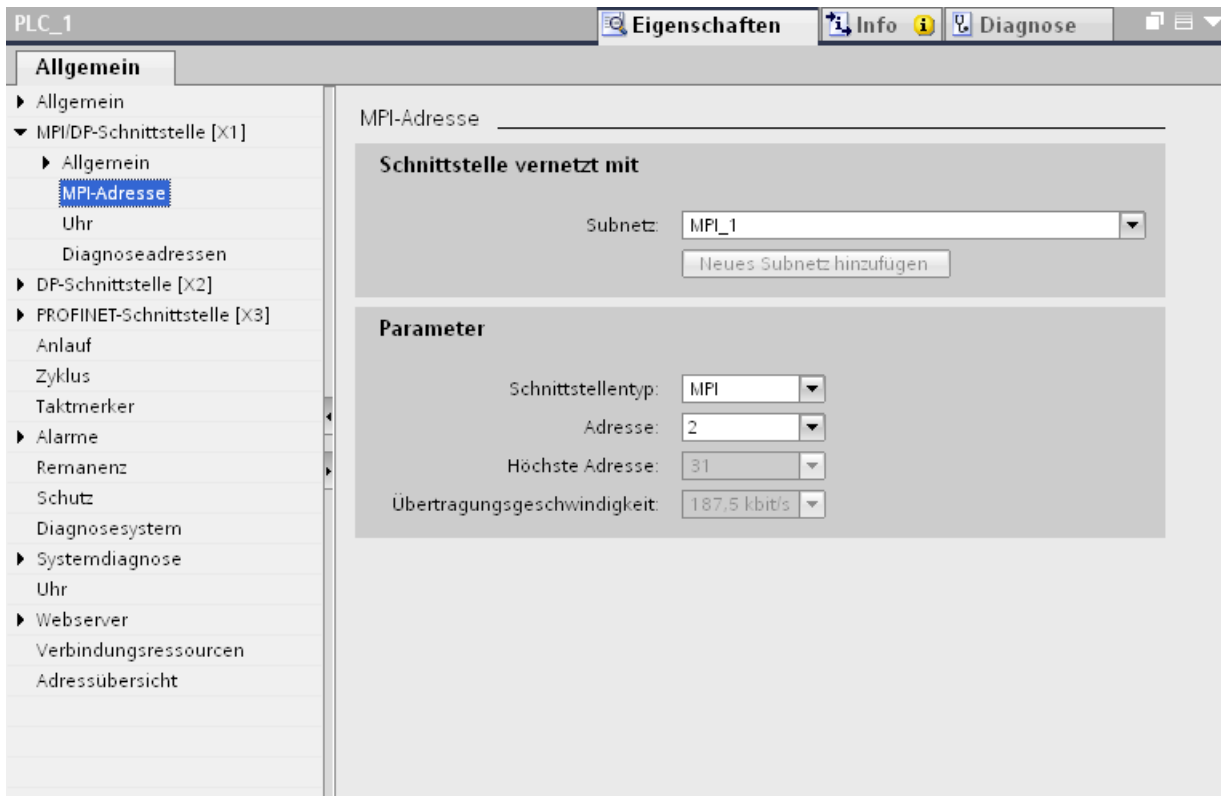
### **MPI-Parameter für die Steuerung**

#### **MPI-Parameter für Steuerung**

Eine Übersicht über die projektierten Parameter finden Sie in den Eigenschaften der Steuerung.

## PROFIBUS-Parameter der Steuerung anzeigen und ändern

1. Klicken Sie im Editor "Geräte & Netze" auf die Steuerung.
2. Ändern Sie im Inspektorfenster unter "Eigenschaften > Allgemein > Allgemein" die Parameter der Steuerung.



### "Schnittstelle vernetzt mit"

Im Bereich "Subnetz" wählen Sie das Subnetz der HMI-Verbindung aus, über die das Bediengerät an das Netzwerk angeschlossen ist. Mit der Schaltfläche "Neues Subnetz hinzufügen" legen Sie ein neues Subnetz an.

### "Parameter"

- "Schnittstellen-Typ"  
Abhängig vom Bediengerätetyp stehen verschiedene Schnittstellen zur Auswahl.
- "Adresse"  
Im Bereich "Adresse" parametrieren Sie die MPI-Adresse des Bediengeräts. Die MPI-Adresse muss im MPI-Netzwerk eindeutig sein
- "Höchste Adresse"  
Im Bereich "Höchste Adresse" wird die höchste Adresse des MPI-Netzwerks angezeigt.
- "Übertragungsgeschwindigkeit"  
Die "Übertragungsgeschwindigkeit" wird durch das langsamste ans Netzwerk angeschlossene Gerät festgelegt. Die Einstellung ist im gesamten Netzwerk gleich.

## Adressierung der Steuerung über MPI

### Einleitung

Damit die über das MPI-Netz verbundenen Kommunikationspartner miteinander kommunizieren können, wird jedem Partner eine MPI-Adresse zugewiesen.

Jede kommunikationsfähige S7-Baugruppe in der SIMATIC-S7-300/400 hat eine eindeutige MPI-Adresse. Pro Baugruppenträger darf nur eine CPU eingesetzt werden.

---

#### Hinweis

**Bediengeräte sind bei falscher Adressvergabe nicht bedienbar.**

Vermeiden Sie eine mehrfache Vergabe von Adressen am MPI-Bus.

---

### MPI-Adresse des Kommunikationspartners bei SIMATIC S7-300

Bei der Adressierung unterscheiden Sie zwischen Kommunikationspartnern mit eigener MPI-Adresse und ohne eigene MPI-Adresse.

- Bei Kommunikationspartnern mit eigener MPI-Adresse geben Sie nur die MPI-Adresse an.
- Bei Kommunikationspartnern ohne eigene MPI-Adresse geben Sie die MPI-Adresse des Kommunikationspartners an, über den die Kopplung erfolgt. Außerdem geben Sie den Steckplatz und den Baugruppenträger des Kommunikationspartners ohne MPI-Adresse an.

### MPI-Adresse des Kommunikationspartners bei SIMATIC S7-400

Nur die S7-Baugruppen, die einen MPI-Stecker haben, haben auch eine MPI-Adresse. Baugruppen, die keinen MPI-Stecker haben, werden indirekt adressiert:

- MPI-Adresse der Baugruppe, an der das Bediengerät angeschlossen ist.
- Steckplatz und Baugruppenträger, in dem die Baugruppe steckt, mit der das Bediengerät kommuniziert.

#### 10.8.8.5 Datenaustausch

### Datenaustausch über Bereichszeiger

#### Allgemeines zu Bereichszeigern

### Einleitung

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu. In diese Datenbereiche wird während der Kommunikation wechselseitig von der Steuerung und dem Bediengerät geschrieben und gelesen.

Durch die Auswertung der abgelegten Daten lösen Steuerung und Bediengerät gegenseitig fest definierte Aktionen aus.

### Projektierung von Bereichszeigern

Bevor Sie den Bereichszeiger verwenden, aktivieren Sie die Bereichszeiger unter "Verbindungen > Bereichszeiger". Danach parametrieren Sie die Bereichszeiger.

Nähere Hinweise zum Projektieren von Bereichszeigern finden Sie unter:

Bereichszeiger projektieren (Seite 3491)

### Bereichszeiger "Bildnummer"

#### Funktion

Die Bediengeräte legen im Bereichszeiger "Bildnummer" Informationen über das am Bediengerät aufgerufene Bild ab.

Dadurch ist es möglich, Informationen zum aktuellen Bildinhalt des Bediengeräts zur Steuerung zu übertragen. In der Steuerung können bestimmte Reaktionen getriggert werden, z. B. den Aufruf eines anderen Bildes.

#### Verwendung

Bevor der Bereichszeiger "Bildnummer" verwendet werden kann, muss dieser unter "Kommunikation > Verbindungen" eingerichtet und aktiviert werden. Der Bereichszeiger "Bildnummer" kann nur in **einer** Steuerung und in dieser Steuerung nur **einmal** angelegt werden.

Die Bildnummer wird immer zur Steuerung übertragen, wenn ein neues Bild aktiviert wird oder der Fokus innerhalb eines Bildes von einem Bildobjekt zu einem anderen wechselt.

#### Aufbau

Der Bereichszeiger ist ein Datenbereich im Speicher der Steuerung mit einer festen Länge von 5 Worten.

|        | 15                  | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1.Wort | Aktueller Bildtyp   |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 2.Wort | Aktuelle Bildnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 3.Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 4.Wort | Aktuelle Feldnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 5.Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

- Aktueller Bildtyp  
"1" für Grundbild oder  
"4" für Permanentfenster
- Aktuelle Bildnummer  
1 bis 32767
- Aktuelle Feldnummer  
1 bis 32767

## Bereichszeiger "Datum/Uhrzeit"

### Funktion

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von dem Bediengerät zur Steuerung verwendet.

Die Steuerung schreibt den Steuerauftrag "41" oder "40" in das Auftragsfach.

Mit der Auswertung des Steuerauftrags schreibt das Bediengerät sein aktuelles Datum und die Uhrzeit in den im Bereichszeiger "Datum/Uhrzeit" projektierten Datenbereich. Alle Angaben sind BCD-codiert.

Wenn in einem Projekt mehrere Verbindungen projektiert sind, und in einer der Verbindungen der Bereichszeiger "Datum / Uhrzeit" verwendet werden soll, so muss der Bereichszeiger für jede projektierte Verbindung aktiviert werden.

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" projektiert haben, dann können Sie den Bereichszeiger "Datum/Uhrzeit PLC" nicht verwenden.

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" verwenden, dann ist eine symbolische Adressierung nicht möglich.

Wenn in einem Projekt mehrere Verbindungen projektiert sind, und in einer der Verbindungen der Bereichszeiger "Datum / Uhrzeit" verwendet werden soll, so muss der Bereichszeiger für jede projektierte Verbindung aktiviert werden.

Der Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

| Datenwort | Höherwertiges Byte |  |  |  |  |  |   | Niederwertiges Byte   |  |  |  |  |  |   |         |
|-----------|--------------------|--|--|--|--|--|---|-----------------------|--|--|--|--|--|---|---------|
|           | 7                  |  |  |  |  |  | 0 | 7                     |  |  |  |  |  | 0 |         |
| n+0       | Reserviert         |  |  |  |  |  |   | Stunde (0-23)         |  |  |  |  |  |   | Uhrzeit |
| n+1       | Minute (0-59)      |  |  |  |  |  |   | Sekunde (0-59)        |  |  |  |  |  |   |         |
| n+2       | Reserviert         |  |  |  |  |  |   | Reserviert            |  |  |  |  |  |   |         |
| n+3       | Reserviert         |  |  |  |  |  |   | Wochentag (1-7, 1=So) |  |  |  |  |  |   | Datum   |
| n+4       | Tag (1-31)         |  |  |  |  |  |   | Monat (1-12)          |  |  |  |  |  |   |         |
| n+5       | Jahr (80-99/0-29)  |  |  |  |  |  |   | Reserviert            |  |  |  |  |  |   |         |

**Hinweis**

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

**Bereichszeiger "Datum/Uhrzeit PLC"**

**Funktion**

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von der Steuerung zum Bediengerät verwendet. Sie setzen diesen Bereichszeiger ein, wenn die Steuerung Master für die Zeit ist.

Die Steuerung lädt den Datenbereich des Bereichszeigers. Alle Angaben sind BCD-codiert.

Das Bediengerät liest zyklisch die Daten über den projektierten Erfassungszyklus und synchronisiert sich.

**Hinweis**

Wählen Sie in der Projektierung den Erfassungszyklus für den Bereichszeiger Datum/Uhrzeit nicht zu klein, da dies die Performance des Bediengeräts beeinflusst.  
 Empfehlung: Erfassungszyklus 1 Minute, wenn dies Ihr Prozess erlaubt.

Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" projektiert haben, können Sie den Bereichszeiger "Datum/Uhrzeit" nicht verwenden.

"Datum/Uhrzeit PLC" ist ein globaler Bereichszeiger und kann in einem Projekt nur ein Mal projektiert werden.

**Hinweis**

Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" projektiert haben, können Sie den Bereichszeiger "Datum/Uhrzeit" nicht verwenden.

Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

**Format DATE\_AND\_TIME (BCD-codiert)**

| Datenwort | Höherwertiges Byte |       |   | Niederwertiges Byte |                       |   |
|-----------|--------------------|-------|---|---------------------|-----------------------|---|
|           | 7                  | ..... | 0 | 7                   | .....                 | 0 |
| n+0       | Jahr (80-99/0-29)  |       |   | Monat (1-12)        |                       |   |
| n+1       | Tag (1-31)         |       |   | Stunde (0-23)       |                       |   |
| n+2       | Minute (0-59)      |       |   | Sekunde (0-59)      |                       |   |
| n+3       | Reserviert         |       |   | Reserviert          | Wochentag (1-7, 1=So) |   |



| Datenwort         | Höherwertiges Byte |       |   | Niederwertiges Byte |       |   |
|-------------------|--------------------|-------|---|---------------------|-------|---|
|                   | 7                  | ..... | 0 | 7                   | ..... | 0 |
| n+4 <sup>1)</sup> | Reserviert         |       |   | Reserviert          |       |   |
| n+5 <sup>1)</sup> | Reserviert         |       |   | Reserviert          |       |   |

- 1) Die beiden Datenwörter müssen im Datenbereich vorhanden sein, um eine Übereinstimmung des Datenformats mit WinCC flexible sicherzustellen und das Lesen falscher Informationen zu vermeiden.

---

#### Hinweis

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

---

## Bereichszeiger "Koordinierung"

### Funktion

Der Bereichszeiger "Koordinierung" dient zur Realisierung der folgenden Funktionen:

- Anlauf des Bediengeräts im Steuerungsprogramm erkennen
- Aktuelle Betriebsart des Bediengeräts im Steuerungsprogramm erkennen
- Kommunikationsbereitschaft des Bediengeräts im Steuerungsprogramm erkennen

Der Bereichszeiger "Koordinierung" standardmäßig hat eine Länge von einem Wort und kann nicht verändert werden.

### Verwendung

---

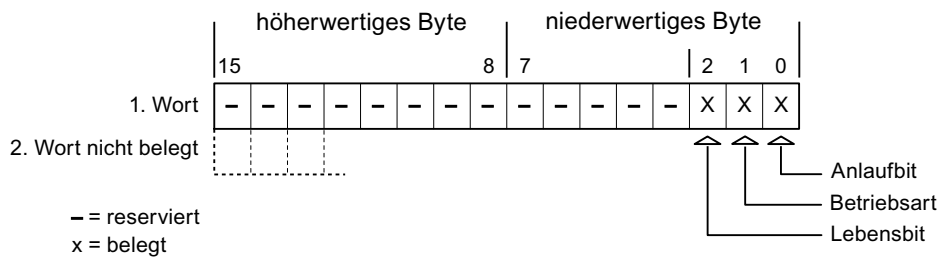
#### Hinweis

Bei jeder Aktualisierung des Bereichszeigers durch das Bediengerät wird immer der komplette Koordinierungsbereich geschrieben.

Das SPS-Programm darf deshalb im Koordinierungsbereich keine Änderungen vornehmen.

---

### Belegung der Bits im Bereichszeiger "Koordinierung"



#### Anlaufbit

Das Anlaufbit wird durch das Bediengerät während des Anlaufvorgangs kurzfristig auf "0" gesetzt. Nach dem Anlaufvorgang steht das Bit dauerhaft auf "1".

#### Betriebsart

Sobald das Bediengerät durch den Benutzer offline geschaltet wird, wird das Betriebsartenbit auf 1 gesetzt. Im Normalbetrieb des Bediengeräts ist der Zustand des Betriebsartenbits "0". Im Steuerungsprogramm können Sie durch Abfrage dieses Bits die aktuelle Betriebsart des Bediengeräts ermitteln.

#### Lebensbit

Das Lebensbit wird durch das Bediengerät im zeitlichen Abstand von ca. einer Sekunde invertiert. Im Steuerungsprogramm können Sie durch Abfrage dieses Bits überprüfen, ob die Verbindung zum Bediengerät noch besteht.

### Bereichszeiger "Projektkennung"

#### Funktion

Beim Start der Runtime kann überprüft werden, ob das Bediengerät an die richtige Steuerung angeschlossen ist. Diese Überprüfung ist beim Einsatz mehrerer Bediengeräte wichtig.

Dazu vergleicht das Bediengerät einen in der Steuerung hinterlegten Wert mit dem in der Projektierung angegebenen Wert. Damit wird die Kompatibilität der Projektierungsdaten mit dem Steuerungsprogramm sichergestellt. Eine fehlende Übereinstimmung führt zur Anzeige einer Systemmeldung am Bediengerät und zum Stopp der Runtime.

## Verwendung

Um diesen Bereichszeiger zu verwenden, richten Sie bei der Projektierung Folgendes ein:

- Angabe der Version, welche die Projektierung hat. Möglicher Wert zwischen 1 und 255. Sie geben die Version ein im Editor "Runtime-Einstellungen > Allgemein" im Bereich "Identifizierung".
- Datenadresse des Werts für die Version, die in der Steuerung hinterlegt ist: Sie geben die Datenadresse ein im Editor "Kommunikation > Verbindungen" unter "Adresse".

## Ausfall einer Verbindung

Ein Verbindungsausfall zu einem Gerät, auf dem der Bereichszeiger "Projektkennung" projektiert ist, hat zur Folge, dass auch alle anderen Verbindungen des Geräts "Offline" geschaltet werden.

Dieses Verhalten hat folgende Voraussetzungen:

- Sie haben in einem Projekt mehrere Verbindungen projektiert.
- Sie verwenden in mindestens einer Verbindung den Bereichszeiger "Projektkennung".

Folgende Ursachen können Verbindungen in den Zustand "Offline" setzen:

- Die Steuerung ist nicht erreichbar.
- Die Verbindung wurde im Engineering System offline geschaltet.

## Bereichszeiger "Steuerungsauftrag"

### Funktion

Über das Steuerungsauftrags-Fach können dem Bediengerät Steuerungsaufträge gegeben und damit Aktionen am Bediengerät getriggert werden. Zu diesen Funktionen gehören z. B.:

- Bild anzeigen
- Datum und Uhrzeit stellen

### Datenstruktur

Im ersten Wort des Steuerungsauftrags-Fachs steht die Auftragsnummer. Je nach Steuerungsauftrag können dann bis zu drei Parameter übergeben werden.

| Wort | Höherwertiges Byte | Niederwertiges Byte |
|------|--------------------|---------------------|
| n+0  | 0                  | Auftragsnummer      |
| n+1  |                    | Parameter 1         |
| n+2  |                    | Parameter 2         |
| n+3  |                    | Parameter 3         |

Wenn das erste Wort des Steuerungsauftrags-Fachs ungleich 0 ist, wertet das Bediengerät den Steuerungsauftrag aus. Aus diesem Grund müssen zuerst die Parameter in das Steuerungsauftrags-Fach eingetragen werden und dann erst die Auftragsnummer.

Wenn das Bediengerät den Steuerungsauftrag angenommen hat, wird das erste Wort wieder auf 0 gesetzt. Die Ausführung des Steuerungsauftrags ist zu diesem Zeitpunkt im Allgemeinen noch nicht abgeschlossen.

### Steuerungsaufträge

Nachfolgend sind alle Steuerungsaufträge und deren Parameter aufgelistet. Die Spalte "Nr." enthält die Auftragsnummer des Steuerungsauftrags. Generell können Steuerungsaufträge nur dann von der Steuerung getriggert werden, wenn das Bediengerät im Online-Betrieb ist.

| Nr.       | Funktion   |   |
|-----------|--|---|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>   |   |
|           | Parameter 1  | Linkes Byte: -<br>Rechtes Byte: Stunden (0-23)                      |
|           | Parameter 2  | Linkes Byte: Minuten (0-59)<br>Rechtes Byte: Sekunden (0-59)        |
|           | Parameter 3  | -   |
| <b>15</b> | <b>Datum stellen (BCD-codiert) <sup>3) 4)</sup></b>  |   |
|           | Parameter 1  | Linkes Byte: -<br>Rechtes Byte: Wochentag<br>(1-7: Sonntag-Samstag) |
|           | Parameter 2  | Linkes Byte: Tag (1-31)<br>Rechtes Byte: Monat (1-12)               |
|           | Parameter 3  | Linkes Byte: Jahr   |
| <b>23</b> | <b>Benutzer anmelden</b>   |   |
|           | Meldet den Benutzer "PLC User" mit der im Parameter 1 übergebenen Gruppennummer am Bediengerät an.<br>Voraussetzung für die Anmeldung ist, dass die übergebene Gruppennummer im Projekt vorhanden ist. |   |
|           | Parameter 1  | Gruppennummer 1 - 255   |
|           | Parameter 2, 3   | -   |
| <b>24</b> | <b>Benutzer abmelden</b>   |   |
|           | Meldet den aktuell angemeldeten Benutzer ab.<br>(Funktion entspricht der Systemfunktion "Abmelden")  |   |
|           | Parameter 1, 2, 3  | -   |
| <b>40</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |   |
|           | (Im S7-Format DATE_AND_TIME)<br>Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.   |   |
|           | Parameter 1, 2, 3  | -   |
| <b>41</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |   |
|           | (Im Format OP/MP)<br>Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.  |   |

| Nr.       | Funktion  |  |
|-----------|---|--|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>  |  |
|           | Parameter 1, 2, 3   | -  |
| <b>46</b> | <b>Variable aktualisieren</b>   |  |
|           | Veranlasst das Bediengerät den aktuellen Wert der Variablen aus der Steuerung zu lesen, deren Aktualisierungskennung mit dem im Parameter 1 übergebenen Wert übereinstimmt. (Funktion entspricht der Systemfunktion "AktualisiereVariable") |  |
|           | Parameter 1   | 1 - 100  |
| <b>49</b> | <b>Meldepuffer löschen</b>  |  |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Warnings" aus dem meldepuffer.   |  |
|           | Parameter 1, 2, 3   | -  |
| <b>50</b> | <b>Meldepuffer löschen</b>  |  |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Errors" aus dem Meldepuffer  |  |
|           | Parameter 1, 2, 3   | -  |
| <b>51</b> | <b>Bildanwahl <sup>2)</sup></b>   |  |
|           | Parameter 1   | Bildnummer   |
|           | Parameter 2   | -  |
|           | Parameter 3   | Feldnummer   |
| <b>69</b> | <b>Datensatz aus Steuerung lesen <sup>1)</sup></b>  |  |
|           | Parameter 1   | Rezeptnummer (1-999)   |
|           | Parameter 2   | Datensatznummer (1-65535)  |
|           | Parameter 3   | 0: Vorhandenen Datensatz nicht überschreiben<br>1: Vorhandenen Datensatz überschreiben |
| <b>70</b> | <b>Datensatz in Steuerung schreiben <sup>1)</sup></b>   |  |
|           | Parameter 1   | Rezeptnummer (1-999)   |
|           | Parameter 2   | Datensatznummer (1-65535)  |
|           | Parameter 3   | -  |

|    |   |
|----|---|
| 1) | Nur bei Geräten, die Rezepturen unterstützen.   |
| 2) | Auf den Bediengeräten OP 73, OP 77A und TP 177A wird der Steuerungsauftrag "Bildanwahl" auch dann ausgeführt, wenn die Bildschirmtastatur geöffnet ist. |
| 3) | Beim Bediengerät KTP 600 BASIC PN wird der Wochentag ignoriert.   |
| 4) | Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" verwenden, wird der Wochentag ignoriert.  |

## Bereichszeiger "Datensatz"

## Bereichszeiger "Datensatz"

### Funktion

Bei der Übertragung von Datensätzen zwischen Bediengerät und Steuerung greifen beide Kommunikationspartner wechselseitig auf gemeinsame Kommunikationsbereiche in der Steuerung zu.

### Übertragungsarten

Für die Übertragung von Datensätzen zwischen Bediengerät und Steuerung gibt es zwei Möglichkeiten:

- Übertragung ohne Synchronisation
- Übertragung mit Synchronisation über das Datenfach

Datensätze werden immer direkt übertragen. D. h., die Variablenwerte werden direkt, ohne den Umweg über eine Zwischenablage, aus der Adresse gelesen oder in die Adresse geschrieben, die für die Variable projektiert ist.

### Übertragung von Datensätzen anstoßen

Für den Anstoß der Übertragung gibt es drei Möglichkeiten:

- Bedienung in der Rezepturanzeige
- Steuerungsaufträge  
Die Übertragung der Datensätze kann auch durch die Steuerung getriggert werden.
- Auslösen projektierte Funktionen

Wenn die Übertragung von Datensätzen durch eine projektierte Funktion oder einen Steuerungsauftrag getriggert wird, ist die Rezepturanzeige am Bediengerät weiterhin ungehindert bedienbar. Die Datensätze werden im Hintergrund übertragen.

Das gleichzeitige Abarbeiten mehrerer Übertragungsanforderungen ist jedoch nicht möglich. In diesem Fall lehnt das Bediengerät eine weitere Übertragung mit einer Systemmeldung ab.

### Übertragung ohne Synchronisation

Bei der asynchronen Übertragung von Datensätzen zwischen Bediengerät und Steuerung findet keine Koordination über gemeinsam benutzte Datenbereiche statt. Die Einrichtung eines Datenbereichs beim Projektieren ist daher nicht nötig.

Die asynchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Ein unkontrolliertes Überschreiben der Daten durch die Kommunikationspartner kann systembedingt ausgeschlossen werden.
- Die Steuerung braucht keine Informationen über die Rezepturnummer und die Datensatznummer.
- Die Übertragung von Datensätzen wird durch Bedienung am Bediengerät getriggert.

### **Werte lesen**

Beim Anstoß der Übertragung zum Lesen werden die Werte aus den Steuerungsadressen gelesen und zum Bediengerät übertragen.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die Werte werden in das Bediengerät geladen. Im Bediengerät können Sie diese weiterverarbeiten, z. B. Werte ändern, speichern etc.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte werden sofort auf dem Datenträger gespeichert.

### **Werte schreiben**

Beim Anstoß der Übertragung zum Schreiben werden die Werte in die Steuerungsadressen geschrieben.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die aktuellen Werte werden in die Steuerung geschrieben.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte vom Datenträger werden in die Steuerung geschrieben.

### **Übertragung mit Synchronisation**

Bei der synchronen Übertragung setzen beide Kommunikationspartner Status-Bits im gemeinsam benutzten Datenbereich. Dadurch können Sie in Ihrem Steuerungsprogramm ein unkontrolliertes gegenseitiges Überschreiben der Daten verhindern.

### **Anwendung**

Die synchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Die Steuerung ist der "aktive Partner" bei der Übertragung von Datensätzen.
- In der Steuerung werden Informationen über die Rezepturnummer und die Datensatznummer ausgewertet.
- Die Übertragung von Datensätzen wird per Steuerungsauftrag getriggert.

### Voraussetzungen

Damit Datensätze synchronisiert zwischen Bediengerät und Steuerung übertragen werden, müssen beim Projektieren folgende Voraussetzungen erfüllt sein:

- Ein Bereichszeiger ist eingerichtet: Editor "Kommunikation > Verbindungen" unter "Bereichszeiger".
- In der Rezeptur ist die Steuerung angegeben, mit der das Bediengerät die Übertragung der Datensätze synchronisiert:  
Editor "Rezepturen" im Inspektorfenster unter "Allgemein > Synchronisation > Einstellungen" die Auswahl "Koordinierte Übertragung der Datensätze"

### Aufbau des Datenbereichs

Der Datenbereich hat eine feste Länge von 5 Worten. Der Datenbereich ist wie folgt aufgebaut:

|         |                                       |  |          |
|---------|---------------------------------------|--|----------|
|         | <b>15</b>                             |  | <b>0</b> |
| 1. Wort | Aktuelle Rezepturnummer (1 - 999)     |  |          |
| 2. Wort | Aktuelle Datensatznummer (0 - 65.535) |  |          |
| 3. Wort | Reserviert                            |  |          |
| 4. Wort | Status (0, 2, 4, 12)                  |  |          |
| 5. Wort | Reserviert                            |  |          |

- Status  
Das Statuswort (Wort 4) kann folgende Werte annehmen:

| Wert    |           | Bedeutung                            |
|---------|-----------|--------------------------------------|
| Dezimal | Binär     |                                      |
| 0       | 0000 0000 | Übertragung zulässig, Datenfach frei |
| 2       | 0000 0010 | Übertragung läuft.                   |
| 4       | 0000 0100 | Übertragung fehlerfrei beendet       |
| 12      | 0000 1100 | Übertragung mit Fehler beendet       |

### Ablauf der Übertragung bei Bedienung in der Rezepturanzeige

#### Lesen aus der Steuerung durch Bedienung in der Rezepturanzeige

| Schritt | Aktion   |                            |
|---------|--|----------------------------|
| 1       | Prüfung: Statuswort = 0?   |                            |
|         | Ja   | Nein                       |
| 2       | Das Bediengerät trägt die zu lesende Rezepturnummer sowie den Status "Übertragung läuft" im Datenfach ein und setzt die Datensatznummer auf 0. | Abbruch mit Systemmeldung. |



| Schritt | Aktion  |
|---------|---|
| 3       | Das Bediengerät liest die Werte aus der Steuerung und zeigt diese in der Rezepturanzeige an.<br>Bei Rezepturen mit synchronisierten Variablen werden die Werte aus der Steuerung auch in die Variablen geschrieben. |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".   |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |

### Schreiben in die Steuerung durch Bedienung in der Rezepturanzeige

| Schritt   | Aktion   |    |      |   |                            |
|---|--|----|------|---|----------------------------|
|   | Prüfung: Statuswort = 0?   |    |      |   |                            |
| 1   | <table border="1"> <thead> <tr> <th>Ja</th> <th>Nein</th> </tr> </thead> <tbody> <tr> <td>Das Bediengerät trägt die zu schreibende Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.</td> <td>Abbruch mit Systemmeldung.</td> </tr> </tbody> </table> | Ja | Nein | Das Bediengerät trägt die zu schreibende Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein. | Abbruch mit Systemmeldung. |
| Ja  | Nein   |    |      |   |                            |
| Das Bediengerät trägt die zu schreibende Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein. | Abbruch mit Systemmeldung.   |    |      |   |                            |
| 2   | Das Bediengerät schreibt die aktuellen Werte in die Steuerung.<br>Bei Rezepturen mit synchronisierten Variablen werden die geänderten Werte zwischen Rezepturanzeige und Variablen abgeglichen und dann in die Steuerung geschrieben.  |    |      |   |                            |
| 3   | Das Bediengerät setzt den Status "Übertragung beendet".  |    |      |   |                            |
| 4   | Das Steuerungsprogramm kann jetzt ggf. die übertragenen Daten auswerten.   |    |      |   |                            |
| 5   | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.  |    |      |   |                            |

#### Hinweis

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

#### Hinweis

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach ist auf "Übertragung beendet" gesetzt.
- Der Status im Datenfach ist auf "Übertragung mit Fehler beendet" gesetzt.

### Ablauf der Übertragung bei Steuerungsauftrag

Die Übertragung von Datensätzen zwischen Bediengerät und Steuerung kann vom Bediengerät oder von der Steuerung initiiert werden.

Für diese Art der Übertragung stehen die beiden Steuerungsaufträge Nr. 69 und Nr. 70 zur Verfügung.

**Nr. 69: Datensatz aus Steuerung lesen ("SPS → DAT")**

Der Steuerungsauftrag Nr. 69 überträgt Datensätze von der Steuerung zum Bediengerät. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Linkes Byte (LB)   | Rechtes Byte (RB) |
|--------|--|-------------------|
| Wort 1 | 0  | 69                |
| Wort 2 | Rezeptnummer (1-999)   |                   |
| Wort 3 | Datensatznummer (1-65.535)   |                   |
| Wort 4 | Vorhandenen Datensatz nicht überschreiben: 0<br>Vorhandenen Datensatz überschreiben: 1 |                   |

**Nr. 70: Datensatz in Steuerung schreiben ("DAT → SPS")**

Der Steuerungsauftrag Nr. 70 überträgt Datensätze vom Bediengerät zur Steuerung. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Linkes Byte (LB)           | Rechtes Byte (RB) |
|--------|----------------------------|-------------------|
| Wort 1 | 0                          | 70                |
| Wort 2 | Rezeptnummer (1-999)       |                   |
| Wort 3 | Datensatznummer (1-65.535) |                   |
| Wort 4 | —                          |                   |

**Ablauf bei Lesen aus der Steuerung durch Steuerungsauftrag "SPS → DAT" (Nr. 69)**

| Schritt | Aktion  |                           |
|---------|---|---------------------------|
| 1       | Prüfung: Statuswort = 0?  |                           |
|         | Ja  | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese Werte in dem Datensatz, der im Steuerungsauftrag angegeben ist.   |                           |
| 4       | <ul style="list-style-type: none"> <li>Falls im Auftrag "Überschreiben" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben. Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>Falls im Auftrag "Nicht überschreiben" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                           |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                           |

**Ablauf bei Schreiben in die Steuerung durch Steuerungsauftrag "DAT → SPS" (Nr. 70)**

| Schritt | Aktion   |                           |
|---------|--|---------------------------|
| 1       | Prüfung: Statuswort = 0?   |                           |
|         | Ja   | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät holt die Werte des im Auftrag angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.   |                           |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                           |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                           |

**Ablauf der Übertragung bei Anstoß durch projizierte Funktion**

**Lesen aus der Steuerung durch projizierte Funktion**

| Schritt | Aktion  |                            |
|---------|---|----------------------------|
| 1       | Prüfung: Statuswort = 0?  |                            |
|         | Ja  | Nein                       |
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese in dem Datensatz ab, der in der Funktion angegeben ist.   |                            |
| 4       | <ul style="list-style-type: none"> <li>Falls in der Funktion bei "Überschreiben" "Ja" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben.<br/>Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>Falls in der Funktion bei "Überschreiben" "Nein" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                            |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                            |

**Schreiben in die Steuerung durch projizierte Funktion**

| Schritt | Aktion                   |      |
|---------|--------------------------|------|
| 1       | Prüfung: Statuswort = 0? |      |
|         | Ja                       | Nein |

| Schritt | Aktion   |                            |
|---------|--|----------------------------|
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät holt die Werte des in der Funktion angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.  |                            |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                            |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                            |

### Mögliche Fehlerursachen bei der Übertragung von Datensätzen

#### Mögliche Fehlerursachen

Falls die Übertragung von Datensätzen mit Fehler beendet wird, so kann dies u. a. folgende Ursachen haben:

- Variablen-Adresse in der Steuerung nicht eingerichtet
- Überschreiben von Datensätzen nicht möglich
- Rezepturnummer nicht vorhanden
- Datensatznummer nicht vorhanden

---

#### Hinweis

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

---

#### Hinweis

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach wurde auf "Übertragung beendet" gesetzt.
  - Der Status im Datenfach wurde auf "Übertragung mit Fehler beendet" gesetzt.
- 

### Reaktion auf fehlerbedingten Abbruch

Das Bediengerät reagiert auf einen fehlerbedingten Abbruch der Übertragung von Datensätzen wie folgt:

- Anstoß über Bedienung in der Rezepturanzeige  
Hinweise in der Statuszeile der Rezepturanzeige und Ausgabe von Systemmeldungen
- Anstoß über Funktion  
Ausgabe von Systemmeldungen
- Anstoß über Steuerungsauftrag  
Keine Rückmeldung am Bediengerät

Unabhängig davon können Sie den Status der Übertragung durch Abfragen des Statusworts im Datenfach auswerten.

## Kurven

## Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt bei den Basic Panels zeitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 3201)

## Zeitgetriggerte Kurven

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein.

Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

## Meldungen

## Meldungen projektieren

## Meldungen projektieren

Um Meldungen wie Betriebs-, Störmeldungen und Quittierung zu projektieren sind mehrere Schritte notwendig.

- Schritt 1: Variablen anlegen
- Schritt 2: Meldungen projektieren
- Schritt 3: Quittierung projektieren

Nähere Hinweise hierzu finden Sie im Kapitel:

Arbeiten mit Meldungen (Seite 3213)

## Besonderheiten bei der Projektierung von Meldungen

Wenn Sie Verbindungen von Bediengeräten mit Steuerungen anderer Hersteller projektieren, beachten Sie folgende Besonderheiten bei der Projektierung:

- Datentypen der Variablen
- Adressierung der Variablen
- Zählweise der Bitstellen

## Datentypen

Für Verbindungen mit SIMATIC Kommunikationstreiber, werden folgende Datentypen unterstützt:

| Steuerung             | Zulässige Datentypen |  |
|-----------------------|----------------------|--|
|                       | Bitmeldungen         | Analogmeldungen  |
| SIMATIC S7<br>300/400 | WORD, INT            | BYTE, CHAR, WORD, INT, DWORD, DINT, REAL,<br>COUNTER, TIME |

## Zählweise der Bitstellen

Für Verbindungen mit SIMATIC Kommunikationstreiber, gilt folgende Zählweise:

| Zählweise der Bitstellen      | Byte 0             |  |  |  |  |  |   | Byte 1              |  |  |  |  |  |   |
|-------------------------------|--------------------|--|--|--|--|--|---|---------------------|--|--|--|--|--|---|
|                               | Höherwertiges Byte |  |  |  |  |  |   | Niederwertiges Byte |  |  |  |  |  |   |
| In SIMATIC S7-<br>Steuerungen | 7                  |  |  |  |  |  | 0 | 7                   |  |  |  |  |  | 0 |
| Im WinCC projektieren<br>Sie: | 15                 |  |  |  |  |  | 8 | 7                   |  |  |  |  |  | 0 |

## Quittierung von Meldungen

### Vorgehensweise

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

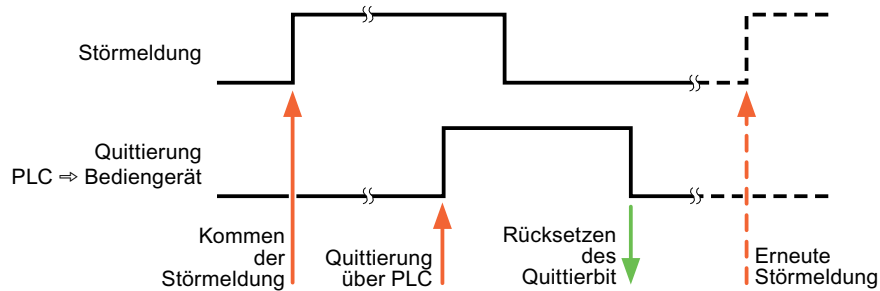
### Quittieren von der Steuerung

Unter "PLC-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulsdiagramm.



### Quittierung am Bediengerät

Unter "HMI-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittvariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittvariable in die Steuerung eine gewisse Zeitspanne.

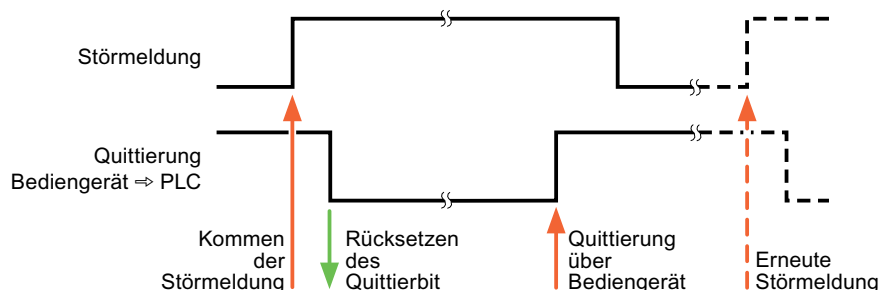
Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittvariablen gesetzt. Die gesamte Quittvariable wird dann vom Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

#### Hinweis

Alle seit dem letzten Runtime-Start quittierten Meldebits bleiben in der Quittvariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittvariablen der gesamte Bereich vom Bediengerät überschrieben wird.

Das nachfolgende Bild zeigt das Impulssdiagramm.



### 10.8.8.6 Leistungsmerkmale der Kommunikation

#### Zulässige Datentypen für SIMATIC S7 300/400

#### Zulässige Datentypen für Verbindungen mit SIMATIC S7 300/400

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp         | Länge                      |
|------------------|----------------------------|
| BOOL             | 1 Bit                      |
| BYTE             | 1 Byte                     |
| WORD             | 2 Byte                     |
| DWORD            | 4 Byte                     |
| CHAR             | 1 Byte                     |
| INT              | 2 Byte                     |
| DINT             | 4 Byte                     |
| REAL             | 4 Byte                     |
| TIME             | 4 Byte                     |
| DATE             | 2 Byte                     |
| TIME_OF_DAY, TOD | 4 Byte                     |
| S5TIME           | 2 Byte                     |
| COUNTER          | 2 Byte                     |
| TIMER            | 2 Byte                     |
| DATE_AND_TIME    | 8 Byte                     |
| STRING           | (2+n) Bytes, n = 0 bis 254 |

### 10.8.8.7 Verbindungen im Editor "Verbindungen" anlegen

#### PROFINET-Verbindung anlegen

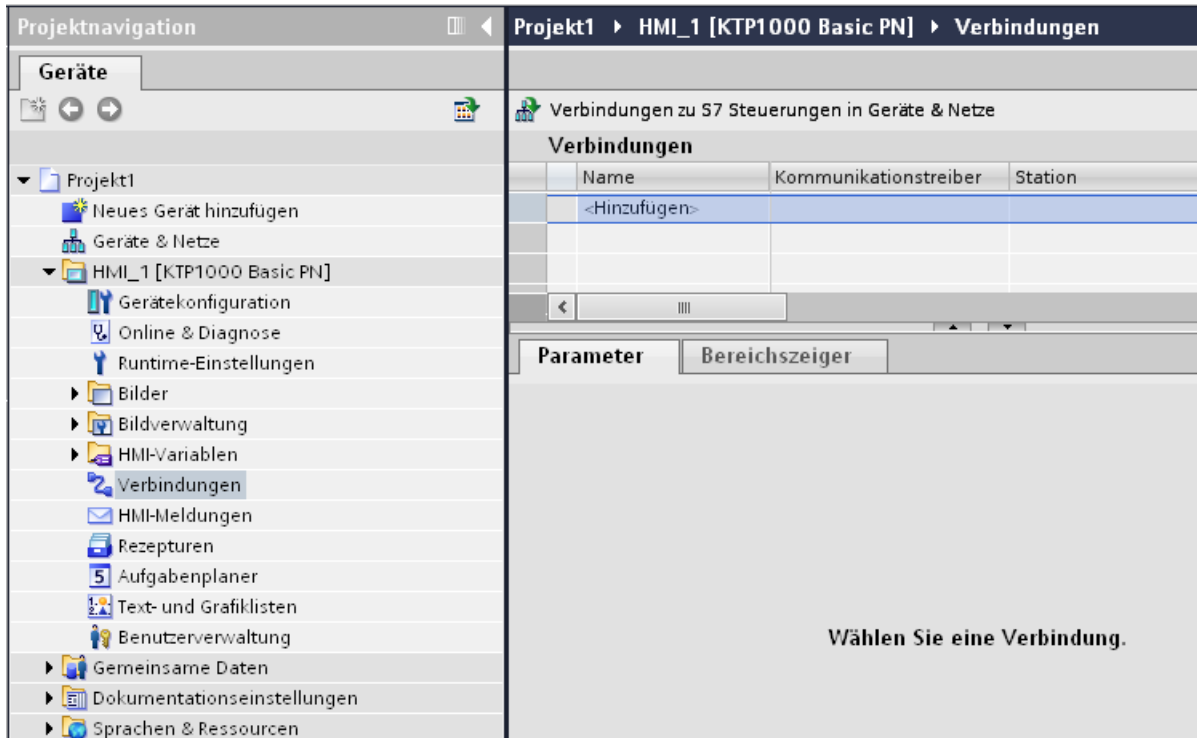
##### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät mit PROFINET-Schnittstelle ist angelegt.

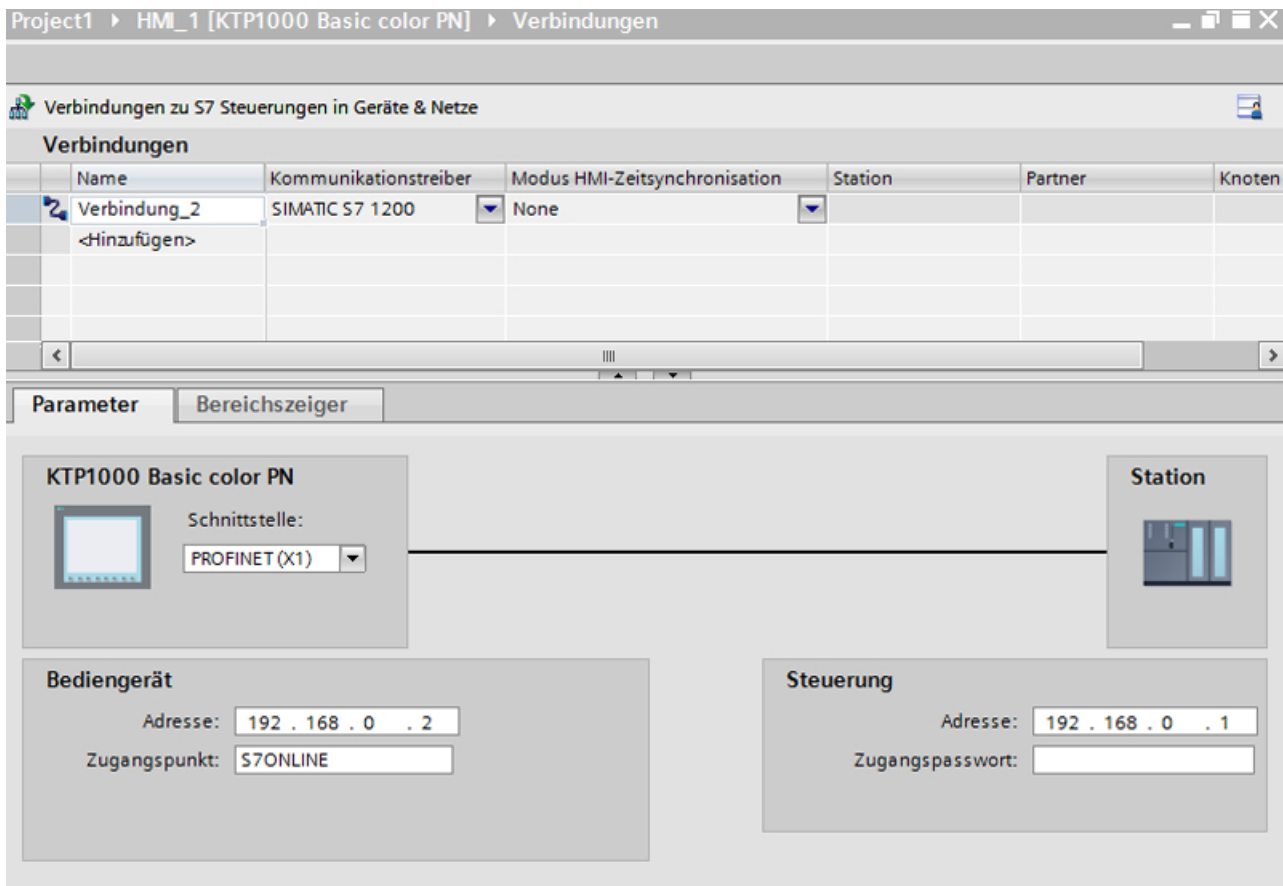


## Vorgehensweise

1. Öffnen Sie den Editor "Verbindungen" des Bediengeräts.
2. Doppelklicken Sie auf "<Hinzufügen>".



3. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber aus.



4. Klicken Sie auf den Namen der Verbindung.
5. Wählen Sie im Inspektorfenster unter "Parameter > Schnittstelle" eine PROFINET-Schnittstelle des Bediengeräts aus.
6. Stellen Sie im Inspektorfenster die IP-Adressen der Kommunikationspartner ein:
  - Bediengerät: "Parameter > Bediengerät > Adresse"
  - Steuerung: "Parameter > Steuerung > Adresse"

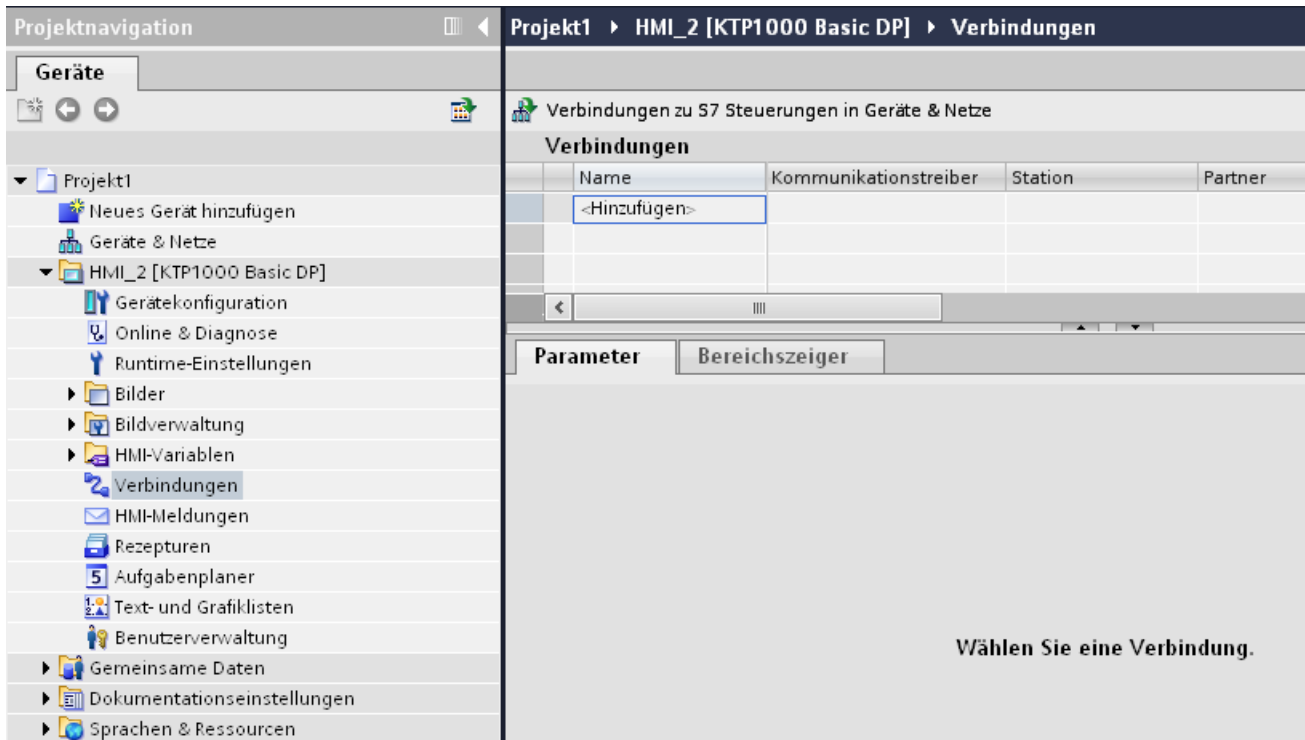
## PROFIBUS-Verbindung anlegen

### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät mit PROFIBUS-Schnittstelle ist angelegt.

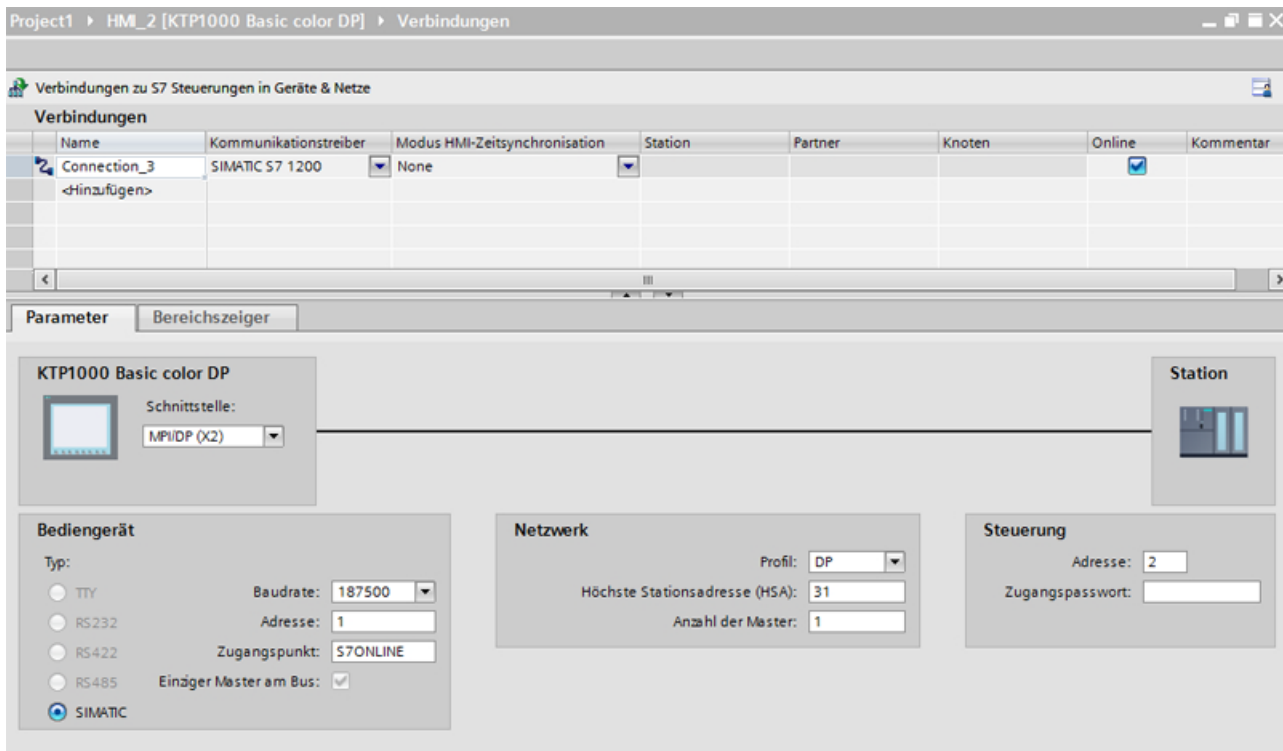
## Vorgehensweise

1. Öffnen Sie den Editor "Verbindungen" des Bediengeräts.
2. Doppelklicken Sie auf "<Hinzufügen>".



3. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber aus.
4. Klicken Sie auf den Namen der Verbindung.
5. Wählen Sie im Inspektorfenster unter "Parameter > Schnittstelle" die Schnittstelle "MPI/DP" aus.

6. Wählen Sie im Inspektorfenster unter "Parameter > Netzwerk" das Profil "DP" aus.



7. Stellen Sie im Inspektorfenster die Adressen der Kommunikationspartner ein:
- Bediengerät: "Parameter > Bediengerät > Adresse"
  - Steuerung: "Parameter > Steuerung > Adresse"

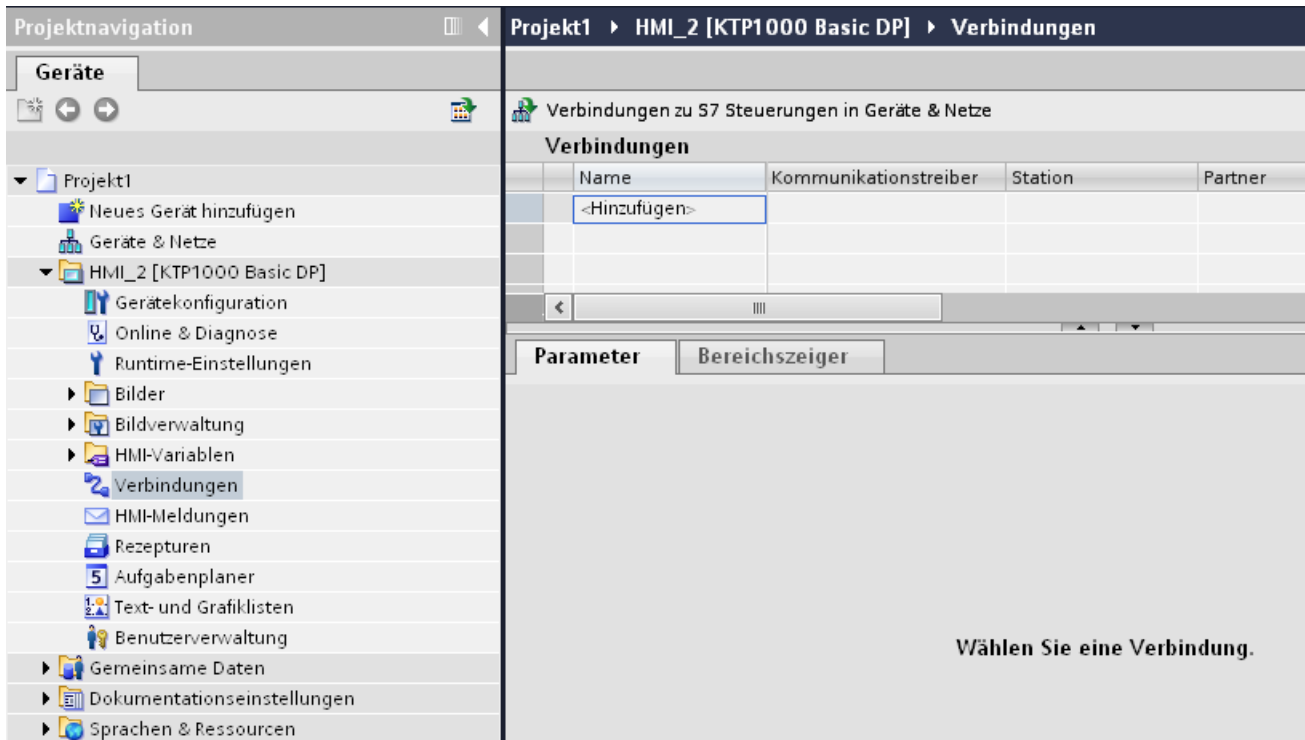
## MPI-Verbindung anlegen

### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät mit MPI-Schnittstelle ist angelegt.

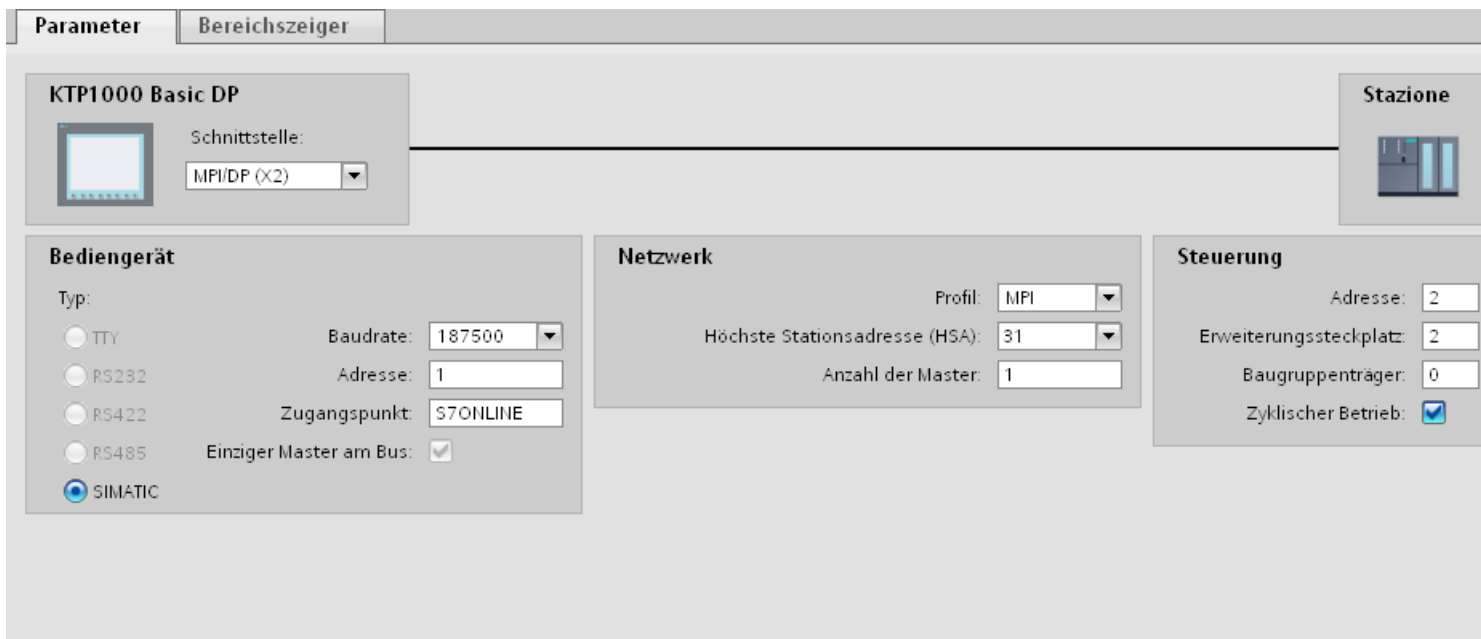
## Vorgehensweise

1. Öffnen Sie den Editor "Verbindungen" des Bediengeräts.
2. Doppelklicken Sie auf "<Hinzufügen>".



3. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber aus.
4. Klicken Sie auf den Namen der Verbindung.
5. Wählen Sie im Inspektorfenster unter "Parameter > Schnittstelle" die Schnittstelle "MPI/DP" aus.

6. Wählen Sie im Inspektorfenster unter "Parameter > Netzwerk" das Profil "MPI" aus.



7. Stellen Sie im Inspektorfenster die Adressen der Kommunikationspartner ein:

- Bediengerät: "Parameter > Bediengerät > Adresse"
- Steuerung: "Parameter > Steuerung > Adresse"

## Parameter für die Verbindung

### Parameter für die Verbindung (SIMATIC S7 300/400)

#### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät", "Netzwerk" und "Steuerung".

Project1 ▶ HMI\_2 [KTP1000 Basic color DP] ▶ Verbindungen

Verbindungen zu S7 Steuerungen in Geräte & Netze

**Verbindungen**

| Name                         | Kommunikationstreiber | Modus HMI-Zeitsynchronisation | Station | Partner | Knoten | Online                              | Kommentar |
|------------------------------|-----------------------|-------------------------------|---------|---------|--------|-------------------------------------|-----------|
| Connection_3<br><Hinzufügen> | SIMATIC S7 1200       | None                          |         |         |        | <input checked="" type="checkbox"/> |           |

Parameter Bereichszeiger

**KTP1000 Basic color DP**

Schnittstelle: MPI/DP (X2)

**Station**

**Bediengerät**

Typ:

- TTY
- RS232
- RS422
- RS485
- SIMATIC

Baudrate: 187500

Adresse: 1

Zugangspunkt: S7ONLINE

Einziges Master am Bus:

**Netzwerk**

Profil: DP

Höchste Stationsadresse (HSA): 31

Anzahl der Master: 1

**Steuerung**

Adresse: 2

Zugangspasswort:

## Ethernet-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie die Parameter für das Bediengerät im Netzwerk ein. Die Änderungen werden nicht automatisch auf das Bediengerät übertragen. Sie müssen die Einstellungen in der Systemsteuerung des Bediengeräts ändern.

- "Schnittstelle"  
Wenn Sie beim Projektieren direkt mit dem Bediengerät verbunden sind, können Sie die IP-Adresse des Bediengeräts in WinCC einrichten.

---

#### Hinweis

Wenn Sie die IP-Adresse bereits im Control-Panel des Bediengeräts eingerichtet haben, wird die IP-Adresse im Control Panel beim nächsten Laden überschrieben.

Wenn Sie "IP-Adresse auf anderem Weg beziehen" aktivieren, bleibt die bereits eingerichtete IP-Adresse im Control Panel beim nächsten Laden erhalten.

---

Beim Projekttransfer wird die IP-Adresse auf das Bediengerät übertragen. Die IP-Adresse des Bediengeräts richten Sie folgendermaßen ein:

- Klicken Sie auf das Bediengerät.
- Öffnen Sie den Editor "Gerätekonfiguration"
- Klicken Sie auf die Ethernet-Schnittstelle.
- Vergeben Sie im Inspektorfenster die IP-Adresse unter:  
"Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"
- "Adresse"  
Im Bereich "Adresse" vergeben Sie die IP-Adresse des Bediengeräts. Wenn Sie das WinCC-Projekt auf das Bediengerät transferieren, dann wird diese IP-Adresse direkt im Bediengerät eingerichtet.
- "Zugangspunkt"  
Der Zugangspunkt legt einen logischen Gerätenamen fest, über den der Kommunikationspartner erreicht wird.

### Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die IP-Adresse der S7-Baugruppe ein, an die das Bediengerät angeschlossen ist.
- "Erweiterungssteckplatz"  
Legt die Nummer des Erweiterungssteckplatzes der zu adressierenden CPU fest.



- "Baugruppenträger"  
Legt die Baugruppenträger-Nummer der zu adressierenden CPU fest.
- "Zyklischer Betrieb"

---

**Hinweis**

Die Einstellung "Zyklischer Betrieb" ist bei der Steuerung SIMATIC S7 1200 nicht projektierbar.

---

Wenn Sie den zyklischen Betrieb einschalten, optimiert die Steuerung die Datenübertragung zwischen Bediengerät und Steuerung. Damit wird eine bessere Performance erreicht.

Bei Parallelbetrieb mehrerer Bediengeräte schalten Sie den zyklischen Betrieb aus.

## PROFIBUS-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie einmal die Parameter für das Bediengerät im Netzwerk ein. Die Änderung gilt für jeden Kommunikationspartner.

- "Typ"  
Legt den benutzten physikalischen Anschluss fest.
- "Schnittstelle"  
Unter "Schnittstelle" wählen Sie die Schnittstelle des Bediengeräts aus, über die das Bediengerät am PROFIBUS-Netz angeschlossen ist.
- "Baudrate"  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit der Daten im Netzwerk ein. Die Baudrate wird durch das langsamste ans Netzwerk angeschlossene Bediengerät festgelegt. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

**Hinweis**

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5 Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Adresse"  
Unter "Adresse" stellen Sie die PROFIBUS-Adresse des Bediengeräts ein. Die PROFIBUS-Adresse muss im PROFIBUS-Netz eindeutig sein.
- "Einziges Master am Bus"  
Deaktiviert eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des Bediengeräts an das Netz. Eine passive Station (Slave) kann nur dann Daten senden, wenn Sie von einer aktiven Station (Master) dazu aufgefordert wird.  
Bei S7-200 müssen Sie ein Bediengerät als Master einstellen.
- "Zugangspunkt"  
Der Zugangspunkt legt einen logischen Gerätenamen fest, über den der Kommunikationspartner erreicht wird.

### Parameter für das Netzwerk

Unter "Netzwerk" stellen Sie die Parameter für das PROFIBUS-Netz ein, an dem das Bediengerät eingebunden ist.

- "Profil"  
Unter "Profil" wählen Sie das Netzprofil aus, das im Netzwerk verwendet wird. Stellen Sie unter "Profil" "DP", "Universal" oder "Standard" ein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Höchste Stationsadresse"  
Unter "Höchste Stationsadresse" stellen Sie die höchste Stationsadresse ein. Die höchste Stationsadresse muss größer gleich der größten tatsächlichen PROFIBUS-Adresse sein. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

#### Hinweis

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Anzahl der Master"  
Unter "Anzahl der Master" stellen Sie die Anzahl der Master im PROFIBUS-Netz ein. Diese Angabe ist erforderlich, damit die Busparameter korrekt berechnet werden.

### Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die PROFIBUS-Adresse der S7-Baugruppe (CPU, FM oder CP) ein, an die das Bediengerät angeschlossen ist.
- "Zyklischer Betrieb"

---

#### Hinweis

Die Einstellung "Zyklischer Betrieb" ist bei der Steuerung SIMATIC S7 1200 nicht projektierbar.

---

Wenn Sie den zyklischen Betrieb einschalten, optimiert die Steuerung die Datenübertragung zwischen Bediengerät und Steuerung. Damit wird eine bessere Performance erreicht. Bei Parallelbetrieb mehrerer Bediengeräte schalten Sie den zyklischen Betrieb aus. Diese Einstellung ist für SIMATIC S7-200 nicht notwendig.

## MPI-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie einmal die Parameter für das Bediengerät im Netzwerk ein. Die Änderung gilt für jeden Kommunikationspartner.

- "Typ"  
Legt den benutzten physikalischen Anschluss fest.
- "Schnittstelle"  
Unter "Schnittstelle" wählen Sie die Schnittstelle des Bediengeräts aus, über die das Bediengerät am MPI-Netz angeschlossen ist.
- "Baudrate"  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit der Daten im Netzwerk ein. Die Baudrate wird durch das langsamste ans Netzwerk angeschlossene Bediengerät festgelegt. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

#### **Hinweis**

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5 Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Adresse"  
Unter "Adresse" stellen Sie die MPI-Adresse des Bediengeräts ein. Die MPI-Adresse muss im MPI-Netz eindeutig sein.
- "Einziges Master am Bus"  
Deaktiviert eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des Bediengeräts an das Netz. Eine passive Station (Slave) kann nur dann Daten senden, wenn Sie von einer aktiven Station (Master) dazu aufgefordert wird. Wenn Sie nur Slaves am Bediengerät angeschlossen haben, müssen Sie deshalb die Sicherheitsfunktion "Einziges Master am Bus" deaktivieren.  
Bei S7-200 müssen Sie ein Bediengerät als Master einstellen.

### Parameter für das Netzwerk

Unter "Netzwerk" stellen Sie die Parameter für das MPI-Netz ein, mit dem das Bediengerät verbunden ist.

- "Profil"  
Unter "Profil" wählen Sie das Netzprofil aus, das im Netzwerk verwendet wird. Stellen Sie unter "Profil" "MPI" ein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Höchste Stationsadresse"  
Unter "Höchste Stationsadresse" stellen Sie die höchste Stationsadresse ein. Die höchste Stationsadresse muss größer oder gleich der größten tatsächlichen MPI-Adresse sein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Anzahl der Master"  
Diese Angabe ist bei MPI nicht erforderlich.

## Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die MPI-Adresse der S7-Baugruppe (CPU, FM oder CP) ein, an die das Bediengerät angeschlossen ist.
- "Zyklischer Betrieb"  
Wenn Sie den zyklischen Betrieb einschalten, optimiert die Steuerung die Datenübertragung zwischen Bediengerät und Steuerung. Damit wird eine bessere Performance erreicht. Bei Parallelbetrieb mehrerer Bediengeräte schalten Sie den zyklischen Betrieb aus. Diese Einstellung ist für SIMATIC S7-200 nicht notwendig.

## Zyklischer Betrieb

### Umgang mit der Auswahl "Zyklischer Betrieb"

Wenn der "zyklische Betrieb" eingeschaltet ist, dann schickt das Bediengerät zu Beginn der Kommunikation an die Steuerung ein Telegramm mit der Information, dass bestimmte Variablen ständig benötigt werden.

Die Steuerung verschickt daraufhin die Daten immer im gleichen Zyklus. Das Bediengerät erspart sich dadurch die ständige Neuankündigung der Daten.

Wenn der zyklische Betrieb abgeschaltet ist, verschickt das Bediengerät für jede benötigte Information eine extra Anforderung.

Weitere Eigenschaften:

- Der zyklische Betrieb entlastet bei der Übertragung von Daten das Bediengerät. Zur Entlastung des Bediengeräts werden die vorhandenen Ressourcen der Steuerung genutzt.
- Die Steuerung unterstützt nur eine bestimmte Anzahl an zyklischen Diensten. Wenn die Steuerung keine Ressourcen mehr für die zyklischen Dienste hat, dann übernimmt das Bediengerät wieder den Vorgang.
- Wenn die Steuerung keinen zyklischen Betrieb unterstützt, dann übernimmt das Bediengerät die Bildung des Zyklus.
- Bildvariablen werden nicht in den zyklischen Betrieb eingebunden.
- Der zyklische Betrieb wird nur bei Neustart der Runtime eingerichtet.
- Wenn der zyklische Betrieb aktiviert ist, dann wird steuerungsabhängig mehr als ein Auftrag vom Bediengerät an die Steuerung übertragen.
- Wenn der zyklische Betrieb abgewählt ist, dann wird immer nur ein Auftrag vom Bediengerät an die Steuerung übertragen.

## 10.8.9 Mit SIMATIC S7 200 kommunizieren

### 10.8.9.1 Kommunikation mit SIMATIC S7 200

#### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und der Steuerung SIMATIC S7 200 beschrieben.

Folgende Kommunikationskanäle können Sie für die Steuerung SIMATIC S7 200 projektieren:

- PROFINET und Ethernet
- PROFIBUS
- MPI
- PPI

#### HMI-Verbindung für die Kommunikation

Verbindungen von Bediengerät und SIMATIC S7 200 projektieren Sie im Editor "Verbindungen" des Bediengeräts.

### 10.8.9.2 Verbindung mit SIMATIC S7 200 anlegen

#### Einleitung

Eine Verbindung zu der Steuerung SIMATIC S7 200 projektieren Sie im Editor "Verbindungen" des Bediengeräts. Abhängig vom Bediengerät sind die Schnittstellen unterschiedlich benannt.

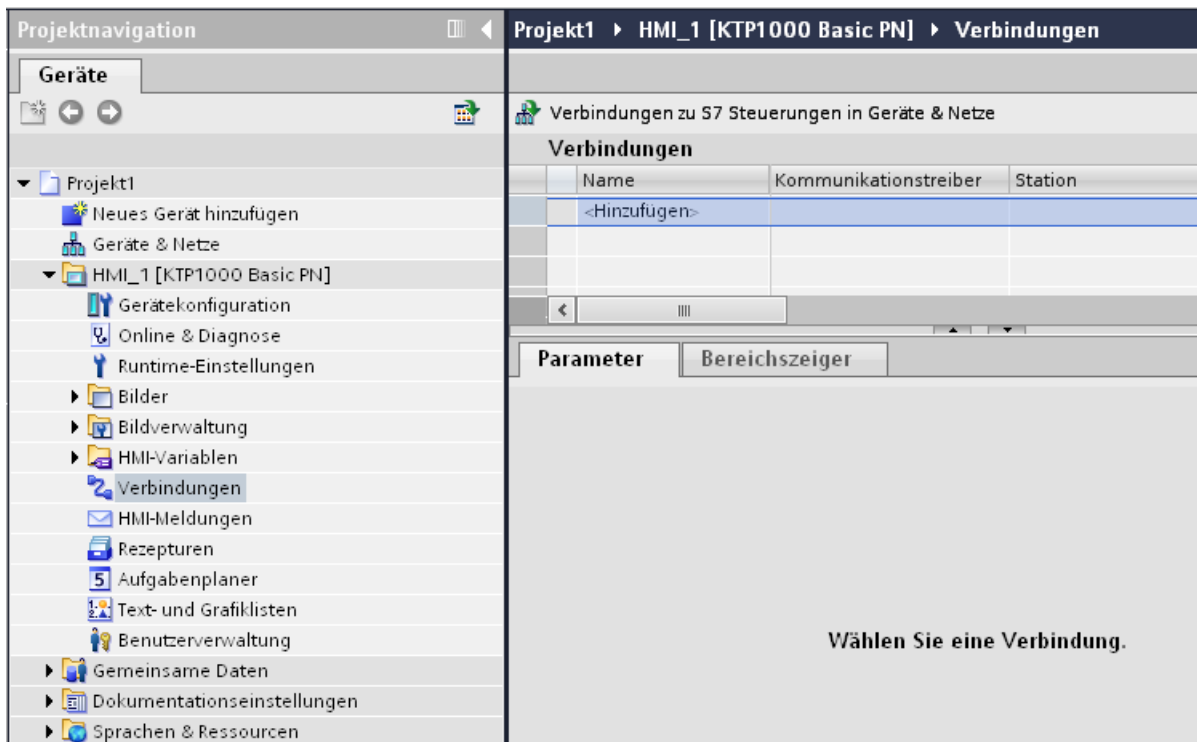
#### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

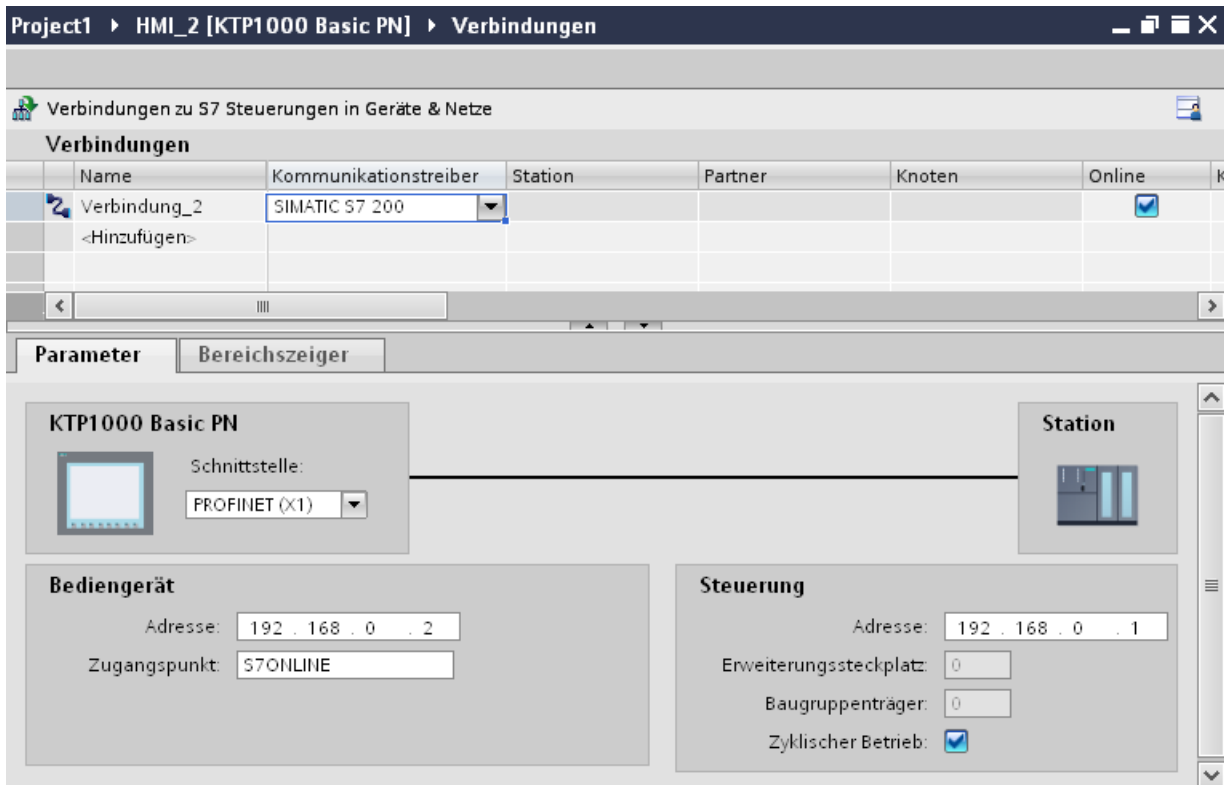
#### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".

3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".



4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "SIMATIC S7 200" aus.
5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.



Nähere Hinweise hierzu finden Sie im Kapitel "Parameter für die Verbindung (Seite 3665)".

### 10.8.9.3 Parameter für die Verbindung

#### Zyklischer Betrieb

#### Umgang mit der Auswahl "Zyklischer Betrieb"

Wenn der "zyklische Betrieb" eingeschaltet ist, dann schickt das Bediengerät zu Beginn der Kommunikation an die Steuerung ein Telegramm mit der Information, dass bestimmte Variablen ständig benötigt werden.

Die Steuerung verschickt daraufhin die Daten immer im gleichen Zyklus. Das Bediengerät erspart sich dadurch die ständige Neuanforderung der Daten.

Wenn der zyklische Betrieb abgeschaltet ist, verschickt das Bediengerät für jede benötigte Information eine extra Anforderung.

Weitere Eigenschaften:

- Der zyklische Betrieb entlastet bei der Übertragung von Daten das Bediengerät. Zur Entlastung des Bediengeräts werden die vorhandenen Ressourcen der Steuerung genutzt.
- Die Steuerung unterstützt nur eine bestimmte Anzahl an zyklischen Diensten. Wenn die Steuerung keine Ressourcen mehr für die zyklischen Dienste hat, dann übernimmt das Bediengerät wieder den Vorgang.
- Wenn die Steuerung keinen zyklischen Betrieb unterstützt, dann übernimmt das Bediengerät die Bildung des Zyklus.
- Bildvariablen werden nicht in den zyklischen Betrieb eingebunden.
- Der zyklische Betrieb wird nur bei Neustart der Runtime eingerichtet.
- Wenn der zyklische Betrieb aktiviert ist, dann wird steuerungsabhängig mehr als ein Auftrag vom Bediengerät an die Steuerung übertragen.
- Wenn der zyklische Betrieb abgewählt ist, dann wird immer nur ein Auftrag vom Bediengerät an die Steuerung übertragen.

## Parameter für die Verbindung (SIMATIC S7 200)

### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät" und "Steuerung".



Project1 > HMI\_2 [KTP1000 Basic PN] > Verbindungen

Verbindungen zu S7 Steuerungen in Geräte & Netze

**Verbindungen**


| Name                         | Kommunikationstreiber | Station | Partner | Knoten | Online                              |
|------------------------------|-----------------------|---------|---------|--------|-------------------------------------|
| Verbindung_2<br><Hinzufügen> | SIMATIC S7 200        |         |         |        | <input checked="" type="checkbox"/> |

Parameter Bereichszeiger

**KTP1000 Basic PN**

Schnittstelle:  
 PROFINET (X1)

**Station**



**Bediengerät**

Adresse: 192 . 168 . 0 . 2

Zugangspunkt: S7ONLINE

**Steuerung**

Adresse: 192 . 168 . 0 . 1

Erweiterungssteckplatz: 0

Baugruppenträger: 0

Zyklischer Betrieb:

## Ethernet-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie die Parameter für das Bediengerät im Netzwerk ein. Die Änderungen werden nicht automatisch auf das Bediengerät übertragen. Sie müssen die Einstellungen in der Systemsteuerung des Bediengeräts ändern.

- "Schnittstelle"  
Wenn Sie beim Projektieren direkt mit dem Bediengerät verbunden sind, können Sie die IP-Adresse des Bediengeräts in WinCC einrichten.

---

#### Hinweis

Wenn Sie die IP-Adresse bereits im Control-Panel des Bediengeräts eingerichtet haben, wird die IP-Adresse im Control Panel beim nächsten Laden überschrieben.

Wenn Sie "IP-Adresse auf anderem Weg beziehen" aktivieren, bleibt die bereits eingerichtete IP-Adresse im Control Panel beim nächsten Laden erhalten.

---

Beim Projekttransfer wird die IP-Adresse auf das Bediengerät übertragen. Die IP-Adresse des Bediengeräts richten Sie folgendermaßen ein:

- Klicken Sie auf das Bediengerät.
- Öffnen Sie den Editor "Gerätekonfiguration"
- Klicken Sie auf die Ethernet-Schnittstelle.
- Vergeben Sie im Inspektorfenster die IP-Adresse unter:  
"Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"
- "Adresse"  
Im Bereich "Adresse" vergeben Sie die IP-Adresse des Bediengeräts. Wenn Sie das WinCC-Projekt auf das Bediengerät transferieren, dann wird diese IP-Adresse direkt im Bediengerät eingerichtet.
- "Zugangspunkt"  
Der Zugangspunkt legt einen logischen Gerätenamen fest, über den der Kommunikationspartner erreicht wird.

### Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die IP-Adresse der S7-Baugruppe ein, an die das Bediengerät angeschlossen ist.
- "Erweiterungssteckplatz"  
Legt die Nummer des Erweiterungssteckplatzes der zu adressierenden CPU fest.

- "Baugruppenträger"  
Legt die Baugruppenträger-Nummer der zu adressierenden CPU fest.
- "Zyklischer Betrieb"

---

**Hinweis**

Die Einstellung "Zyklischer Betrieb" ist bei der Steuerung SIMATIC S7 1200 nicht projektierbar.

---

Wenn Sie den zyklischen Betrieb einschalten, optimiert die Steuerung die Datenübertragung zwischen Bediengerät und Steuerung. Damit wird eine bessere Performance erreicht.

Bei Parallelbetrieb mehrerer Bediengeräte schalten Sie den zyklischen Betrieb aus.

## PROFIBUS-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie einmal die Parameter für das Bediengerät im Netzwerk ein. Die Änderung gilt für jeden Kommunikationspartner.

- "Typ"  
Legt den benutzten physikalischen Anschluss fest.
- "Schnittstelle"  
Unter "Schnittstelle" wählen Sie die Schnittstelle des Bediengeräts aus, über die das Bediengerät am PROFIBUS-Netz angeschlossen ist.
- "Baudrate"  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit der Daten im Netzwerk ein. Die Baudrate wird durch das langsamste ans Netzwerk angeschlossene Bediengerät festgelegt. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

**Hinweis**

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5 Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Adresse"  
Unter "Adresse" stellen Sie die PROFIBUS-Adresse des Bediengeräts ein. Die PROFIBUS-Adresse muss im PROFIBUS-Netz eindeutig sein.
- "Einziges Master am Bus"  
Deaktiviert eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des Bediengeräts an das Netz. Eine passive Station (Slave) kann nur dann Daten senden, wenn Sie von einer aktiven Station (Master) dazu aufgefordert wird.  
Bei S7-200 müssen Sie ein Bediengerät als Master einstellen.
- "Zugangspunkt"  
Der Zugangspunkt legt einen logischen Gerätenamen fest, über den der Kommunikationspartner erreicht wird.

### Parameter für das Netzwerk

Unter "Netzwerk" stellen Sie die Parameter für das PROFIBUS-Netz ein, an dem das Bediengerät eingebunden ist.

- "Profil"  
Unter "Profil" wählen Sie das Netzprofil aus, das im Netzwerk verwendet wird. Stellen Sie unter "Profil" "DP", "Universal" oder "Standard" ein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Höchste Stationsadresse"  
Unter "Höchste Stationsadresse" stellen Sie die höchste Stationsadresse ein. Die höchste Stationsadresse muss größer gleich der größten tatsächlichen PROFIBUS-Adresse sein. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

#### Hinweis

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Anzahl der Master"  
Unter "Anzahl der Master" stellen Sie die Anzahl der Master im PROFIBUS-Netz ein. Diese Angabe ist erforderlich, damit die Busparameter korrekt berechnet werden.

### Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die PROFIBUS-Adresse der S7-Baugruppe (CPU, FM oder CP) ein, an die das Bediengerät angeschlossen ist.
- "Zyklischer Betrieb"

---

#### Hinweis

Die Einstellung "Zyklischer Betrieb" ist bei der Steuerung SIMATIC S7 1200 nicht projektierbar.

---

Wenn Sie den zyklischen Betrieb einschalten, optimiert die Steuerung die Datenübertragung zwischen Bediengerät und Steuerung. Damit wird eine bessere Performance erreicht. Bei Parallelbetrieb mehrerer Bediengeräte schalten Sie den zyklischen Betrieb aus. Diese Einstellung ist für SIMATIC S7-200 nicht notwendig.

## MPI-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie einmal die Parameter für das Bediengerät im Netzwerk ein. Die Änderung gilt für jeden Kommunikationspartner.

- "Typ"  
Legt den benutzten physikalischen Anschluss fest.
- "Schnittstelle"  
Unter "Schnittstelle" wählen Sie die Schnittstelle des Bediengeräts aus, über die das Bediengerät am MPI-Netz angeschlossen ist.
- "Baudrate"  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit der Daten im Netzwerk ein. Die Baudrate wird durch das langsamste ans Netzwerk angeschlossene Bediengerät festgelegt. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

#### Hinweis

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5 Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Adresse"  
Unter "Adresse" stellen Sie die MPI-Adresse des Bediengeräts ein. Die MPI-Adresse muss im MPI-Netz eindeutig sein.
- "Einzigster Master am Bus"  
Deaktiviert eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des Bediengeräts an das Netz. Eine passive Station (Slave) kann nur dann Daten senden, wenn Sie von einer aktiven Station (Master) dazu aufgefordert wird. Wenn Sie nur Slaves am Bediengerät angeschlossen haben, müssen Sie deshalb die Sicherheitsfunktion "Einzigster Master am Bus" deaktivieren.  
Bei S7-200 müssen Sie ein Bediengerät als Master einstellen.

### Parameter für das Netzwerk

Unter "Netzwerk" stellen Sie die Parameter für das MPI-Netz ein, mit dem das Bediengerät verbunden ist.

- "Profil"  
Unter "Profil" wählen Sie das Netzprofil aus, das im Netzwerk verwendet wird. Stellen Sie unter "Profil" "MPI" ein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Höchste Stationsadresse"  
Unter "Höchste Stationsadresse" stellen Sie die höchste Stationsadresse ein. Die höchste Stationsadresse muss größer oder gleich der größten tatsächlichen MPI-Adresse sein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Anzahl der Master"  
Diese Angabe ist bei MPI nicht erforderlich.

### Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die MPI-Adresse der S7-Baugruppe (CPU, FM oder CP) ein, an die das Bediengerät angeschlossen ist.
- "Zyklischer Betrieb"  
Wenn Sie den zyklischen Betrieb einschalten, optimiert die Steuerung die Datenübertragung zwischen Bediengerät und Steuerung. Damit wird eine bessere Performance erreicht. Bei Parallelbetrieb mehrerer Bediengeräte schalten Sie den zyklischen Betrieb aus. Diese Einstellung ist für SIMATIC S7-200 nicht notwendig.

### PPI-Parameter

#### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie einmal die Parameter für das Bediengerät im Netzwerk ein. Die Änderung gilt für jeden Kommunikationspartner.

- "Typ"  
Legt den benutzten physikalischen Anschluss fest.
- "Schnittstelle"  
Unter "Schnittstelle" wählen Sie die Schnittstelle des Bediengeräts aus, über die das Bediengerät am PPI-Netz angeschlossen ist.
- "Baudrate"  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit der Daten im Netzwerk ein. Die Baudrate wird durch das langsamste ans Netzwerk angeschlossene Bediengerät festgelegt. Die Einstellung muss im gesamten Netzwerk gleich sein.

---

#### Hinweis

Wenn Sie beim OP 73 oder beim OP 77A eine Baudrate von 1,5 Mbaud einstellen, muss die höchste Stationsadresse kleiner oder gleich 63 sein.

---

- "Adresse"  
Unter "Adresse" stellen Sie die PPI-Adresse des Bediengeräts ein. Die PPI-Adresse muss im PPI-Netz eindeutig sein.
- "Zugangspunkt"  
Unter "Zugangspunkt" stellen Sie den Zugangspunkt ein, über welchen der Kommunikationspartner erreicht wird.
- "Einziges Master am Bus"  
Deaktiviert eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des Bediengeräts an das Netz. Eine passive Station (Slave) kann nur dann Daten senden, wenn Sie von einer aktiven Station (Master) dazu aufgefordert wird. Wenn Sie nur Slaves am Bediengerät angeschlossen haben, müssen Sie deshalb die Sicherheitsfunktion "Einziges Master am Bus" deaktivieren.  
Bei S7-200 müssen Sie ein Bediengerät als Master einstellen.

## Parameter für das Netzwerk

Unter "Netzwerk" stellen Sie die Netzparameter ein, an dem das Bediengerät eingebunden ist.

- "Profil"  
Unter "Profil" wählen Sie das Netzprofil aus, das im Netzwerk verwendet wird. Stellen Sie unter "Profil" "PPI" ein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Höchste Stationsadresse"  
Unter "Höchste Stationsadresse" stellen Sie die höchste Stationsadresse ein. Die höchste Stationsadresse muss größer gleich der größten tatsächlichen MPI-Adresse sein. Die Einstellung muss im gesamten Netzwerk gleich sein.
- "Anzahl der Master"  
Stellen Sie die Anzahl der im Netz befindlichen Master auf "1" ein.

## Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die PPI-Adresse der S7-Baugruppe (CPU) ein, an die das Bediengerät angeschlossen ist.
- "Zyklischer Betrieb"  
Dieser Parameter ist für eine Kommunikation über PPI nicht notwendig.

### 10.8.9.4 Datenaustausch

#### Datenaustausch über Bereichszeiger

#### Allgemeines zu Bereichszeigern

##### Einleitung

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu. In diese Datenbereiche wird während der Kommunikation wechselseitig von der Steuerung und dem Bediengerät geschrieben und gelesen.

Durch die Auswertung der abgelegten Daten lösen Steuerung und Bediengerät gegenseitig fest definierte Aktionen aus.

#### Projektierung von Bereichszeigern

Bevor Sie den Bereichszeiger verwenden, aktivieren Sie die Bereichszeiger unter "Verbindungen > Bereichszeiger". Danach parametrieren Sie die Bereichszeiger.

Nähere Hinweise zum Projektieren von Bereichszeigern finden Sie unter:

Bereichszeiger projektieren (Seite 3491)

## Bereichszeiger "Bildnummer"

### Funktion

Die Bediengeräte legen im Bereichszeiger "Bildnummer" Informationen über das am Bediengerät aufgerufene Bild ab.

Dadurch ist es möglich, Informationen zum aktuellen Bildinhalt des Bediengeräts zur Steuerung zu übertragen. In der Steuerung können bestimmte Reaktionen getriggert werden, z. B. den Aufruf eines anderen Bildes.

### Verwendung

Bevor der Bereichszeiger "Bildnummer" verwendet werden kann, muss dieser unter "Kommunikation > Verbindungen" eingerichtet und aktiviert werden. Der Bereichszeiger "Bildnummer" kann nur in **einer** Steuerung und in dieser Steuerung nur **einmal** angelegt werden.

Die Bildnummer wird immer zur Steuerung übertragen, wenn ein neues Bild aktiviert wird oder der Fokus innerhalb eines Bildes von einem Bildobjekt zu einem anderen wechselt.

### Aufbau

Der Bereichszeiger ist ein Datenbereich im Speicher der Steuerung mit einer festen Länge von 5 Worten.

|        | 15                  | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1.Wort | Aktueller Bildtyp   |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 2.Wort | Aktuelle Bildnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 3.Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 4.Wort | Aktuelle Feldnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 5.Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

- Aktueller Bildtyp  
"1" für Grundbild oder  
"4" für Permanentfenster
- Aktuelle Bildnummer  
1 bis 32767
- Aktuelle Feldnummer  
1 bis 32767



## Bereichszeiger "Datum/Uhrzeit"

### Funktion

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von dem Bediengerät zur Steuerung verwendet.

Die Steuerung schreibt den Steuerauftrag "41" in das Auftragsfach.

Mit der Auswertung des Steuerauftrags schreibt das Bediengerät sein aktuelles Datum und die Uhrzeit in den im Bereichszeiger "Datum/Uhrzeit" projektierten Datenbereich. Alle Angaben sind BCD-codiert.

Wenn in einem Projekt mehrere Verbindungen projektiert sind, und in einer der Verbindungen der Bereichszeiger "Datum / Uhrzeit" verwendet werden soll, so muss der Bereichszeiger für jede projektierte Verbindung aktiviert werden.

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" projektiert haben, dann können Sie den Bereichszeiger "Datum/Uhrzeit PLC" nicht verwenden.

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" verwenden, dann ist eine symbolische Adressierung nicht möglich.

Wenn in einem Projekt mehrere Verbindungen projektiert sind, und in einer der Verbindungen der Bereichszeiger "Datum / Uhrzeit" verwendet werden soll, so muss der Bereichszeiger für jede projektierte Verbindung aktiviert werden.

Der Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

| Datenwort | Höherwertiges Byte |  |  |  |   | Niederwertiges Byte   |  |  |  |   |         |
|-----------|--------------------|--|--|--|---|-----------------------|--|--|--|---|---------|
|           | 7                  |  |  |  | 0 | 7                     |  |  |  | 0 |         |
| n+0       | Reserviert         |  |  |  |   | Stunde (0-23)         |  |  |  |   | Uhrzeit |
| n+1       | Minute (0-59)      |  |  |  |   | Sekunde (0-59)        |  |  |  |   |         |
| n+2       | Reserviert         |  |  |  |   | Reserviert            |  |  |  |   |         |
| n+3       | Reserviert         |  |  |  |   | Wochentag (1-7, 1=So) |  |  |  |   | Datum   |
| n+4       | Tag (1-31)         |  |  |  |   | Monat (1-12)          |  |  |  |   |         |
| n+5       | Jahr (80-99/0-29)  |  |  |  |   | Reserviert            |  |  |  |   |         |

#### Hinweis

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

## Bereichszeiger "Datum/Uhrzeit PLC"

### Funktion

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von der Steuerung zum Bediengerät verwendet. Sie setzen diesen Bereichszeiger ein, wenn die Steuerung Master für die Zeit ist.

Die Steuerung lädt den Datenbereich des Bereichszeigers. Alle Angaben sind BCD-codiert.

Das Bediengerät liest zyklisch die Daten über den projektierten Erfassungszyklus und synchronisiert sich.

---

#### Hinweis

Wählen Sie in der Projektierung den Erfassungszyklus für den Bereichszeiger Datum/Uhrzeit nicht zu klein, da dies die Performance des Bediengeräts beeinflusst.  
Empfehlung: Erfassungszyklus 1 Minute, wenn dies Ihr Prozess erlaubt.

---

"Datum/Uhrzeit PLC" ist ein globaler Bereichszeiger und kann in einem Projekt nur ein Mal projektiert werden.

---

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" projektiert haben, können Sie den Bereichszeiger "Datum/Uhrzeit" nicht verwenden.

---

Der Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

### Format DATE\_AND\_TIME (BCD-codiert)

| Datenwort         | Höherwertiges Byte |       |   | Niederwertiges Byte |                       |   |
|-------------------|--------------------|-------|---|---------------------|-----------------------|---|
|                   | 7                  | ..... | 0 | 7                   | .....                 | 0 |
| n+0               | Jahr (80-99/0-29)  |       |   | Monat (1-12)        |                       |   |
| n+1               | Tag (1-31)         |       |   | Stunde (0-23)       |                       |   |
| n+2               | Minute (0-59)      |       |   | Sekunde (0-59)      |                       |   |
| n+3               | Reserviert         |       |   | Reserviert          | Wochentag (1-7, 1=So) |   |
| n+4 <sup>1)</sup> | Reserviert         |       |   | Reserviert          |                       |   |
| n+5 <sup>1)</sup> | Reserviert         |       |   | Reserviert          |                       |   |

- 1) Die beiden Datenwörter müssen im Datenbereich vorhanden sein, um eine Übereinstimmung des Datenformats mit WinCC flexible sicherzustellen und das Lesen falscher Informationen zu vermeiden.

---

### Hinweis

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

---

## Bereichszeiger "Koordinierung"

### Funktion

Der Bereichszeiger "Koordinierung" dient zur Realisierung der folgenden Funktionen:

- Anlauf des Bediengeräts im Steuerungsprogramm erkennen
- Aktuelle Betriebsart des Bediengeräts im Steuerungsprogramm erkennen
- Kommunikationsbereitschaft des Bediengeräts im Steuerungsprogramm erkennen

Der Bereichszeiger "Koordinierung" standardmäßig hat eine Länge von einem Wort und kann nicht verändert werden.

### Verwendung

---

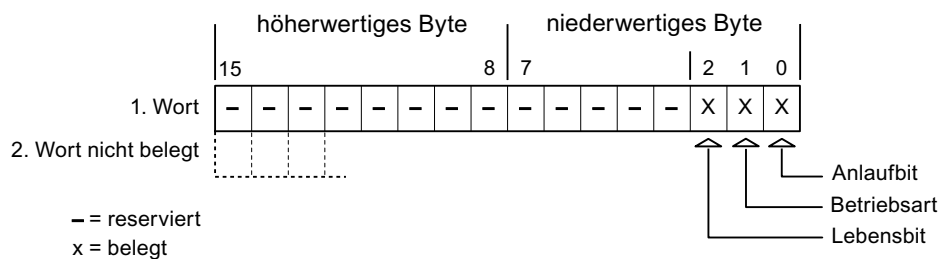
### Hinweis

Bei jeder Aktualisierung des Bereichszeigers durch das Bediengerät wird immer der komplette Koordinierungsbereich geschrieben.

Das SPS-Programm darf deshalb im Koordinierungsbereich keine Änderungen vornehmen.

---

## Belegung der Bits im Bereichszeiger "Koordinierung"



### Anlaufbit

Das Anlaufbit wird durch das Bediengerät während des Anlaufvorgangs kurzfristig auf "0" gesetzt. Nach dem Anlaufvorgang steht das Bit dauerhaft auf "1".

## Betriebsart

Sobald das Bediengerät durch den Benutzer offline geschaltet wird, wird das Betriebsartenbit auf 1 gesetzt. Im Normalbetrieb des Bediengeräts ist der Zustand des Betriebsartenbits "0". Im Steuerungsprogramm können Sie durch Abfrage dieses Bits die aktuelle Betriebsart des Bediengeräts ermitteln.

## Lebensbit

Das Lebensbit wird durch das Bediengerät im zeitlichen Abstand von ca. einer Sekunde invertiert. Im Steuerungsprogramm können Sie durch Abfrage dieses Bits überprüfen, ob die Verbindung zum Bediengerät noch besteht.

## Bereichszeiger "Projektkennung"

### Funktion

Beim Start der Runtime kann überprüft werden, ob das Bediengerät an die richtige Steuerung angeschlossen ist. Diese Überprüfung ist beim Einsatz mehrerer Bediengeräte wichtig.

Dazu vergleicht das Bediengerät einen in der Steuerung hinterlegten Wert mit dem in der Projektierung angegebenen Wert. Damit wird die Kompatibilität der Projektierungsdaten mit dem Steuerungsprogramm sichergestellt. Eine fehlende Übereinstimmung führt zur Anzeige einer Systemmeldung am Bediengerät und zum Stopp der Runtime.

### Verwendung

Um diesen Bereichszeiger zu verwenden, richten Sie bei der Projektierung Folgendes ein:

- Angabe der Version, welche die Projektierung hat. Möglicher Wert zwischen 1 und 255. Sie geben die Version ein im Editor "Runtime-Einstellungen > Allgemein" im Bereich "Identifizierung".
- Datenadresse des Werts für die Version, die in der Steuerung hinterlegt ist: Sie geben die Datenadresse ein im Editor "Kommunikation > Verbindungen" unter "Adresse".

### Ausfall einer Verbindung

Ein Verbindungsausfall zu einem Gerät, auf dem der Bereichszeiger "Projektkennung" projiziert ist, hat zur Folge, dass auch alle anderen Verbindungen des Geräts "Offline" geschaltet werden.

Dieses Verhalten hat folgende Voraussetzungen:

- Sie haben in einem Projekt mehrere Verbindungen projiziert.
- Sie verwenden in mindestens einer Verbindung den Bereichszeiger "Projektkennung".

Folgende Ursachen können Verbindungen in den Zustand "Offline" setzen:

- Die Steuerung ist nicht erreichbar.
- Die Verbindung wurde im Engineering System offline geschaltet.

## Bereichszeiger "Steuerungsauftrag"

### Funktion

Über das Steuerungsauftrags-Fach können dem Bediengerät Steuerungsaufträge gegeben und damit Aktionen am Bediengerät getriggert werden. Zu diesen Funktionen gehören z. B.:

- Bild anzeigen
- Datum und Uhrzeit stellen

### Datenstruktur

Im ersten Wort des Steuerungsauftrags-Fachs steht die Auftragsnummer. Je nach Steuerungsauftrag können dann bis zu drei Parameter übergeben werden.

| Wort | Höherwertiges Byte | Niederwertiges Byte |
|------|--------------------|---------------------|
| n+0  | 0                  | Auftragsnummer      |
| n+1  |                    | Parameter 1         |
| n+2  |                    | Parameter 2         |
| n+3  |                    | Parameter 3         |

Wenn das erste Wort des Steuerungsauftrags-Fachs ungleich 0 ist, wertet das Bediengerät den Steuerungsauftrag aus. Aus diesem Grund müssen zuerst die Parameter in das Steuerungsauftrags-Fach eingetragen werden und dann erst die Auftragsnummer.

Wenn das Bediengerät den Steuerungsauftrag angenommen hat, wird das erste Wort wieder auf 0 gesetzt. Die Ausführung des Steuerungsauftrags ist zu diesem Zeitpunkt im Allgemeinen noch nicht abgeschlossen.

### Steuerungsaufträge

Nachfolgend sind alle Steuerungsaufträge und deren Parameter aufgelistet. Die Spalte "Nr." enthält die Auftragsnummer des Steuerungsauftrags. Generell können Steuerungsaufträge nur dann von der Steuerung getriggert werden, wenn das Bediengerät im Online-Betrieb ist.

---

#### Hinweis

Beachten Sie, dass nicht alle Bediengeräte Steuerungsaufträge unterstützen. So gibt es z. B. für das TP 170A und Micro Panels keine Steuerungsaufträge.

---

| Nr.       | Funktion   |   |
|-----------|--|---|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>   |   |
|           | Parameter 1  | Linkes Byte: -<br>Rechtes Byte: Stunden (0-23)                      |
|           | Parameter 2  | Linkes Byte: Minuten (0-59)<br>Rechtes Byte: Sekunden (0-59)        |
|           | Parameter 3  | -   |
| <b>15</b> | <b>Datum stellen (BCD-codiert) <sup>3)</sup></b>   |   |
|           | Parameter 1  | Linkes Byte: -<br>Rechtes Byte: Wochentag<br>(1-7: Sonntag-Samstag) |
|           | Parameter 2  | Linkes Byte: Tag (1-31)<br>Rechtes Byte: Monat (1-12)               |
|           | Parameter 3  | Linkes Byte: Jahr   |
| <b>23</b> | <b>Benutzer anmelden</b>   |   |
|           | Meldet den Benutzer "PLC User" mit der im Parameter 1 übergebenen Gruppennummer am Bediengerät an.<br>Voraussetzung für die Anmeldung ist, dass die übergebene Gruppennummer im Projekt vorhanden ist.   |   |
|           | Parameter 1  | Gruppennummer 1 - 255   |
|           | Parameter 2, 3   | -   |
| <b>24</b> | <b>Benutzer abmelden</b>   |   |
|           | Meldet den aktuell angemeldeten Benutzer ab.<br>(Funktion entspricht der Systemfunktion "Abmelden")  |   |
|           | Parameter 1, 2, 3  | -   |
| <b>40</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |   |
|           | (Im S7-Format DATE_AND_TIME)<br>Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.   |   |
|           | Parameter 1, 2, 3  | -   |
| <b>41</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |   |
|           | (Im Format OP/MP)<br>Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.  |   |
|           | Parameter 1, 2, 3  | -   |
| <b>46</b> | <b>Variable aktualisieren</b>  |   |
|           | Veranlasst das Bediengerät den aktuellen Wert der Variablen aus der Steuerung zu lesen, deren Aktualisierungskennung mit dem im Parameter 1 übergebenen Wert übereinstimmt.<br>(Funktion entspricht der Systemfunktion "AktualisiereVariable") |   |
|           | Parameter 1  | 1 - 100   |
| <b>49</b> | <b>Meldepuffer löschen</b>   |   |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Warnings" aus dem Meldepuffer.  |   |
|           | Parameter 1, 2, 3  | -   |
| <b>50</b> | <b>Meldepuffer löschen</b>   |   |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Errors" aus dem Meldepuffer.  |   |
|           | Parameter 1, 2, 3  | -   |

| Nr. | Funktion                                       |  |
|-----|--|--|
| 14  | Uhrzeit stellen (BCD-codiert)                  |  |
| 51  | Bildanwahl <sup>2)</sup>                       |  |
|     | Parameter 1                                    | Bildnummer   |
|     | Parameter 2                                    | -  |
|     | Parameter 3                                    | Feldnummer   |
| 69  | Datensatz aus Steuerung lesen <sup>1)</sup>    |  |
|     | Parameter 1                                    | Rezepturnummer (1-999)   |
|     | Parameter 2                                    | Datensatznummer (1-65535)  |
|     | Parameter 3                                    | 0: Vorhandenen Datensatz nicht überschreiben<br>1: Vorhandenen Datensatz überschreiben |
| 70  | Datensatz in Steuerung schreiben <sup>1)</sup> |  |
|     | Parameter 1                                    | Rezepturnummer (1-999)   |
|     | Parameter 2                                    | Datensatznummer (1-65535)  |
|     | Parameter 3                                    | -  |

|               |   |
|---------------|---|
| <sup>1)</sup> | Nur bei Geräten, die Rezepturen unterstützen  |
| <sup>2)</sup> | Auf den Bediengeräten OP 73, OP 77A und TP 177A wird der Steuerungsauftrag "Bildanwahl" auch dann ausgeführt, wenn die Bildschirmtastatur geöffnet ist. |
| <sup>3)</sup> | Beim Bediengerät KTP 600 BASIC PN wird der Wochentag ignoriert.   |

## Bereichszeiger "Datensatz"

## Bereichszeiger "Datensatz"

## Funktion

Bei der Übertragung von Datensätzen zwischen Bediengerät und Steuerung greifen beide Kommunikationspartner wechselseitig auf gemeinsame Kommunikationsbereiche in der Steuerung zu.

## Übertragungsarten

Für die Übertragung von Datensätzen zwischen Bediengerät und Steuerung gibt es zwei Möglichkeiten:

- Übertragung ohne Synchronisation
- Übertragung mit Synchronisation über das Datenfach

Datensätze werden immer direkt übertragen. D. h., die Variablenwerte werden direkt, ohne den Umweg über eine Zwischenablage, aus der Adresse gelesen oder in die Adresse geschrieben, die für die Variable projektiert ist.

### Übertragung von Datensätzen anstoßen

Für den Anstoß der Übertragung gibt es drei Möglichkeiten:

- Bedienung in der Rezepturanzeige
- Steuerungsaufträge  
Die Übertragung der Datensätze kann auch durch die Steuerung getriggert werden.
- Auslösen projektierter Funktionen

Wenn die Übertragung von Datensätzen durch eine projektierte Funktion oder einen Steuerungsauftrag getriggert wird, ist die Rezepturanzeige am Bediengerät weiterhin ungehindert bedienbar. Die Datensätze werden im Hintergrund übertragen.

Das gleichzeitige Abarbeiten mehrerer Übertragungsanforderungen ist jedoch nicht möglich. In diesem Fall lehnt das Bediengerät eine weitere Übertragung mit einer Systemmeldung ab.

### Ablauf der Übertragung bei Bedienung in der Rezepturanzeige

#### Lesen aus der Steuerung durch Bedienung in der Rezepturanzeige

| Schritt | Aktion  |                            |
|---------|---|----------------------------|
| 1       | Prüfung: Statuswort = 0?  |                            |
|         | Ja  | Nein                       |
| 2       | Das Bediengerät trägt die zu lesende Rezepturnummer sowie den Status "Übertragung läuft" im Datenfach ein und setzt die Datensatznummer auf 0.  | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und zeigt diese in der Rezepturanzeige an.<br>Bei Rezepturen mit synchronisierten Variablen werden die Werte aus der Steuerung auch in die Variablen geschrieben. |                            |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".   |                            |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                            |

#### Schreiben in die Steuerung durch Bedienung in der Rezepturanzeige

| Schritt | Aktion  |                            |
|---------|---|----------------------------|
| 1       | Prüfung: Statuswort = 0?  |                            |
|         | Ja  | Nein                       |
|         | Das Bediengerät trägt die zu schreibende Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch mit Systemmeldung. |
| 2       | Das Bediengerät schreibt die aktuellen Werte in die Steuerung.<br>Bei Rezepturen mit synchronisierten Variablen werden die geänderten Werte zwischen Rezepturanzeige und Variablen abgeglichen und dann in die Steuerung geschrieben. |                            |
| 3       | Das Bediengerät setzt den Status "Übertragung beendet".   |                            |



| Schritt | Aktion  |
|---------|---|
| 4       | Das Steuerungsprogramm kann jetzt ggf. die übertragenen Daten auswerten.                                    |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |

**Hinweis**

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

**Hinweis**

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach ist auf "Übertragung beendet" gesetzt.
- Der Status im Datenfach ist auf "Übertragung mit Fehler beendet" gesetzt.

**Ablauf der Übertragung bei Steuerungsauftrag**

Die Übertragung von Datensätzen zwischen Bediengerät und Steuerung kann vom Bediengerät oder von der Steuerung initiiert werden.

Für diese Art der Übertragung stehen die beiden Steuerungsaufträge Nr. 69 und Nr. 70 zur Verfügung.

**Nr. 69: Datensatz aus Steuerung lesen ("SPS → DAT")**

Der Steuerungsauftrag Nr. 69 überträgt Datensätze von der Steuerung zum Bediengerät. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Linkes Byte (LB)   | Rechtes Byte (RB) |
|--------|--|-------------------|
| Wort 1 | 0  | 69                |
| Wort 2 | Rezepturnummer (1-999)   |                   |
| Wort 3 | Datensatznummer (1-65.535)   |                   |
| Wort 4 | Vorhandenen Datensatz nicht überschreiben: 0<br>Vorhandenen Datensatz überschreiben: 1 |                   |

**Nr. 70: Datensatz in Steuerung schreiben ("DAT → SPS")**

Der Steuerungsauftrag Nr. 70 überträgt Datensätze vom Bediengerät zur Steuerung. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Linkes Byte (LB)       | Rechtes Byte (RB) |
|--------|------------------------|-------------------|
| Wort 1 | 0                      | 70                |
| Wort 2 | Rezepturnummer (1-999) |                   |

|        | Linkes Byte (LB)           | Rechtes Byte (RB) |
|--------|----------------------------|-------------------|
| Wort 3 | Datensatznummer (1-65.535) |                   |
| Wort 4 | —                          |                   |

**Ablauf bei Lesen aus der Steuerung durch Steuerungsauftrag "SPS → DAT" (Nr. 69)**

| Schritt | Aktion  |                           |
|---------|---|---------------------------|
| 1       | Prüfung: Statuswort = 0?  |                           |
|         | Ja  | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese Werte in dem Datensatz, der im Steuerungsauftrag angegeben ist.   |                           |
| 4       | <ul style="list-style-type: none"> <li>Falls im Auftrag "Überschreiben" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben. Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>Falls im Auftrag "Nicht überschreiben" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                           |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                           |

**Ablauf bei Schreiben in die Steuerung durch Steuerungsauftrag "DAT → SPS" (Nr. 70)**

| Schritt | Aktion   |                           |
|---------|--|---------------------------|
| 1       | Prüfung: Statuswort = 0?   |                           |
|         | Ja   | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät holt die Werte des im Auftrag angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.   |                           |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                           |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                           |

## Ablauf der Übertragung bei Anstoß durch projizierte Funktion

### Lesen aus der Steuerung durch projizierte Funktion

| Schritt | Aktion  |                            |
|---------|---|----------------------------|
| 1       | Prüfung: Statuswort = 0?  |                            |
|         | Ja  | Nein                       |
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese in dem Datensatz ab, der in der Funktion angegeben ist.   |                            |
| 4       | <ul style="list-style-type: none"> <li>Falls in der Funktion bei "Überschreiben" "Ja" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben. Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>Falls in der Funktion bei "Überschreiben" "Nein" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                            |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                            |

### Schreiben in die Steuerung durch projizierte Funktion

| Schritt | Aktion   |                            |
|---------|--|----------------------------|
| 1       | Prüfung: Statuswort = 0?   |                            |
|         | Ja   | Nein                       |
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät holt die Werte des in der Funktion angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.  |                            |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                            |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                            |

## Mögliche Fehlerursachen bei der Übertragung von Datensätzen

### Mögliche Fehlerursachen

Falls die Übertragung von Datensätzen mit Fehler beendet wird, so kann dies u. a. folgende Ursachen haben:

- Variablen-Adresse in der Steuerung nicht eingerichtet
- Überschreiben von Datensätzen nicht möglich
- Rezepturnummer nicht vorhanden
- Datensatznummer nicht vorhanden

---

#### Hinweis

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

---

#### Hinweis

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach wurde auf "Übertragung beendet" gesetzt.
  - Der Status im Datenfach wurde auf "Übertragung mit Fehler beendet" gesetzt.
- 

### Reaktion auf fehlerbedingten Abbruch

Das Bediengerät reagiert auf einen fehlerbedingten Abbruch der Übertragung von Datensätzen wie folgt:

- Anstoß über Bedienung in der Rezepturanzeige  
Hinweise in der Statuszeile der Rezepturanzeige und Ausgabe von Systemmeldungen
- Anstoß über Funktion  
Ausgabe von Systemmeldungen
- Anstoß über Steuerungsauftrag  
Keine Rückmeldung am Bediengerät

Unabhängig davon können Sie den Status der Übertragung durch Abfragen des Statusworts im Datenfach auswerten.

### Übertragung ohne Synchronisation

Bei der asynchronen Übertragung von Datensätzen zwischen Bediengerät und Steuerung findet keine Koordination über gemeinsam benutzte Datenbereiche statt. Die Einrichtung eines Datenbereichs beim Projektieren ist daher nicht nötig.

Die asynchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Ein unkontrolliertes Überschreiben der Daten durch die Kommunikationspartner kann systembedingt ausgeschlossen werden.
- Die Steuerung braucht keine Informationen über die Rezepturnummer und die Datensatznummer.
- Die Übertragung von Datensätzen wird durch Bedienung am Bediengerät getriggert.

### **Werte lesen**

Beim Anstoß der Übertragung zum Lesen werden die Werte aus den Steuerungsadressen gelesen und zum Bediengerät übertragen.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die Werte werden in das Bediengerät geladen. Im Bediengerät können Sie diese weiterverarbeiten, z. B. Werte ändern, speichern etc.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte werden sofort auf dem Datenträger gespeichert.

### **Werte schreiben**

Beim Anstoß der Übertragung zum Schreiben werden die Werte in die Steuerungsadressen geschrieben.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die aktuellen Werte werden in die Steuerung geschrieben.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte vom Datenträger werden in die Steuerung geschrieben.

### **Übertragung mit Synchronisation**

Bei der synchronen Übertragung setzen beide Kommunikationspartner Status-Bits im gemeinsam benutzten Datenbereich. Dadurch können Sie in Ihrem Steuerungsprogramm ein unkontrolliertes gegenseitiges Überschreiben der Daten verhindern.

### **Anwendung**

Die synchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Die Steuerung ist der "aktive Partner" bei der Übertragung von Datensätzen.
- In der Steuerung werden Informationen über die Rezepturnummer und die Datensatznummer ausgewertet.
- Die Übertragung von Datensätzen wird per Steuerungsauftrag getriggert.

### Voraussetzungen

Damit Datensätze synchronisiert zwischen Bediengerät und Steuerung übertragen werden, müssen beim Projektieren folgende Voraussetzungen erfüllt sein:

- Ein Bereichszeiger ist eingerichtet: Editor "Kommunikation > Verbindungen" unter "Bereichszeiger".
- In der Rezeptur ist die Steuerung angegeben, mit der das Bediengerät die Übertragung der Datensätze synchronisiert:  
Editor "Rezepturen" im Inspektorfenster unter "Allgemein > Synchronisation > Einstellungen" die Auswahl "Koordinierte Übertragung der Datensätze".

### Aufbau des Datenbereichs

Der Datenbereich hat eine feste Länge von 5 Worten. Der Datenbereich ist wie folgt aufgebaut:

|         |                                       |  |          |
|---------|---------------------------------------|--|----------|
|         | <b>15</b>                             |  | <b>0</b> |
| 1. Wort | Aktuelle Rezepturnummer (1 - 999)     |  |          |
| 2. Wort | Aktuelle Datensatznummer (0 - 65.535) |  |          |
| 3. Wort | Reserviert                            |  |          |
| 4. Wort | Status (0, 2, 4, 12)                  |  |          |
| 5. Wort | Reserviert                            |  |          |

- Status  
Das Statuswort (Wort 4) kann folgende Werte annehmen:

| Wert    |           | Bedeutung                            |
|---------|-----------|--------------------------------------|
| Dezimal | Binär     |                                      |
| 0       | 0000 0000 | Übertragung zulässig, Datenfach frei |
| 2       | 0000 0010 | Übertragung läuft.                   |
| 4       | 0000 0100 | Übertragung fehlerfrei beendet       |
| 12      | 0000 1100 | Übertragung mit Fehler beendet       |

### Kurven

### Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt bei den Basic Panels zeitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 3201)

## Zeitgetriggerte Kurven

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein.

Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

## Meldungen

### Meldungen projektieren

### Meldungen projektieren

Um Meldungen wie Betriebs-, Störmeldungen und Quittierung zu projektieren sind mehrere Schritte notwendig.

- Schritt 1: Variablen anlegen
- Schritt 2: Meldungen projektieren
- Schritt 3: Quittierung projektieren

Nähere Hinweise hierzu finden Sie im Kapitel:

Arbeiten mit Meldungen (Seite 3213)

### Besonderheiten bei der Projektierung von Meldungen

Wenn Sie Verbindungen von Bediengeräten mit Steuerungen anderer Hersteller projektieren, beachten Sie folgende Besonderheiten bei der Projektierung:

- Datentypen der Variablen
- Adressierung der Variablen
- Zählweise der Bitstellen

## Datentypen

Für Verbindungen mit SIMATIC Kommunikationstreiber, werden folgende Datentypen unterstützt:

| Steuerung              | Zulässige Datentypen |   |
|------------------------|----------------------|---|
|                        | Bitmeldungen         | Analogmeldungen                                 |
| SIMATIC S7-Steuerungen | WORD, INT            | BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER |

### Zählweise der Bitstellen

Für Verbindungen mit SIMATIC Kommunikationstreiber, gilt folgende Zählweise:

| Zählweise der Bitstellen   | Byte 0             |  |  |  |  |  |  |   | Byte 1              |  |  |  |  |  |  |  |   |
|----------------------------|--------------------|--|--|--|--|--|--|---|---------------------|--|--|--|--|--|--|--|---|
|                            | Höherwertiges Byte |  |  |  |  |  |  |   | Niederwertiges Byte |  |  |  |  |  |  |  |   |
| In SIMATIC S7-Steuerungen  | 7                  |  |  |  |  |  |  | 0 | 7                   |  |  |  |  |  |  |  | 0 |
| Im WinCC projektieren Sie: | 15                 |  |  |  |  |  |  | 8 | 7                   |  |  |  |  |  |  |  | 0 |

### Quittierung von Meldungen

#### Vorgehensweise

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

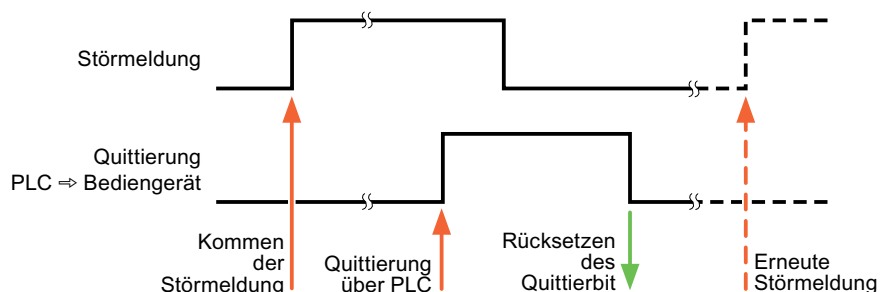
#### Quittieren von der Steuerung

Unter "PLC-Quittiervariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulsdiagramm.





## Quittierung am Bediengerät

Unter "HMI-Quittiervariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittiervariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittiervariable in die Steuerung eine gewisse Zeitspanne.

Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittiervariablen gesetzt. Die gesamte Quittiervariable wird dann vom Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

---

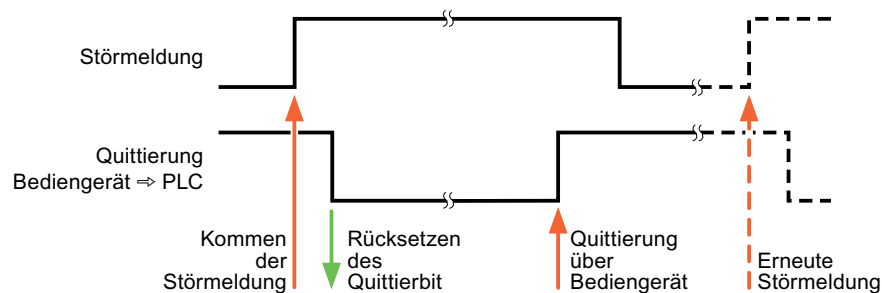
### Hinweis

Alle seit dem letzten Runtime-Start quittierten Meldebites bleiben in der Quittiervariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittiervariablen der gesamte Bereich vom Bediengerät überschrieben wird.

---

Das nachfolgende Bild zeigt das Impulsdiagramm.



### 10.8.9.5 Leistungsmerkmale der Kommunikation

#### Zulässige Datentypen für SIMATIC S7 200

#### Zulässige Datentypen für Verbindungen mit SIMATIC S7 200

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp   | Länge  |
|------------|--------|
| Bool       | 1 Bit  |
| Byte       | 1 Byte |
| Char       | 1 Byte |
| Word       | 2 Byte |
| Int        | 2 Byte |
| DWord      | 4 Byte |
| DInt       | 4 Byte |
| Real       | 4 Byte |
| StringChar | --     |
| Timer      | 2 Byte |
| Array      | --     |

#### Hinweis

##### Verbindungsabbruch beim PPI-Netzwerk

Wenn Sie Arrays in der Projektierung verwenden, kann es bei einer Array-Größe von ca. 1000 Byte zur Unterbrechung der Verbindung kommen.

Verwenden Sie bei ihrer Projektierung kleinere Arrays.

### 10.8.10 Mit SIMATIC LOGO! kommunizieren

#### 10.8.10.1 Kommunikation mit SIMATIC LOGO!

##### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und der Steuerung SIMATIC LOGO! beschrieben.

Folgende Kommunikationskanäle können Sie für die Steuerung SIMATIC LOGO! projektieren:

- PROFINET
- Ethernet

### **HMI-Verbindung für die Kommunikation**

Verbindungen von Bediengerät und SIMATIC LOGO! projektieren Sie im Editor "Verbindungen" des Bediengeräts.

### **Datenaustausch**

Der Datenaustausch mit der Steuerung SIMATIC LOGO! ist über Variablen möglich.  
Ein Datenaustausch über Bereichszeiger ist nicht möglich.

## **10.8.10.2 Verbindung mit SIMATIC LOGO! anlegen**

### **Einleitung**

Eine Verbindung zu der Steuerung SIMATIC LOGO! projektieren Sie im Editor "Verbindungen" des Bediengeräts. Abhängig vom Bediengerät sind die Schnittstellen unterschiedlich benannt.

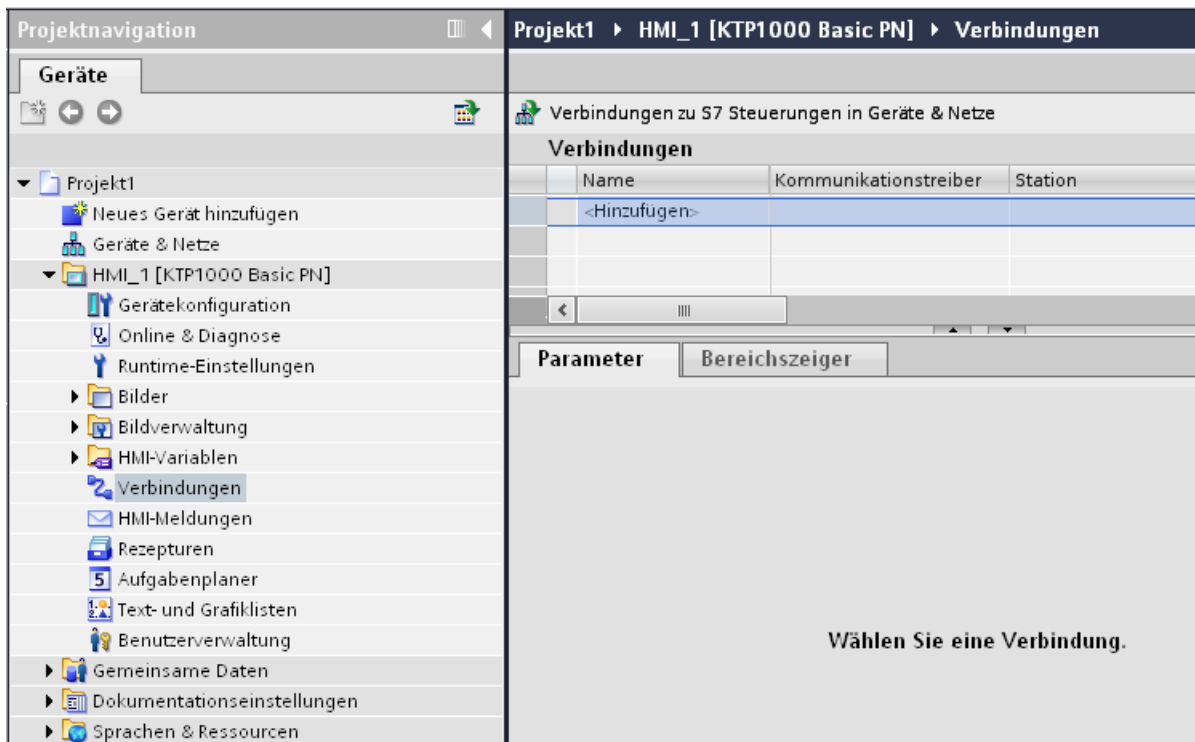
### **Voraussetzungen**

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

### **Vorgehensweise**

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".

3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".

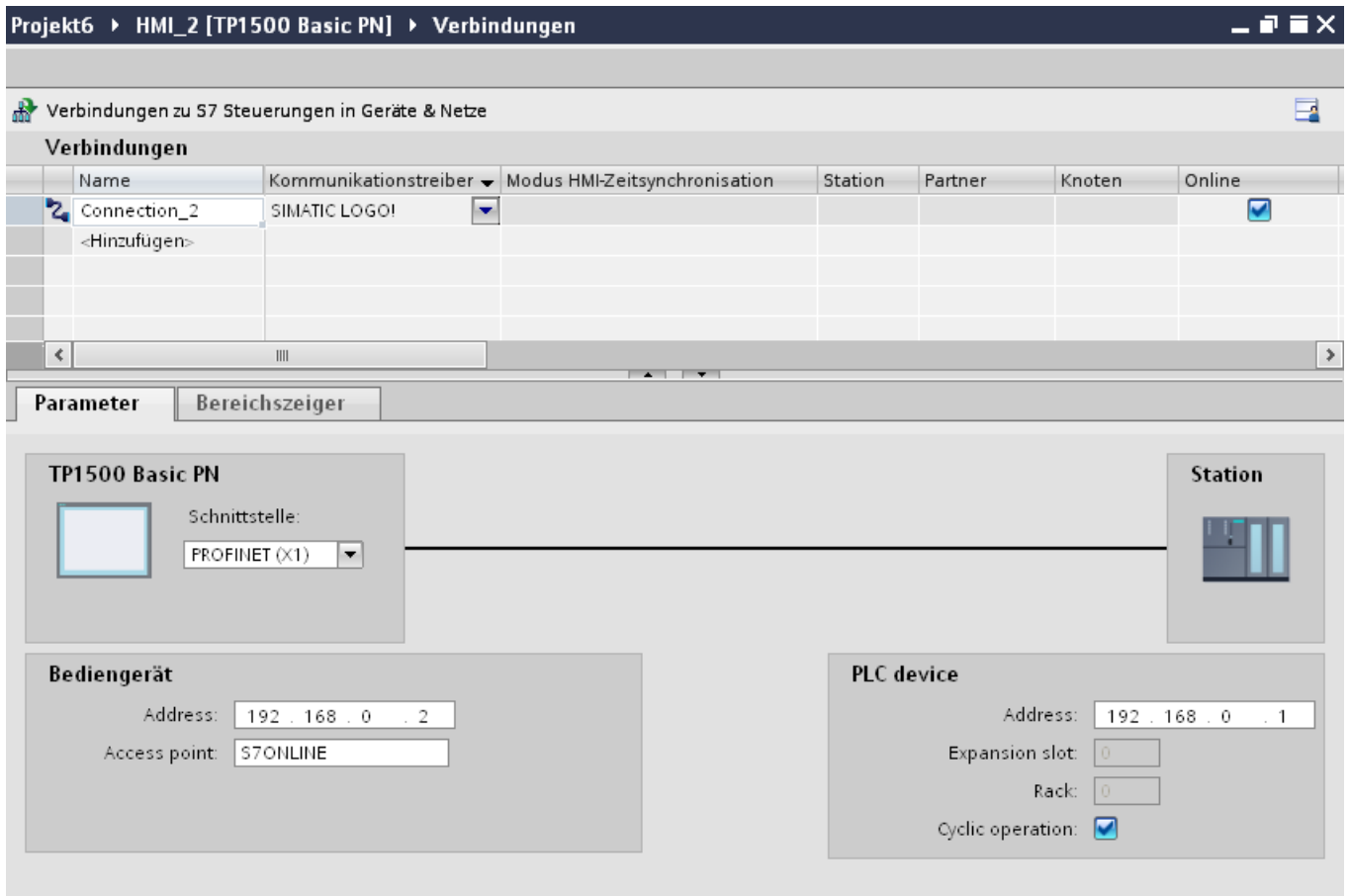


The screenshot displays the WinCC Basic interface for configuring connections. On the left, the 'Projektnavigation' pane shows a tree view where 'Verbindungen' is selected under the 'HMI\_1 [KTP1000 Basic PN]' project. The main editor area is titled 'Projekt1 > HMI\_1 [KTP1000 Basic PN] > Verbindungen'. It features a table for defining connections:

| Name         | Kommunikationstreiber | Station |
|--------------|-----------------------|---------|
| <Hinzufügen> |                       |         |

Below the table, there are two tabs: 'Parameter' and 'Bereichszeiger'. A large grey area at the bottom contains the instruction: 'Wählen Sie eine Verbindung.'

4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "SIMATIC LOGO!" aus.
5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.



Nähere Hinweise hierzu finden Sie im Kapitel "Auto-Hotspot".

### 10.8.10.3 Parameter für die Verbindung

#### Parameter für die Verbindung

##### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungsektor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät" und "Steuerung".

Projekt6 ▶ HMI\_2 [TP1500 Basic PN] ▶ Verbindungen

Verbindungen zu S7 Steuerungen in Geräte & Netze

**Verbindungen**


| Name         | Kommunikationstreiber | Modus HMI-Zeitsynchronisation | Station | Partner | Knoten | Online                              |
|--------------|-----------------------|-------------------------------|---------|---------|--------|-------------------------------------|
| Connection_2 | SIMATIC LOGO!         |                               |         |         |        | <input checked="" type="checkbox"/> |
| <Hinzufügen> |                       |                               |         |         |        |                                     |

Parameter | Bereichszeiger

**TP1500 Basic PN**

Schnittstelle: PROFINET (X1)

**Station**



**Bediengerät**

Address: 192 . 168 . 0 . 2

Access point: S7ONLINE

**PLC device**

Address: 192 . 168 . 0 . 1

Expansion slot: 0

Rack: 0

Cyclic operation:

## Ethernet-Parameter

### Parameter für das Bediengerät

Unter "Bediengerät" stellen Sie die Parameter für das Bediengerät im Netzwerk ein. Die Änderungen werden nicht automatisch auf das Bediengerät übertragen. Sie müssen die Einstellungen in der Systemsteuerung des Bediengeräts ändern.

- "Schnittstelle"  
Wenn Sie beim Projektieren direkt mit dem Bediengerät verbunden sind, können Sie die IP-Adresse des Bediengeräts in WinCC einrichten.

---

#### Hinweis

Wenn Sie die IP-Adresse bereits im Control-Panel des Bediengeräts eingerichtet haben, wird die IP-Adresse im Control Panel beim nächsten Laden überschrieben.

Wenn Sie "IP-Adresse auf anderem Weg beziehen" aktivieren, bleibt die bereits eingerichtete IP-Adresse im Control Panel beim nächsten Laden erhalten.

---

Beim Projekttransfer wird die IP-Adresse auf das Bediengerät übertragen. Die IP-Adresse des Bediengeräts richten Sie folgendermaßen ein:

- Klicken Sie auf das Bediengerät.
- Öffnen Sie den Editor "Gerätekonfiguration"
- Klicken Sie auf die Ethernet-Schnittstelle.
- Vergeben Sie im Inspektorfenster die IP-Adresse unter:  
"Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"
- "Adresse"  
Im Bereich "Adresse" vergeben Sie die IP-Adresse des Bediengeräts. Wenn Sie das WinCC-Projekt auf das Bediengerät transferieren, dann wird diese IP-Adresse direkt im Bediengerät eingerichtet.
- "Zugangspunkt"  
Der Zugangspunkt legt einen logischen Gerätenamen fest, über den der Kommunikationspartner erreicht wird.

### Parameter für die Steuerung

Unter "Steuerung" adressieren Sie die S7-Baugruppe, mit der das Bediengerät Daten austauscht. Vergeben Sie für jeden Kommunikationspartner einen Namen für die Verbindung.

- "Adresse"  
Unter "Adresse" stellen Sie die IP-Adresse der S7-Baugruppe ein, an die das Bediengerät angeschlossen ist.
- "Erweiterungssteckplatz"  
Legt die Nummer des Erweiterungssteckplatzes der zu adressierenden CPU fest.

- "Baugruppenträger"  
Legt die Baugruppenträger-Nummer der zu adressierenden CPU fest.
- "Zyklischer Betrieb"

---

**Hinweis**

Die Einstellung "Zyklischer Betrieb" ist bei der Steuerung SIMATIC S7 1200 nicht projektierbar.

---

Wenn Sie den zyklischen Betrieb einschalten, optimiert die Steuerung die Datenübertragung zwischen Bediengerät und Steuerung. Damit wird eine bessere Performance erreicht.

Bei Parallelbetrieb mehrerer Bediengeräte schalten Sie den zyklischen Betrieb aus.

## Zyklischer Betrieb

### Umgang mit der Auswahl "Zyklischer Betrieb"

Wenn der "zyklische Betrieb" eingeschaltet ist, dann schickt das Bediengerät zu Beginn der Kommunikation an die Steuerung ein Telegramm mit der Information, dass bestimmte Variablen ständig benötigt werden.

Die Steuerung verschickt daraufhin die Daten immer im gleichen Zyklus. Das Bediengerät erspart sich dadurch die ständige Neuansforderung der Daten.

Wenn der zyklische Betrieb abgeschaltet ist, verschickt das Bediengerät für jede benötigte Information eine extra Anforderung.

Weitere Eigenschaften:

- Der zyklische Betrieb entlastet bei der Übertragung von Daten das Bediengerät. Zur Entlastung des Bediengeräts werden die vorhandenen Ressourcen der Steuerung genutzt.
- Die Steuerung unterstützt nur eine bestimmte Anzahl an zyklischen Diensten. Wenn die Steuerung keine Ressourcen mehr für die zyklischen Dienste hat, dann übernimmt das Bediengerät wieder den Vorgang.
- Wenn die Steuerung keinen zyklischen Betrieb unterstützt, dann übernimmt das Bediengerät die Bildung des Zyklus.
- Bildvariablen werden nicht in den zyklischen Betrieb eingebunden.
- Der zyklische Betrieb wird nur bei Neustart der Runtime eingerichtet.
- Wenn der zyklische Betrieb aktiviert ist, dann wird steuerungsabhängig mehr als ein Auftrag vom Bediengerät an die Steuerung übertragen.
- Wenn der zyklische Betrieb abgewählt ist, dann wird immer nur ein Auftrag vom Bediengerät an die Steuerung übertragen.



#### 10.8.10.4 Datenaustausch

### Kurven

#### Allgemeines zu Kurven

### Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt je nach Projektierung zeit- oder bitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Auto-Hotspot

---

#### Hinweis

Bei Basic Panels erfolgt das Auslesen des Werts zeitgetriggert.

---

### Zeitgetriggerte Kurven

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein. Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

### Bitgetriggerte Kurven

Durch Setzen eines Triggerbits in der Variablen Kurvenanforderung liest das Bediengerät entweder einen Kurvenwert oder einen gesamten Kurvenpuffer ein. Diese Einstellung wird in der Projektierung festgelegt. Bitgetriggerte Kurven werden in der Regel zur Darstellung sich schnell ändernder Werte verwendet. Ein Beispiel hierfür ist der Einspritzdruck bei der Fertigung von Kunststoffteilen.

Zum Auslösen bitgetriggelter Kurven müssen bei der Projektierung entsprechende externe Variablen im Editor "HMI-Variablen" angelegt und mit Kurvenbereichen verbunden werden. Über diese Kurvenbereiche kommunizieren dann Bediengerät und Steuerung miteinander.

Folgende Bereiche stehen für Kurven zur Verfügung:

- Kurvenanforderungsbereich
- Kurvenübertragungsbereich 1
- Kurvenübertragungsbereich 2 (nur bei Wechsellpuffer erforderlich)

## Kurvenanforderung und Kurvenübertragung

### Kurvenanforderungsbereich

Wenn am Bediengerät ein Bild mit einer oder mehreren Kurven aufgeschlagen wird, setzt das Bediengerät die zugehörigen Bits im Kurvenanforderungsbereich. Nach Abwahl des Bildes setzt das Bediengerät die entsprechenden Bits im Kurvenanforderungsbereich zurück.

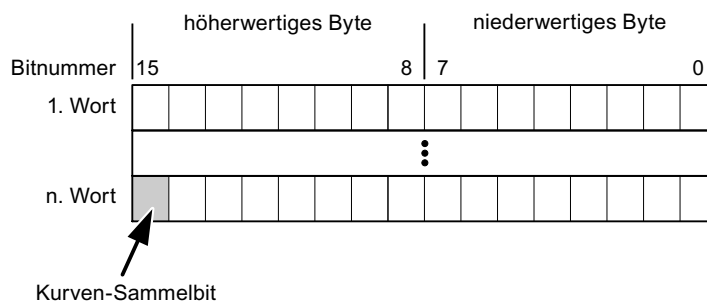
Über den Kurvenanforderungsbereich kann in der Steuerung ausgewertet werden, welche Kurve am Bediengerät gerade dargestellt wird. Kurven können auch ohne Auswertung des Kurvenanforderungsbereichs getriggert werden.

### Kurvenübertragungsbereich 1

Dieser Bereich dient zur Triggerung von Kurven. Sie müssen in Ihrem Steuerungsprogramm das der Kurve zugeordnete Bit im Kurvenübertragungsbereich sowie das Kurven-Sammelbit setzen. Das Kurven-Sammelbit ist das letzte Bit im Kurvenübertragungsbereich.

Das Bediengerät erkennt die Triggerung. Das Bediengerät liest entweder einen Wert oder den gesamten Puffer aus der Steuerung. Danach setzt es das Kurvenbit und das Kurven-Sammelbit zurück.

Das folgende Bild zeigt den Aufbau eines Kurvenübertragungsbereichs.



Solange das Kurven-Sammelbit nicht zurückgesetzt wurde, darf der Kurvenübertragungsbereich nicht durch das Steuerungsprogramm verändert werden.

### Kurvenübertragungsbereich 2

Der Kurvenübertragungsbereich 2 ist für Kurven erforderlich, die mit Wechselpuffer projektiert werden. Der Kurvenübertragungsbereich 2 ist genauso aufgebaut wie der Kurvenübertragungsbereich 1.

### Wechselpuffer

Der Wechselpuffer ist ein zweiter Puffer für dieselbe Kurve, der bei der Projektierung eingerichtet werden kann.

Während das Bediengerät die Werte aus dem Puffer 1 liest, schreibt die Steuerung in den Puffer 2. Während das Bediengerät den Puffer 2 liest, schreibt die Steuerung in den Puffer 1. Dadurch wird verhindert, dass während des Auslesens der Kurve durch das Bediengerät die Kurvenwerte von der Steuerung überschrieben werden.

## Zulässige Datentypen für Kurven

### Für SIMATIC S7

In der Projektierung ordnen Sie jeder Kurve ein Bit zu. Zulässig sind Variablen vom Datentyp "Word" oder "Int" und Arrayvariablen vom Datentyp "Word" oder "Int".

## Meldungen

### Meldungen projektieren

### Meldungen projektieren

Um Meldungen wie Betriebs-, Störmeldungen und Quittierung zu projektieren sind mehrere Schritte notwendig.

- Schritt 1: Variablen anlegen
- Schritt 2: Meldungen projektieren
- Schritt 3: Quittierung projektieren

Nähere Hinweise hierzu finden Sie im Kapitel:

Auto-Hotspot

### Besonderheiten bei der Projektierung von Meldungen

Wenn Sie Verbindungen von Bediengeräten mit Steuerungen anderer Hersteller projektieren, beachten Sie folgende Besonderheiten bei der Projektierung:

- Datentypen der Variablen
- Adressierung der Variablen
- Zählweise der Bitstellen

## Datentypen

Für Verbindungen mit SIMATIC Kommunikationstreiber, werden folgende Datentypen unterstützt:

| Steuerung              | Zulässige Datentypen |   |
|------------------------|----------------------|---|
|                        | Bitmeldungen         | Analogmeldungen                                 |
| SIMATIC S7-Steuerungen | WORD, INT            | BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER |

### Zählweise der Bitstellen

Für Verbindungen mit SIMATIC Kommunikationstreiber, gilt folgende Zählweise:

| Zählweise der Bitstellen   | Byte 0             |  |  |  |  |  |  |   | Byte 1              |  |  |  |  |  |  |  |   |
|----------------------------|--------------------|--|--|--|--|--|--|---|---------------------|--|--|--|--|--|--|--|---|
|                            | Höherwertiges Byte |  |  |  |  |  |  |   | Niederwertiges Byte |  |  |  |  |  |  |  |   |
| In SIMATIC S7-Steuerungen  | 7                  |  |  |  |  |  |  | 0 | 7                   |  |  |  |  |  |  |  | 0 |
| Im WinCC projektieren Sie: | 15                 |  |  |  |  |  |  | 8 | 7                   |  |  |  |  |  |  |  | 0 |

### Quittierung von Meldungen

#### Vorgehensweise

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

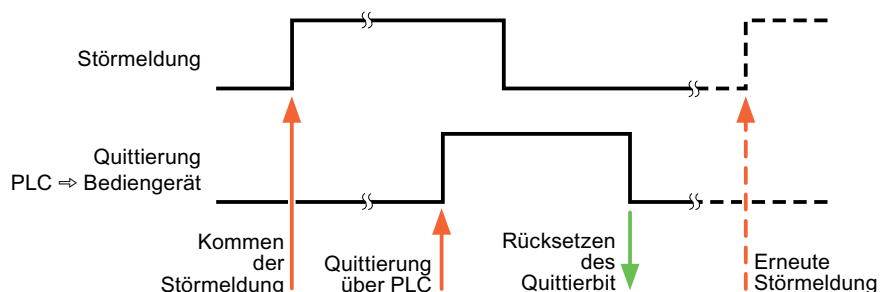
#### Quittieren von der Steuerung

Unter "PLC-Quittiervariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulsdiagramm.



## Quittierung am Bediengerät

Unter "HMI-Quittiervariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittiervariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittiervariable in die Steuerung eine gewisse Zeitspanne.

Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittiervariablen gesetzt. Die gesamte Quittiervariable wird dann vom Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

---

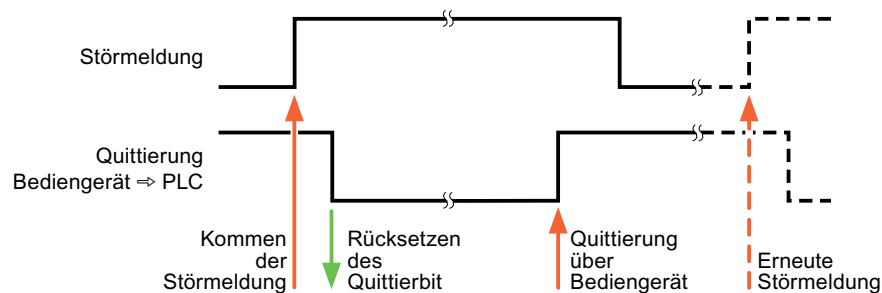
### Hinweis

Alle seit dem letzten Runtime-Start quittierten Meldebites bleiben in der Quittiervariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittiervariablen der gesamte Bereich vom Bediengerät überschrieben wird.

---

Das nachfolgende Bild zeigt das Impulsdiagramm.



### 10.8.10.5 Leistungsmerkmale der Kommunikation

#### Zulässige Datentypen für SIMATIC LOGO!

#### Zulässige Datentypen für Verbindungen mit SIMATIC LOGO!

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp | Länge  |
|----------|--------|
| Bool     | 1 Bit  |
| Byte     | 1 Byte |
| Int      | 2 Byte |
| DInt     | 4 Byte |
| Word     | 2 Byte |
| DWord    | 4 Byte |
| Array    | --     |

### 10.8.11 Kommunikation mit anderen Steuerungen

#### 10.8.11.1 Kommunikation mit anderen Steuerungen

##### Einleitung

Kommunikation mit anderen Steuerungen ist die Kommunikation mit Steuerungen, die nicht zur SIMATIC-Familie gehören.

Diese Steuerungen verfügen über firmeninterne Protokolle zum Datenaustausch. Die Protokolle werden in WinCC als Kommunikationstreiber projiziert.

##### Kommunikationstreiber

Folgende Kommunikationstreiber werden in WinCC unterstützt und sind bereits installiert:

- Allen-Bradley
  - Allen-Bradley EtherNet/IP
  - Allen-Bradley DF1
- Mitsubishi
  - Mitsubishi MC TCP/IP
  - Mitsubishi FX

- Modicon Modbus
  - Modicon Modbus TCP/IP
  - Modicon Modbus RTU
- Omron
  - Omron Host Link

### **Kommunikationstreiber in WinCC RT Professional**

Folgende Kommunikationstreiber werden für RT Professional unterstützt:

- Allen-Bradley
  - Allen-Bradley EtherNet/IP
- Mitsubishi
  - Mitsubishi MC TCP/IP
- Modicon Modbus
  - Modicon Modbus TCP

### **Verbindungen zwischen Bediengeräten und anderen Steuerungen**

Verbindungen zwischen Bediengeräten und anderen Steuerungen projektieren Sie im Editor "Verbindungen" des Bediengeräts. Diese Verbindungen gehören zu den nicht-integrierten Verbindungen.

#### **10.8.11.2 Besonderheiten bei der Projektierung**

##### **Besonderheiten beim Datenaustausch**

Bei der Projektierung von Verbindungen mit anderen Steuerungen bestehen Besonderheiten im Gegensatz zur Projektierung von integrierten Verbindungen.

Beachten Sie bei der Projektierung folgende Besonderheiten:

- Adressierung von Variablen
- Zulässige Datentypen
- Besonderheiten bei der Projektierung von Bereichszeigern
- Besonderheiten bei der Projektierung von Meldungen
- Besonderheiten bei der Projektierung von Kurven

Nähere Hinweise zu den Besonderheiten bei der Projektierung finden Sie im Kapitel "Datenaustausch" des jeweiligen Kommunikationstreibers.

### 10.8.11.3 Kommunikationstreiber

#### Allen-Bradley

#### Allen-Bradley Kommunikationstreiber

##### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und Steuerungen beschrieben, die Allen-Bradley Kommunikationstreiber verwenden.

Folgende Kommunikationstreiber werden unterstützt:

- Allen-Bradley EtherNet/IP
- Allen-Bradley DF1

##### Datenaustausch

Datenaustausch findet über Variablen oder Bereichszeiger statt.

- Variablen  
Die Steuerung und das Bediengerät tauschen ihre Daten über Prozesswerte aus. Legen Sie in der Projektierung Variablen an, die auf Adressen in der Steuerung zeigen. Das Bediengerät liest aus der angegebenen Adresse den Wert und zeigt ihn an. Genauso kann der Bediener eine Eingabe am Bediengerät vornehmen, die dann in die Adresse in der Steuerung geschrieben wird.
- Bereichszeiger  
Bereichszeiger dienen dem Austausch spezieller Daten und werden nur bei Verwendung dieser Daten eingerichtet.

#### Allen-Bradley EtherNet/IP

#### Verbindung über Allen-Bradley EtherNet/IP projektieren

##### Einleitung

Eine Verbindung zu einer der Steuerung mit Allen-Bradley EtherNet/IP Kommunikationstreiber, projektieren Sie im Editor "Verbindungen" des Bediengeräts.

Abhängig vom Bediengerät sind die Ethernet-Schnittstellen unterschiedlich benannt.

Beispiel: PROFINET-Schnittstelle entspricht der Ethernet-Schnittstelle

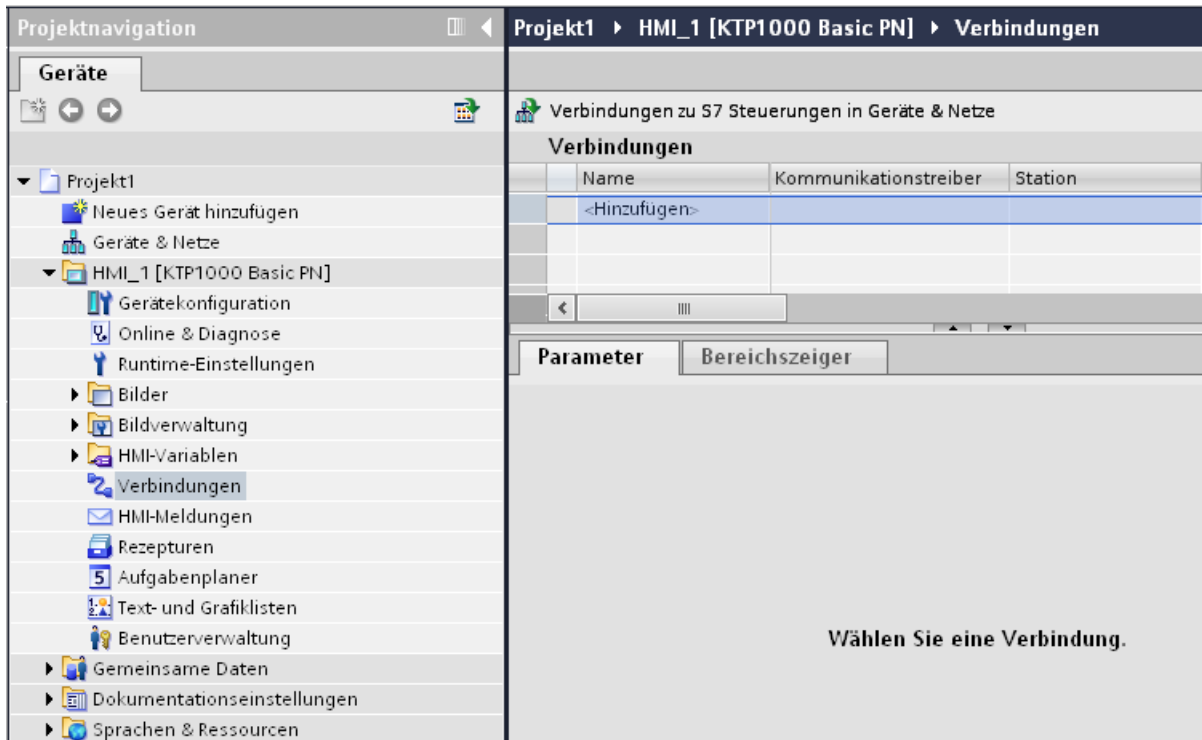
##### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

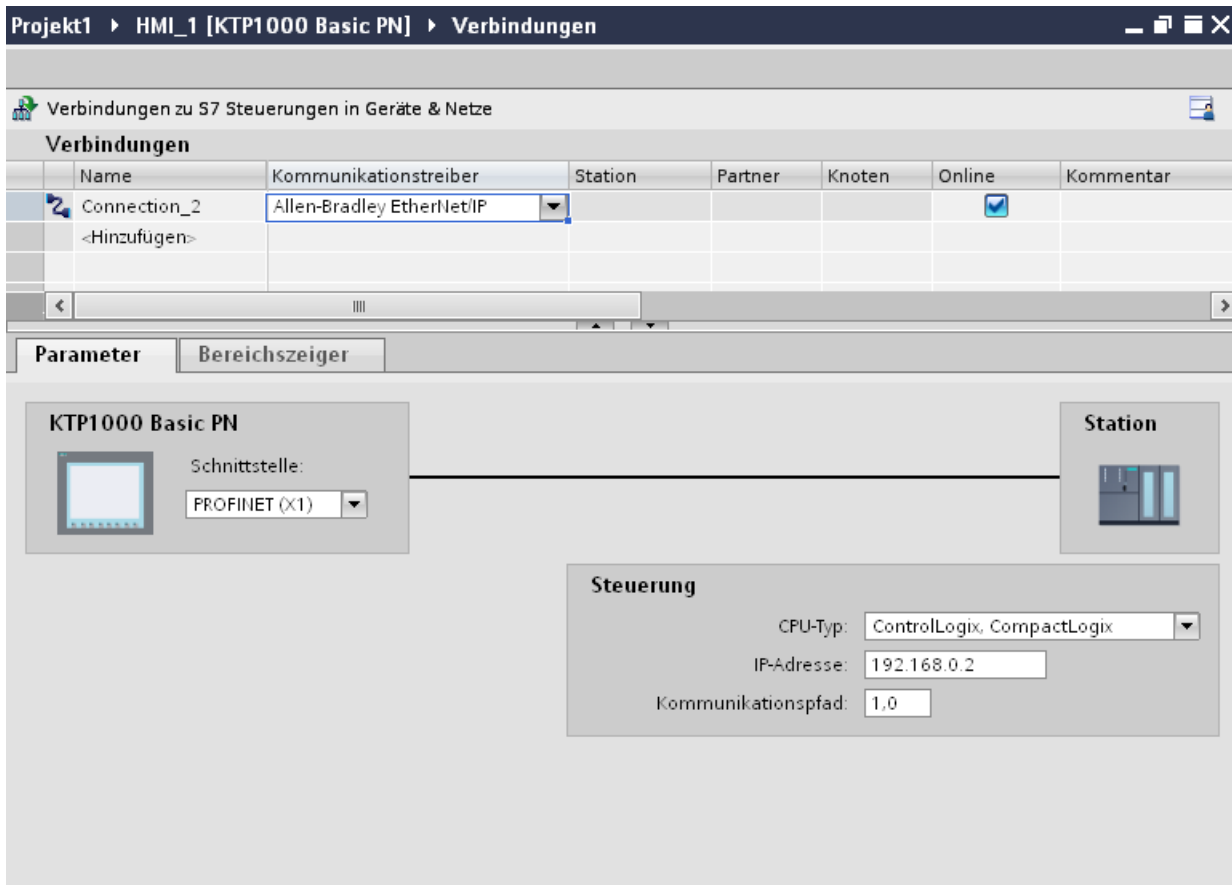


## Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".
3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".



4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "Allen-Bradley EtherNet/IP" aus.



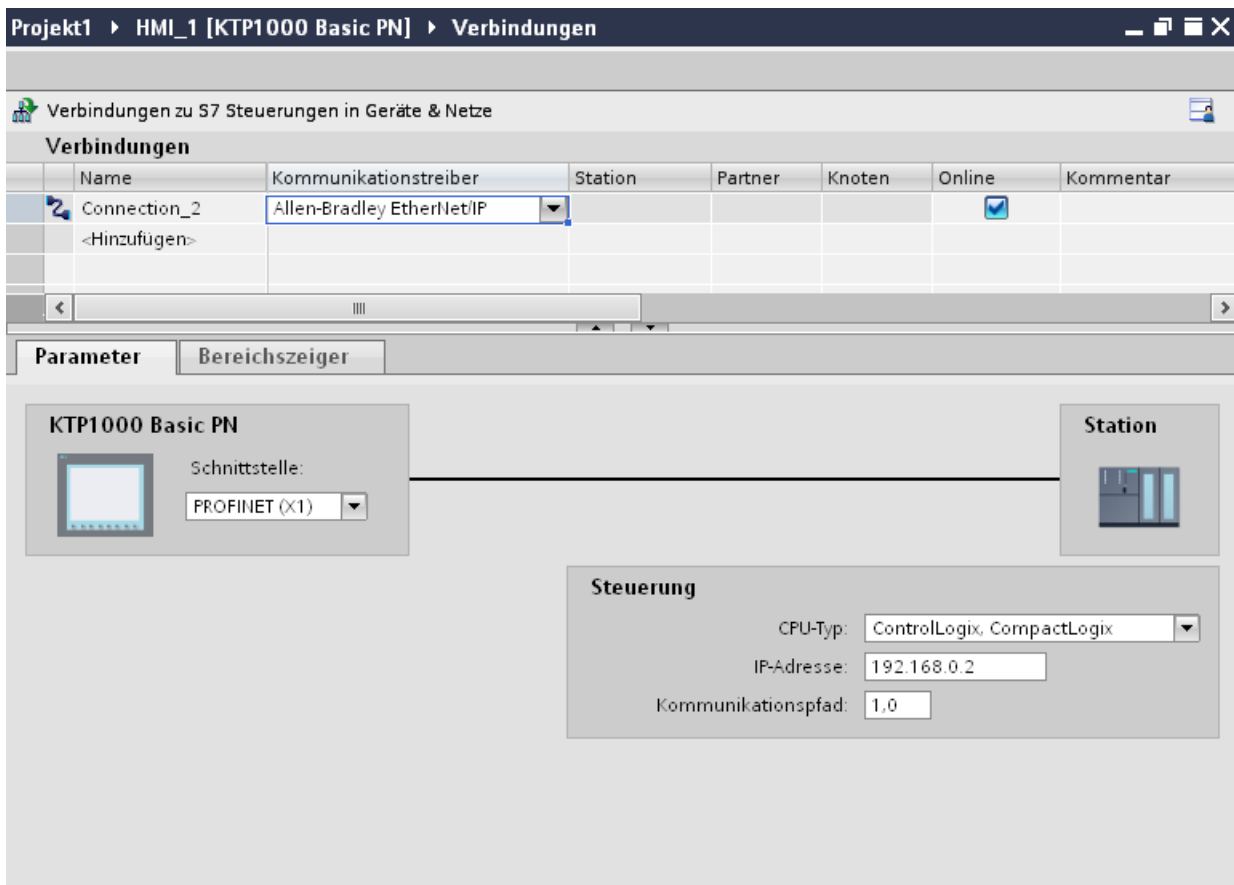
5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.

### Parameter für die Verbindung (Allen-Bradley EtherNet/IP)

#### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle den Bereich "Steuerung".



## Parameter für das Bediengerät

Sie können im Inspektorfenster "Parameter" für das Bediengerät nur eine Schnittstelle auswählen. Abhängig vom Bediengerät stehen auch mehrere Schnittstellen zur Verfügung.

Wenn Sie beim Projektieren direkt mit dem Bediengerät verbunden sind, können Sie die IP-Adresse des Bediengeräts in WinCC einrichten. Beim nächsten Laden wird die IP-Adresse auf das Bediengerät übertragen.

---

### Hinweis

Wenn Sie die IP-Adresse bereits im Control-Panel des Bediengeräts eingerichtet haben, wird die IP-Adresse im Control Panel beim nächsten Laden überschrieben.

Wenn Sie "IP-Adresse auf anderem Weg beziehen" aktivieren, bleibt die bereits eingerichtete IP-Adresse im Control Panel beim nächsten Laden erhalten.

---

Die IP-Adresse des Bediengeräts richten Sie folgendermaßen ein:

1. Klicken Sie auf das Bediengerät.
2. Öffnen Sie den Editor "Gerätekonfiguration"

3. Klicken Sie auf die Ethernet-Schnittstelle.
4. Vergeben Sie im Inspektorfenster die IP-Adresse unter:  
"Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"

### Parameter für die Steuerung

- CPU-Typ  
Unter "CPU-Typ" stellen Sie den CPU-Typ der verwendeten Steuerung ein.
- IP-Adresse  
Stellen Sie die IP-Adresse oder den Host-Namen des Ethernet/IP Moduls der Steuerung ein. Auf einem Basic Panel kann nur die IP-Adresse verwendet werden.
- Kommunikationspfad  
Stellen Sie den CIP-Pfad vom Ethernet-Modul zur Steuerung ein. Sie stellen hiermit eine logische Verbindung zwischen Ethernet-Modul und SPS her, auch wenn beide sich in verschiedenen CIP-Netzen befinden.  
Nähere Information hierzu finden Sie im Kapitel: Beispiele: Kommunikationspfad

### Bediengerät mit Steuerung verbinden

#### Verbindungen über Allen-Bradley EtherNet/IP

#### Anschluss

Das Bediengerät kann über folgende Komponenten an die Allen-Bradley-Steuerung angebunden werden:

- Bestehendes Ethernet-Netz, in dem sich auch die Steuerungen befinden
- Gekreuztes Ethernet-Kabel (Cross-Over), direkt an die Ethernetschnittstelle der CPU bzw. des Kommunikationsmoduls

Die Kopplung des Bediengeräts an eine Allen-Bradley-Steuerung beschränkt sich hauptsächlich auf den physikalischen Anschluss des Bediengeräts. Spezielle Bausteine für die Kopplung sind in der Steuerung nicht erforderlich.

#### Kommunikationsarten

#### Freigegebene Kommunikationsarten mit Allen-Bradley EtherNet/IP

Folgende Kommunikationsarten sind systemgetestet und freigegeben:

- Punkt-zu-Punkt-Kopplung zu den freigegebenen Steuerungen
- Mehrpunkt-Kopplung von einem Bediengerät (Allen-Bradley Ethernet/IP-Client) mit bis zu 4 Steuerungen, mit den jeweils freigegebenen Steuerungen. Das Mischen der CPU-Typen ist möglich.

## Kopplung

Die Kopplung zu folgenden Steuerungen ist mit Allen-Bradley EtherNet/IP freigegeben:

- CPU-Typ: "ControlLogix, Compact Logix"
  - ControlLogix  
556x(1756-L6x) mit Ethernet-Modul 1756-ENBT
  - Guard Logix-System ControlLogix  
556xS(1756-L6xS) mit Ethernet-Modul 1756-ENBT
  - CompactLogix
  - 533xE(1769-L3xE) mit Ethernet-Schnittstelle on Board
  - 532xE(1769-L2xE) mit Ethernet-Schnittstelle on Board
  - 534x (1768-L4x) mit Ethernet-Modul 1768-ENBT
- CPU-Typ: "SLC, MicroLogix"
  - MicroLogix 1100 (mit Ethernet-Schnittstelle on Board)
  - MicroLogix 1400 (mit Ethernet-Schnittstelle on Board)
  - SLC 5/05 (mit Ethernet-Schnittstelle on Board)

## Leistungsmerkmale der Kommunikation

### Zulässige Datentypen für Allen-Bradley EtherNet/IP

#### Zulässige Datentypen

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeichern verwendet werden können.

#### CPU-Typ: ControlLogix, CompactLogix

| Datentyp | Länge            |
|----------|------------------|
| Bool     | 1 Bit            |
| DInt     | 4 Byte           |
| Int      | 2 Byte           |
| Real     | 4 Byte           |
| SInt     | 1 Byte           |
| String   | 1 bis 80 Zeichen |
| UDInt    | 4 Byte           |
| UInt     | 2 Byte           |
| USInt    | 1 Byte           |

#### Zulässige Datentypen Arrays

| Adresse   | Zulässige Datentypen                      |
|---|---|
| Array   | SInt, USInt, Int, UInt, DInt, UDInt, Real |
| einzelne Bits aus den Basisdatentypen der Steuerung SInt, USInt, Int, UInt, DInt, UDInt | Bool*                                     |

\* Bei einzelnen Bits wird nach der Änderung des angegebenen Bits der gesamte Wert wieder in die Steuerung zurückgeschrieben. Dabei wird nicht geprüft, ob sich zwischenzeitlich andere Bits im Wert geändert haben. Deshalb darf die Steuerung (oder andere) auf den Wert nur lesend zugreifen.

### CPU-Typ: SLC, MicroLogix

| Datentyp | Operandentyp           | Länge            |
|----------|------------------------|------------------|
| ASCII    | A                      | 0 bis 80 Zeichen |
| Bool     | N, R, C, T, B, S, I, O | 1 Bit            |
| DInt     | N                      | 4 Byte           |
| Int      | N, R, C, T, S          | 2 Byte           |
| Real     | N, F                   | 4 Byte           |
| String   | ST                     | 1 bis 80 Zeichen |
| UDInt    | N                      | 4 Byte           |
| UInt     | N, R, C, T, B, I, O    | 2 Byte           |

#### Hinweis

Die Standardlänge eines Strings in RSLogix 5000 sind 82 Zeichen. In WinCC sind maximal 80 Zeichen darstellbar. Verwenden Sie nur Strings, die eine Maximallänge von 80 Zeichen nicht überschreiten.

#### Zulässige Datentypen Arrays

| Adresse | Zulässige Datentypen         |
|---------|------------------------------|
| Array   | Int, UInt, DInt, UDInt, Real |

## **Besonderheiten bei Kopplung mit Allen Bradley Ethernet/IP**

Mit dem Kommunikationstreiber Allen Bradley Ethernet/IP und dem CPU-Typ SLC, MicroLogix dürfen Sie Arrayvariablen nur für Bitmeldungen und Kurven verwenden.

---

### **Hinweis**

Ein-/Ausgangsmodule mit 8 oder 16 Ports belegen ein ganzes Wort in der Steuerung.

Ein-/Ausgangsmodule mit 24 oder 32 Ports belegen zwei Wörter.

Wenn am Bediengerät nicht vorhandene Bits belegt sind, so gibt das Bediengerät keine Fehlermeldung aus.

Achten Sie daher bei der Projektierung darauf, dass bei den Ein-/Ausgangsmodulen mit 8 oder 24 Ports nur die Bits belegt werden, die auch einem Port zugeordnet sind.

---

## **Unterstützte CPU-Typen für Allen-Bradley EtherNet/IP**

### **CPU-Typen**

Folgende CPU-Typen werden bei der Projektierung des Kommunikationstreibers Allen-Bradley EtherNet/IP unterstützt.

- CompactLogix
  - 1769-L2xE mit Ethernet-Schnittstelle on Board
  - 1769-L3xE mit Ethernet-Schnittstelle on Board
  - 1768-L4x mit Ethernet-Modul 1768-ENBT
- ControlLogix
  - 1756-L6x mit Ethernet-Modul 1756-ENBT
- GuardLogix
  - 1756-L61S mit Ethernet-Modul 1756-ENBT
  - 1756-L62S mit Ethernet-Modul 1756-ENBT
  - 1756-L63S mit Ethernet-Modul 1756-ENBT
- MicroLogix
  - MicroLogix 1100 / 1400
- SLC50x
  - SLC5/05

## Adressierung beim CPU-Typ C.Logix

### Adressierung

### Adressierung

Eine Variable wird in WinCC durch eine Adresse in der Steuerung eindeutig referenziert. Die Adresse muss dabei dem Variablennamen in der Steuerung entsprechen. Die Adresse einer Variable wird durch eine Folge von maximal 128 Zeichen angegeben.

### Verwendung von Zeichen für Adressierung

Die zulässigen Zeichen für die Adresse einer Variable sind:

- Buchstaben (a-z, A-Z)
- Zahlen (0-9)
- Unterstrich (\_)

Die Adresse einer Variable besteht aus einem Variablennamen und weiteren Zeichenfolgen zur genaueren Spezifizierung der Variablen in der Steuerung.

Ein Variablenname muss folgende Eigenschaften haben:

- Der Variablenname kann mit einem Unterstrich beginnen, darf aber nicht mit einem Unterstrich abgeschlossen sein.
- Mehrere aufeinander folgende Unterstriche und Leerzeichen sind nicht zulässig.
- Die Adresse darf eine Maximallänge von 128 Zeichen nicht überschreiten.

---

#### Hinweis

Die Zeichen, die für die Adressierung von Variablen reserviert sind, dürfen nicht von Programm- und Variablennamen oder anderweitig in der Adresse verwendet werden.

---

Nachfolgend sind reservierte Zeichen aufgeführt:

| Reserviertes Zeichen | Funktion   |
|----------------------|--|
| .                    | Trennung zwischen Elementen                                |
| :                    | Angabe einer Programmvariable                              |
| ,                    | Trennung bei der Adressierung von mehrdimensionalen Arrays |
| /                    | Reserviert für die Bitadressierung.                        |
| [ ]                  | Adressierung von Arrayelementen oder Arrays                |

### Controller- und Programmvariablen

Über den Allen-Bradley EtherNet/IP Kommunikationstreiber können Controllervariablen (projektglobale Variablen) und/oder Programmvariablen (programmglobale Variablen) angesprochen werden.



Eine Programmvariable wird durch den Programmnamen in der Steuerung und dem eigentlichen Variablennamen voneinander durch einen Doppelpunkt getrennt deklariert. Controllervariablen hingegen werden einfach über ihren Namen angesprochen.

---

**Hinweis**

**Adressierfehler**

Wenn die Variablennamen und Datentypen nicht übereinstimmen, treten Adressierfehler auf.

Beachten Sie bei der Adressierung, dass der Variablenname in WinCC im Adressfeld mit dem Variablennamen in der Steuerung übereinstimmt. Stellen Sie sicher, dass Datentypen von Variablen in WinCC mit den Datentypen in der Steuerung übereinstimmen.

---

**Hinweis**

Modulspezifische Tags z. B. für Daten auf Eingangs- und Ausgangsmodulen können Sie nicht direkt adressieren. Verwenden Sie stattdessen in der Steuerung ein Alias-Tag.

Beispiel: Local:3:O.Data ist in WinCC nicht adressierbar.

Wenn für Local:3:O in der Steuerung der Alias "MyOut" definiert wird, können Sie mit WinCC über MyOut.Data adressieren.

---

## Syntax für Adressierung

### Schreibregeln für Adressierungen

In den folgenden Tabellen sind die Schreibweisen für die einzelnen Adressierungsmöglichkeiten bei Allen-Bradley EtherNet/IP definiert.

Tabelle 10-13 Zugriff auf Arrays, Basisdatentypen und Strukturelemente

| <b>Datentypen</b> | <b>Typ</b>         | <b>Adresse</b>                                |
|-------------------|--------------------|---|
| Basisdatentypen   | Controllervariable | Variablenname                                 |
|                   | Programmvariable   | Programmname:Variablenname                    |
| Arrays            | Controllervariable | Arrayvariable                                 |
|                   | Programmvariable   | Programmname:Arrayvariable                    |
| Bits              | Controllervariable | Variablenname/Bitnummer                       |
|                   | Programmvariable   | Programmname:Variablenname/Bitnummer          |
| Strukturelemente  | Controllervariable | Strukturvariable.Strukturelement              |
|                   | Programmvariable   | Programmname:Strukturvariable.Strukturelement |

---

**Hinweis**

Bitadressierungen bei den Datentypen Bool, Real und String sind nicht zulässig und haben Adressierungsfehler zur Folge.

---

## Syntaxbeschreibung

Nachfolgend eine zusammenfassende Syntaxbeschreibung.

```
(Programmname:) Variablenname ([x(, y) (, z)]) { .Variablenname ([x(, y) (, z)]) } (/bitNummer)
```

- Der Ausdruck "( )" bedeutet, dass der Ausdruck optional ist und einmal vorkommen kann.
- Der Ausdruck "{ }" bedeutet, dass der Ausdruck optional ist und beliebig oft vorkommen kann.

Die Anzahl der Zeichen einer Adresse darf die obere Grenze von 128 Zeichen nicht überschreiten.

## Adressierungstypen

### Arrays

Ein Array ist eine Datenstruktur, die eine Anzahl von Daten gleichen Typs beinhaltet. In WinCC sind nur eindimensionale Arrays anlegbar.

Tragen Sie in der Adress-Spalte des Variableneditors den Arraynamen eventuell mit Angabe eines Startelements ein. Die Länge wird über das Eingabefeld Array-Elemente im Variableneditor bestimmt. Wenn Arraygrenzen in der Steuerung überschritten werden (durch Fehlindizierung), entstehen Adressierfehler.

Diese Arrays müssen als Controller- oder Programmvariablen in der Steuerung deklariert sein.

Zwei- oder dreidimensionale Arrays in der Steuerung sind in WinCC nur adressierbar, wenn diese bereichsweise auf eindimensionale Arrays abgebildet werden können.

---

#### Hinweis

Bei allen Lese-Zugriffen und bei allen Schreibzugriffen werden stets alle Array-Elemente einer Variablen gelesen oder geschrieben. Wenn eine Arrayvariable mit einer Steuerung verbunden ist, werden bei einer Änderung grundsätzlich alle Inhalte übertragen. Deswegen können Sie nicht zur selben Zeit vom Bediengerät und von der Steuerung Werte in dieselbe Arrayvariable schreiben. Beim Schreiben eines Elements wird das komplette Array in die Steuerung geschrieben und nicht nur das betroffene Element.

---

### Arrayelemente

Elemente von ein-, zwei-, dreidimensionalen Arrays in der Steuerung werden durch einen Index im Variableneditor über die entsprechende Schreibweise indiziert. Die Adressierung eines Arrays beginnt mit dem Element 0. Bei der Adressierung von Elementen sind Arrays von sämtlichen Basistypen zulässig. Hierbei wird nur das angesprochene Element geschrieben bzw. gelesen und nicht das komplette Array.

### Bits und Bitvariablen

Der Zugriff auf einzelne Bits ist für alle Basisdatentypen außer Bool, Real und String zulässig. Die Adressierung von Bits ist auch bei Array- und Strukturelementen erlaubt. Bei der

Adressierung von Bits und Bitvariablen in Basisdatentypen wird in WinCC der Datentyp Bool eingestellt.

Einstellige Bitnummern werden mit "/x" oder "/0x" (x = bitNummer) adressiert. Bitnummern werden maximal zweistellig angegeben.

---

#### Hinweis

Beim Datentyp "Bool" in den Datentypen SInt, Int und DInt wird nach der Änderung des angegebenen Bits die komplette Variable wieder in die Steuerung zurückgeschrieben. Dabei wird nicht geprüft, ob zwischenzeitlich andere Bits der Variable geändert wurden. Deshalb darf die Steuerung auf die angegebene Variable nur lesend zugreifen.

---

## Strukturen

Um benutzerdefinierte Datentypen zu erstellen, werden Strukturen verwendet. In Strukturen werden Variablen mit verschiedenen Datentypen zusammengefasst. Strukturen können aus Basistypen, Arrays und anderen Strukturen bestehen. In WinCC werden nur Strukturelemente und nicht gesamte Strukturen adressiert.

## Strukturelemente

Strukturelemente werden durch den Namen der Struktur und den Namen des gewünschten Strukturelements adressiert. Die Schreibweise dieser Adressierung erfolgt voneinander getrennt durch einen Punkt. Strukturelemente können neben Basisdatentypen auch Arrays und wiederum andere Strukturen sein. Arrays als Strukturelement sind nur eindimensional zulässig.

---

#### Hinweis

Die Schachtelungstiefe von Strukturen wird nur durch die maximale Länge von 128 Zeichen für die Adresse beschränkt.

---

## Adress-Multiplexen

### Adress-Multiplexen

Mit dem CPU-Typ CompactLogix, ControlLogix ist Adress-Multiplexen möglich.

Für Adress-Multiplexen werden zwei Variablen benötigt:

- "Tag\_1" ist eine Variable vom Datentyp "String" und enthält als Wert eine logische Adresse z. B. "HMI:Robot5.Block5".  
Der Wert kann sich jedoch ändern in eine 2. gültige Adresse z. B. "HMI:Robot4.Block3".
- "Tag\_2" ist eine Variable, bei der als Verbindung der Kommunikationstreiber "Allen-Bradley EtherNet/IP" eingerichtet ist.  
Als Adresse geben Sie einen gültigen Namen einer HMI\_Variable in eckigen Klammern ein.
  - z. B: "[Tag\_1]"
  - Die Variable muss vom Datentyp String sein.
  - Die eckigen Klammern zeigen dann an, dass die Adresse gemultiplext wird.
  - Die Adresse ist der jeweils aktuelle Wert der Variablen "Tag\_1".

**Hinweis**

Sie können nur komplette Allen-Bradley EtherNet/IP-Adressen multiplexen und nicht Teile davon. "HMI:Robot[Tag\_1].Block5" ist eine ungültige Adresse.

Alternativ klicken Sie in der Spalte "Adresse" auf den Pfeil rechts. Wählen Sie im darauf folgenden Adress-Dialog über den Pfeil am linken Rand statt "Konstant" den Eintrag "Multiplex". In der Variablenauswahlliste stehen dann nur Variablen vom Datentyp "String" zur Auswahl.

Auch bei gemultiplexten Variablen können Sie an das Ereignis "Wertänderung" eine Funktion projektieren.

**Beispiele für Adressierung**

**Beispieltabelle für Adressierungen**

Die nachfolgende Tabelle zeigt grundlegende Adressierungsvarianten für Controllervariablen auf. Andere Adressierungsvarianten sind durch Kombinationen möglich.

| Typ       | Art  | Adresse  |
|-----------|--|--|
| Allgemein | Controllervariable                               | Variablenname  |
|           | Programmvariable                                 | Programm:Variablenname                                 |
| Array     | Zugriff auf Element eines 2-dimensionalen Arrays | Arrayvariable[Dim1,Dim2]                               |
|           | Element von Struktur-Array (1-dimensional)       | Arrayvariable[Dim1].Strukturelement                    |
|           | Bit in Element Basistyp-Array (2-dimensional)    | Arrayvariable[Dim1,Dim2]/Bit                           |
| Struktur  | Array in Struktur                                | Strukturvariable.Arrayvariable                         |
|           | Bit in Element eines Arrays in Unterstruktur     | Strukturvariable.Struktur2.Arrayvariable [Element]/Bit |

**Hinweis**

Wenn Sie Programmvariablen adressieren wollen, dann müssen Sie den Programmnamen in der Steuerung der Adresse, getrennt durch einen Doppelpunkt, voranstellen.

Beispiel: Programmname:Arrayvariable[Dim1,Dim2]

**Zugriff auf Arrayelemente**

| Typ                | Adresse                                    |
|--------------------|--|
| Controllervariable | Arrayvariable[Dim1]                        |
|                    | Arrayvariable[Dim1,Dim2]                   |
|                    | Arrayvariable[Dim1,Dim2,Dim3]              |
| Programmvariable   | Programmname:Arrayvariable[Dim1]           |
|                    | Programmname:Arrayvariable[Dim1,Dim2]      |
|                    | Programmname:Arrayvariable[Dim1,Dim2,Dim3] |

**Beispiele: Kommunikationspfad**

**Beispiel 1:**

Verbindung mit einer SPS, die sich in demselben Allen-Bradley-Baugruppenträger befindet.  
1,0

| Zahl | Bedeutung                            |
|------|--------------------------------------|
| 1    | Steht für eine Backplane-Verbindung. |
| 0    | Steht für die Slot Nummer der CPU.   |

**Beispiel 2:**

Verbindung mit einer SPS, die sich in anderen Allen-Bradley-Baugruppenträger befindet. Zwei Allen-Bradley-Baugruppenträger sind mit Ethernet verbunden.

1,2,2,190.130.3.101,1,5

| Zahl          | Bedeutung  |
|---------------|--|
| 1             | Backplane- Verbindung  |
| 2             | Steht für die Slot Nummer des zweiten Ethernet Moduls.   |
| 2             | Steht für eine Ethernet-Netz Verbindung.   |
| 190.130.3.101 | IP-Adresse eines anderen AB-Baugruppenträgers im Netz – insbesondere das dritte Ethernet Modul |

---

| Zahl | Bedeutung             |
|------|-----------------------|
| 1    | Backplane- Verbindung |
| 5    | Slot Nummer der CPU   |

---

## Komponenten in Betrieb nehmen

### Projekt auf das Bediengerät übertragen

1. Versetzen Sie das Bediengerät in den "Transfermodus".
2. Stellen Sie alle notwendigen Transferparameter ein:
  - Schnittstelle
  - Übertragungsparameter
  - Zielspeicher
3. Starten Sie die Übertragung des Projekts.  
Das Projekt wird automatisch generiert.  
In einem Fenster werden die einzelnen Schritte der Generierung und der Übertragung mitprotokolliert.

---

#### Hinweis

Wenn die Firmware der Steuerung CompactLogix kleiner als Version 18 ist, dann müssen Sie das Bediengerät unter Umständen nach einem Transfer des SPS-Programms neu starten.

Oder Sie trennen vor dem Transfer des SPS-Programms die Verbindung und stellen die Verbindung nach dem Transfer des SPS-Programms wieder her.

---

### Steuerung und Bediengerät miteinander verbinden

1. Verbinden Sie die Steuerung und das Bediengerät mit einem geeigneten Anschlusskabel.
2. Am Bediengerät erscheint die Meldung "Verbindung zur Steuerung .... ist hergestellt".

### Projektierung optimieren

#### Erfassungszyklus und Aktualisierungszeit

Die in der Projektierungssoftware angegebenen Erfassungszyklen für die "Bereichszeiger" und die Erfassungszyklen der Variablen sind wesentliche Faktoren für die tatsächlich erreichbaren Aktualisierungszeiten.

Die Aktualisierungszeit ist Erfassungszyklus plus Übertragungszeit plus Verarbeitungszeit.

Um optimale Aktualisierungszeiten zu erreichen, beachten Sie bei der Projektierung Folgendes:

- Richten Sie die einzelnen Datenbereiche so klein wie möglich und so groß wie nötig ein.
- Zu klein gewählte Erfassungszyklen beeinträchtigen unnötigerweise die Gesamtperformance. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs. Richtwert für den Erfassungszyklus ca. 1 Sekunde.
- Legen Sie die Variablen einer Meldung oder eines Bildes ohne Lücken in einen Datenbereich.
- Damit Änderungen in der Steuerung sicher erkannt werden, müssen diese mindestens während des tatsächlichen Erfassungszyklus anstehen.

## Bilder

Bei Bildern hängt die tatsächlich erreichbare Aktualisierungsrate von Art und Anzahl der darzustellenden Daten ab.

Beachten Sie bei der Projektierung, dass Sie nur für diejenigen Objekte kurze Erfassungszyklen projektieren, die tatsächlich schnell aktualisiert werden müssen. Damit werden die Aktualisierungszeiten kürzer.

## Kurven

Wenn bei bitgetriggerten Kurven das Sammelbit im "Kurvenübertragungsbereich" gesetzt wird, so aktualisiert das Bediengerät jedes Mal alle Kurven, deren Bit in diesem Bereich gesetzt ist. Danach setzt es die Bits wieder zurück.

Erst wenn vom Bediengerät alle Bits zurückgesetzt wurden, darf das Sammelbit im SPS-Programm wieder gesetzt werden.

## Steuerungsaufträge

Wenn viele Steuerungsaufträge in schneller Folge zum Bedienen gesendet werden, so kann dadurch die Kommunikation zwischen Bediengerät und Steuerung überlastet werden.

Wenn das Bediengerät den Wert 0 in das erste Datenwort des Auftragsfachs einträgt, hat das Bediengerät den Steuerungsauftrag entgegengenommen. Jetzt arbeitet das Bediengerät den Auftrag ab, wofür es noch Zeit benötigt. Wenn sofort wieder ein neuer Steuerungsauftrag in das Auftragsfach eingetragen wird, kann es eine Zeit dauern, bis das Bediengerät den nächsten Steuerungsauftrag ausführt. Der nächste Steuerungsauftrag wird erst wieder entgegengenommen, wenn Rechnerleistung zur Verfügung steht.

## Allen-Bradley DF1

### Verbindung über Allen-Bradley DF1 projektieren

#### Einleitung

Eine Verbindung zu einer der Steuerung mit Allen-Bradley DF1 Kommunikationstreiber projektieren Sie im Editor "Verbindungen" des Bediengeräts.

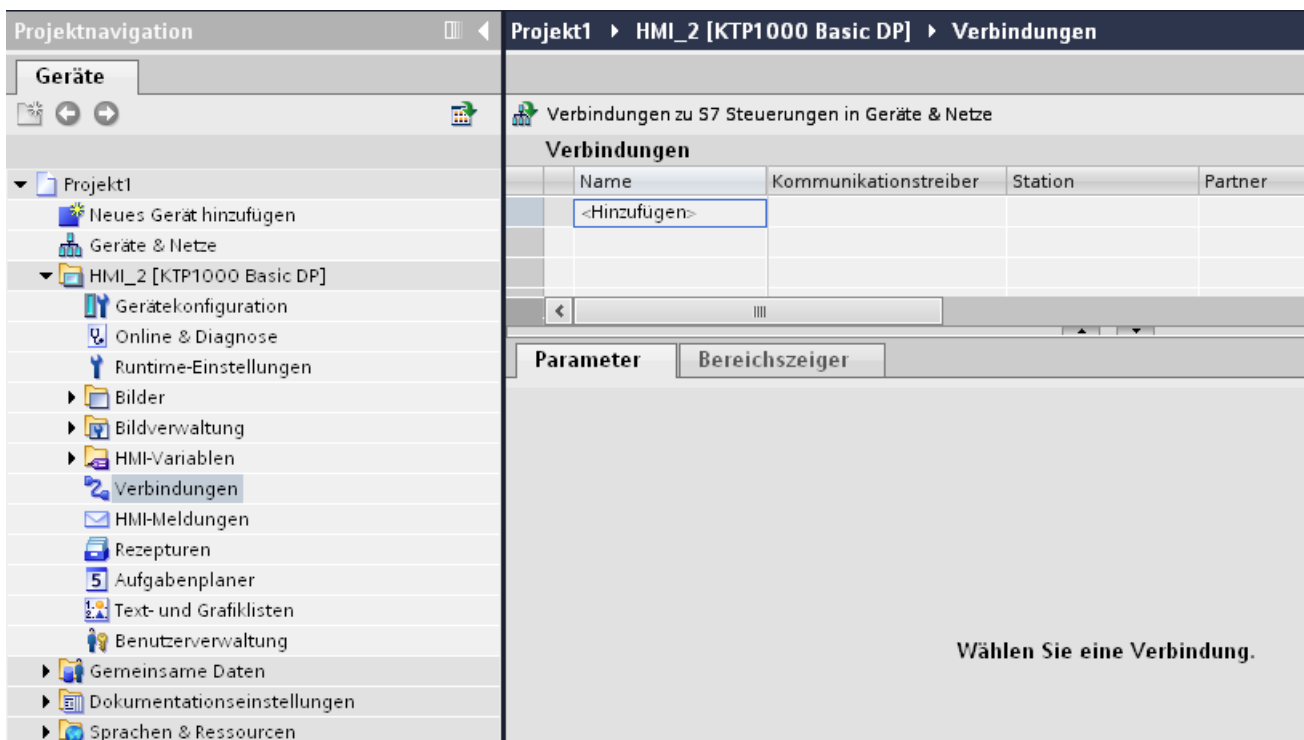
Abhängig vom Bediengerät sind die Schnittstellen unterschiedlich benannt.

#### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

#### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".
3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".





4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "Allen-Bradley DF1" aus.

| Name         | Kommunikationstreiber | Station | Partner | Knoten | Online                              | Kommentar |
|--------------|-----------------------|---------|---------|--------|-------------------------------------|-----------|
| Connection_2 | Allen-Bradley DF1     |         |         |        | <input checked="" type="checkbox"/> |           |
| <Hinzufügen> |                       |         |         |        |                                     |           |

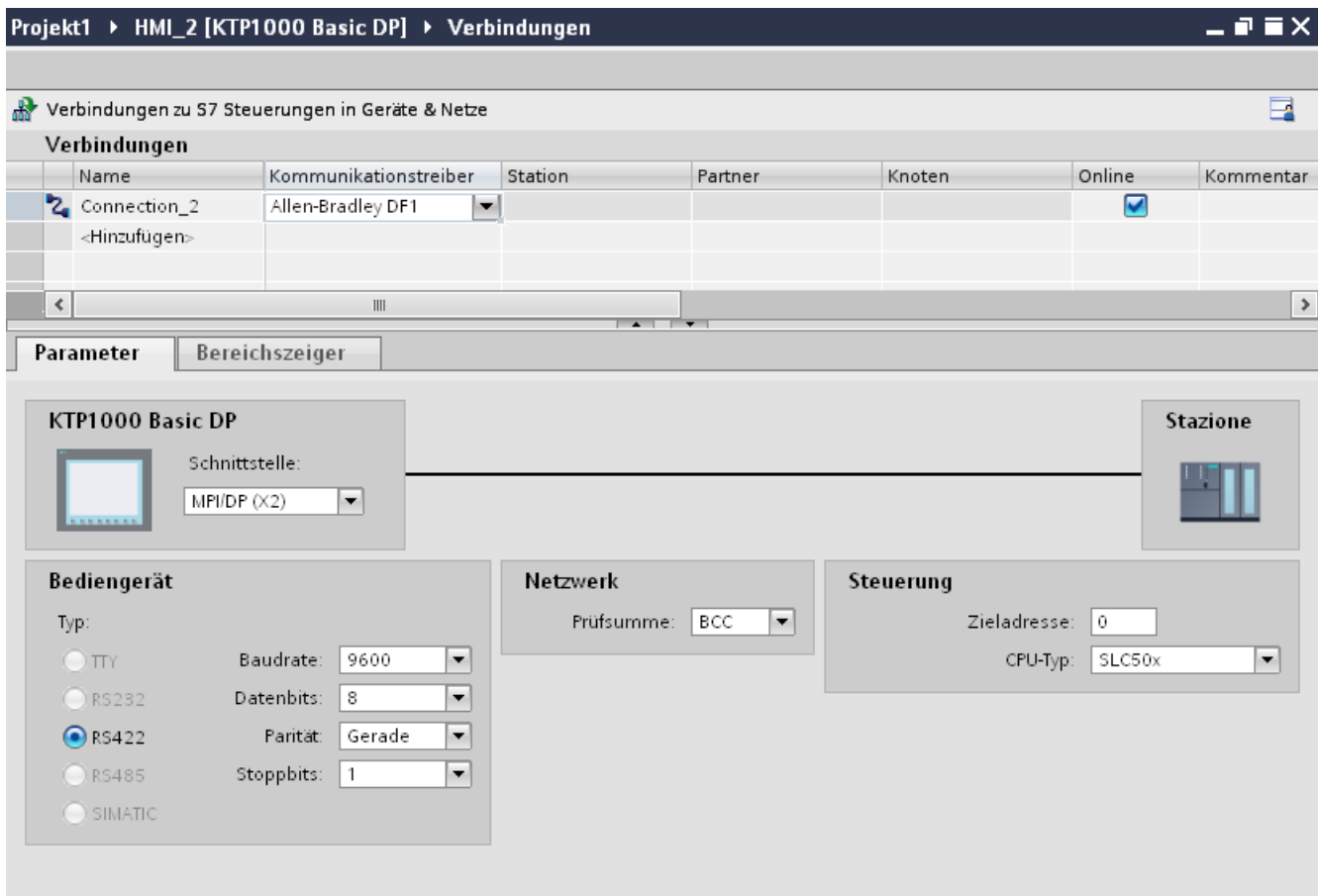
5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.

## Parameter für die Verbindung (Allen-Bradley DF1)

### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungsektor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät", "Netzwerk" und "Steuerung".



### Parameter für das Bediengerät

- Schnittstelle  
Unter "Schnittstelle" wählen Sie die Schnittstelle des Bediengeräts aus, an der die Steuerung angeschlossen ist.  
Weitere Informationen finden sie im Gerätehandbuch des Bediengeräts.
- Typ  
Legt den benutzten physikalischen Anschluss fest.

---

#### Hinweis

Wenn Sie die Schnittstelle IF1B verwenden, müssen Sie zusätzlich die RS-485-Empfangsdaten und das RTS-Signal über 4 DIL-Schalter an der Rückseite der Bediengeräte umschalten.

---

- Baudrate  
Unter "Baudrate" wählen Sie die Übertragungsgeschwindigkeit zwischen Bediengerät und Steuerung aus.
- Datenbits  
Unter "Datenbits" können Sie zwischen "7 Bit" oder "8 Bit" wählen.

- Parität  
Unter "Parität" können Sie zwischen "Keine", "Gerade" und "Ungerade" wählen.
- Stoppbits  
Unter "Stoppbits" können Sie zwischen 1 und 2 Bit wählen.

### Parameter für das Netzwerk

- Prüfsumme  
Unter "Prüfsumme" wählen Sie das Verfahren zur Ermittlung des Fehlercodes aus: "BCC" oder "CRC".

### Parameter für die Steuerung

- Zieladresse  
Unter "Zieladresse" wählen Sie die Steuerungsadresse. Bei einer Punkt-zu-Punkt-DF1-Kopplung stellen Sie die Adresse 0 ein.
- CPU-Typ  
Unter "CPU-Typ" stellen Sie den CPU-Typ der verwendeten Steuerung ein.

---

#### Hinweis

Parametrieren Sie den Treiber DF1 FULL-DUPLEX in der CPU wie folgt : "NO HANDSHAKING" bei "Control Line" und "AUTO-DETECT" bei "Embedded Responses".

---

## Bediengerät mit Steuerung verbinden

### Verbindungen über Allen-Bradley DF1

#### Anschluss

Wenn Sie die Schnittstellenparameter von Steuerung und Bediengerät aufeinander abgestimmt haben, wird die Verbindung aufgebaut. Spezielle Bausteine für die Kopplung sind in der Steuerung nicht erforderlich.

---

#### Hinweis

Die Firma Rockwell bietet eine Vielzahl von Kommunikationsadaptern zur Integration von "DF1-Teilnehmern" für die Netze DH485, DH und DH+ an. Von diesen Kopplungen sind die direkte Kopplung und die Kopplungen über das KF2- und KF3-Modul freigegeben. Alle anderen sind von der SIEMENS AG nicht systemgetestet und daher nicht freigegeben.

---

## Kommunikationspartner für Allen-Bradley DF1

### Koppelbare Steuerungen

Für folgende Allen-Bradley-Steuerungen stehen die aufgezeigten Kommunikationstreiber zur Verfügung: :

| Steuerung           | DF1<br>(Punkt-Punkt) | DF1<br>(Punkt-Punkt) | DF1<br>(Mehr-Punkt)<br>über KF2-Modul<br>an DH+ LAN<br>RS 232/RS 422 | DF1<br>(Mehr-Punkt)<br>über KF3-Modul<br>an DH485 LAN<br>RS 232 |
|---------------------|----------------------|----------------------|--|---|
|                     | RS 232               | RS 422               |  |   |
| SLC500              | –                    | –                    | –  | X   |
| SLC501              | –                    | –                    | –  | X   |
| SLC502              | –                    | –                    | –  | X   |
| SLC503              | X                    | –                    | –  | X   |
| SLC504              | X                    | –                    | X  | X   |
| SLC505              | X                    | –                    | –  | X   |
| MicroLogix          | X                    | –                    | –  | X   |
| PLC-5 <sup>1)</sup> | X                    | X                    | X  | –   |

<sup>1)</sup> Für PLC-5 sind nur folgende Prozessoren freigegeben: PLC-5/11, PLC-5/20, PLC-5/30, PLC-5/40, PLC-5/60 und PLC-5/80.

### Kommunikationsarten

#### Steuerungen mit Kommunikationstreiber Allen-Bradley DF1

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und folgenden Allen Bradley-Steuerungen beschrieben:

- SLC500
- SLC501
- SLC502
- SLC503
- SLC504
- SLC505
- PLC5
- MicroLogix

Bei diesen Steuerungen erfolgt die Kopplung über die steuerungseigenen Protokolle Allen Bradley DF1, Allen Bradley DH485 und Allen Bradley DH+.

Verwendung findet hier immer der Kommunikationstreiber Allen-Bradley DF1, dessen Protokoll bei Mehrpunkt-Kommunikation mit den Kommunikationsmodulen KF2 (Allen Bradley

DH+) und KF3(Allen Bradley DH485) in eines der anderen beiden steuerungseigenen Protokolle konvertiert wird.

### Freigegebene Kommunikationsarten mit Allen-Bradley DF1

Folgende Kommunikationsarten sind systemgetestet und freigegeben:

- HMI (Allen Bradley DF1)  
Punkt-zu-Punkt-Kopplung
- HMI (Allen Bradley DF1)  
Über KF2-Modul an Allen Bradley DH+ (Kommunikation mit bis zu 4 Steuerungen)
- HMI (Allen Bradley DF1)  
Über KF3-Modul an Allen Bradley DH485 (Kommunikation mit bis zu 4 Steuerungen)

### Koppelbare Steuerungen

Für folgende Allen-Bradley-Steuerungen steht der Kommunikationstreiber Allen Bradley DF1 zur Verfügung:

| Steuerung           | DF1<br>(Punkt-Punkt) | DF1<br>(Punkt-Punkt) | DF1<br>(Mehr-Punkt)<br>über KF2-Modul<br>an DH+ LAN | DF1<br>(Mehr-Punkt)<br>über KF3-Modul<br>an DH485 LAN |
|---------------------|----------------------|----------------------|---|---|
|                     | RS 232               | RS 422               | RS 232/RS 422                                       | RS 232 <sup>2)</sup>                                  |
| SLC500              | –                    | –                    | –   | X   |
| SLC501              | –                    | –                    | –   | X   |
| SLC502              | –                    | –                    | –   | X   |
| SLC503              | X <sup>2)</sup>      | –                    | –   | X   |
| SLC504              | X <sup>2)</sup>      | –                    | X   | X   |
| SLC505              | X <sup>2)</sup>      | –                    | –   | X   |
| MicroLogix          | X <sup>2)</sup>      | –                    | –   | X   |
| PLC-5 <sup>1)</sup> | X                    | X                    | X   | –   |

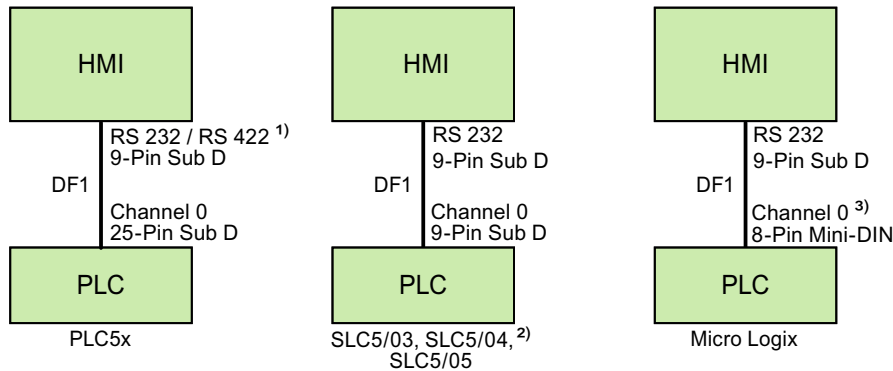
<sup>1)</sup> Für PLC-5 sind nur folgende Prozessoren freigegeben: PLC-5/11, PLC-5/20, PLC-5/30, PLC-5/40, PLC-5/60 und PLC-5/80.

<sup>2)</sup> Bei Bediengeräten, die nur eine RS 422/485-Schnittstelle haben und der Kommunikationspartner eine RS 232-Schnittstelle ist, ist der RS 422/232-Konverter getestet und freigegeben.  
Bestellnummer: 6AV6 671-8XE00-0AX0

## Protokoll DF1 mit Mehr-Punkt-Kopplung

### Punkt-zu-Punkt-Kopplung mit Protokoll DF1

Mit dem Protokoll DF1 können nur Punkt-zu-Punkt-Kopplungen aufgebaut werden.



- 1) Bei Panel PC und PC ist nur RS 232 möglich.
- 2) Eine Punkt-zu-Punkt-Kopplung zu den Steuerungen SLC500, SLC501 und SLC502 ist über DF1 nicht möglich.
- 3) Bei MicroLogix ML1500 LRP ist auch Channel 1 (9-polig Sub D) möglich.

### Verbindungskabel

| Verwendete HMI Panel-Schnittstelle | Für Verbindung zu PLC5x              | Für Verbindung zu SLC5/03, SLC5/04, SLC5/05 | Für Verbindung zu MicroLogix      |
|------------------------------------|--------------------------------------|---|-----------------------------------|
| RS 232 9-polig                     | Allen-Bradley-Kabel 1784-CP10        | Allen-Bradley-Kabel 1747-CP3                | Allen-Bradley-Kabel 1761-CBL-PM02 |
| RS 422 9-polig                     | Verbindungskabel 9-pol. Sub D RS 422 | —   | —                                 |

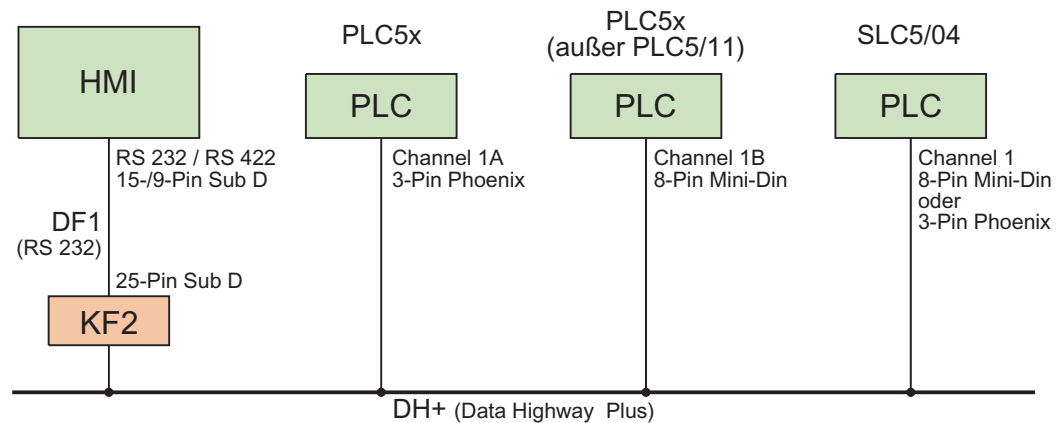
Welche Schnittstelle am Bediengerät zu verwenden ist, entnehmen Sie dem entsprechenden Gerätehandbuch.

Die Anschlussbelegungen der Kabel finden Sie im Kapitel "Verbindungskabel für Allen-Bradley".

## Protokoll DF1 mit Mehr-Punkt-Kopplung über KF2-Modul

### Protokoll DF1 mit Mehr-Punkt-Kopplung über KF2-Modul an DH+ LAN

Durch den Einsatz eines Protokoll Interface Modul KF2 ist eine Kopplung zu Steuerungen im DH+ LAN (Data Highway Plus Local Area Network) möglich.



#### Verbindungskabel

| Verwendete HMI Panel-Schnittstelle | Für Verbindung zu Interface-Modul KF2                                   |
|------------------------------------|---|
| RS 232 9-polig                     | Allen-Bradley-Kabel 1784-CP10 und Adapter 25-polig Buchse/Buchse        |
| RS 422 9-polig                     | Verbindungskabel 9-pol. Sub D RS 422 und Adapter 25-polig Buchse/Buchse |

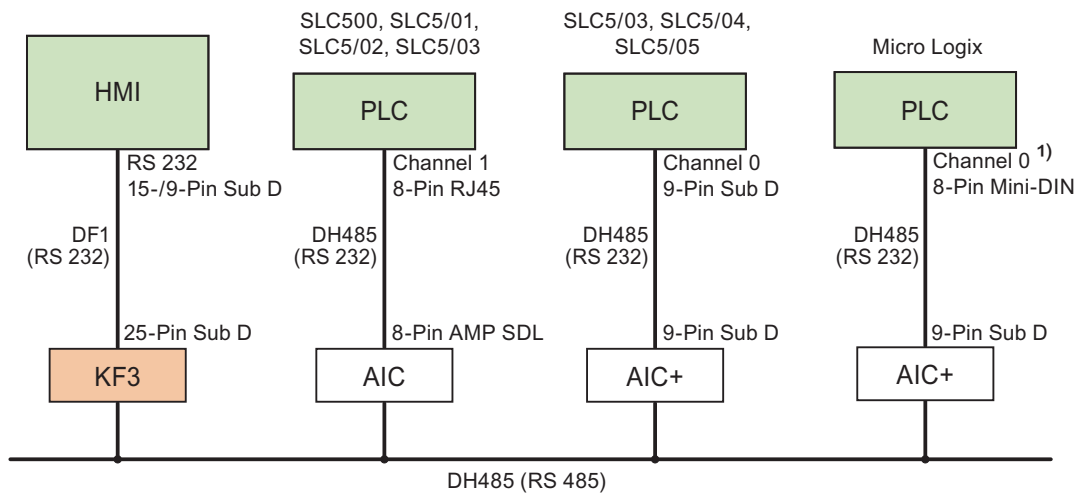
Die Kabelverbindung von den Steuerungen zum Datenbus DH+ entnehmen Sie aus der Allen-Bradley-Dokumentation.

Welche Schnittstelle am Bediengerät zu verwenden ist, entnehmen Sie dem entsprechenden Gerätehandbuch.

Die Anschlussbelegungen der Kabel finden Sie im Kapitel "Verbindungskabel für Allen-Bradley".

**Protokoll DF1 mit Mehr-Punkt-Kopplung über KF3-Modul**

**Protokoll DF1 mit Mehr-Punkt-Kopplung über KF3-Modul an DH485 LAN**



1) Bei MicroLogix ML1500 LRP auch Channel 1 (9-polig Sub D) möglich.

**Verbindungskabel**

| Verwendete HMI Panel-Schnittstelle | Für Verbindung zu Interface-Modul KF3                            |
|------------------------------------|--|
| RS 232 9-polig                     | Allen-Bradley-Kabel 1784-CP10 und Adapter 25-polig Buchse/Buchse |

Welche Schnittstelle am Bediengerät zu verwenden ist, entnehmen Sie dem entsprechenden Gerätehandbuch.

Die Anschlussbelegungen der Kabel finden Sie im Kapitel "Verbindungskabel für Allen-Bradley".

**Verbindungskabel für Allen-Bradley DF1**

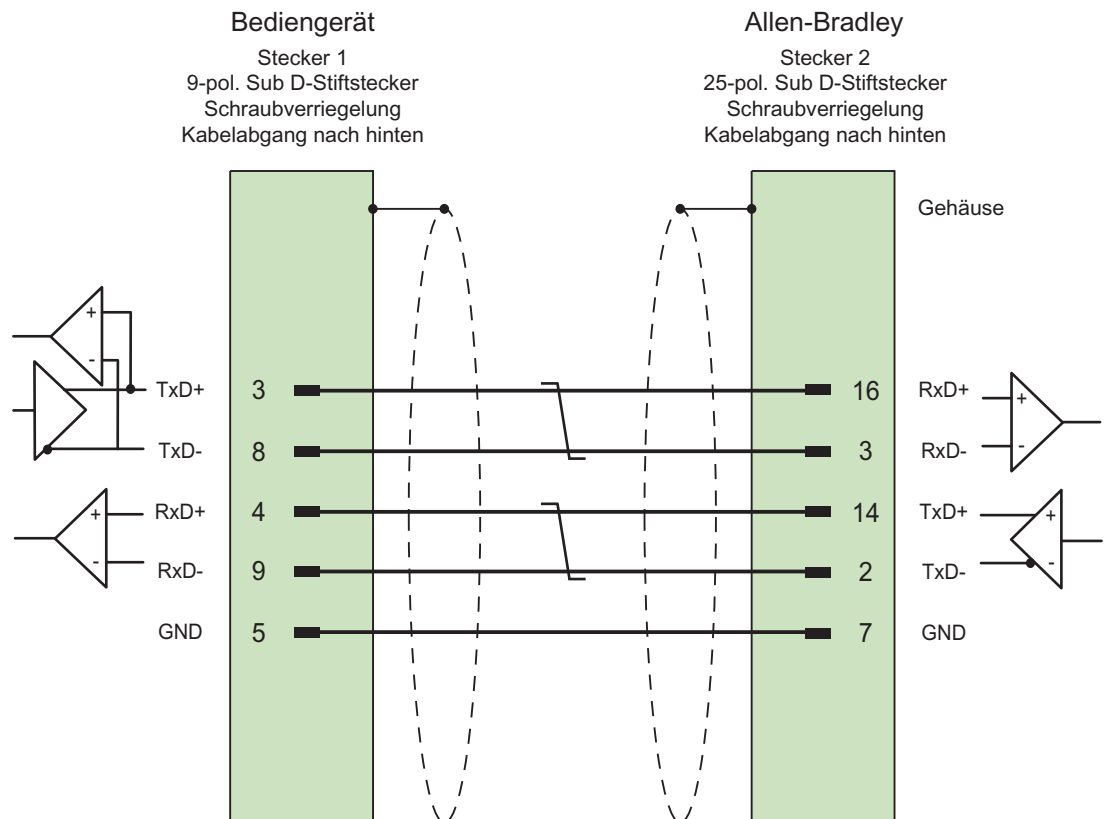
**Verbindungskabel 9-pol. Sub D, RS 422, für Allen-Bradley**

**Verbindungskabel 9-pol. Sub D RS 422**

Für Verbindung Bediengerät (RS 422, 9-pol. Sub D) - PLC5x, KF2, KF3.

Bei Verbindung zum KF2 und KF3 wird zusätzlich ein Adapter (Gender Changer) 25-polig, Buchse / Buchse benötigt.





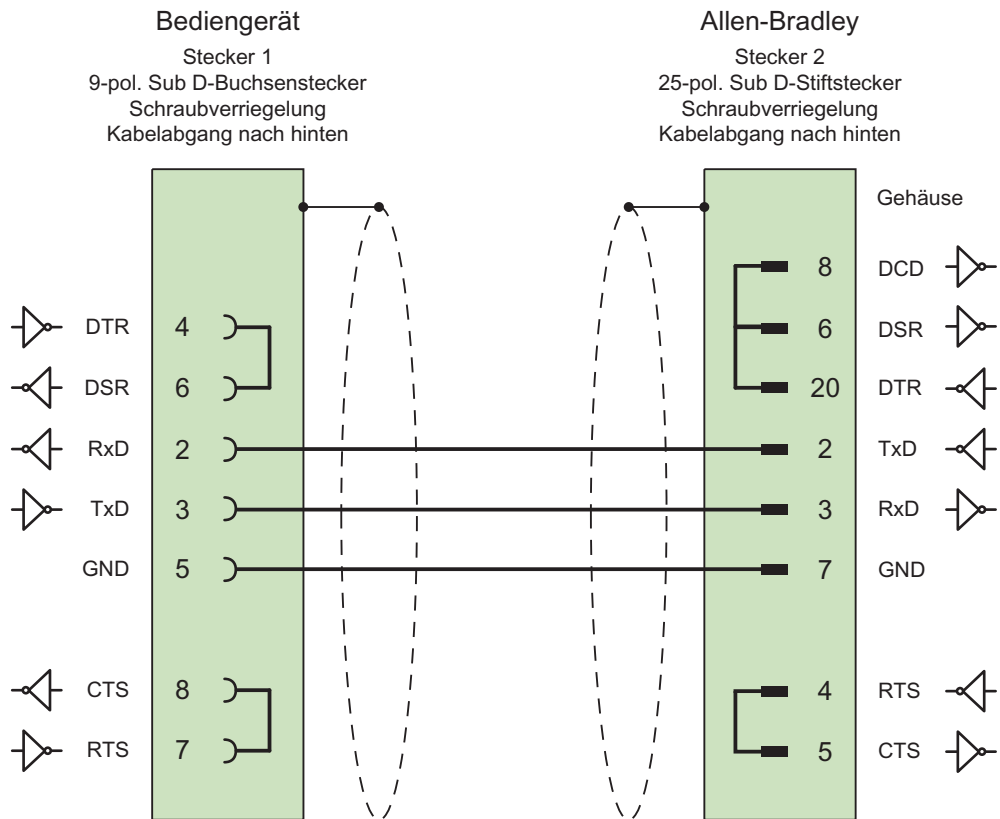
Schirm beidseitig großflächig mit Gehäuse verbunden, Schirmkontakte verbunden  
 Kabel: 3 x 2 x 0,14 mm<sup>2</sup>, geschirmt,  
 max. Länge 60 m

### Verbindungskabel 1784-CP10, RS 232, für Allen-Bradley

#### Allen-Bradley-Kabel 1784-CP10

Für Verbindung Bediengerät (RS 232, 9-pol. Sub D) - PLC5x, KF2, KF3

Bei Verbindung zum KF2 und KF3 wird zusätzlich ein Adapter (Gender Changer) 25-polig, Buchse / Buchse benötigt.

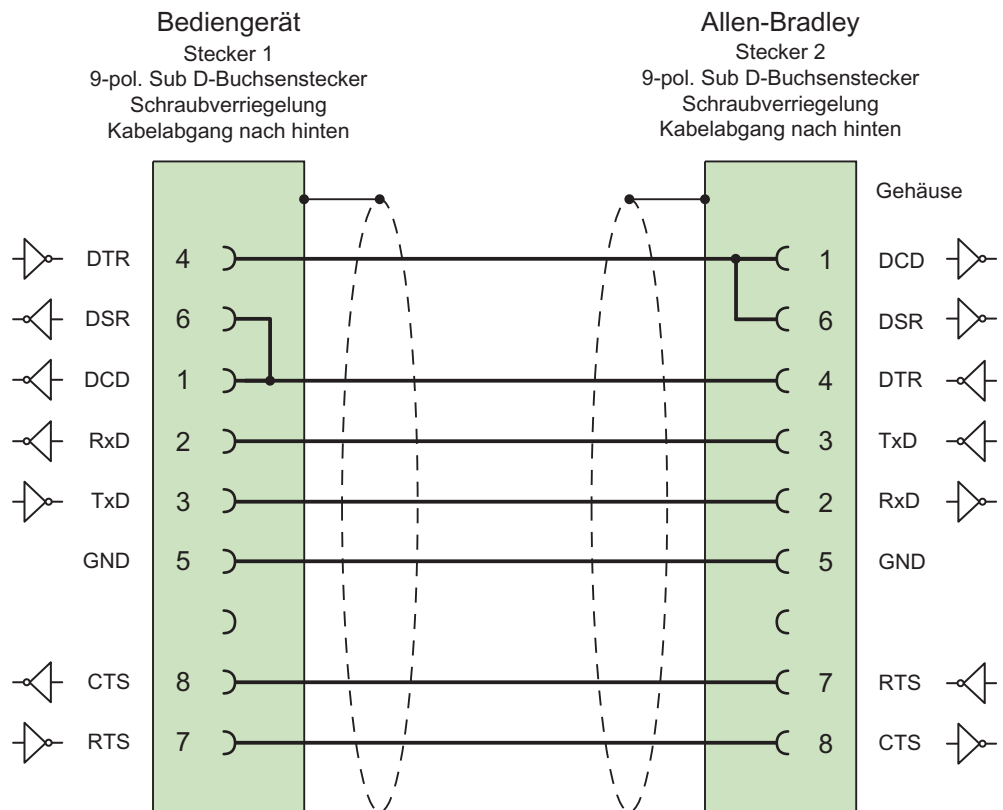


Schirm beidseitig großflächig mit Gehäuse verbunden  
max. Länge 15 m

### Verbindungskabel 1747-CP3, RS 232, für Allen-Bradley

#### Allen-Bradley-Kabel 1747-CP3

Für Verbindung Bediengerät (RS 232, 9-pol. Sub D) - SLC503, SLC504, SLC505 (Channel 0), AIC+

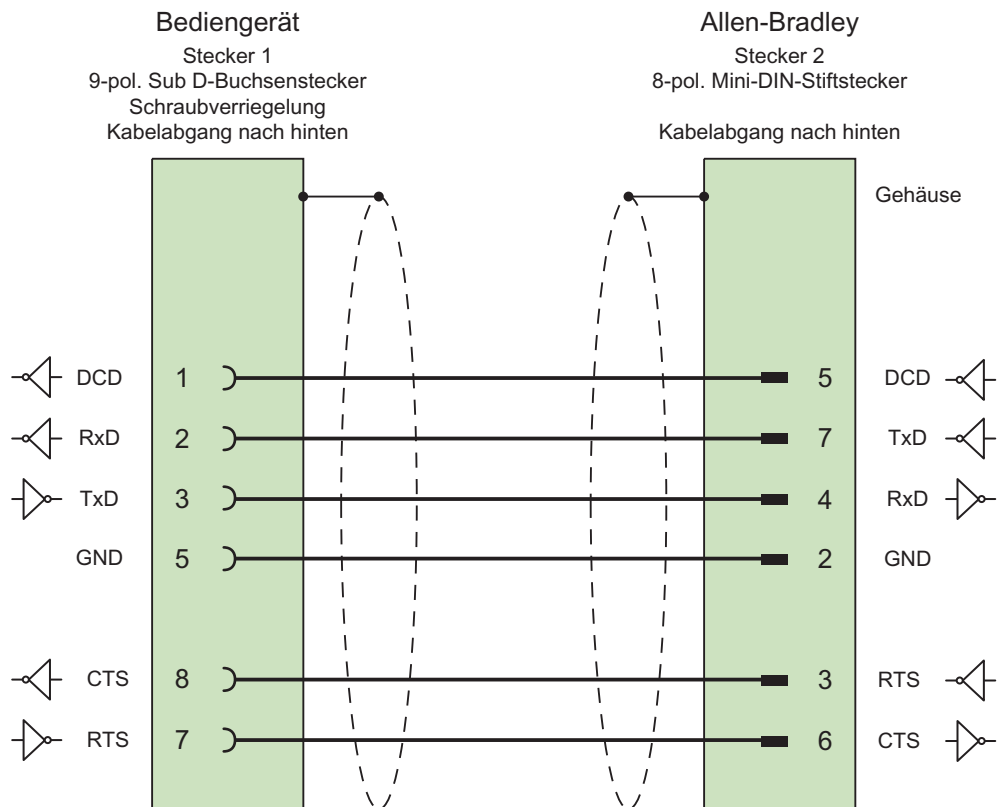


Schirm beidseitig großflächig mit Gehäuse verbunden  
 max. Länge 3 m

### Verbindungskabel 1761-CBL-PM02, RS 232, für Allen-Bradley

### Allen-Bradley-Kabel 1761-CBL-PM02

Für Verbindung Bediengerät (RS 232, 9-pol. Sub D) - Micro Logix, AIC+



Schirm beidseitig großflächig mit Gehäuse verbunden  
 max. Länge 15 m

### Leistungsmerkmale der Kommunikation

### Zulässige Datentypen für Allen-Bradley DF1

### Zulässige Datentypen für Allen-Bradley DF1

In der Tabelle sind die Anwender-Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp | Operandentyp                         | Länge            |
|----------|--------------------------------------|------------------|
| ASCII    | A <sup>1)</sup>                      | 1 bis 80 Zeichen |
| Bool     | N, R, C, T, B, S, I, O               | 1 Bit            |
| Int      | N, R, C, T, S                        | 2 Byte           |
| DInt     | N, D <sup>2)</sup>                   | 4 Byte           |
| UInt     | N, R, C, T, B, I, O, D <sup>2)</sup> | 2 Byte           |

---

| Datentyp | Operandentyp       | Länge  |
|----------|--------------------|--------|
| UDInt    | N, D <sup>2)</sup> | 4 Byte |
| Real     | N, F <sup>1)</sup> | 4 Byte |

---

1) Abhängig vom gewählten CPU-Typ auswählbar.

2) Nur bei CPU-Typ PLC5

## Abkürzungen

In WinCC werden Formate der Datentypen folgendermaßen abgekürzt:

- UNSIGNED INT = UInt
- UNSIGNED LONG = UDInt
- SIGNED INT = Int
- SIGNED LONG = DInt

## Besonderheiten bei Kopplung mit Allen-Bradley DF1

Mit Allen Bradley DF1 dürfen Arrayvariablen nur für Bitmeldungen und Kurven verwendet werden.

---

### Hinweis

Ein-/Ausgangsmodule mit 8 oder 16 Ports belegen ein ganzes Wort in der Steuerung.

Ein-/Ausgangsmodule mit 24 oder 32 Ports belegen zwei Wörter.

Wenn am Bediengerät nicht vorhandene Bits belegt sind, so gibt das Bediengerät keine Fehlermeldung aus.

Achten Sie daher bei der Projektierung darauf, dass bei den Ein-/Ausgangsmodulen mit 8 oder 24 Ports nur die Bits belegt werden, die auch einem Port zugeordnet sind.

---

## Unterstützte CPU-Typen für Allen-Bradley DF1

### CPU-Typen

Folgende CPU-Typen werden bei der Projektierung des Kommunikationstreibers Allen-Bradley DF1 unterstützt.

- SLC
  - SLC500
  - SLC501
  - SLC502
  - SLC503
  - SLC504
  - SLC505
- MicroLogix
  - MicroLogix 1x00
  - MicroLogix 1100 / 1400
- PLC 5
  - PLC-5/11
  - PLC-5/20
  - PLC-5/40
  - PLC-5/60
  - PLC-5/80

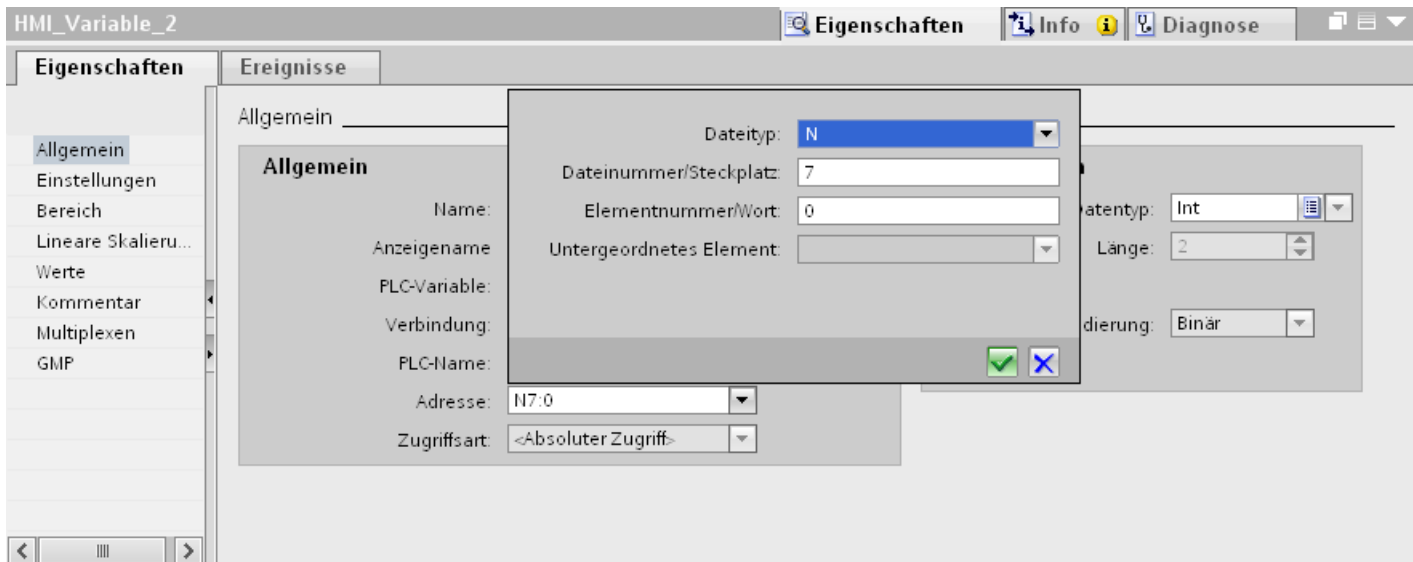
### Adressierung

### Adressierung

Die Adressierung wird beim Kommunikationstreiber Allen-Bradley DF1 in folgender Reihenfolge eingegeben:

- Operandentyp
- Dateinummer
- Elementnummer

- Untergeordnetes Element
- Bitnummer



Die Adresse erscheint dann in folgendem Format ohne Leerzeichen:

- Dateityp Dateinummer : Elementnummer . Untergeordnetes Element
- z. B. T8:2.ACC

## Operandentyp

Unter Operandentyp haben Sie folgende Auswahl:

- I
- O
- S
- B
- T
- C
- R
- N
- A
- D nur bei CPU-Typ PLC5

## Dateinummer

Unter Dateinummer wählen Sie Zahl zwischen zwei Grenzwerten:

- Unterer Grenzwert
- Oberer Grenzwert

Die Grenzwerte sind abhängig vom gewählten Dateityp.

### Untergeordnetes Element

Ein untergeordnetes Element können Sie wählen, wenn Sie einen der folgenden Dateitypen gewählt haben:

- R
- C
- T

### Adressbereiche für Allen-Bradley DF1

#### MicroLogix

| Adressbereiche | Datentypen   |                            |                            |                    |                    |                    |
|----------------|--|----------------------------|----------------------------|--------------------|--------------------|--------------------|
|                | Bool   | Int                        | UInt                       | DInt               | UDInt              | Real               |
| N              | N7:0/0 -<br>N255:255/15  | N7:0 -<br>N255:255         | N7:0 -<br>N255:255         | N7:0 -<br>N255:254 | N7:0 -<br>N255:254 | N7:0 -<br>N255:254 |
| F              | --   | --                         | --                         | --                 | F8:0 -<br>F255:255 | --                 |
| R              | R6:0.EN -<br>R255:255.ER -<br>R255:255.DN -<br>R255:255.FD -<br>R255:255.IN -<br>R255:255.EU -<br>R255:255.EM -<br>R255:255.UL | R6:0.LEN -<br>R255:255.POS | R6:0.LEN -<br>R255:255.POS | --                 | --                 | --                 |
| C              | C5:0.CU -<br>C255:255.CD -<br>C255:255.DN -<br>C255:255.OV -<br>C255:255.UN  | C5:0.PRE -<br>C255:255.ACC | C5:0.PRE -<br>C255:255.ACC | --                 | --                 | --                 |
| T              | T4:0.DN -<br>T255:255.TT -<br>T255:255.EN  | T4:0.PRE -<br>T255:255.ACC | T4:0.PRE -<br>T255:255.ACC | --                 | --                 | --                 |
| B              | B3:0/0 -<br>B255:255/15  | --                         | B3:0 -<br>B255:255         | --                 | --                 | --                 |
| S              | S2:0/0 -<br>S2:65/15   | S2:0 -<br>S2:65            | --                         | --                 | --                 | --                 |
| I              | I0:0/0 -<br>I38:255/15   | --                         | I0:0 -<br>I38:255          | --                 | --                 | --                 |
| O              | O0:0/0 -<br>O38:255/15   | --                         | O0:0/0 -<br>O38:255        | --                 | --                 | --                 |



## SLC500

| Adressbereiche | Datentypen   |                            |                            |                    |                    |                    |
|----------------|--|----------------------------|----------------------------|--------------------|--------------------|--------------------|
|                | Bool   | Int                        | UInt                       | DInt               | UDInt              | Real               |
| N              | N7:0/0 -<br>N255:255/15  | N7:0 -<br>N255:255         | N7:0 -<br>N255:255         | N7:0 -<br>N255:254 | N7:0 -<br>N255:254 | N7:0 -<br>N255:254 |
| R              | R6:0.EN -<br>R255:255.ER -<br>R255:255.DN -<br>R255:255.FD -<br>R255:255.IN -<br>R255:255.EU -<br>R255:255.EM -<br>R255:255.UL | R6:0.LEN -<br>R255:255.POS | R6:0.LEN -<br>R255:255.POS | --                 | --                 | --                 |
| C              | C5:0.CU -<br>C255:255.CD -<br>C255:255.DN -<br>C255:255.OV -<br>C255:255.UN  | C5:0.PRE -<br>C255:255.ACC | C5:0.PRE -<br>C255:255.ACC | --                 | --                 | --                 |
| T              | T4:0.DN -<br>T255:255.TT -<br>T255:255.EN  | T4:0.PRE -<br>T255:255.ACC | T4:0.PRE -<br>T255:255.ACC | --                 | --                 | --                 |
| B              | B3:0/0 -<br>B255:255/15  | --                         | B3:0 -<br>B255:255         | --                 | --                 | --                 |
| S              | S2:0/0 -<br>S2:15/15   | S2:0 -<br>S2:15            | --                         | --                 | --                 | --                 |
| I              | I0:0/0 -<br>I38:255/15   | --                         | I0:0 -<br>I38:255          | --                 | --                 | --                 |
| O              | O0:0/0 -<br>O38:255/15   | --                         | O0:0 -<br>O38:255          | --                 | --                 | --                 |

## SLC501/502

| Adressbereiche | Datentypen   |                            |                            |                    |                    |                    |
|----------------|--|----------------------------|----------------------------|--------------------|--------------------|--------------------|
|                | Bool   | Int                        | UInt                       | DInt               | UDInt              | Real               |
| N              | N7:0/0 -<br>N255:255/15  | N7:0 -<br>N255:255         | N7:0 -<br>N255:255         | N7:0 -<br>N255:254 | N7:0 -<br>N255:254 | N7:0 -<br>N255:254 |
| R              | R6:0.EN -<br>R255:255.ER -<br>R255:255.DN -<br>R255:255.FD -<br>R255:255.IN -<br>R255:255.EU -<br>R255:255.EM -<br>R255:255.UL | R6:0.LEN -<br>R255:255.POS | R6:0.LEN -<br>R255:255.POS | --                 | --                 | --                 |

| Adressbereiche | Datentypen  |                            |                            |      |       |      |
|----------------|---|----------------------------|----------------------------|------|-------|------|
|                | Bool  | Int                        | UInt                       | DInt | UDInt | Real |
| C              | C5:0.CU -<br>C255:255.CD -<br>C255:255.DN -<br>C255:255.OV -<br>C255:255.UN | C5:0.PRE -<br>C255:255.ACC | C5:0.PRE -<br>C255:255.ACC | --   | --    | --   |
| T              | T4:0.DN -<br>T255:255.TT -<br>T255:255.EN                                   | T4:0.PRE -<br>T255:255.ACC | T4:0.PRE -<br>T255:255.ACC | --   | --    | --   |
| B              | B3:0/0 -<br>B255:255/15   | --                         | B3:0 -<br>B255:255         | --   | --    | --   |
| S              | S2:0/0 -<br>S2:32/15  | S2:0 -<br>S2:32            | --                         | --   | --    | --   |
| I              | I0:0/0 -<br>I38:255/15  | --                         | I0:0 -<br>I38:255          | --   | --    | --   |
| O              | O0:0/0 -<br>O38:255/15  | --                         | O0:0 -<br>O38:255          | --   | --    | --   |

PLC5

| Adressbereich | Datentypen   |                                |                                |                    |                    |                    |                    |
|---------------|--|--------------------------------|--------------------------------|--------------------|--------------------|--------------------|--------------------|
|               | Bool   | Int                            | UInt                           | DInt               | UDInt              | Real               | ASCII              |
| N             | N3:0/0 -<br>N999:999/15  | N3:0 -<br>N999:999             | N3:0 -<br>N999:999             | N3:0 -<br>N999:999 | N3:0 -<br>N999:998 | N3:0 -<br>N999:998 | --                 |
| F             | --   | --                             | --                             | --                 | --                 | F3:0 -<br>F999:999 | --                 |
| A             | --   | --                             | --                             | --                 | --                 | --                 | A3:0 -<br>A999:999 |
| R             | R3:0.EN -<br>R999:999.ER<br>-<br>R999:999.DN<br>-<br>R999:999.FD<br>-<br>R999:999.IN<br>-<br>R999:999.EU<br>-<br>R999:999.EM<br>-<br>R999:999.UL | R3:0.LEN -<br>R999:999.PO<br>S | R3:0.LEN -<br>R999:999.PO<br>S | --                 | --                 | --                 | --                 |

| Adressbereich | Datentypen   |                                |                                |      |                    |      |       |
|---------------|--|--------------------------------|--------------------------------|------|--------------------|------|-------|
|               | Bool   | Int                            | UInt                           | DInt | UDInt              | Real | ASCII |
| C             | C3:0.CU -<br>C999:999.CD<br>-<br>C999:999.DN<br>-<br>C999:999.OV<br>-<br>C999:999.UN | C3:0.PRE -<br>C999:999.AC<br>C | C3:0.PRE -<br>C999:999.AC<br>C | --   | --                 | --   | --    |
| T             | T3:0.DN -<br>T999:999.TT<br>-<br>T999:999.EN   | T3:0.PRE -<br>T999:999.AC<br>C | T3:0.PRE -<br>T999:999.AC<br>C | --   | --                 | --   | --    |
| B             | B3:0/0 -<br>B999:999/15  | --                             | B3:0 -<br>B999:999             | --   | --                 | --   | --    |
| S             | S2:0/0 -<br>S2:127/15  | S2:0 -<br>S2:127               | --                             | --   | --                 | --   | --    |
| I             | I1:0/0 -<br>I1:277/17  | --                             | I1:0 -<br>I1:277               | --   | --                 | --   | --    |
| O             | O0:0/0 -<br>O0:277/17  | --                             | O0:0 -<br>O0:277               | --   | --                 | --   | --    |
| D             | D3:0/0 -<br>D999:999/15  | D3:0 -<br>D999:999             | D3:0 -<br>D999:999             | --   | D3:0 -<br>D999:998 | --   | --    |

## Komponenten in Betrieb nehmen

### Projekt auf das Bediengerät übertragen

1. Versetzen Sie das Bediengerät in den "Transfermodus".
2. Stellen Sie alle notwendigen Transferparameter ein:
  - Schnittstelle
  - Übertragungsparameter
  - Zielspeicher
3. Starten Sie die Übertragung des Projekts.  
 Das Projekt wird automatisch generiert.  
 In einem Fenster werden die einzelnen Schritte der Generierung und der Übertragung mitprotokolliert.

### Steuerung und Bediengerät miteinander verbinden

1. Verbinden Sie die Steuerung und das Bediengerät mit einem geeigneten Anschlusskabel.
2. Am Bediengerät erscheint die Meldung "Verbindung zur Steuerung .... ist hergestellt".

## Projektierung optimieren

### Erfassungszyklus und Aktualisierungszeit

Die in der Projektierungssoftware angegebenen Erfassungszyklen für die "Bereichszeiger" und die Erfassungszyklen der Variablen sind wesentliche Faktoren für die tatsächlich erreichbaren Aktualisierungszeiten.

Die Aktualisierungszeit ist Erfassungszyklus plus Übertragungszeit plus Verarbeitungszeit.

Um optimale Aktualisierungszeiten zu erreichen, beachten Sie bei der Projektierung Folgendes:

- Richten Sie die einzelnen Datenbereiche so klein wie möglich und so groß wie nötig ein.
- Zu klein gewählte Erfassungszyklen beeinträchtigen unnötigerweise die Gesamtleistung. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs. Richtwert für den Erfassungszyklus ca. 1 Sekunde.
- Legen Sie die Variablen einer Meldung oder eines Bildes ohne Lücken in einen Datenbereich.
- Damit Änderungen in der Steuerung sicher erkannt werden, müssen diese mindestens während des tatsächlichen Erfassungszyklus anstehen.

### Bilder

Bei Bildern hängt die tatsächlich erreichbare Aktualisierungsrate von Art und Anzahl der darzustellenden Daten ab.

Beachten Sie bei der Projektierung, dass Sie nur für diejenigen Objekte kurze Erfassungszyklen projektieren, die tatsächlich schnell aktualisiert werden müssen. Damit werden die Aktualisierungszeiten kürzer.

### Kurven

Wenn bei bitgetriggerten Kurven das Sammelbit im "Kurvenübertragungsbereich" gesetzt wird, so aktualisiert das Bediengerät jedes Mal alle Kurven, deren Bit in diesem Bereich gesetzt ist. Danach setzt es die Bits wieder zurück.

Erst wenn vom Bediengerät alle Bits zurückgesetzt wurden, darf das Sammelbit im SPS-Programm wieder gesetzt werden.

### Steuerungsaufträge

Wenn viele Steuerungsaufträge in schneller Folge zum Bedienen gesendet werden, so kann dadurch die Kommunikation zwischen Bediengerät und Steuerung überlastet werden.

Wenn das Bediengerät den Wert 0 in das erste Datenwort des Auftragsfachs einträgt, hat das Bediengerät den Steuerungsauftrag entgegengenommen. Jetzt arbeitet das Bediengerät den Auftrag ab, wofür es noch Zeit benötigt. Wenn sofort wieder ein neuer Steuerungsauftrag in das Auftragsfach eingetragen wird, kann es eine Zeit dauern, bis das Bediengerät den

nächsten Steuerungsauftrag ausführt. Der nächste Steuerungsauftrag wird erst wieder entgegengenommen, wenn Rechnerleistung zur Verfügung steht.

## Datenaustausch

### Bereichszeiger bei Allen-Bradley

#### Bereichszeiger bei Verbindungen über Allen-Bradley Kommunikationstreiber

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu.

Nähere Hinweise zu Bereichszeigern und deren Projektierung finden Sie im Kapitel "Datenaustausch über Bereichszeiger".

#### Besonderheiten bei Verbindungen über Allen-Bradley EtherNet/IP

Sie können folgende Bereichszeiger projektieren

| Bereichszeiger    | Allen-Bradley EtherNet/IP | Allen-Bradley DF1 |
|-------------------|---------------------------|-------------------|
| Bildnummer        | ja                        | ja                |
| Datum/Uhrzeit     | ja                        | ja                |
| Datum/Uhrzeit PLC | ja                        | ja                |
| Koordinierung     | ja                        | ja                |
| Projektkennung    | ja                        | ja                |
| Steuerungsauftrag | ja                        | ja                |
| Datensatz         | ja                        | ja                |

#### Einschränkungen Allen-Bradley Ethernet/IP

Folgende Einschränkungen gelten für die Projektierung von Bereichszeigern.

| CPU-Typ                    | Datentypen | Dateitypen |
|----------------------------|------------|------------|
| ControlLogix, CompactLogix | Int, UInt  | --         |
| SLC, MicroLogix            | Int, UInt  | N, B       |

## Einschränkungen Allen-Bradley DF1

Folgende Einschränkungen gelten für die Projektierung von Bereichszeigern.

| CPU-Typ    | Datentypen | Dateitypen |
|------------|------------|------------|
| MicroLogix | --         | N, O, I, B |
| SLC50x     | --         | N, O, I, B |
| PLC5       | --         | N, O, I, B |

## Siehe auch

Datenaustausch über Bereichszeiger (Seite 3821)

## Kurven

## Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt bei den Basic Panels zeitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 3201)

## Zeitgetriggerte Kurven

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein.

Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

## Meldungen

## Meldungen projektieren

## Meldungen projektieren bei nicht integrierten Verbindungen

Um Meldungen wie Warnungen, Fehlermeldungen und Quittierung zu projektieren sind mehrere Schritte notwendig.

- Schritt 1: Variablen anlegen
- Schritt 2: Meldungen projektieren
- Schritt 3: Quittierung projektieren

Nähere Hinweise hierzu finden Sie im Kapitel:

Arbeiten mit Meldungen (Seite 3213)

### Besonderheiten bei der Projektierung von Meldungen

Wenn Sie Verbindungen von Bediengeräten mit Steuerungen anderer Hersteller projektieren, beachten Sie folgende Besonderheiten bei der Projektierung:

- Datentypen der Variablen
- Adressierung der Variablen
- Zählweise der Bitstellen

### Einschränkungen

Als Trigger-Variable für Bitmeldungen sind nur Variablen in "Dateityp" "N", "O", "I", "S" und "B" zulässig. Die Variablen sind nur für die Datentypen "Int" und "UInt" zulässig.

### Datentypen

Für Verbindungen mit Allen-Bradley Kommunikationstreiber, werden folgende Datentypen unterstützt:

| Kommunikationstreiber     | Steuerung  | Zulässige Datentypen |   |
|---------------------------|--|----------------------|---|
|                           |  | Bitmeldungen         | Analogmeldungen                           |
| Allen-Bradley DF1         | SLC500, SLC501, SLC502, SLC503, SLC504, SLC505, PLC5, MicroLogix | Int, UInt            | Int, UInt, Long, ULong, Real              |
| Allen-Bradley EtherNet/IP | ControlLogix, CompactLogix, SLC, Micrologix                      | Int, UInt            | SInt, USInt, Int, UInt, DInt, UDInt, Real |

### Zählweise der Bitstellen

Für Verbindungen mit Allen-Bradley Kommunikationstreiber, gilt folgende Zählweise:

| Zählweise der Bitstellen     | linkes Byte |  |  |  |  |  |  |   | rechtes Byte |  |  |  |  |  |  |  |   |
|------------------------------|-------------|--|--|--|--|--|--|---|--------------|--|--|--|--|--|--|--|---|
| In Allen-Bradley-Steuerungen | 15          |  |  |  |  |  |  | 8 | 7            |  |  |  |  |  |  |  | 0 |
| Im WinCC projektieren Sie:   | 15          |  |  |  |  |  |  | 8 | 7            |  |  |  |  |  |  |  | 0 |

### Siehe auch

Das Meldesystem in WinCC (Seite 3203)

## Quittierung von Meldungen

### Vorgehensweise

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

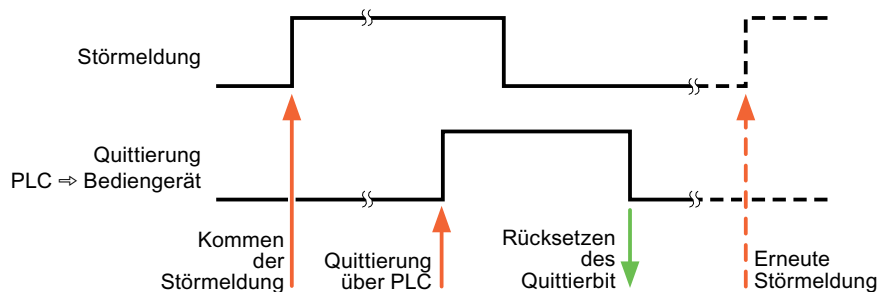
### Quittieren von der Steuerung

Unter "PLC-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulssdiagramm.



### Quittierung am Bediengerät

Unter "HMI-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittvariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittvariable in die Steuerung eine gewisse Zeitspanne.



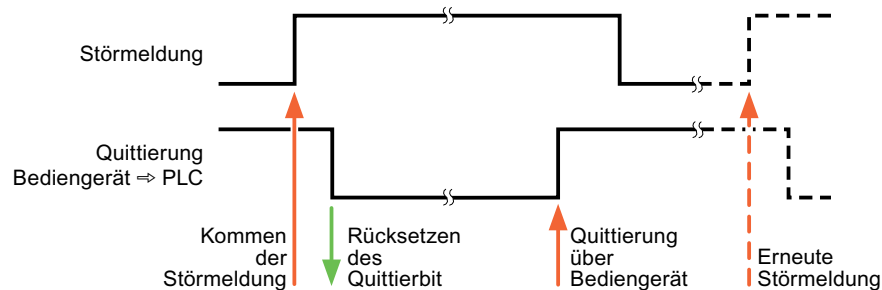
Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittiervariablen gesetzt. Die gesamte Quittiervariable wird dann vom Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

#### Hinweis

Alle seit dem letzten Runtime-Start quittierten Meldebits bleiben in der Quittiervariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittiervariablen der gesamte Bereich vom Bediengerät überschrieben wird.

Das nachfolgende Bild zeigt das Impulsdigramm.



## Mitsubishi

### Mitsubishi Kommunikationstreiber

#### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und Steuerungen beschrieben, die Mitsubishi Kommunikationstreiber verwenden.

Folgende Kommunikationstreiber werden unterstützt:

- Mitsubishi MC TCPI/IP
- Mitsubishi FX

## Datenaustausch

Datenaustausch findet über Variablen oder Bereichszeiger statt.

- Variablen  
Die Steuerung und das Bediengerät tauschen ihre Daten über Prozesswerte aus. Legen Sie in der Projektierung Variablen an, die auf Adressen in der Steuerung zeigen. Das Bediengerät liest aus der angegebenen Adresse den Wert und zeigt ihn an. Genauso kann der Bediener eine Eingabe am Bediengerät vornehmen, die dann in die Adresse in der Steuerung geschrieben wird.
- Bereichszeiger  
Bereichszeiger dienen dem Austausch spezieller Daten und werden nur bei Verwendung dieser Daten eingerichtet.

## Mitsubishi MC TCP/IP

### Verbindung über Mitsubishi MC TCPI/IP projektieren

#### Einleitung

Eine Verbindung zu einer der Steuerung mit Mitsubishi MC TCPI/IP Kommunikationstreiber, projektieren Sie im Editor "Verbindungen" des Bediengeräts.

Abhängig vom Bediengerät sind die Ethernet-Schnittstellen unterschiedlich benannt.

Beispiel: PROFINET-Schnittstelle entspricht der Ethernet-Schnittstelle

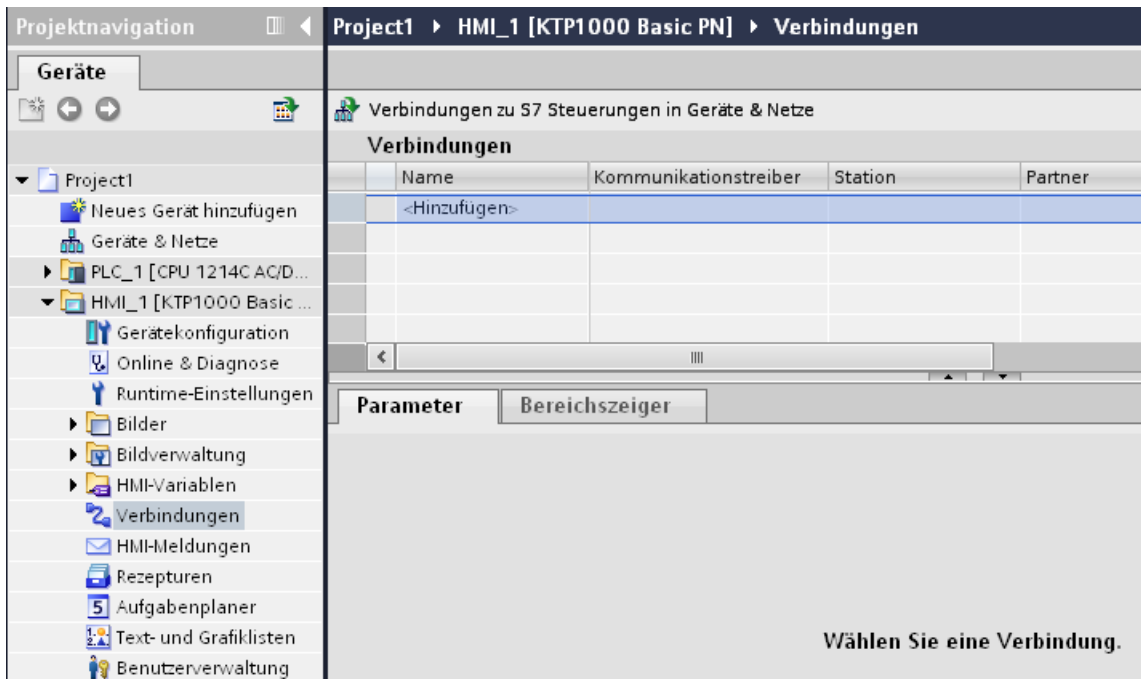
#### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

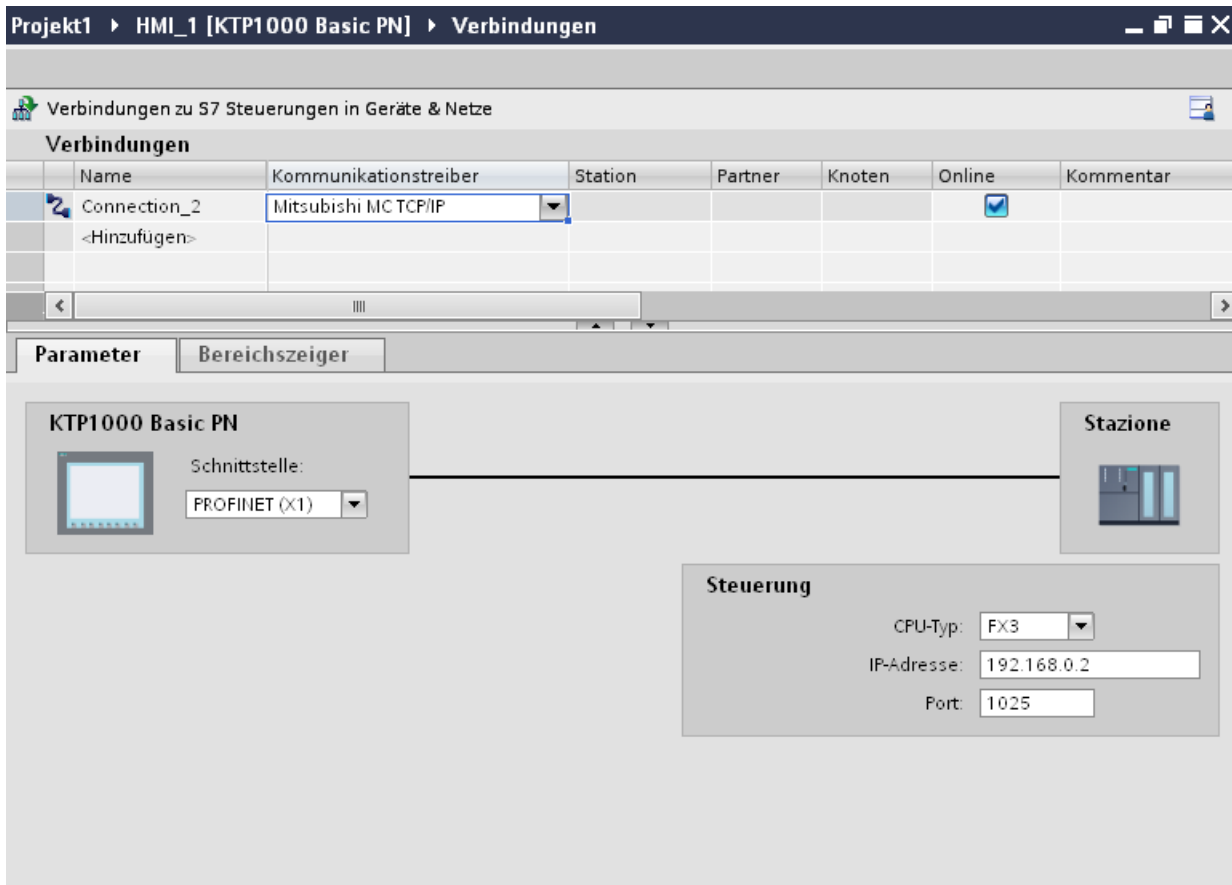
#### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".

3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".



4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "Mitsubishi MC TCP/IP" aus.



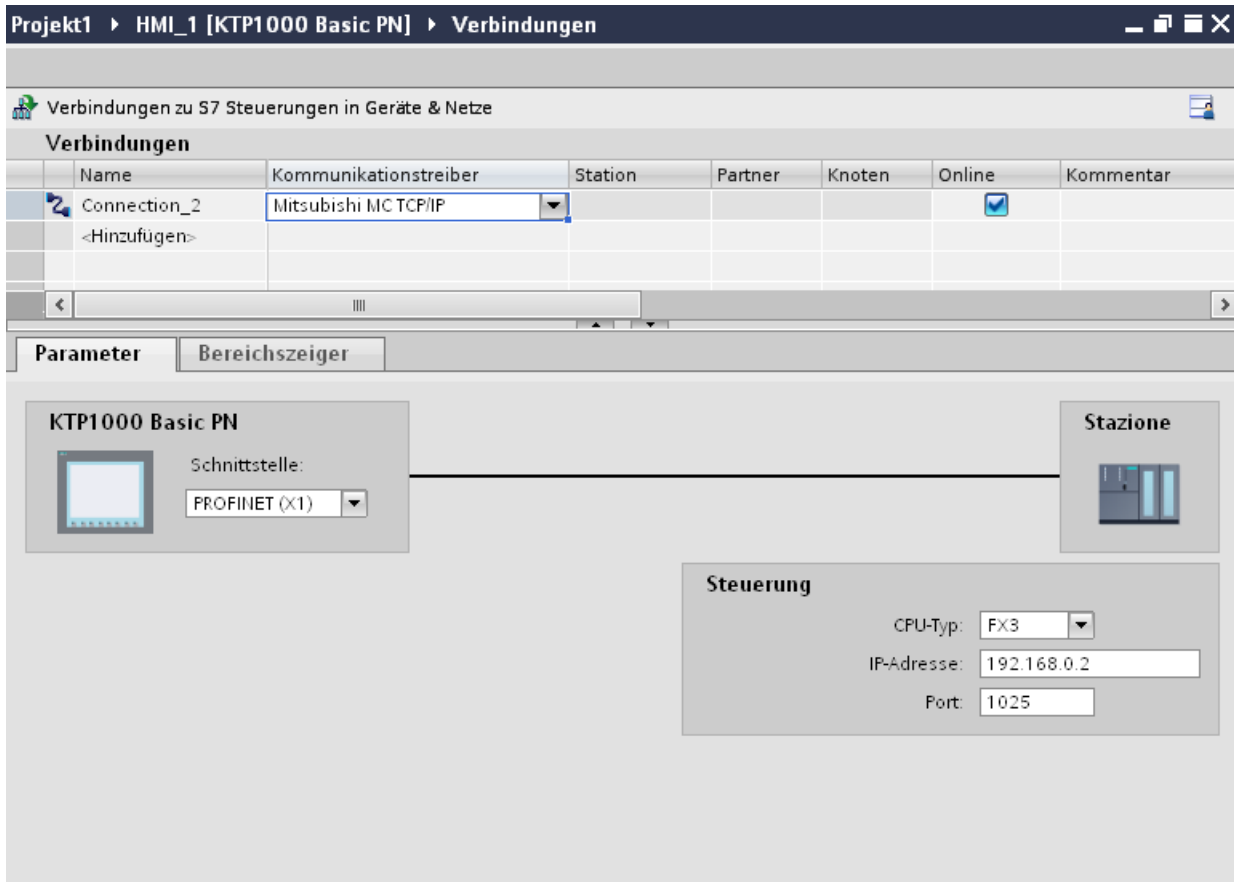
5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.

### Parameter für die Verbindung (Mitsubishi MC TCP/IP)

#### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät" und "Steuerung".



## Parameter für das Bediengerät

Sie können im Inspektorfenster "Parameter" für das Bediengerät nur eine Schnittstelle auswählen. Abhängig vom Bediengerät stehen auch mehrere Schnittstellen zur Verfügung.

Wenn Sie beim Projektieren direkt mit dem Bediengerät verbunden sind, können Sie die IP-Adresse des Bediengeräts in WinCC einrichten. Beim Projekttransfer wird die IP-Adresse auf das Bediengerät übertragen.

---

### Hinweis

Wenn Sie die IP-Adresse bereits im Control-Panel des Bediengeräts eingerichtet haben, wird die IP-Adresse im Control Panel beim nächsten Laden überschrieben.

Wenn Sie "IP-Adresse auf anderem Weg beziehen" aktivieren, bleibt die bereits eingerichtete IP-Adresse im Control Panel beim nächsten Laden erhalten.

---

Die IP-Adresse des Bediengeräts richten Sie folgendermaßen ein:

1. Klicken Sie auf das Bediengerät.
2. Öffnen Sie den Editor "Gerätekonfiguration"

3. Klicken Sie auf die Ethernet-Schnittstelle.
4. Vergeben Sie im Inspektorfenster die IP-Adresse unter:  
"Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"

### Parameter für die Steuerung

- CPU-Typ  
Unter "CPU-Typ" stellen Sie ein, an welchen Steuerungstyp das Bediengerät angeschlossen ist.  
Folgende Einträge sind möglich:  
–FX3  
–Q  
Wenn Sie den CPU-Typ FX3 auswählen, dann wird das Mitsubishi MC Protokoll "1E" und beim CPU-Typ "Q" wird "3E" verwendet.  
Es wird immer die Protokoll-Variante "Binärer Code" verwendet.

---

#### Hinweis

Wenn bei einer projektierten Verbindung der CPU-Typ geändert wird, müssen Variablen mit folgenden Eigenschaften überarbeitet werden:

- Operanden, die für den neuen CPU-Typ nicht existieren wie "W", "B", "F".
  - Ein- und Ausgänge mit abweichender Adressierung (hexadezimal/oktal)
  - Adressen größer als der zugelassene Adressbereich des neuen CPU-Typs.
- 

- IP-Adresse  
Stellen Sie die IP-Adresse oder den Host-Namen des Ethernet/IP Moduls der Steuerung ein. Auf einem Basic Panel kann nur die IP-Adresse verwendet werden.
- Port  
Stellen Sie die Port-Nummer des Moduls der Steuerung ein.

### Bediengerät mit Steuerung verbinden

#### Verbindungen über Mitsubishi MC TCP/IP

#### Anschluss

Das Bediengerät kann über folgende Komponenten an die Mitsubishi Steuerung angebunden werden:

- Bestehendes Ethernet-Netz, in dem sich auch die Steuerungen befinden
- Gekreuztes Ethernet-Kabel (Cross-Over), direkt an die Ethernetschnittstelle der CPU bzw. des Kommunikationsmoduls

Die Kopplung des Bediengeräts an eine Mitsubishi Steuerung beschränkt sich hauptsächlich auf den physikalischen Anschluss des Bediengeräts. Spezielle Bausteine für die Kopplung sind in der Steuerung nicht erforderlich.

Schließen Sie das Bediengerät an eine oder mehrere Steuerungen der Q-Serie und/oder der FX3 an. Das Bediengerät schließen Sie über folgende Schnittstellen an:

- Kommunikationsschnittstelle OnBoard
- Freigegebenes Kommunikationsmodul passend zur Steuerung

---

#### **Hinweis**

##### **Timeout-verhalten bei TCP/IP (Ethernet)**

Durch die Verwendung des TCP/IP-Protokolls werden Verbindungsunterbrechungen frühestens nach ca. einer Minute erkannt. Werden keine Variablen angefordert, z. B. keine Ausgabevariable im aktuellen Bild, so wird eine Verbindungsunterbrechung nicht sicher erkannt.

Projektieren Sie einen Bereichszeiger Koordination für jede Steuerung. Diese Einstellung stellt sicher, dass auch im beschriebenen Fall die Unterbrechung der Verbindung nach ca. zwei Minuten erkannt wird.

---

## **Kommunikationsarten**

### **Freigegebene Kommunikationsarten**

- Gilt nur für Mitsubishi FX(PG-Protokoll):  
Die Punkt-zu-Punkt-Kopplung von einem Bediengerät zu einer freigegebenen Mitsubishi FX-CPU über Mitsubishi FX ist von der Siemens AG systemgetestet und freigegeben.
- Gilt nur für Mitsubishi MC TCP/IP:  
Folgende Kommunikationsarten sind systemgetestet und freigegeben:
  - Punkt-zu-Punkt-Kopplung zu den freigegebenen Steuerungen
  - Mehrpunkt-Kopplung von einem Bediengerät mit bis zu 4 Steuerungen, mit den jeweils freigegebenen Steuerungen. Das Mischen der CPU-Typen (FX3 und Q) ist möglich.

---

#### **Hinweis**

Das Bediengerät ist Client und die Steuerung muss als Server arbeiten.

---

## Koppelbare Steuerungen

Für folgende Mitsubishi-Steuerungen können Kopplungen realisiert werden:

|  | Mitsubishi FX<br>(PG-Protokoll) | Mitsubishi MC TCP/IP |
|--|---------------------------------|----------------------|
| <b>Steuerung</b>   |                                 |                      |
| MELSEC FX1n, FX2n  | ja                              | nein                 |
| MELSEC FX3U, FX3UC, FX3G mit<br>Kommunikations-Modul FX3U-ENET   | nein                            | ja                   |
| MELSEC System Q  | nein                            | ja                   |
| <ul style="list-style-type: none"><li>• Q-Series mit dem Kommunikations-Modul QJ71E71-100</li><li>• QnUDEH CPU mit Ethernet-Schnittstelle on Board</li></ul> |                                 |                      |

## Parametrierung der Kommunikationsmodule

### FX3 Steuerungen

#### Vorgehensweise

1. Starten Sie den FX-Configurator.
2. Wählen Sie das Modul aus.
3. Vergeben Sie im Dialog "Operational settings" folgende Einstellungen:
  - Communication data code:  
Binary code
  - Initial timing:  
Always wait for OPEN
  - IP address:  
IP-Adresse
  - Send frame setting:  
Ethernet(V2.0)
  - TCP Existence confirmation setting:  
Use the Ping



4. Vergeben Sie im Dialog "Open settings" folgende Einstellungen:

- Protocol:  
TCP
- Open system:  
Unpassive
- Fixed buffer:  
Receive
- Fixed buffer communication procedure:  
Procedure exist(MC)
- Pairing open  
Disable
- Existence confirmation  
No confirm
- Host station Port No. (DEC)  
Port-Nummer

---

**Hinweis**

Die im Kommunikationsmodul gewählte Port-Nummer muss mit der in WinCC übereinstimmen. Für jedes angeschlossene Bediengerät muss eine Verbindung mit einer Portnummer parametrisiert werden.

Sie müssen die Port-Nummer in Dezimalwerten angeben.

---

5. Bestätigen Sie die Standard-Einstellungen der weiteren Dialoge.

Die Parameter Netzwerk Nr. und Stations Nr. sind für die Kopplung nicht relevant und können beliebig gewählt werden.

## Q Steuerungen

### Vorgehensweise

1. Klicken Sie auf Netzwerkparameter bearbeiten.
2. Wählen Sie den Netzwerktyp:
  - Ethernet  
Die Netzwerknummer und die Gruppen- / Stationsnummer werden nicht ausgewertet und können beliebig vergeben werden

3. Vergeben Sie im Dialog "Operational settings" folgende Einstellungen:
  - Communication data code:  
Binary code
  - Initial timing:  
Always wait for OPEN
  - IP address:  
IP-Adresse
  - Send frame setting:  
Ethernet(V2.0)
  - Schreiben während RUN ermöglichen
4. Vergeben Sie im Dialog "Open settings" folgende Einstellungen:
  - Protocol:  
TCP
  - Open system:  
Unpassive
  - Pairing open  
Disable
  - Existence confirmation  
No confirm
  - Host station Port No. (HEX)  
Port-Nummer

---

**Hinweis**

Die im Kommunikationsmodul gewählte Port-Nummer muss mit der in WinCC übereinstimmen. Für jedes angeschlossene Bediengerät muss eine Verbindung mit einer Portnummer parametrisiert werden.

Sie müssen die Port-Nummer in Hexadezimalwerten angeben.

---

## Interner Ethernet Port der Q0xUDEH CPU

### Vorgehensweise

1. Vergeben Sie im Dialog "Internal Ethernet Port" folgende Einstellungen:
  - IP address:  
IP-Adresse
  - Communication data code:  
Binary code
  - Online-Änderungen einschalten
2. Vergeben Sie im Dialog "Open settings" folgende Einstellungen:
  - Protocol:  
TCP
  - Open system:  
MC-Protocol
  - Host station Port No. (HEX)  
Port-Nummer

---

#### Hinweis

Die im Kommunikationsmodul gewählte Port-Nummer muss mit der in WinCC übereinstimmen. Für jedes angeschlossene Bediengerät muss eine Verbindung mit einer Portnummer parametrieren werden.

---

## Leistungsmerkmale der Kommunikation

### Zulässige Datentypen für Mitsubishi MC TCPI/IP

#### Zulässige Datentypen

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp     | Operandentyp     | Länge  |
|--------------|------------------|--------|
| 4-bit block  | M, X, Y, B, F    | 1 Byte |
| 8-bit block  | M, X, Y, B, F    | 1 Byte |
| 12-bit block | M, X, Y, B, F    | 2 Byte |
| 16-bit block | M, X, Y, B, F    | 2 Byte |
| 20-bit block | M, X, Y, B, F    | 4 Byte |
| 24-bit block | M, X, Y, B, F    | 4 Byte |
| 28-bit block | M, X, Y, B, F    | 4 Byte |
| 32-bit block | M, X, Y, B, F    | 4 Byte |
| Bool         | M, D, X, Y, B, F | 1 Bit  |

---

| Datentyp  | Operandentyp | Länge            |
|-----------|--------------|------------------|
| DInt      | D, W         | 4 Byte           |
| DWord     | D, C, W      | 4 Byte           |
| Int       | D, W         | 2 Byte           |
| Real 1)   | D, W         | 4 Byte           |
| String 1) | D            | 1 bis 80 Zeichen |
| Word      | D, T, C, W   | 2 Byte           |

---

- 1) Die Datentypen "String" und "Real" sind nicht bei allen CPUs verfügbar.
- 2) Die Operandentypen B, F und W sind nur für den CPU-Typ "Q" verfügbar.

---

#### Hinweis

Beachten Sie bei schreibenden Zugriffen:

Variablen können nur geschrieben werden, wenn bei der Parametrierung der Mitsubishi-Kommunikationsmodule "Online-Änderungen einschalten" bzw. "Schreiben während RUN ermöglichen" ausgewählt wurde.

Bei Datentyp "Bool" im Operandentyp "D" wird nach der Änderung des angegebenen Bits das gesamte Wort wieder in die Steuerung zurückgeschrieben. Dabei wird nicht geprüft, ob sich zwischenzeitlich andere Bits im Wort geändert haben. Deshalb darf die Steuerung auf das angegebene Wort nur lesend zugreifen.

---

#### Hinweis

Array-Elemente in E/A-Feldern sind bei einer Kommunikation mit Mitsubishi-Steuerung nicht zulässig.

---

## Unterstützte CPU-Typen für Mitsubishi MC TCP/IP

### CPU-Typen

Folgende CPU-Typen werden bei der Projektierung des Kommunikationstreibers Mitsubishi MC TCP/IP unterstützt.

- FX3 series
  - FX 3G / FX 3G with communication modul FX3U-ENET
  - FX 3U / FX 3U with communication modul FX3U-ENET
  - FX 3UC / FX 3UC with communication modul FX3U-ENET
- Q series
  - Q-Series with QJ71E71-100 communication module
- iQ series / QnUD
  - QnUDEHCPU with built in ethernet module

## Adressen bei Mitsubishi MC TCP/IP

### Adressbereiche bei Verbindungen über Mitsubishi MC TCP/IP

Für die CPUs der verschiedenen Serien gibt es unterschiedliche Grenzen für die Adressbereiche, die Sie den MITSUBISHI-Computerlink-Handbüchern entnehmen.

Beispiele für die CPU- und kommunikationsformat-abhängigen Grenzen der Adressbereiche:

| Bezeichnung   | Operandentyp | Max. Adresse FX3                | Max. Adresse Q-Serie                                  |
|---------------|--------------|---------------------------------|---|
| Output/Input  | Y/X          | Octal X/Y 0 - 777               | HEX X/Y 0 - 7FF                                       |
| Merker        | M            | M0 - M3071 und<br>M8000 - M8255 | M/L/S 0 - 8191  |
| Data Register | D            | D0 - 7999<br>D8000 - D8255      | D0 - 8191<br>D9000 - D9255 wird zu SD1000 -<br>SD1255 |
| Counter       | C            | C0 - 255                        | C0 - 1023   |
| Timer         | T            | T0 - 255                        | T0 - 2047   |
| Link Register | W            | --                              | Hex: W0 - FFF   |
| Link Merker   | B            | --                              | Hex: B0 - FFF   |
| Fehlermerker  | F            | --                              | F0 - 2047   |

### Adressbereiche für Mitsubishi MC TCP/IP

#### FX0

| Adressbereich | Datentypen         |                           |           |           |           |                |                |                |                |              |              |              |              |
|---------------|--------------------|---------------------------|-----------|-----------|-----------|----------------|----------------|----------------|----------------|--------------|--------------|--------------|--------------|
|               | Bool               | Word                      | DWord     | Real      | String    | 4-bit block    | 8-bit block    | 12-bit block   | 16-bit block   | 20-bit block | 24-bit block | 28-bit block | 32-bit block |
| M             | M0 - M9999         | --                        | --        | --        | --        | M0 - M999<br>6 | M0 - M999<br>2 | M0 - M998<br>8 | M0 - M998<br>4 | M0 - M9980   | M0 - M9976   | M0 - M9972   | M0 - M9968   |
| D             | D0.0 - D999.1<br>5 | D0 - D999                 | D0 - D998 | D0 - D998 | D0 - D998 | --             | --             | --             | --             | --           | --           | --           | --           |
| T             |                    | T0 - T255                 | --        | --        | --        | --             | --             | --             | --             | --           | --           | --           | --           |
| C-16-Bit      | --                 | C-16-Bit 0 - C-16-Bit 199 | --        | --        | --        | --             | --             | --             | --             | --           | --           | --           | --           |

| Adressbereich | Datentypen |      |              |      |        |             |             |              |              |              |              |              |              |
|---------------|------------|------|--------------|------|--------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|--------------|
|               | Bool       | Word | DWord        | Real | String | 4-bit block | 8-bit block | 12-bit block | 16-bit block | 20-bit block | 24-bit block | 28-bit block | 32-bit block |
| C-32-Bit      | --         | --   | C-32-Bit 200 | --   | --     | --          | --          | --           | --           | --           | --           | --           | --           |
|               |            |      | -            |      |        |             |             |              |              |              |              |              |              |
|               |            |      | C-32-Bit 255 |      |        |             |             |              |              |              |              |              |              |
| X             | X0 - X255  | --   | --           | --   | --     | X0 - X252   | X0 - X248   | X0 - X244    | X0 - X240    | X0 - X236    | X0 - X232    | X0 - X228    | X0 - X224    |
| Y             | Y0 - X255  | --   | --           | --   | --     | Y0 - Y252   | Y0 - Y248   | Y0 - Y244    | Y0 - Y240    | Y0 - Y236    | Y0 - Y232    | Y0 - Y228    | Y0 - Y224    |

### Komponenten in Betrieb nehmen

#### Projekt auf das Bediengerät übertragen

1. Versetzen Sie das Bediengerät in den "Transfermodus".
2. Stellen Sie alle notwendigen Transferparameter ein:
  - Schnittstelle
  - Übertragungsparameter
  - Zielspeicher
3. Starten Sie die Übertragung des Projekts.  
Das Projekt wird automatisch generiert.  
In einem Fenster werden die einzelnen Schritte der Generierung und der Übertragung mitprotokolliert.

#### Steuerung und Bediengerät miteinander verbinden

1. Verbinden Sie die Steuerung und das Bediengerät mit einem geeigneten Anschlusskabel.
2. Am Bediengerät erscheint die Meldung "Verbindung zur Steuerung .... ist hergestellt".

#### Projektierung optimieren

##### Erfassungszyklus und Aktualisierungszeit

Die in der Projektierungssoftware angegebenen Erfassungszyklen für die "Bereichszeiger" und die Erfassungszyklen der Variablen sind wesentliche Faktoren für die tatsächlich erreichbaren Aktualisierungszeiten.

Die Aktualisierungszeit ist Erfassungszyklus plus Übertragungszeit plus Verarbeitungszeit.

Um optimale Aktualisierungszeiten zu erreichen, beachten Sie bei der Projektierung Folgendes:

- Richten Sie die einzelnen Datenbereiche so klein wie möglich und so groß wie nötig ein.
- Zu klein gewählte Erfassungszyklen beeinträchtigen unnötigerweise die Gesamtperformance. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs. Richtwert für den Erfassungszyklus ca. 1 Sekunde.
- Legen Sie die Variablen einer Meldung oder eines Bildes ohne Lücken in einen Datenbereich.
- Damit Änderungen in der Steuerung sicher erkannt werden, müssen diese mindestens während des tatsächlichen Erfassungszyklus anstehen.

## Bilder

Bei Bildern hängt die tatsächlich erreichbare Aktualisierungsrate von Art und Anzahl der darzustellenden Daten ab.

Beachten Sie bei der Projektierung, dass Sie nur für diejenigen Objekte kurze Erfassungszyklen projektieren, die tatsächlich schnell aktualisiert werden müssen. Damit werden die Aktualisierungszeiten kürzer.

## Kurven

Wenn bei bitgetriggerten Kurven das Sammelbit im "Kurvenübertragungsbereich" gesetzt wird, so aktualisiert das Bediengerät jedes Mal alle Kurven, deren Bit in diesem Bereich gesetzt ist. Danach setzt es die Bits wieder zurück.

Erst wenn vom Bediengerät alle Bits zurückgesetzt wurden, darf das Sammelbit im SPS-Programm wieder gesetzt werden.

## Steuerungsaufträge

Wenn viele Steuerungsaufträge in schneller Folge zum Bedienen gesendet werden, so kann dadurch die Kommunikation zwischen Bediengerät und Steuerung überlastet werden.

Wenn das Bediengerät den Wert 0 in das erste Datenwort des Auftragsfachs einträgt, hat das Bediengerät den Steuerungsauftrag entgegengenommen. Jetzt arbeitet das Bediengerät den Auftrag ab, wofür es noch Zeit benötigt. Wenn sofort wieder ein neuer Steuerungsauftrag in das Auftragsfach eingetragen wird, kann es eine Zeit dauern, bis das Bediengerät den nächsten Steuerungsauftrag ausführt. Der nächste Steuerungsauftrag wird erst wieder entgegengenommen, wenn Rechnerleistung zur Verfügung steht.

## Mitsubishi FX

### Verbindung über Mitsubishi FX projektieren

#### Einleitung

Eine Verbindung zu einer der Steuerung mit Mitsubishi FX Kommunikationstreiber projektieren Sie im Editor "Verbindungen" des Bediengeräts.

Das Mitsubishi FX-Protokoll wird auch als Mitsubishi PG-Protokoll bezeichnet.

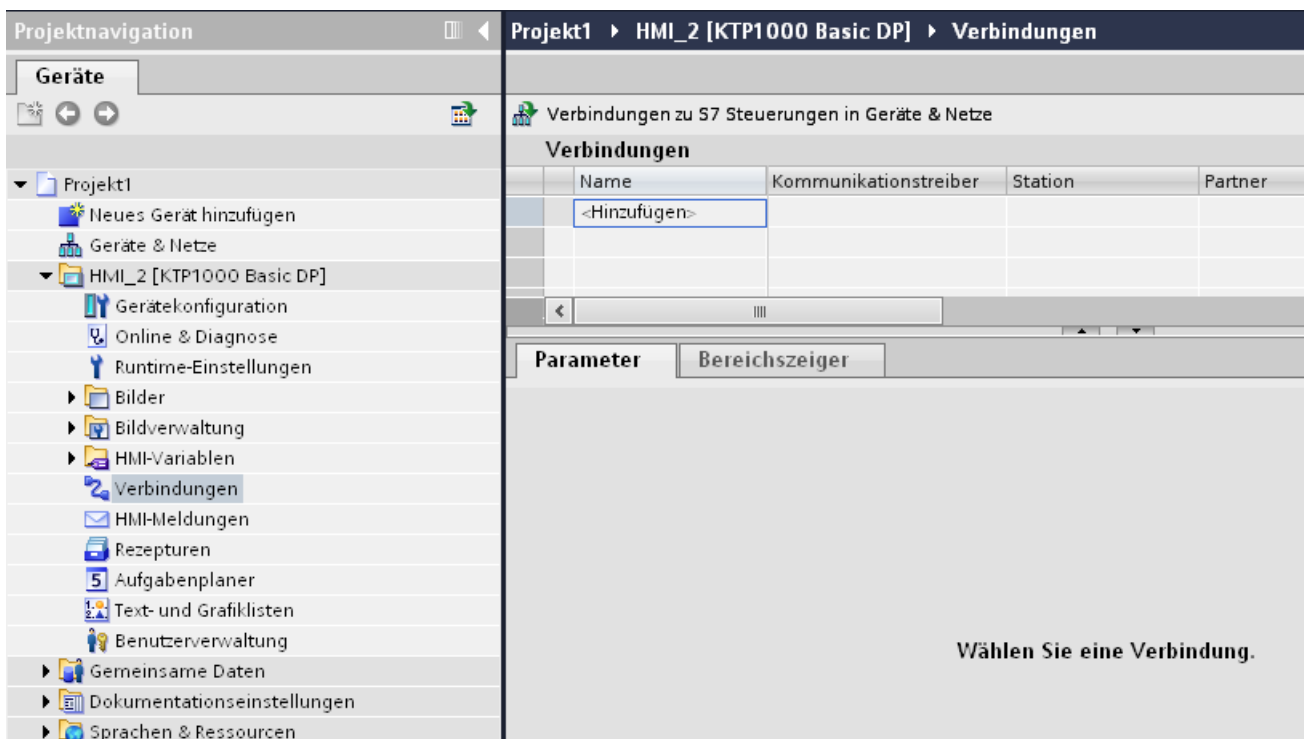
Abhängig vom Bediengerät sind die Schnittstellen unterschiedlich benannt.

#### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

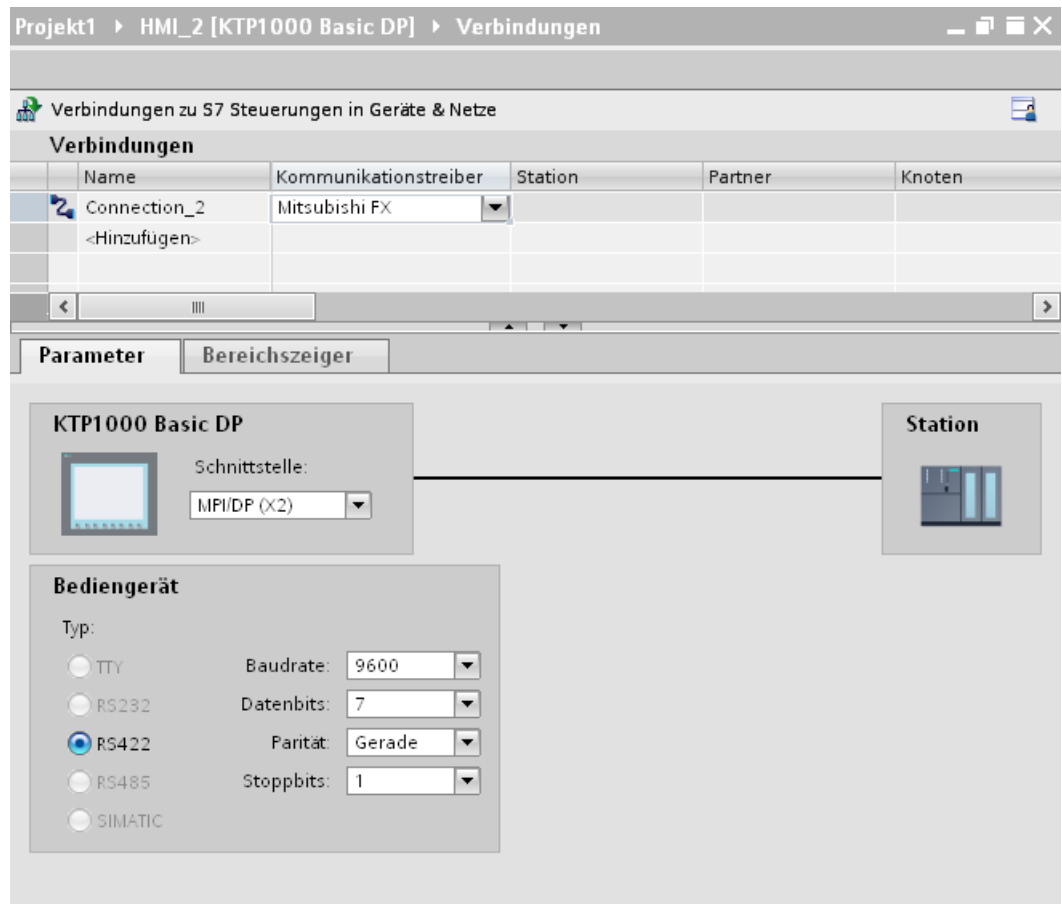
#### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".
3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".





4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "Mitsubishi FX" aus.



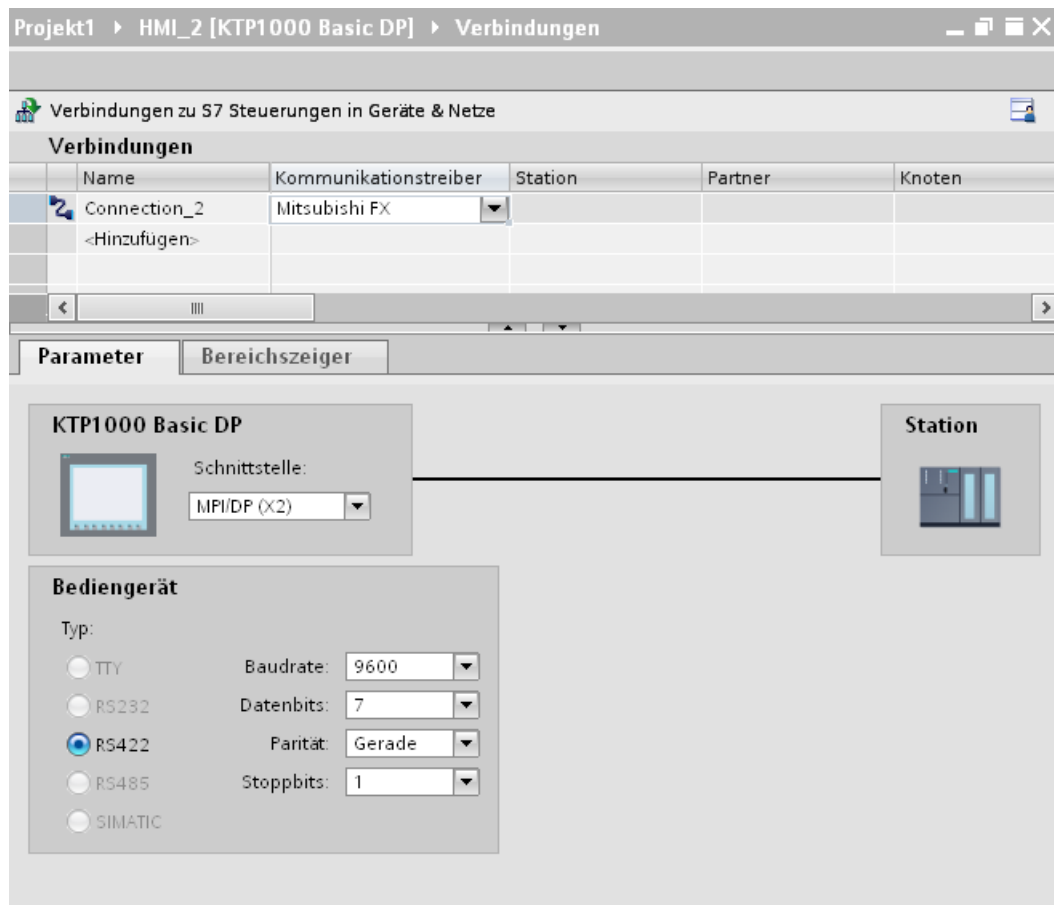
5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.

## Parameter für die Verbindung (Mitsubishi FX)

### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungsektor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät", "Netzwerk" und "Steuerung".



### Parameter für das Bediengerät

Sie können im Inspektorfenster "Parameter" für das Bediengerät eine Schnittstelle auswählen. Abhängig vom Bediengerät stehen auch mehrere Schnittstellen zur Verfügung.

- "Typ"  
Legt den benutzten physikalischen Anschluss fest.

#### Hinweis

Wenn Sie die Schnittstelle IF1B verwenden müssen Sie die RS422-Empfangsdaten und das RTS-Signal zusätzlich über 4 DIL-Schalter an der Rückseite des Bediengeräts umschalten.

### Parameter für die Steuerung

- Baudrate: Unter "Baudrate" wählen Sie die Übertragungsgeschwindigkeit zwischen Bediengerät und Steuerung aus. Wählen Sie die Baudrate 9600 aus.
- Datenbits: Unter "Datenbits" wählen Sie "7 Bit" aus.
- Parität: Unter "Parität" wählen Sie "Gerade" aus.
- Stoppbits: Unter "Stoppbits" wählen Sie "1 Bit" aus.

## Bediengerät mit Steuerung verbinden

### Kommunikationsarten

#### Freigegebene Kommunikationsarten

- Gilt nur für Mitsubishi FX(PG-Protokoll):  
Die Punkt-zu-Punkt-Kopplung von einem Bediengerät zu einer freigegebenen Mitsubishi FX-CPU über Mitsubishi FX (PG-Protokoll := Protocol for access to the program and memory elements of the FX series PC CPU version V1.21 and after) ist von der Siemens AG systemgetestet und freigegeben.
- Gilt nur für Mitsubishi MC TCP/IP:  
Folgende Kommunikationsarten sind systemgetestet und freigegeben:
  - Punkt-zu-Punkt-Kopplung zu den freigegebenen Steuerungen
  - Mehrpunkt-Kopplung von einem Bediengerät mit bis zu 4 Steuerungen, mit den jeweils freigegebenen Steuerungen. Das Mischen der CPU-Typen (FX3 und Q) ist möglich.

#### Hinweis

Das Bediengerät ist Client und die Steuerung muss als Server arbeiten.

## Koppelbare Steuerungen

Für folgende Mitsubishi-Steuerungen können Kopplungen realisiert werden:

| Steuerung   | Mitsubishi FX<br>(PG-Protokoll) | Mitsubishi MC TCP/IP |
|---|---------------------------------|----------------------|
| MELSEC FX1n, FX2n   | ja                              | nein                 |
| MELSEC FX3U, FX3UC, FX3G mit<br>Kommunikations-Modul FX3U-ENET  | nein                            | ja                   |
| MELSEC System Q   | nein                            | ja                   |
| <ul style="list-style-type: none"> <li>• Q-Series mit dem Kommunikations-Modul QJ71E71-100</li> <li>• QnUDEH CPU mit Ethernet-Schnittstelle on Board</li> </ul> |                                 |                      |

## Verbindungen über Mitsubishi FX

### Anschluss

Schließen Sie das Bediengerät an die Programmierschnittstelle der CPU (RS 422) an (siehe Dokumentation der Steuerung).

Die Kopplung zwischen dem Bediengerät und der Mitsubishi Steuerung beschränkt sich im Wesentlichen darauf, die Schnittstellenparameter einzustellen. Spezielle Bausteine für die Kopplung sind in der Steuerung nicht erforderlich.

### Verbindungskabel

Für den Anschluss des Bediengeräts an die Steuerung stehen folgende Verbindungskabel zur Verfügung:

| Schnittstelle am Bediengerät bzw. Adapter | Mitsubishi Electric Steuerung über FX-Protokoll |
|---|---|
|   | FX1n, Fx2n, Mini DIN, 8-polig                   |
| RS 232, 9-polig                           | Mitsubishi SC-09 <sup>1)</sup>                  |
| RS 422, 9-polig                           | Verbindungskabel RS422-2P                       |

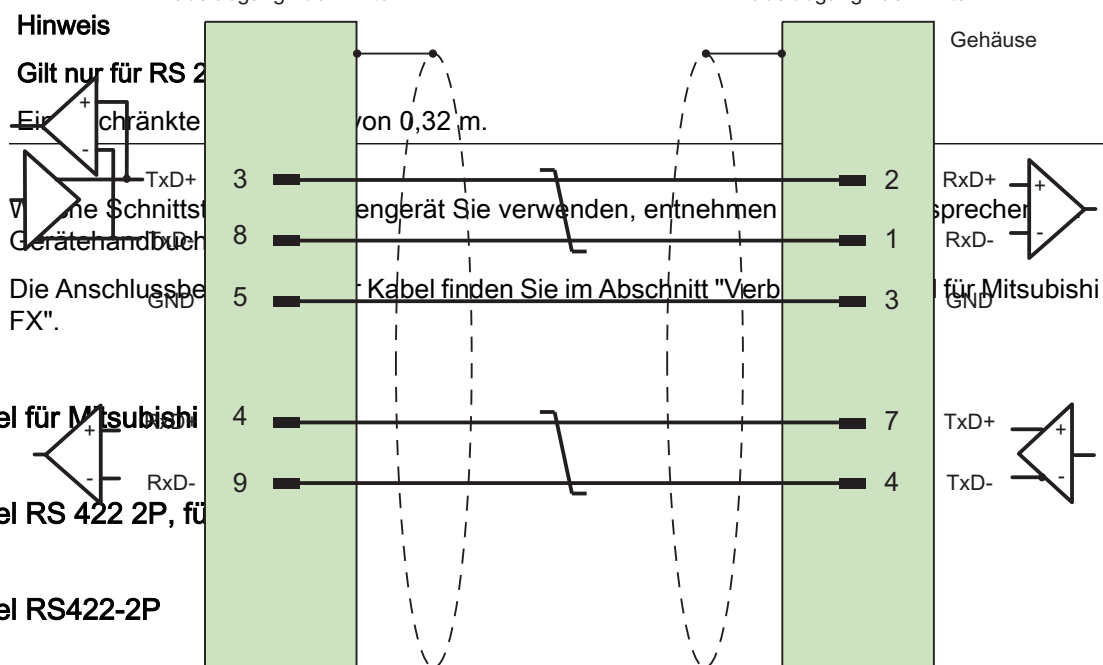
<sup>1)</sup> Da die Mitsubishi-Steuerungen standardmäßig über RS 422 kommunizieren, ist für den Anschluss eines Bediengeräts über RS 232 das Mitsubishi-Programmierkabel SC-09 mit spezieller Anpassung RS 422/RS 232 erforderlich.

| Stecker 1                 | Stecker 2                    |
|---------------------------|------------------------------|
| 9-pol. Sub D-Stiftstecker | 8-pol. Mini-DIN-Stiftstecker |
| Schraubverriegelung       | Schraubverriegelung          |
| Kabelabgang nach hinten   | Kabelabgang nach hinten      |

#### Hinweis

Gilt nur für RS 232

Die Anschlüsse sind durch die Gerätehandbücher der Steuerung zu finden.



Schirm beidseitig großflächig mit Gehäuse verbunden  
Kabel: 3 x 2 x 0,14 mm<sup>2</sup>, geschirmt,  
max. Länge 500 m

## Leistungsmerkmale der Kommunikation

## Zulässige Datentypen für Mitsubishi FX

### Zulässige Datentypen

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp     | Operandentyp   | Länge            |
|--------------|----------------|------------------|
| 4-bit block  | M, X, Y        | 1 Byte           |
| 8-bit block  | M, X, Y        | 1 Byte           |
| 12-bit block | M, X, Y        | 2 Byte           |
| 16-bit block | M, X, Y        | 2 Byte           |
| 20-bit block | M, X, Y        | 4 Byte           |
| 24-bit block | M, X, Y        | 4 Byte           |
| 28-bit block | M, X, Y        | 4 Byte           |
| 32-bit block | M, X, Y        | 4 Byte           |
| Bool         | D, M, X, Y     | 1 Bit            |
| DWord        | D, C-32-Bit    | 4 Byte           |
| Real         | D              | 4 Byte           |
| String       | D              | 1 bis 50 Zeichen |
| Word         | D, T, C-16-Bit | 2 Byte           |

### Hinweis

Beachten Sie bei schreibenden Zugriffen:

Bei Datentyp "Bool" im Operandentyp "D" wird nach der Änderung des angegebenen Bits das gesamte Wort wieder in die Steuerung zurückgeschrieben. Dabei wird nicht geprüft, ob sich zwischenzeitlich andere Bits im Wort geändert haben. Deshalb darf die Steuerung auf das angegebene Wort nur lesend zugreifen.

### Hinweis

Array-Elemente in E/A-Feldern sind bei einer Kommunikation mit Mitsubishi-Steuerung nicht zulässig.

## Unterstützte CPU-Typen für Mitsubishi FX

### CPU-Typen

Folgende CPU-Typen werden bei der Projektierung des Kommunikationstreibers Mitsubishi FX unterstützt.

- FX1 series
  - FX1n
- FX2 series
  - FX2n

### Adressbereiche für Mitsubishi FX

#### FX3

| Adressbereich | Datentypen      |            |            |            |            |            |            |             |             |              |              |              |              |              |              |  |
|---------------|-----------------|------------|------------|------------|------------|------------|------------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--|
|               | Bool            | Int        | Word       | DInt       | DWord      | Real       | String     | 4-bit block | 8-bit block | 12-bit block | 16-bit block | 20-bit block | 24-bit block | 28-bit block | 32-bit block |  |
| M             | M0 - M9999      | --         | --         | --         | --         | --         | --         | M0 - M9999  | M0 - M9999  | M0 - M9999   | M0 - M9999   | M0 - M9999   | M0 - M9999   | M0 - M9999   | M0 - M9999   |  |
| D             | D0.0 - D9999.15 | D0 - D9999 | D0 - D9999 | D0 - D9999 | D0 - D9999 | D0 - D9999 | D0 - D9999 | --          | --          | --           | --           | --           | --           | --           | --           |  |
| T             | --              | --         | T0 - T999  | --         | --         | --         | --         | --          | --          | --           | --           | --           | --           | --           | --           |  |
| C             | --              | C0 - C999  | C0 - C999  | C0 - C998  | C0 - C998  | --         | --         | --          | --          | --           | --           | --           | --           | --           | --           |  |
| X             | X0 - X777       | --         | --         | --         | --         | --         | --         | X0 - X774   | X0 - X770   | X0 - X764    | X0 - X760    | X0 - X754    | X0 - X750    | X0 - X744    | X0 - X740    |  |
| Y             | Y0 - Y777       | --         | --         | --         | --         | --         | --         | Y0 - Y774   | Y0 - Y770   | Y0 - Y764    | Y0 - Y760    | Y0 - Y754    | Y0 - Y750    | Y0 - Y744    | Y0 - Y740    |  |

Q

| Adressbereich | Datentypen       |             |             |             |             |             |             |             |             |              |              |              |              |              |              |
|---------------|------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|--------------|
|               | Bool             | Int         | Word        | Dint        | DWord       | Real        | String      | 4-bit block | 8-bit block | 12-bit block | 16-bit block | 20-bit block | 24-bit block | 28-bit block | 32-bit block |
| M             | M0 - M9999       | --          | --          | --          | --          | --          | --          | M0 - M9996  | M0 - M9992  | M0 - M9988   | M0 - M9984   | M0 - M9980   | M0 - M9976   | M0 - M9972   | M0 - M9968   |
| F             | F0 - F9999       | --          | --          | --          | --          | --          | --          | F0 - F9996  | F0 - F9992  | F0 - F9988   | F0 - F9984   | F0 - F9980   | F0 - F9976   | F0 - F9972   | F0 - F9968   |
| B             | B0 - BFFFF       | --          | --          | --          | --          | --          | --          | B0 - BFFF   | B0 - BFFF   | B0 - BFFF    | B0 - BFFF    | B0 - BFFE    | B0 - BFFE    | B0 - BFFE    | B0 - BFFE    |
| D             | D0.0 - D65534.15 | D0 - D65534 | D0 - D65534 | D0 - D65533 | D0 - D65533 | D0 - D65533 | D0 - D65534 | --          | --          | --           | --           | --           | --           | --           | --           |
| T             | --               | --          | T0 - T2047  | --          | --          | --          | --          | --          | --          | --           | --           | --           | --           | --           | --           |
| C             | --               | C0 - C2047  | C0 - C2047  | C0 - C2046  | C0 - C2046  | --          | --          | --          | --          | --           | --           | --           | --           | --           | --           |
| W             | --               | W0 - WFFF   | W0 - WFFF   | W0 - WFFE   | W0 - WFFE   | W0 - WFFE   | --          | --          | --          | --           | --           | --           | --           | --           | --           |
| X             | X0 - XFFFF       | --          | --          | --          | --          | --          | --          | X0 - XFFF   | X0 - XFFF   | X0 - XFFF    | X0 - XFFF    | X0 - XFFE    | X0 - XFFE    | X0 - XFFE    | X0 - XFFE    |
| Y             | Y0 - YFFFF       | --          | --          | --          | --          | --          | --          | Y0 - YFFF   | Y0 - YFFF   | Y0 - YFFF    | Y0 - YFFF    | Y0 - YFFE    | Y0 - YFFE    | Y0 - YFFE    | Y0 - YFFE    |

FX0

| Adressbereich | Datentypen      |            |            |            |            |             |             |              |              |              |              |              |              |  |
|---------------|-----------------|------------|------------|------------|------------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--|
|               | Bool            | Word       | DWord      | Real       | String     | 4-bit block | 8-bit block | 12-bit block | 16-bit block | 20-bit block | 24-bit block | 28-bit block | 32-bit block |  |
| M             | M0 - M9999      |            |            |            |            | M0 - M9996  | M0 - M9992  | M0 - M9988   | M0 - M9984   | M0 - M9980   | M0 - M9976   | M0 - M9972   | M0 - M9968   |  |
| D             | D0.0 - D9999.15 | D0 - D9999 | D0 - D9998 | D0 - D9998 | D0 - D9998 |             |             |              |              |              |              |              |              |  |

| Adressbereich | Datentypen |                           |                             |      |        |             |             |              |              |              |              |              |              |
|---------------|------------|---------------------------|-----------------------------|------|--------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|--------------|
|               | Bool       | Word                      | DWord                       | Real | String | 4-bit block | 8-bit block | 12-bit block | 16-bit block | 20-bit block | 24-bit block | 28-bit block | 32-bit block |
| T             |            | T0 - T255                 |                             |      |        |             |             |              |              |              |              |              |              |
| C-16-Bit      |            | C-16-Bit 0 - C-16-Bit 199 |                             |      |        |             |             |              |              |              |              |              |              |
| C-32-Bit      |            |                           | C-32-Bit 200 - C-32-Bit 255 |      |        |             |             |              |              |              |              |              |              |
| X             | X0 - X255  |                           |                             |      |        | X0 - X252   | X0 - X248   | X0 - X244    | X0 - X240    | X0 - X236    | X0 - X232    | X0 - X228    | X0 - X224    |
| Y             | Y0 - X255  |                           |                             |      |        | Y0 - Y252   | Y0 - Y248   | Y0 - Y244    | Y0 - Y240    | Y0 - Y236    | Y0 - Y232    | Y0 - Y228    | Y0 - Y224    |

### Komponenten in Betrieb nehmen

#### Projekt auf das Bediengerät übertragen

1. Versetzen Sie das Bediengerät in den "Transfermodus".
2. Stellen Sie alle notwendigen Transferparameter ein:
  - Schnittstelle
  - Übertragungsparameter
  - Zielspeicher
3. Starten Sie die Übertragung des Projekts.  
Das Projekt wird automatisch generiert.  
In einem Fenster werden die einzelnen Schritte der Generierung und der Übertragung mitprotokolliert.

#### Steuerung und Bediengerät miteinander verbinden

1. Verbinden Sie die Steuerung und das Bediengerät mit einem geeigneten Anschlusskabel.
2. Am Bediengerät erscheint die Meldung "Verbindung zur Steuerung .... ist hergestellt".



## Projektierung optimieren

### Erfassungszyklus und Aktualisierungszeit

Die in der Projektierungssoftware angegebenen Erfassungszyklen für die "Bereichszeiger" und die Erfassungszyklen der Variablen sind wesentliche Faktoren für die tatsächlich erreichbaren Aktualisierungszeiten.

Die Aktualisierungszeit ist Erfassungszyklus plus Übertragungszeit plus Verarbeitungszeit.

Um optimale Aktualisierungszeiten zu erreichen, beachten Sie bei der Projektierung Folgendes:

- Richten Sie die einzelnen Datenbereiche so klein wie möglich und so groß wie nötig ein.
- Zu klein gewählte Erfassungszyklen beeinträchtigen unnötigerweise die Gesamtperformance. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs. Richtwert für den Erfassungszyklus ca. 1 Sekunde.
- Legen Sie die Variablen einer Meldung oder eines Bildes ohne Lücken in einen Datenbereich.
- Damit Änderungen in der Steuerung sicher erkannt werden, müssen diese mindestens während des tatsächlichen Erfassungszyklus anstehen.

### Bilder

Bei Bildern hängt die tatsächlich erreichbare Aktualisierungsrate von Art und Anzahl der darzustellenden Daten ab.

Beachten Sie bei der Projektierung, dass Sie nur für diejenigen Objekte kurze Erfassungszyklen projektieren, die tatsächlich schnell aktualisiert werden müssen. Damit werden die Aktualisierungszeiten kürzer.

### Kurven

Wenn bei bitgetriggerten Kurven das Sammelbit im "Kurvenübertragungsbereich" gesetzt wird, so aktualisiert das Bediengerät jedes Mal alle Kurven, deren Bit in diesem Bereich gesetzt ist. Danach setzt es die Bits wieder zurück.

Erst wenn vom Bediengerät alle Bits zurückgesetzt wurden, darf das Sammelbit im SPS-Programm wieder gesetzt werden.

### Steuerungsaufträge

Wenn viele Steuerungsaufträge in schneller Folge zum Bedienen gesendet werden, so kann dadurch die Kommunikation zwischen Bediengerät und Steuerung überlastet werden.

Wenn das Bediengerät den Wert 0 in das erste Datenwort des Auftragsfachs einträgt, hat das Bediengerät den Steuerungsauftrag entgegengenommen. Jetzt arbeitet das Bediengerät den Auftrag ab, wofür es noch Zeit benötigt. Wenn sofort wieder ein neuer Steuerungsauftrag in das Auftragsfach eingetragen wird, kann es eine Zeit dauern, bis das Bediengerät den

nächsten Steuerungsauftrag ausführt. Der nächste Steuerungsauftrag wird erst wieder entgegengenommen, wenn Rechnerleistung zur Verfügung steht.

## Datenaustausch

### Bereichszeiger bei Mitsubishi

#### Bereichszeiger bei Verbindungen über Mitsubishi Kommunikationstreiber

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu.

Nähere Hinweise zu Bereichszeigern und deren Projektierung finden Sie im Kapitel "Datenaustausch über Bereichszeiger".

#### Besonderheiten bei Verbindungen über Mitsubishi Kommunikationstreiber

Sie können folgende Bereichszeiger projektieren

| Bereichszeiger    | Mitsubishi MC TCP/IP | Mitsubishi FX |
|-------------------|----------------------|---------------|
| Bildnummer        | ja                   | ja            |
| Datum/Uhrzeit     | ja                   | ja            |
| Datum/Uhrzeit PLC | ja                   | ja            |
| Koordinierung     | ja                   | ja            |
| Projektkennung    | ja                   | ja            |
| Steuerungsauftrag | ja                   | ja            |
| Datensatz         | ja                   | ja            |

#### Einschränkungen Mitsubishi MC TCP/IP

Folgende Einschränkungen gelten für die Projektierung von Bereichszeigern.

| CPU-Typ | Datentypen | Operandentyp |
|---------|------------|--------------|
| FX3     | Int, Word  | D            |
| Q       | Int, Word  | D            |

#### Einschränkungen Mitsubishi FX

Den Operandentyp D können Sie für die Projektierung von Bereichszeigern verwenden.

#### Siehe auch

Datenaustausch über Bereichszeiger (Seite 3821)

## Kurven

### Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt bei den Basic Panels zeitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 3201)

### Zeitgetriggerte Kurven

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein.

Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

## Meldungen

### Meldungen projektieren

#### Meldungen projektieren bei nicht integrierten Verbindungen

Um Meldungen wie Warnungen, Fehlermeldungen und Quittierung zu projektieren sind mehrere Schritte notwendig.

- Schritt 1: Variablen anlegen
- Schritt 2: Meldungen projektieren
- Schritt 3: Quittierung projektieren

Nähere Hinweise hierzu finden Sie im Kapitel:

Arbeiten mit Meldungen (Seite 3213)

#### Besonderheiten bei der Projektierung von Meldungen

Wenn Sie Verbindungen von Bediengeräten mit Steuerungen anderer Hersteller projektieren, beachten Sie folgende Besonderheiten bei der Projektierung:

- Datentypen der Variablen
- Adressierung der Variablen
- Zählweise der Bitstellen

## Datentypen

Für Verbindungen mit Mitsubishi Kommunikationstreiber, werden folgende Datentypen unterstützt:

| Steuerung   | Zulässige Datentypen    |   |
|---|-------------------------|---|
|   | Bitmeldungen            | Analogmeldungen   |
| FX1n, FX2n, FX3 series, Q-Series, iQ-Series                 | Word, Int <sup>1)</sup> | 4 bit-block, 8 bit-block, 12 bit-block, 16 bit-block, 20 bit-block, 24 bit-block, 28 bit-block, 32 bit-block, Word, DWord, Int <sup>1)</sup> , DInt <sup>1)</sup> , Real, |
| <sup>1)</sup> Nicht bei Kommunikationstreiber Mitsubishi FX |                         |   |

## Zählweise der Bitstellen

Für Verbindungen mit Mitsubishi Kommunikationstreiber, gilt folgende Zählweise:

| Zählweise der Bitstellen   | Linkes Byte |  |  |  |  |  |  |   | Rechtes Byte |  |  |  |  |  |  |  |   |
|----------------------------|-------------|--|--|--|--|--|--|---|--------------|--|--|--|--|--|--|--|---|
| In Mitsubishi-Steuerungen  | 15          |  |  |  |  |  |  | 8 | 7            |  |  |  |  |  |  |  | 0 |
| In WinCC projektieren Sie: | 15          |  |  |  |  |  |  | 8 | 7            |  |  |  |  |  |  |  | 0 |

## Einschränkungen zu Meldungen

- Mitsubishi MC TCP/IP**  
 Als Trigger-Variablen für Bitmeldungen sind nur Variablen des Operandentyp "D" und der Datentypen "Word" und "Int" zulässig. Für Bitmeldungen können Sie Arrayvariablen (Operandentyp: "D"; Datentypen: "ARRAY [x..y] of Word" oder "ARRAY [x..y] of Int") verwenden.
- Mitsubishi FX**  
 Als Trigger-Variablen für Bitmeldungen sind nur Variablen des Operandentyp "D" und des Datentypen "Word" zulässig. Für Bitmeldungen können Sie Arrayvariablen (Operandentyp "D"; Datentypen "ARRAY [x..y] of Word") verwenden."

## Quittierung von Meldungen

### Vorgehensweise

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

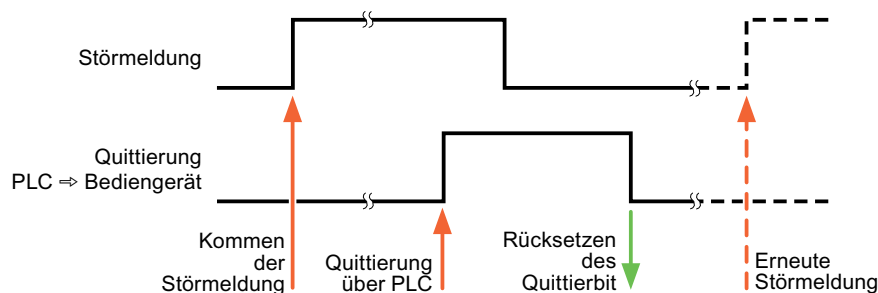
## Quittieren von der Steuerung

Unter "PLC-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulssdiagramm.



## Quittierung am Bediengerät

Unter "HMI-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittvariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittvariablen in die Steuerung eine gewisse Zeitspanne.

Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittvariablen gesetzt. Die gesamte Quittvariable wird dann vom Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

---

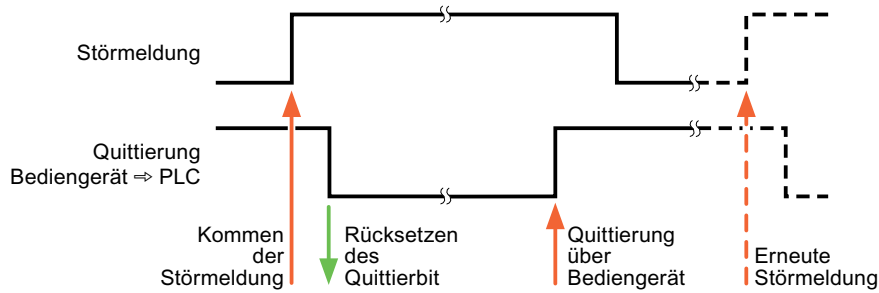
### Hinweis

Alle seit dem letzten Runtime-Start quittierten Meldebites bleiben in der Quittvariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittvariablen der gesamte Bereich vom Bediengerät überschrieben wird.

---

Das nachfolgende Bild zeigt das Impulssdiagramm.



## Modicon Modbus

### Modicon Modbus Kommunikationstreiber

#### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und Steuerungen beschrieben, die Modicon Modbus Kommunikationstreiber verwenden.

Folgende Kommunikationstreiber werden unterstützt:

- Modicon Modbus TCP/IP
- Modicon Modbus RTU

#### Datenaustausch

Datenaustausch findet über Variablen oder Bereichszeiger statt.

- Variablen  
Die Steuerung und das Bediengerät tauschen ihre Daten über Prozesswerte aus. Legen Sie in der Projektierung Variablen an, die auf Adressen in der Steuerung zeigen. Das Bediengerät liest aus der angegebenen Adresse den Wert und zeigt ihn an. Genauso kann der Bediener eine Eingabe am Bediengerät vornehmen, die dann in die Adresse in der Steuerung geschrieben wird.
- Bereichszeiger  
Bereichszeiger dienen dem Austausch spezieller Daten und werden nur bei Verwendung dieser Daten eingerichtet.

## Modicon Modbus TCP/IP

### Verbindung über Modicon Modbus TCP/IP projektieren

#### Einleitung

Eine Verbindung zu einer der Steuerung mit Modicon Modbus TCP/IP Kommunikationstreiber, projektieren Sie im Editor "Verbindungen" des Bediengeräts.

Abhängig vom Bediengerät sind die Ethernet-Schnittstellen unterschiedlich benannt.

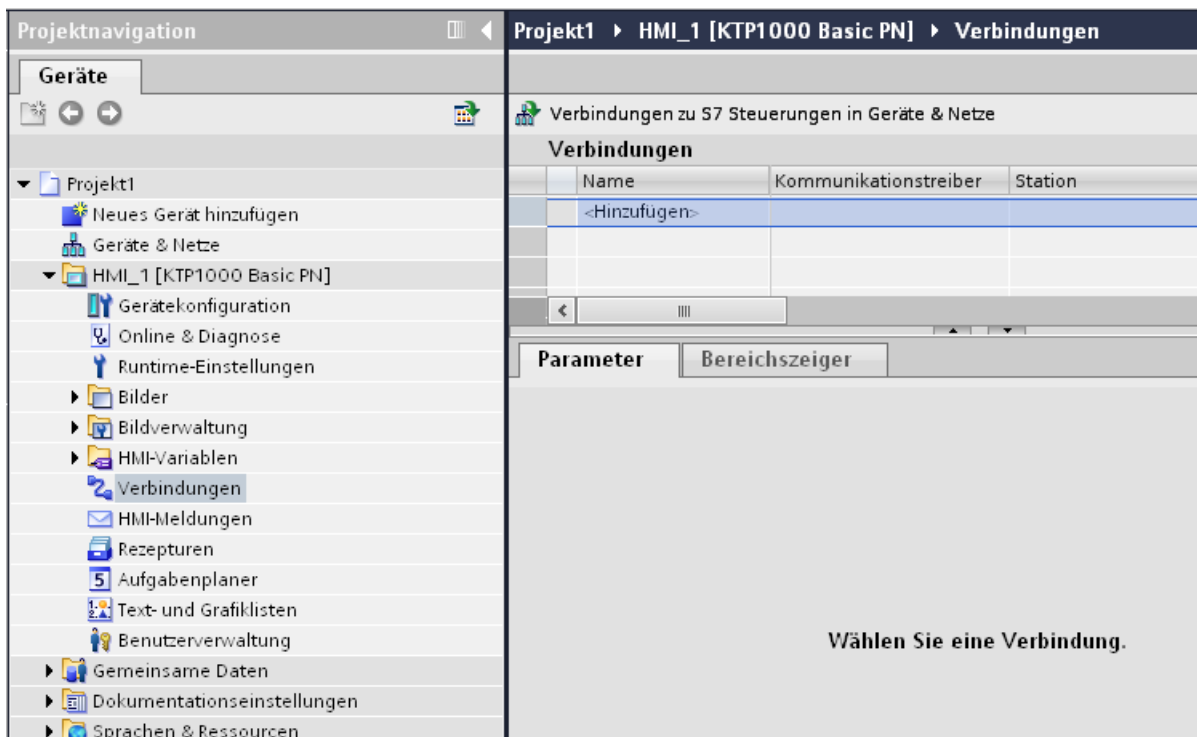
Beispiel: PROFINET-Schnittstelle entspricht der Ethernet-Schnittstelle

#### Voraussetzungen

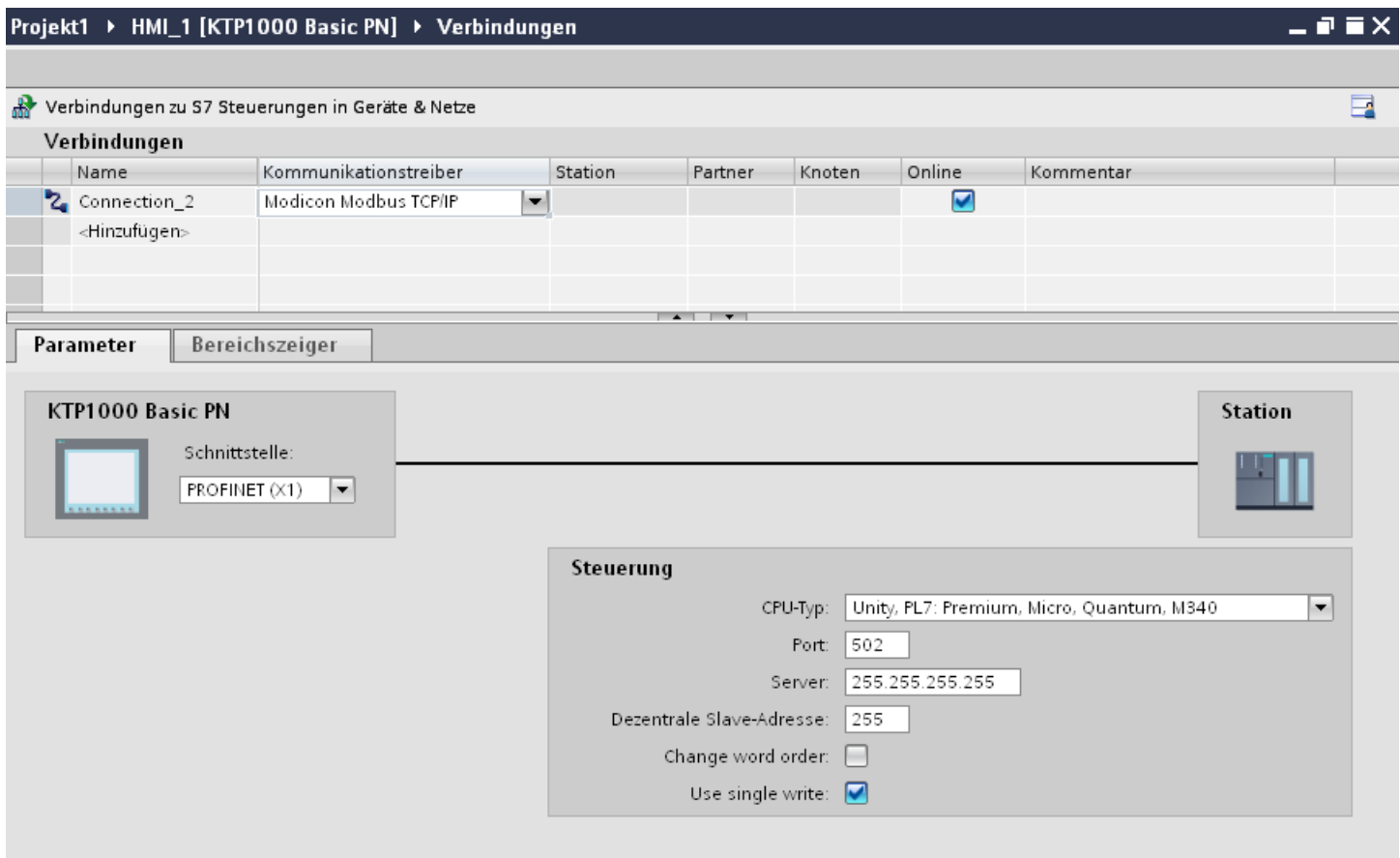
- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

#### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".
3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".



4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "Modicon Modbus TCP" aus.



5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.

## Parameter für die Verbindung (Modicon Modbus TCP/IP)

### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät" und "Steuerung".



The screenshot shows the 'Verbindungen' (Connections) window in WinCC Basic. The title bar reads 'Projekt1 > HMI\_1 [KTP1000 Basic PN] > Verbindungen'. The main area is titled 'Verbindungen zu S7 Steuerungen in Geräte & Netze'. Below this is a table with columns: Name, Kommunikationstreiber, Station, Partner, Knoten, Online, and Kommentar. One entry is visible: 'Connection\_2' with 'Modicon Modbus TCP/IP' as the driver and a checked 'Online' status. Below the table are two tabs: 'Parameter' and 'Bereichszeiger'. The 'Parameter' tab is active, showing a configuration for 'KTP1000 Basic PN'. It includes a 'Schnittstelle:' dropdown set to 'PROFINET (X1)'. To the right is a 'Station' icon. Below these is a 'Steuerung' (Control) section with the following settings: CPU-Typ: 'Unity, PL7: Premium, Micro, Quantum, M340'; Port: '502'; Server: '255.255.255.255'; Dezentrale Slave-Adresse: '255'; Change word order: unchecked; Use single write: checked.

## Parameter für das Bediengerät

Sie können im Inspektorfenster "Parameter" für das Bediengerät nur eine Schnittstelle auswählen. Abhängig vom Bediengerät stehen auch mehrere Schnittstellen zur Verfügung.

Wenn Sie beim Projektieren direkt mit dem Bediengerät verbunden sind, können Sie die IP-Adresse des Bediengeräts in WinCC einrichten. Beim Projekttransfer wird die IP-Adresse auf das Bediengerät übertragen.

---

### Hinweis

Wenn Sie die IP-Adresse bereits im Control-Panel des Bediengeräts eingerichtet haben, wird die IP-Adresse im Control Panel beim nächsten Laden überschrieben.

Wenn Sie "IP-Adresse auf anderem Weg beziehen" aktivieren, bleibt die bereits eingerichtete IP-Adresse im Control Panel beim nächsten Laden erhalten.

---

Die IP-Adresse des Bediengeräts richten Sie folgendermaßen ein:

1. Klicken Sie auf das Bediengerät.
2. Öffnen Sie den Editor "Gerätekonfiguration"

3. Klicken Sie auf die Ethernet-Schnittstelle.
4. Vergeben Sie im Inspektorfenster die IP-Adresse unter:  
"Allgemein > PROFINET-Schnittstelle > Ethernet-Adressen"

### Parameter für die Steuerung

- "CPU-Typ"  
Unter "CPU-Typ" stellen Sie ein, an welcher Modicon-Steuerung das Bediengerät angeschlossen ist.
- "Port"  
Unter "Port" stellen Sie den Port ein, der für die TCP/IP-Verbindung verwendet wird. Der von den Modicon-Steuerungen verwendete Port ist 502.
- "Server"  
Unter "Server" stellen Sie die IP-Adresse oder den Host-Namen der Steuerung ein. Auf einem Basic Panel kann nur die IP-Adresse verwendet werden.
- "Remote Slave address"  
Unter "Remote Slave address" stellen Sie nur bei Verwendung einer Bridge ein, welche Slave-Adresse die entfernte Steuerung besitzt.  
Wenn keine Bridge verwendet wird, muss der Default-Wert 255 (oder 0) stehen bleiben.
- "Change word order"  
Der Parameter "Change word order" hat nur Auswirkung auf die Word-Reihenfolge der Darstellung der 32-Bit-Werte. Die Einstellung betrifft die Datentypen Double, Double+/- und Float. Die Byte-Reihenfolge ist nicht änderbar.
  - "Change word order" nicht aktiviert  
Das Most Significant Byte wird zuerst gesendet.  
Bei Doppelworten wird das "Least Significant Word" vor dem "Most Significant Word" gesendet.  
Diese Einstellung ist systemgetestet für alle freigegebenen Steuerungen.
  - "Change word order" aktiviert  
Das Most Significant Byte wird zuerst gesendet.  
Bei Doppelworten wird das "Most Significant Word" vor dem "Least Significant Word" gesendet.

---

#### Hinweis

Diese Einstellung muss für die SIEMENS Multifunktionsmessgeräte SENTRON PAC3200 und PAC4200 verwendet und kann für weitere Steuerungen anderer Hersteller verwendet werden.

---

- "Use single write"  
Wenn Sie diese Funktion abwählen, dann werden beim Schreiben in die Steuerung nur die Funktionscodes 15H und 16H verwendet.  
Wenn diese Funktion angewählt bleibt, dann werden die Funktionscodes 05H, 06H 16H und 16H verwendet.

## Bediengerät mit Steuerung verbinden

### Verbindungen über Modicon Modbus TCP/IP

#### Anschluss

Das Bediengerät kann über folgende Komponenten an die Modicon Modbus Steuerung angebunden werden:

- Bestehendes Ethernet-Netz, in dem sich auch die Steuerungen befinden
- Gekreuztes Ethernet-Kabel (Cross-Over), direkt an die Ethernetschnittstelle der CPU bzw. des Kommunikationsmoduls

Die Kopplung des Bediengeräts an eine Modicon Modbus Steuerung beschränkt sich hauptsächlich auf den physikalischen Anschluss des Bediengeräts. Spezielle Bausteine für die Kopplung sind in der Steuerung nicht erforderlich.

---

#### Hinweis

Timeout-verhalten bei TCP/IP (Ethernet)

Durch die Verwendung des TCP/IP-Protokolls werden Verbindungsunterbrechungen frühestens nach ca. einer Minute erkannt. Werden keine Variablen angefordert, z. B. keine Ausgabevariable im aktuellen Bild, so wird eine Verbindungsunterbrechung nicht sicher erkannt.

Projektieren Sie einen Bereichszeiger Koordination für jede Steuerung. Diese Einstellung stellt sicher, dass auch im beschriebenen Fall die Unterbrechung der Verbindung nach ca. zwei Minuten erkannt wird.

---

## Kommunikationsarten

### Freigegebene Kommunikationsarten

Folgende Kommunikationsarten sind systemgetestet und freigegeben:

- Punkt-zu-Punkt-Kopplung:
- Mehrpunkt-Kopplung von einem Bediengerät (Modbus TCP/IP-Client) mit bis zu 4 Steuerungen, mit jeweils unterschiedlichen Kopplungen. Das Mischen der CPU-Typen ist möglich.

Folgende Kopplungen sind dabei möglich:

- Kopplung an die Ethernet CPU-Schnittstelle der TSX Unity Quantum.
- Kopplung über die Kommunikationsmodule für Ethernet 140 NOE 771 01 für die Serien TSX Quantum und TSX Unity Quantum
- Kopplung über die Ethernet-Schnittstelle des CPU-Adapters 171 CCC 980 30 der Serie Momentum
- Kopplung an die Ethernet CPU-Schnittstelle der TSX Unity Premium
- Kopplung über die Ethernet TCP/IP-Anschaltmodule TSX ETY 110 für die Serien TSX Premium und TSX Unity Premium
- Kopplung über die Ethernet TCP/IP-Anschaltmodule TSX ETY 410 für die Serie Micro
- Kopplung über die Ethernet TCP/IP-Modbus Plus Bridge 174 CEV 200 40 an die Modbus Plus-Schnittstelle der Compact, der TSX Quantum und der TSX Unity Quantum

Über die TCP/IP-Modbus Plus Bridge 174 CEV 200 40 sind die Steuerungen unter ihrer Remote Slave-Adresse über die Ethernet-Schnittstelle dieser Bridge erreichbar.

---

#### Hinweis

Eine Integration des Bediengeräts über eine Bridge in ein Modbus-Netzwerk ist nicht möglich. Das Bediengerät ist Modbus-Master.

---

### Einschränkungen

Die Kopplung des Bediengeräts an Steuerungen anderer Hersteller, die eine Modbus TCP/IP-Schnittstelle anbieten, ist nicht systemgetestet und damit nicht freigegeben.

Falls Sie dennoch eine andere Steuerung verwenden, beachten Sie folgende Hinweise:

- Verwenden Sie folgende CPU-Typen, da diese ohne Adressoffset und mit der üblicheren Bitzählweise arbeitet.
  - Unity, PL7: Premium, Micro, Quantum, M340
- Folgende Funktionscodes werden für die jeweiligen Datenbereiche verwendet:

| Lesende Funktionscodes |                 | Adressbereich |             |
|------------------------|-----------------|---------------|-------------|
| 01                     | ReadCoilStatus  | 0x / %M       | DIGITAL_OUT |
| 02                     | ReadInputStatus | 1x / %I       | DIGITAL_IN  |

| Lesende Funktionscodes |                      | Adressbereich |  |
|------------------------|----------------------|---------------|--|
| 03                     | ReadHoldingRegisters | 4x / %MW      | USERDATA                                 |
| 04                     | ReadInputRegisters   | 3x / %IW      | ANALOG_IN                                |
| 20 (14Hex)             | ReadGeneralReference | 6x / –        | EXTENDEDMEMORY<br>(Nicht bei allen CPUs) |

| Schreibende Funktionscodes |                         | Adressbereich |  |
|----------------------------|-------------------------|---------------|--|
| 06 <sup>1)</sup>           | PresetSingleRegister    | 4x / %MW      | USERDATA Single                          |
| 16 (10Hex)                 | PresetMultipleRegisters | 4x / %MW      | USERDATA Multiple                        |
| 05 <sup>1)</sup>           | ForceSingelCoil         | 0x / %M       | DIGITAL_OUT mit BIT                      |
| 15 (0FHex)                 | ForceMultipleCoils      | 0x / %M       | DIGITAL_OUT mit 16 BIT GROUP             |
| 21 (15Hex)                 | WriteGeneralReference   | 6x / –        | EXTENDEDMEMORY<br>(Nicht bei allen CPUs) |

<sup>1)</sup> Verwendung über "Use single write" anwählbar.

## Koppelbare Steuerungen

Für folgende Modicon Modbus Steuerungen können Kopplungen realisiert werden:

| Modicon Modbus Steuerung | Unterstütztes Protokoll          |                       |
|--------------------------|----------------------------------|-----------------------|
|                          | Modicon Modbus RTU <sup>2)</sup> | Modicon Modbus TCP/IP |
| TSX Compact              | x                                | x <sup>1)</sup>       |
| TSX Quantum              | x                                | x                     |
| Momentum                 | x                                | x                     |
| Premium                  | -                                | x                     |
| Micro                    | -                                | x                     |
| M340 20x0 (ohne 2010)    | -                                | x                     |

<sup>1)</sup> Nur über Ethernet TCP/IP-Modbus Plus Bridge

<sup>2)</sup> Steuerungsseitig ist die Kommunikation über RS 232 getestet und freigegeben. Bei den HMIs, die nur eine RS 422/485-Schnittstelle haben, wurde der RS 422/232-Konverter mit der Bestellnummer 6AV6 671-8XE00-0AX0 getestet und freigegeben.

## Leistungsmerkmale der Kommunikation

### Zulässige Datentypen für Modicon Modbus TCP/IP

#### Zulässige Datentypen

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

---

#### Hinweis

Wenn Sie den Kommunikationstreiber Modicon Modbus RTU auf Modicon Modbus TCP/IP umstellen, dann kann die Zeichenfolge im Datentyp "String" unterschiedlich sein.

---

### Zulässige Datentypen für CPU-Typ "Unity, PLC: Premium, Micro, Quantum M340"

| Datentyp     | Operandentyp        | Länge            |
|--------------|---------------------|------------------|
| +/- Double   | %MW                 | 4 Byte           |
| +/- Int      | %MW, %IW            | 2 Byte           |
| 16 bit group | %MW, %I             | 2 Byte           |
| ASCII        | %MW                 | 0 bis 80 Zeichen |
| Bit          | %MW, %IW,<br>%M, %I | 1 Bit            |
| Double       | %MW                 | 4 Byte           |
| Float        | %MW                 | 4 Byte           |
| Int          | %MW, %IW            | 2 Byte           |

---

#### Hinweis

Für folgende CPU-Typen werden die Bereiche "%I" und "%IW" nicht unterstützt:

- Premium
  - Micro
  - M340
- 

### Zulässige Datentypen für CPU-Typ "Concept, ProWORX: Compact, Quantum, Momentum"

| Datentyp     | Operandentyp | Länge            |
|--------------|--------------|------------------|
| +/- Double   | 4x, 6x       | 4 Byte           |
| +/- Int      | 3x, 4x, 6x   | 2 Byte           |
| 16 bit group | 0x, 1x       | 2 Byte           |
| ASCII        | 4x, 6x       | 0 bis 80 Zeichen |

| Datentyp | Operandentyp       | Länge  |
|----------|--------------------|--------|
| Bit      | 0x, 1x, 3x, 4x, 6x | 1 Bit  |
| Double   | 4x, 6x             | 4 Byte |
| Float    | 4x, 6x             | 4 Byte |
| Int      | 3x, 4x, 6x         | 2 Byte |

### Bitzählweise

Die bei folgenden CPU-Typen wird die übliche Bitzählweise "16 LSB - 1 MSB" nur im Editor "HMI-Variablen" bei ausgewähltem Datentyp "Bit" verwendet:

- Concept, ProWORX: Compact, Quantum, Momentum

Folgende Zuordnung der Bitstellen gilt:

|                         | Linkes Byte |   |   |   |   |   |   |   | Rechtes Byte |    |    |    |    |    |    |    |
|-------------------------|-------------|---|---|---|---|---|---|---|--------------|----|----|----|----|----|----|----|
| Zählweise bei Variablen | 1           | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9            | 10 | 11 | 12 | 13 | 14 | 15 | 16 |

### Darstellung von "Signed"

Der Platzhalter "+/-" steht für die Datentypen "Signed Int" und "Signed Double".

### Unterstützte CPU-Typen für Modicon Modbus TCP/IP

#### CPU-Typen

Folgende CPU-Typen werden bei der Projektierung des Kommunikationstreibers Modicon Modbus TCP/IP unterstützt..

- Compact
- Momentum
- Quantum
  - Concept Quantum
  - Unity Quantum
- Micro
- Premium
- Modicon M340
  - 20x0 (außer 2010)

Adressbereiche für Modicon Modbus TCP/IP

UnityPI7

| Adressbereich | Datentypen                  |                             |                        |                        |                        |                        |                        |                        |
|---------------|-----------------------------|-----------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
|               | Bool                        | 16 Bit Group                | Int                    | +/- Int                | DInt                   | +/- DInt               | Float                  | ASCII                  |
| %I            | %I0 -<br>%I65535            | %I65535<br>%I0 -<br>%I65520 | --                     | --                     | --                     | --                     | --                     | --                     |
| %M            | %M0 -<br>%M65535            | %M65535<br>%M0 -<br>%M65520 | --                     | --                     | --                     | --                     | --                     | --                     |
| %IW           | %IW0.0 -<br>%IW65535.<br>15 | --                          | %IW0 -<br>%IW65535     | %IW0 -<br>%IW65535     | --                     | --                     | --                     | --                     |
| %MW           | %MW0.0 -<br>%MW6553<br>5.15 | --                          | %MW0 -<br>%MW6553<br>5 | %MW0 -<br>%MW6553<br>5 | %MW0 -<br>%MW6553<br>4 | %MW0 -<br>%MW6553<br>4 | %MW0 -<br>%MW6553<br>4 | %MW0 -<br>%MW6553<br>5 |

ConceptProWORX

| Adressbereich | Datentypen                             |                        |                           |                           |                           |                           |                           |                           |
|---------------|--|------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|
|               | Bool                                   | 16 Bit Group           | Int                       | +/- Int                   | DInt                      | +/- DInt                  | Float                     | ASCII                     |
| 0x            | 0x1 -<br>0x65535                       | 0x1 -<br>0x65520       | --                        | --                        | --                        | --                        | --                        | --                        |
| 1x            | 1x100001 -<br>1x165535                 | 1x100001 -<br>1x165520 | --                        | --                        | --                        | --                        | --                        | --                        |
| 3x            | 3x300001.1<br>-<br>3x365535.1<br>6     | --                     | 3x300001 -<br>3x365535    | 3x300001 -<br>3x365535    | --                        | --                        | --                        | --                        |
| 4x            | 4x400001.1<br>-<br>4x465535.1<br>6     | --                     | 4x400001 -<br>4x465535    | 4x400001 -<br>4x465535    | 4x400001 -<br>4x465534    | 4x400001 -<br>4x465534    | 4x400001 -<br>4x465534    | 4x400001 -<br>4x465535    |
| 6x            | 6x60000.1:<br>1 -<br>6x69999.16<br>:10 | --                     | 6x60000:1 -<br>6x69999:10 | 6x60000:1 -<br>6x69999:10 | 6x60000:1 -<br>6x69998:10 | 6x60000:1 -<br>6x69998:10 | 6x60000:1 -<br>6x69998:10 | 6x60000:1 -<br>6x69999:10 |



## Komponenten in Betrieb nehmen

### Projekt auf das Bediengerät übertragen

1. Versetzen Sie das Bediengerät in den "Transfermodus".
2. Stellen Sie alle notwendigen Transferparameter ein:
  - Schnittstelle
  - Übertragungsparameter
  - Zielspeicher
3. Starten Sie die Übertragung des Projekts.  
Das Projekt wird automatisch generiert.  
In einem Fenster werden die einzelnen Schritte der Generierung und der Übertragung mitprotokolliert.

### Steuerung und Bediengerät miteinander verbinden

1. Verbinden Sie die Steuerung und das Bediengerät mit einem geeigneten Anschlusskabel.
2. Am Bediengerät erscheint die Meldung "Verbindung zur Steuerung .... ist hergestellt".

## Projektierung optimieren

### Erfassungszyklus und Aktualisierungszeit

Die in der Projektierungssoftware angegebenen Erfassungszyklen für die "Bereichszeiger" und die Erfassungszyklen der Variablen sind wesentliche Faktoren für die tatsächlich erreichbaren Aktualisierungszeiten.

Die Aktualisierungszeit ist Erfassungszyklus plus Übertragungszeit plus Verarbeitungszeit.

Um optimale Aktualisierungszeiten zu erreichen, beachten Sie bei der Projektierung Folgendes:

- Richten Sie die einzelnen Datenbereiche so klein wie möglich und so groß wie nötig ein.
- Zu klein gewählte Erfassungszyklen beeinträchtigen unnötigerweise die Gesamtperformance. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs. Richtwert für den Erfassungszyklus ca. 1 Sekunde.
- Legen Sie die Variablen einer Meldung oder eines Bildes ohne Lücken in einen Datenbereich.
- Damit Änderungen in der Steuerung sicher erkannt werden, müssen diese mindestens während des tatsächlichen Erfassungszyklus anstehen.

## Bilder

Bei Bildern hängt die tatsächlich erreichbare Aktualisierungsrate von Art und Anzahl der darzustellenden Daten ab.

Beachten Sie bei der Projektierung, dass Sie nur für diejenigen Objekte kurze Erfassungszyklen projektieren, die tatsächlich schnell aktualisiert werden müssen. Damit werden die Aktualisierungszeiten kürzer.

## Kurven

Wenn bei bitgetriggerten Kurven das Sammelbit im "Kurvenübertragungsbereich" gesetzt wird, so aktualisiert das Bediengerät jedes Mal alle Kurven, deren Bit in diesem Bereich gesetzt ist. Danach setzt es die Bits wieder zurück.

Erst wenn vom Bediengerät alle Bits zurückgesetzt wurden, darf das Sammelbit im SPS-Programm wieder gesetzt werden.

## Steuerungsaufträge

Wenn viele Steuerungsaufträge in schneller Folge zum Bedienen gesendet werden, so kann dadurch die Kommunikation zwischen Bediengerät und Steuerung überlastet werden.

Wenn das Bediengerät den Wert 0 in das erste Datenwort des Auftragsfachs einträgt, hat das Bediengerät den Steuerungsauftrag entgegengenommen. Jetzt arbeitet das Bediengerät den Auftrag ab, wofür es noch Zeit benötigt. Wenn sofort wieder ein neuer Steuerungsauftrag in das Auftragsfach eingetragen wird, kann es eine Zeit dauern, bis das Bediengerät den nächsten Steuerungsauftrag ausführt. Der nächste Steuerungsauftrag wird erst wieder entgegengenommen, wenn Rechnerleistung zur Verfügung steht.

## Modicon Modbus RTU

### Verbindung über Modicon Modbus RTU projektieren

#### Einleitung

Eine Verbindung zu einer der Steuerung mit Modicon Modbus RTU Kommunikationstreiber, projektieren Sie im Editor "Verbindungen" des Bediengeräts.

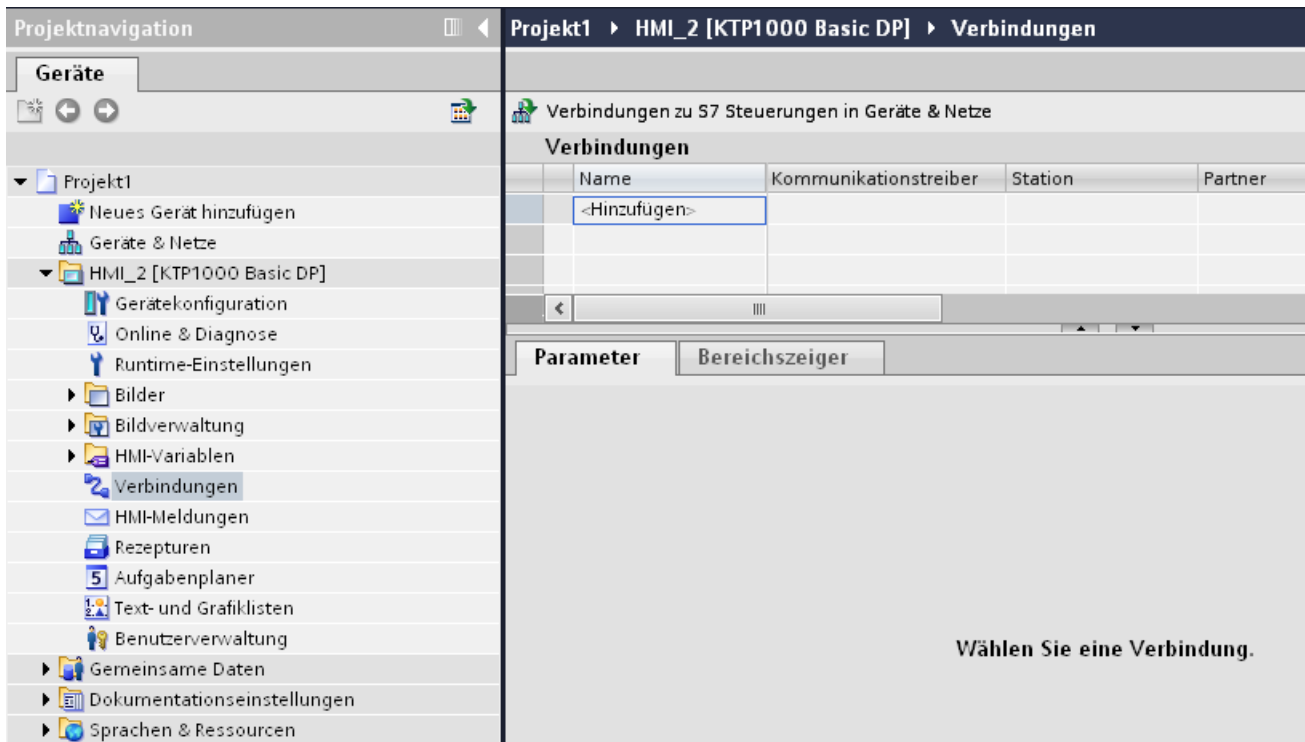
Abhängig vom Bediengerät sind die Schnittstellen unterschiedlich benannt.

#### Voraussetzungen

- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

## Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".
3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".



Projektnavigation

Projekt1 > HMI\_2 [KTP1000 Basic DP] > Verbindungen

Geräte

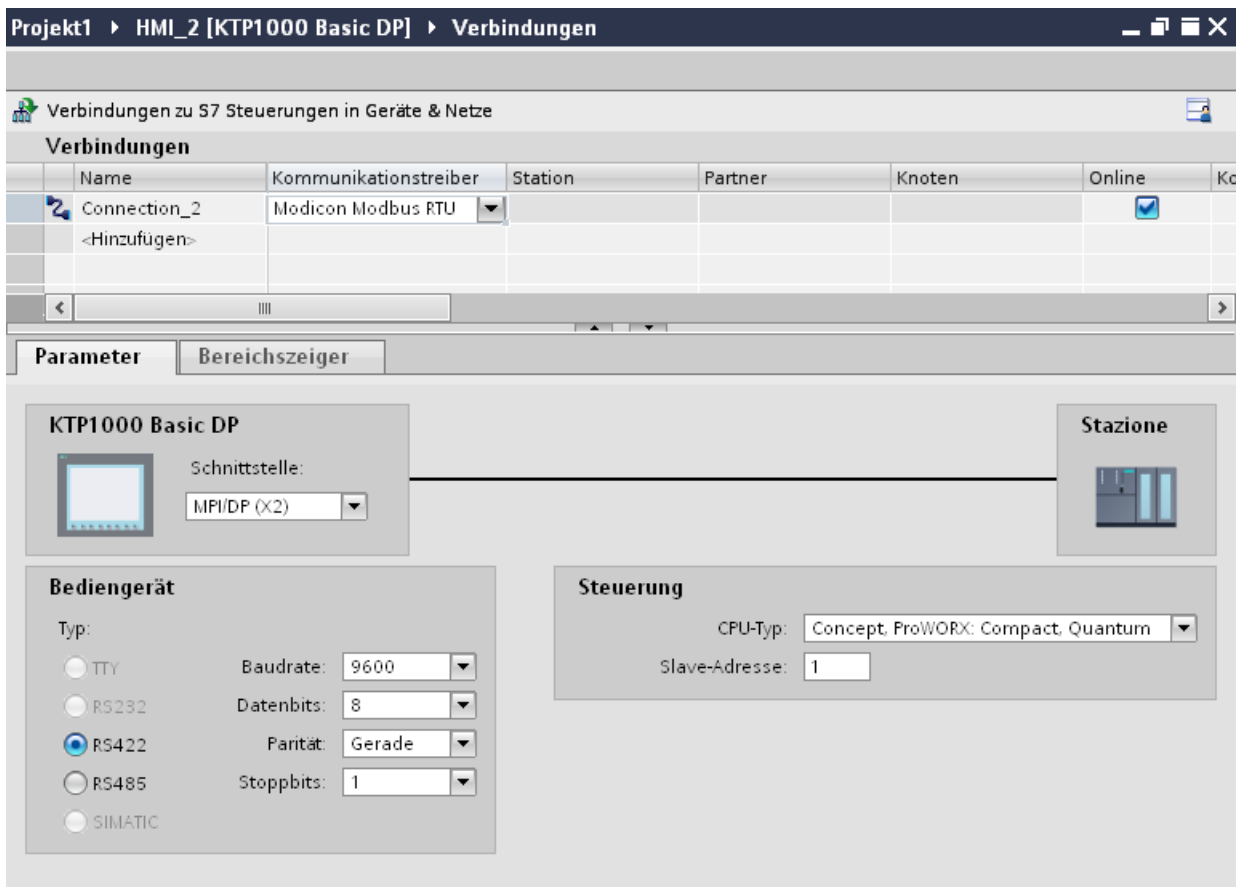
Verbindungen zu 57 Steuerungen in Geräte & Netze

| Name         | Kommunikationstreiber | Station | Partner |
|--------------|-----------------------|---------|---------|
| <Hinzufügen> |                       |         |         |

Parameter Bereichszeiger

Wählen Sie eine Verbindung.

4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "Modicon Modbus RTU" aus.



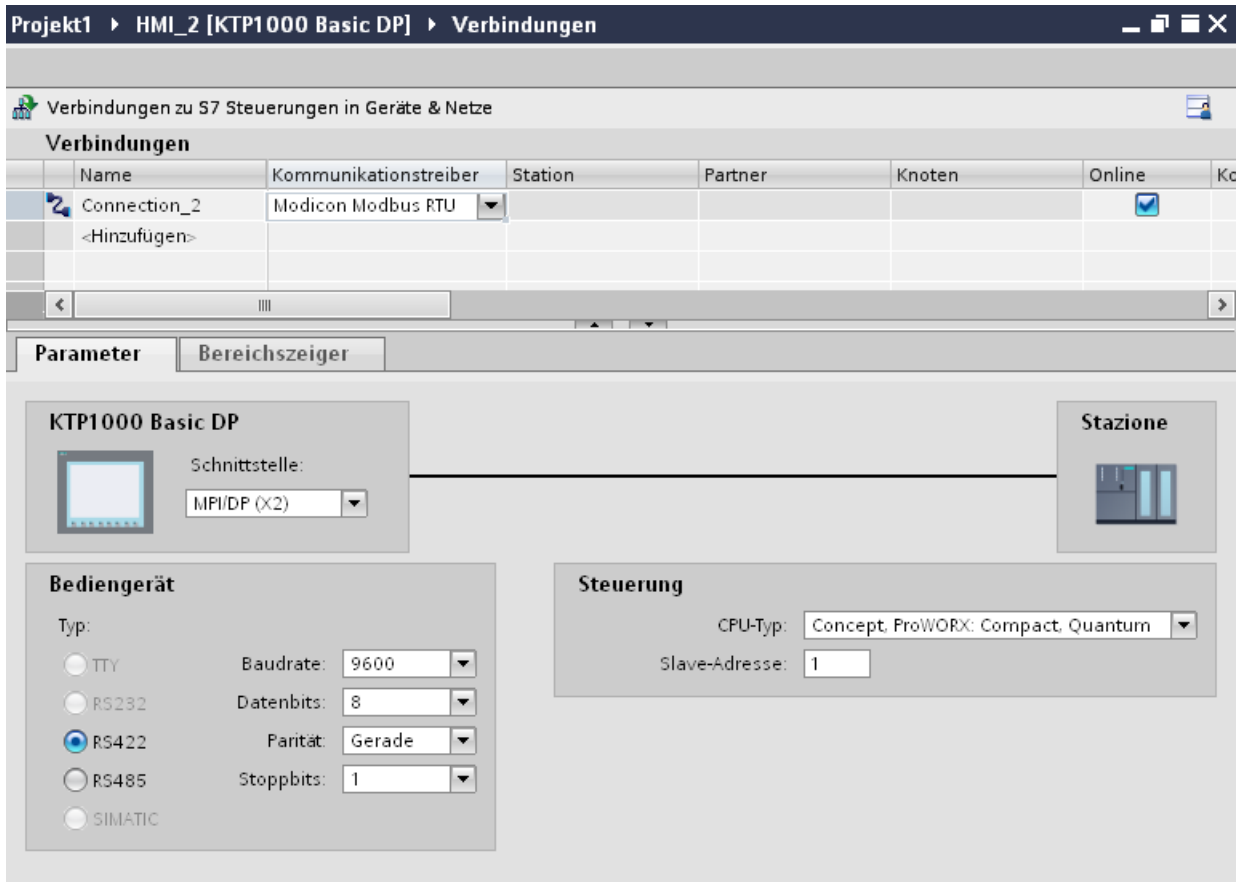
5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.

### Parameter für die Verbindung (Modicon Modbus RTU)

#### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät" und "Steuerung".



## Parameter für das Bediengerät

Sie können im Inspektorfenster "Parameter" für das Bediengerät eine Schnittstelle auswählen. Abhängig vom Bediengerät stehen auch mehrere Schnittstellen zur Verfügung.

- Typ  
Nur RS 232 ist systemgetestet.  
Für RS 485 wird keine Gewährleistung übernommen.

---

### Hinweis

RS 422 ist nur in Kombination mit dem Konverter RS 422-RS 232 freigegeben.

Bestellnummer: 6AV6 671-8XE00-0AX0

---

### Hinweis

Wenn Sie die Schnittstelle IF1B verwenden müssen Sie die RS422-Empfangsdaten zusätzlich über 4 DIL-Schalter an der Rückseite des Bediengeräts umschalten.

---

- Baudrate  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit zwischen Bediengerät und Modicon-Steuerung ein. Die Kommunikation ist mit einer Baudrate von 19200, 9600 Baud möglich.  
Für bestimmte Bediengeräte ist die Baudrate 4800 Baud auswählbar.

- Datenbits  
Unter "Datenbits" können Sie nur "8" wählen.
- Parität  
Unter "Parität" können Sie zwischen "Keine", "Gerade" und "Ungerade" wählen.
- Stoppbits  
Unter "Stoppbits" können Sie zwischen 1 und 2 wählen.

### Parameter für die Steuerung

- CPU-Typ  
Unter "CPU-Typ" stellen Sie ein, an welchen Modicon-Controller das Bediengerät angeschlossen ist.  
Sie können folgende CPUs auswählen:
  - Concept, ProWORX: Compact, Quantum
- Slaveadresse  
Unter "Slaveadresse" stellen Sie ein, welche Slaveadresse die CPU besitzt.

### Bediengerät mit Steuerung verbinden

#### Verbindungen über Modicon Modbus RTU

#### Anschluss

Schließen Sie das Bediengerät an die Modicon Modbus RTU Schnittstelle des Modicon Modbus-RTU Slave an.

Die Kopplung des Bediengeräts an Modicon beschränkt sich hauptsächlich auf den physikalischen Anschluss des Bediengeräts. Spezielle Bausteine für die Kopplung sind in der Steuerung nicht erforderlich.

#### Verbindungskabel

Für den Anschluss des Bediengeräts an Modicon Modbus stehen die folgenden Anschlusskabel zur Verfügung:

| Schnittstelle am Bediengerät | Modicon-Steuerung   |                         |   |
|------------------------------|---|-------------------------|---|
|                              | direkt über Modbus-Schnittstelle (RS232) 9-polig Sub D-Stiftstecker | über MB Bridge (RS 232) | direkt über Modbus-Schnittstelle (RS232) 8 Pin-RJ45-Stecker |
| RS 232, 9-polig              | PP1   | PP1                     | PP2   |

Die Anschlussbelegungen der Kabel finden Sie im Abschnitt "Verbindungskabel für Modicon Modbus RTU".

## Kommunikationsarten

### Freigegebene Kommunikationsarten

Folgende Kommunikationsarten sind systemgetestet und freigegeben:

- Punkt-zu-Punkt-Kopplung nur über die RS-232-Schnittstelle.
- Mehrpunkt-Kopplung von einem Bediengerät (Modbus-Master) mit bis zu 4 Steuerungen: Das Bediengerät muss mit einer Modbus Plus Bridge verbunden sein oder einer Compact, Momentum-CPU oder TSX Quantum-CPU, die als Modbus Plus Bridge konfiguriert ist.
- Die weiteren Steuerungen schließen Sie über den Modbus Plus-Anschluss an der ersten Steuerung an. Die Steuerungen sind unter ihrer Adresse über die Bridge-Funktionalität der ersten Steuerung erreichbar.

---

#### Hinweis

Nicht möglich ist die Integration des Bediengeräts in ein Modbus -Netzwerk, da das Bediengerät Modbus -Master ist.

---

- Die Integration des Bediengeräts in ein Modbus Plus-Netzwerk über den "Bridge-Modus" der Compact, Momentum oder Quantum (logische Punkt-zu-Punkt Kommunikation des Bediengeräts mit einer Compact, Momentum oder Quantum).

### Einschränkungen

Die Kopplung des Bediengeräts an Steuerungen anderer Hersteller, die eine Modicon Modbus-Schnittstelle anbieten, ist nicht systemgetestet und damit nicht freigegeben.

Falls Sie dennoch eine andere Steuerung verwenden, beachten Sie die folgenden Hinweise:

- Dieser Treiber arbeitet nur bei Variablen mit der für Modicon-Steuerungen üblichen Bitzählweise von links (Bit1 = höchst wertiges Bit) nach rechts (Bit16 = niederwertiges Bit bei Datentyp INT).
- Der bei der Projektierung angezeigte Adressoffset wird auf Protokoll-Ebene im Telegramm subtrahiert. Z. B. ist bei dem Holding Register 4x der Offset "40001". Somit wird aus der projektierten Adresse "40006" im Telegramm die Adresse "5". Die Umsetzung der im Telegramm übertragenen Adresse (z. B. "5") auf den steuerungsspezifischen Adressbereich erfolgt in den verschiedenen Nicht-Modicon-Steuerungen unterschiedlich.
- Innerhalb von 500 ms wird ein Antworttelegramm ohne "ExceptionCode" erwartet.
- Folgende Funktionscodes werden für die jeweiligen Datenbereiche verwendet:

| Lesende Funktionscodes |                      | Adressbereich |  |
|------------------------|----------------------|---------------|--|
| 01                     | ReadCoilStatus       | 0x            | DIGITAL_OUT                              |
| 02                     | ReadInputStatus      | 1x            | DIGITAL_IN                               |
| 03                     | ReadHoldingRegisters | 4x            | USERDATA                                 |
| 04                     | ReadInputRegisters   | 3x            | ANALOG_IN                                |
| 20 (14Hex)             | ReadGeneralReference | 6x            | EXTENDEDMEMORY<br>(Nicht bei allen CPUs) |

| Schreibende Funktionscodes |                         | Adressbereich |                                       |
|----------------------------|-------------------------|---------------|---------------------------------------|
| 06                         | PresetSingleRegister    | 4x            | USERDATA Single                       |
| 16 (10Hex)                 | PresetMultipleRegisters | 4x            | USERDATA Multiple                     |
| 05                         | ForceSingelCoil         | 0x            | DIGITAL_OUT mit Datentyp Bit          |
| 15 (0FHex)                 | ForceMultipleCoils      | 0x            | DIGITAL_OUT mit Datentyp 16 bit group |
| 21 (15Hex)                 | WriteGeneralReference   | 6x            | EXTENDEDMEMORY (Nicht bei allen CPUs) |

### Koppelbare Steuerungen

Für folgende Modicon Modbus Steuerungen können Kopplungen realisiert werden:

| Modicon Modbus Steuerung | Unterstütztes Protokoll          |                       |
|--------------------------|----------------------------------|-----------------------|
|                          | Modicon Modbus RTU <sup>2)</sup> | Modicon Modbus TCP/IP |
| TSX Compact              | x                                | x <sup>1)</sup>       |
| TSX Quantum              | x                                | x                     |
| Momentum                 | x                                | x                     |
| Premium                  | -                                | x                     |
| Micro                    | -                                | x                     |
| M340 20x0 (ohne 2010)    | -                                | x                     |

<sup>1)</sup> Nur über Ethernet TCP/IP-Modbus Plus Bridge

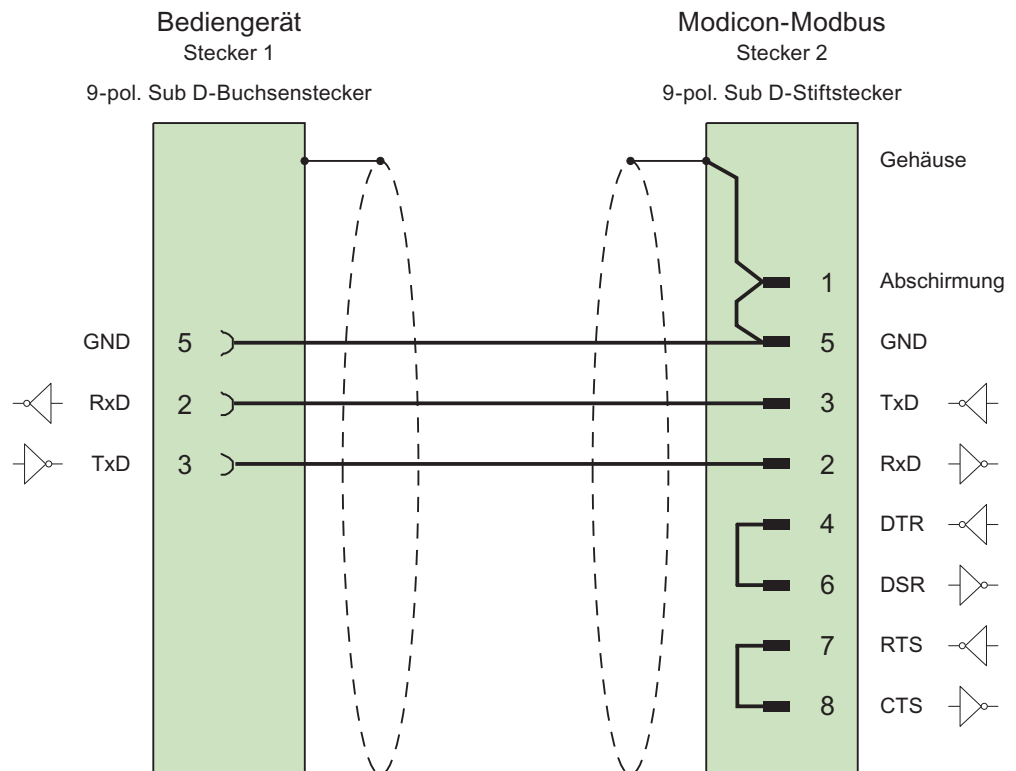
<sup>2)</sup> Steuerungsseitig ist die Kommunikation über RS 232 getestet und freigegeben. Bei den Bediengeräten, die nur eine RS 422/485-Schnittstelle haben, wurde der RS 422/232-Konverter mit der Bestellnummer 6AV6 671-8XE00-0AX0 getestet und freigegeben.



## Verbindungskabel für Modicon Modbus RTU

### Verbindungskabel PP1, RS 232, für Modicon

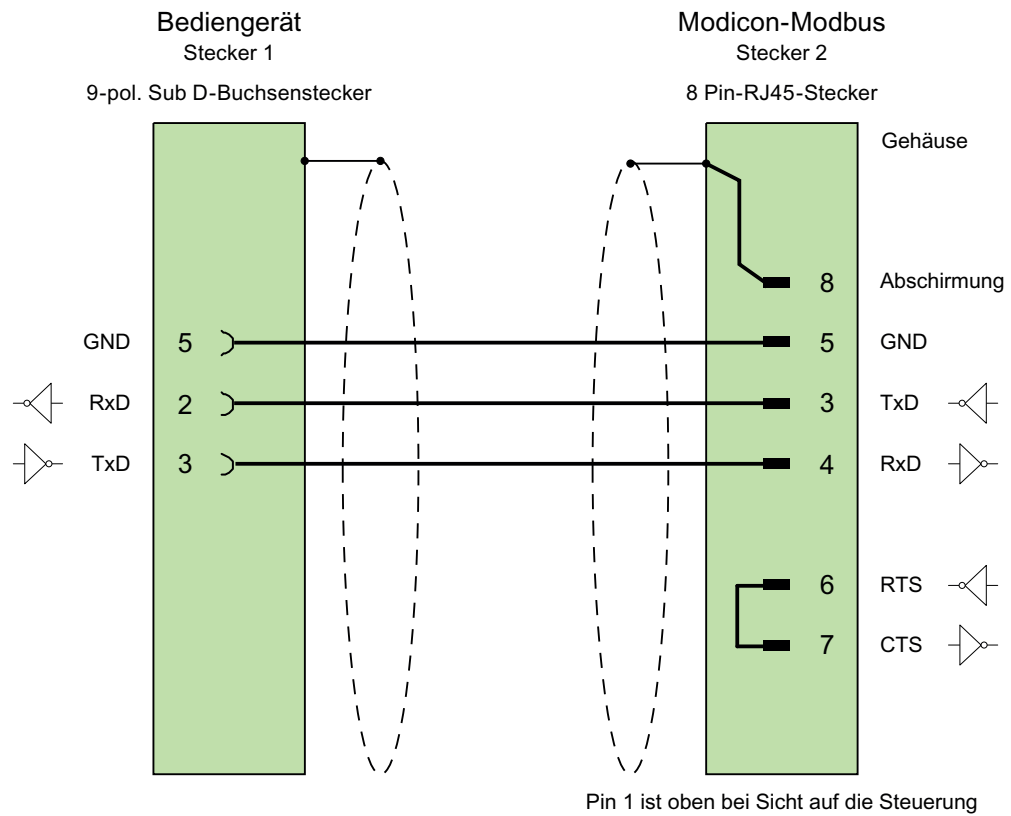
#### Punkt zu Punkt-Kabel 1: SPS > PC ...



Kabel: 3 x 0,14 mm<sup>2</sup>, geschirmt,  
max. Länge 15 m

### Verbindungskabel PP2, RS 232, für Modicon

#### Punkt zu Punkt-Kabel 2: SPS (TSX Compact) > PC...



Kabel: 3 x 0,14 mm<sup>2</sup>, geschirmt,  
max. Länge 15 m

## Leistungsmerkmale der Kommunikation

### Zulässige Datentypen für Modicon Modbus RTU

#### Zulässige Datentypen

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp          | Operandentyp       | Länge            |
|-------------------|--------------------|------------------|
| +/- Double        | 4x, 6x             | 4 Byte           |
| +/- Int           | 3x, 4x, 6x         | 2 Byte           |
| 16 bit group      | 0x, 1x             | 2 Byte           |
| ASCII             | 4x, 6x             | 0 bis 80 Zeichen |
| Bit <sup>1)</sup> | 0x, 1x, 3x, 4x, 6x | 1 Bit            |
| Double            | 4x, 6x             | 4 Byte           |
| Float             | 4x, 6x             | 4 Byte           |
| Int               | 3x, 4x, 6x         | 2 Byte           |

- <sup>1)</sup> Beachten Sie bei schreibenden Zugriffen:  
Bei Datentyp "Bit" mit den Operandentypen "4x" und "6x" wird nach der Änderung des angegebenen Bits das gesamte Wort wieder in die Steuerung zurückgeschrieben. Dabei wird nicht geprüft, ob sich zwischenzeitlich andere Bits im Wort geändert haben. Deshalb darf die Steuerung auf das angegebene Wort nur lesend zugreifen.

Die bei folgenden CPU-Typen übliche Bitzählweise (16 LSB - 1 MSB) wird nur im Editor "HMI-Variablen" bei ausgewähltem Datentyp "Bit" verwendet:

- Concept ProWORX: Compact, Quantum

Folgende Zuordnung der Bitstellen gilt:

|                         | Linkes Byte |   |   |   |   |   |   |   | Rechtes Byte |    |    |    |    |    |    |    |
|-------------------------|-------------|---|---|---|---|---|---|---|--------------|----|----|----|----|----|----|----|
| Zählweise bei Variablen | 1           | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9            | 10 | 11 | 12 | 13 | 14 | 15 | 16 |

#### Darstellung von "Signed"

Der Platzhalter "+/-" steht für die Datentypen "Signed Int" und "Signed Double".

## Unterstützte CPU-Typen für Modicon Modbus RTU

### CPU-Typen

Folgende CPU-Typen werden bei der Projektierung des Kommunikationstreibers Modicon Modbus RTU unterstützt.

- Compact
- Momentum
- Quantum

### Adressbereiche für Modicon Modbus RTU

#### UnityPI7

| Adressbereiche | Datentypen                  |                             |                        |                        |                        |                        |                        |                        |
|----------------|-----------------------------|-----------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
|                | Bool                        | 16 Bit Group                | Int                    | +/- Int                | DInt                   | +/- DInt               | Float                  | ASCII                  |
| %I             | %I0 -<br>%I65535            | %I65535<br>%I0 -<br>%I65520 | --                     | --                     | --                     | --                     | --                     | --                     |
| %M             | %M0 -<br>%M65535            | %M65535<br>%M0 -<br>%M65520 | --                     | --                     | --                     | --                     | --                     | --                     |
| %IW            | %IW0.0 -<br>%IW65535.<br>15 | --                          | %IW0 -<br>%IW65535     | %IW0 -<br>%IW65535     | --                     | --                     | --                     | --                     |
| %MW            | %MW0.0 -<br>%MW6553<br>5.15 | --                          | %MW0 -<br>%MW6553<br>5 | %MW0 -<br>%MW6553<br>5 | %MW0 -<br>%MW6553<br>4 | %MW0 -<br>%MW6553<br>4 | %MW0 -<br>%MW6553<br>4 | %MW0 -<br>%MW6553<br>5 |

#### ConceptProWORX

| Adressbereiche | Datentypen                         |                        |                        |                        |      |          |       |       |
|----------------|------------------------------------|------------------------|------------------------|------------------------|------|----------|-------|-------|
|                | Bool                               | 16 Bit Group           | Int                    | +/- Int                | DInt | +/- DInt | Float | ASCII |
| 0x             | 0x1 -<br>0x65535                   | 0x1 -<br>0x65520       | --                     | --                     | --   | --       | --    | --    |
| 1x             | 1x100001 -<br>1x165535             | 1x100001 -<br>1x165520 | --                     | --                     | --   | --       | --    | --    |
| 3x             | 3x300001.1<br>-<br>3x365535.1<br>6 | --                     | 3x300001 -<br>3x365535 | 3x300001 -<br>3x365535 | --   | --       | --    | --    |

| Adressbereiche | Datentypen                             |              |                           |                           |                           |                           |                           |                           |
|----------------|--|--------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|
|                | Bool                                   | 16 Bit Group | Int                       | +/- Int                   | DInt                      | +/- DInt                  | Float                     | ASCII                     |
| 4x             | 4x400001.1<br>-<br>4x465535.1<br>6     | --           | 4x400001 -<br>4x465535    | 4x400001 -<br>4x465535    | 4x400001 -<br>4x465534    | 4x400001 -<br>4x465534    | 4x400001 -<br>4x465534    | 4x400001 -<br>4x465535    |
| 6x             | 6x60000.1:<br>1 -<br>6x69999.16<br>:10 | --           | 6x60000:1 -<br>6x69999:10 | 6x60000:1 -<br>6x69999:10 | 6x60000:1 -<br>6x69998:10 | 6x60000:1 -<br>6x69998:10 | 6x60000:1 -<br>6x69998:10 | 6x60000:1 -<br>6x69999:10 |

## Komponenten in Betrieb nehmen

### Projekt auf das Bediengerät übertragen

1. Versetzen Sie das Bediengerät in den "Transfermodus".
2. Stellen Sie alle notwendigen Transferparameter ein:
  - Schnittstelle
  - Übertragungsparameter
  - Zielspeicher
3. Starten Sie die Übertragung des Projekts.  
Das Projekt wird automatisch generiert.  
In einem Fenster werden die einzelnen Schritte der Generierung und der Übertragung mitprotokolliert.

### Steuerung und Bediengerät miteinander verbinden

1. Verbinden Sie die Steuerung und das Bediengerät mit einem geeigneten Anschlusskabel.
2. Am Bediengerät erscheint die Meldung "Verbindung zur Steuerung .... ist hergestellt".

## Projektierung optimieren

### Erfassungszyklus und Aktualisierungszeit

Die in der Projektierungssoftware angegebenen Erfassungszyklen für die "Bereichszeiger" und die Erfassungszyklen der Variablen sind wesentliche Faktoren für die tatsächlich erreichbaren Aktualisierungszeiten.

Die Aktualisierungszeit ist Erfassungszyklus plus Übertragungszeit plus Verarbeitungszeit.

Um optimale Aktualisierungszeiten zu erreichen, beachten Sie bei der Projektierung Folgendes:

- Richten Sie die einzelnen Datenbereiche so klein wie möglich und so groß wie nötig ein.
- Zu klein gewählte Erfassungszyklen beeinträchtigen unnötigerweise die Gesamtperformance. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs. Richtwert für den Erfassungszyklus ca. 1 Sekunde.
- Legen Sie die Variablen einer Meldung oder eines Bildes ohne Lücken in einen Datenbereich.
- Damit Änderungen in der Steuerung sicher erkannt werden, müssen diese mindestens während des tatsächlichen Erfassungszyklus anstehen.

## Bilder

Bei Bildern hängt die tatsächlich erreichbare Aktualisierungsrate von Art und Anzahl der darzustellenden Daten ab.

Beachten Sie bei der Projektierung, dass Sie nur für diejenigen Objekte kurze Erfassungszyklen projektieren, die tatsächlich schnell aktualisiert werden müssen. Damit werden die Aktualisierungszeiten kürzer.

## Kurven

Wenn bei bitgetriggerten Kurven das Sammelbit im "Kurvenübertragungsbereich" gesetzt wird, so aktualisiert das Bediengerät jedes Mal alle Kurven, deren Bit in diesem Bereich gesetzt ist. Danach setzt es die Bits wieder zurück.

Erst wenn vom Bediengerät alle Bits zurückgesetzt wurden, darf das Sammelbit im SPS-Programm wieder gesetzt werden.

## Steuerungsaufträge

Wenn viele Steuerungsaufträge in schneller Folge zum Bedienen gesendet werden, so kann dadurch die Kommunikation zwischen Bediengerät und Steuerung überlastet werden.

Wenn das Bediengerät den Wert 0 in das erste Datenwort des Auftragsfachs einträgt, hat das Bediengerät den Steuerungsauftrag entgegengenommen. Jetzt arbeitet das Bediengerät den Auftrag ab, wofür es noch Zeit benötigt. Wenn sofort wieder ein neuer Steuerungsauftrag in das Auftragsfach eingetragen wird, kann es eine Zeit dauern, bis das Bediengerät den nächsten Steuerungsauftrag ausführt. Der nächste Steuerungsauftrag wird erst wieder entgegengenommen, wenn Rechnerleistung zur Verfügung steht.

## Datenaustausch

### Bereichszeiger bei Modicon Modbus

### Bereichszeiger bei Verbindungen über Modicon Modbus Kommunikationstreiber

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu.

Nähere Hinweise zu Bereichszeigern und deren Projektierung finden Sie im Kapitel "Datenaustausch über Bereichszeiger (Seite 3821)".

### Besonderheiten bei Verbindungen über Modicon Modbus Kommunikationstreiber

Sie können folgende Bereichszeiger projektieren

| Bereichszeiger    | Modicon Modbus TCP/IP | Modicon Modbus RTU |
|-------------------|-----------------------|--------------------|
| Bildnummer        | ja                    | ja                 |
| Datum/Uhrzeit     | ja                    | ja                 |
| Datum/Uhrzeit PLC | ja                    | ja                 |
| Koordinierung     | ja                    | ja                 |
| Projektkennung    | ja                    | ja                 |
| Steuerungsauftrag | ja                    | ja                 |
| Datensatz         | ja                    | ja                 |

### Einschränkungen Modicon Modbus TCP/IP

Folgende Einschränkungen gelten für die Projektierung von Bereichszeigern.

| CPU-Typ                                      | Datentypen   | Dateitypen |
|--|--------------|------------|
| Concept, ProWORX: Compact, Quantum, Momentum | +/- Int, Int | 4x, 6x     |
| Unity, PL7: Premium, Micro, Quantum, M340    | +/- Int, Int | %MW        |

### Einschränkungen Modicon Modbus RTU

Folgende Einschränkungen gelten für die Projektierung von Bereichszeigern.

| CPU-Typ                                      | Datentypen   | Dateitypen |
|--|--------------|------------|
| Concept, ProWORX: Compact, Quantum, Momentum | +/- Int, Int | 4x, 6x     |

### Kurven

### Kurven

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt bei den Basic Panels zeitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 3201)

### Zeitgetriggerte Kurven

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein.

Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

### Meldungen

#### Meldungen projektieren

#### Meldungen projektieren bei nicht integrierten Verbindungen

Um Meldungen wie Warnungen, Fehlermeldungen und Quittierung zu projektieren sind mehrere Schritte notwendig.

- Schritt 1: Variablen anlegen
- Schritt 2: Meldungen projektieren
- Schritt 3: Quittierung projektieren

Nähere Hinweise hierzu finden Sie im Kapitel:

Arbeiten mit Meldungen (Seite 3213)

#### Besonderheiten bei der Projektierung von Meldungen

Wenn Sie Verbindungen von Bediengeräten mit Steuerungen anderer Hersteller projektieren, beachten Sie folgende Besonderheiten bei der Projektierung:

- Datentypen der Variablen
- Adressierung der Variablen
- Zählweise der Bitstellen

### Datentypen

Für Verbindungen mit Modicon Modbus Kommunikationstreiber, werden folgende Datentypen unterstützt:

| Steuerung           | Zulässige Datentypen |   |
|---------------------|----------------------|---|
|                     | Bitmeldungen         | Analogmeldungen                                     |
| Alle Modicon-Serien | Int, +/-Int          | 16 Bit Group, Int, +/-Int, Double, +/-Double, Float |

Arrays und Arrayvariablen können für Bitmeldungen verwendet werden.



## Zählweise der Bitstellen

Für Verbindungen mit Modicon Modbus Kommunikationstreiber, gilt folgende Zählweise:

| Zählweise der Bitstellen   | Linkes Byte |    |    |    |    |    |   |   | Rechtes Byte |   |   |   |   |   |   |   |
|----------------------------|-------------|----|----|----|----|----|---|---|--------------|---|---|---|---|---|---|---|
| In WinCC projektieren Sie: | 15          | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7            | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

## Quittierung von Meldungen

### Vorgehensweise

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

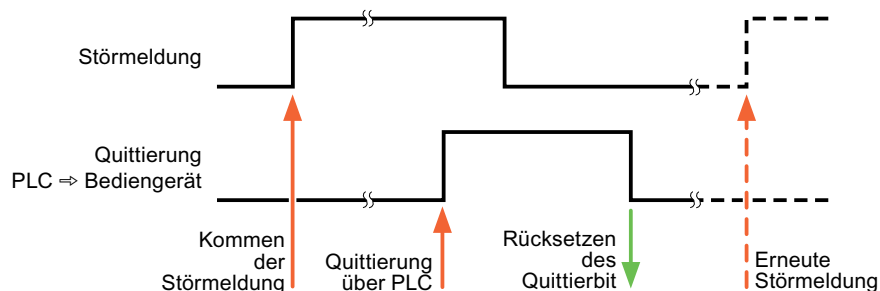
### Quittieren von der Steuerung

Unter "PLC-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulssdiagramm.



### Quittierung am Bediengerät

Unter "HMI-Quittiervariable" projizieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittiervariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittiervariable in die Steuerung eine gewisse Zeitspanne.

Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittiervariablen gesetzt. Die gesamte Quittiervariable wird dann vom Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

---

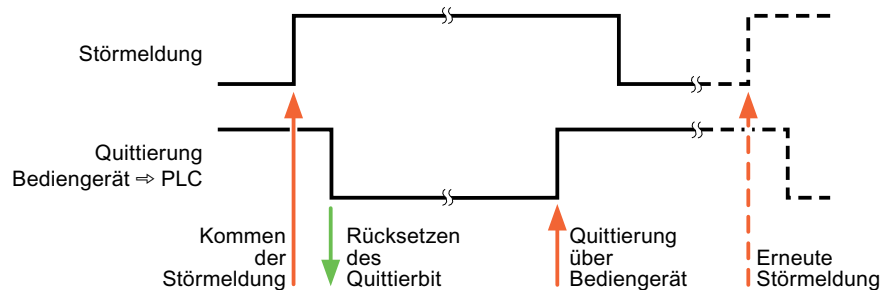
#### Hinweis

Alle seit dem letzten Runtime-Start quittierten Meldebits bleiben in der Quittiervariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittiervariablen der gesamte Bereich vom Bediengerät überschrieben wird.

---

Das nachfolgende Bild zeigt das Impulsdiagramm.



## Omron

### Omron Kommunikationstreiber

#### Einleitung

In diesem Abschnitt ist die Kommunikation zwischen Bediengerät und Steuerungen beschrieben, die Omron Kommunikationstreiber verwenden.

Folgende Kommunikationstreiber werden unterstützt:

- Omron Host Link

## Datenaustausch

Datenaustausch findet über Variablen oder Bereichszeiger statt.

- Variablen  
Die Steuerung und das Bediengerät tauschen ihre Daten über Prozesswerte aus. Legen Sie in der Projektierung Variablen an, die auf Adressen in der Steuerung zeigen. Das Bediengerät liest aus der angegebenen Adresse den Wert und zeigt ihn an. Genauso kann der Bediener eine Eingabe am Bediengerät vornehmen, die dann in die Adresse in der Steuerung geschrieben wird.
- Bereichszeiger  
Bereichszeiger dienen dem Austausch spezieller Daten und werden nur bei Verwendung dieser Daten eingerichtet.

## Omron Hostlink

### Verbindung über Omron Host Link projektieren

#### Einleitung

Eine Verbindung zu einer der Steuerung mit Omron Host Link Kommunikationstreiber, projektieren Sie im Editor "Verbindungen" des Bediengeräts.

---

#### Hinweis

##### Verbindungsaufbau bei Omron Host Link

Wenn Sie eine Verbindung über Omron projektieren, wird beim Starten der Runtime nicht automatisch eine Verbindung aufgebaut.

Im Startbild der Runtime muss eine Variable projektieren sein, die sich im gültigen Speicherbereich der Steuerung befindet.

Andernfalls wird die Verbindung erst aufgebaut, wenn ein entsprechendes Bild angewählt wird.

Beim Starten der Runtime wird diese Variable abgerufen und damit eine Verbindung aufgebaut.

---

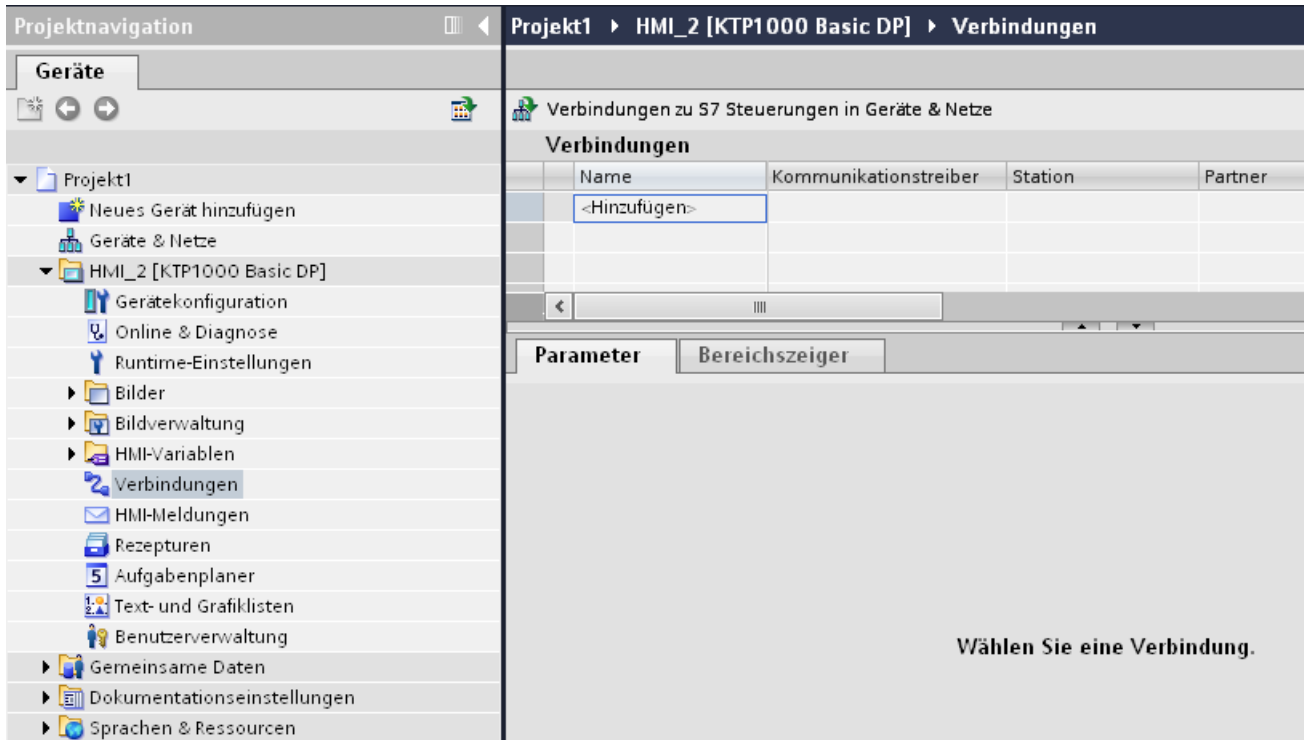
Abhängig vom Bediengerät sind die Schnittstellen unterschiedlich benannt.

#### Voraussetzungen

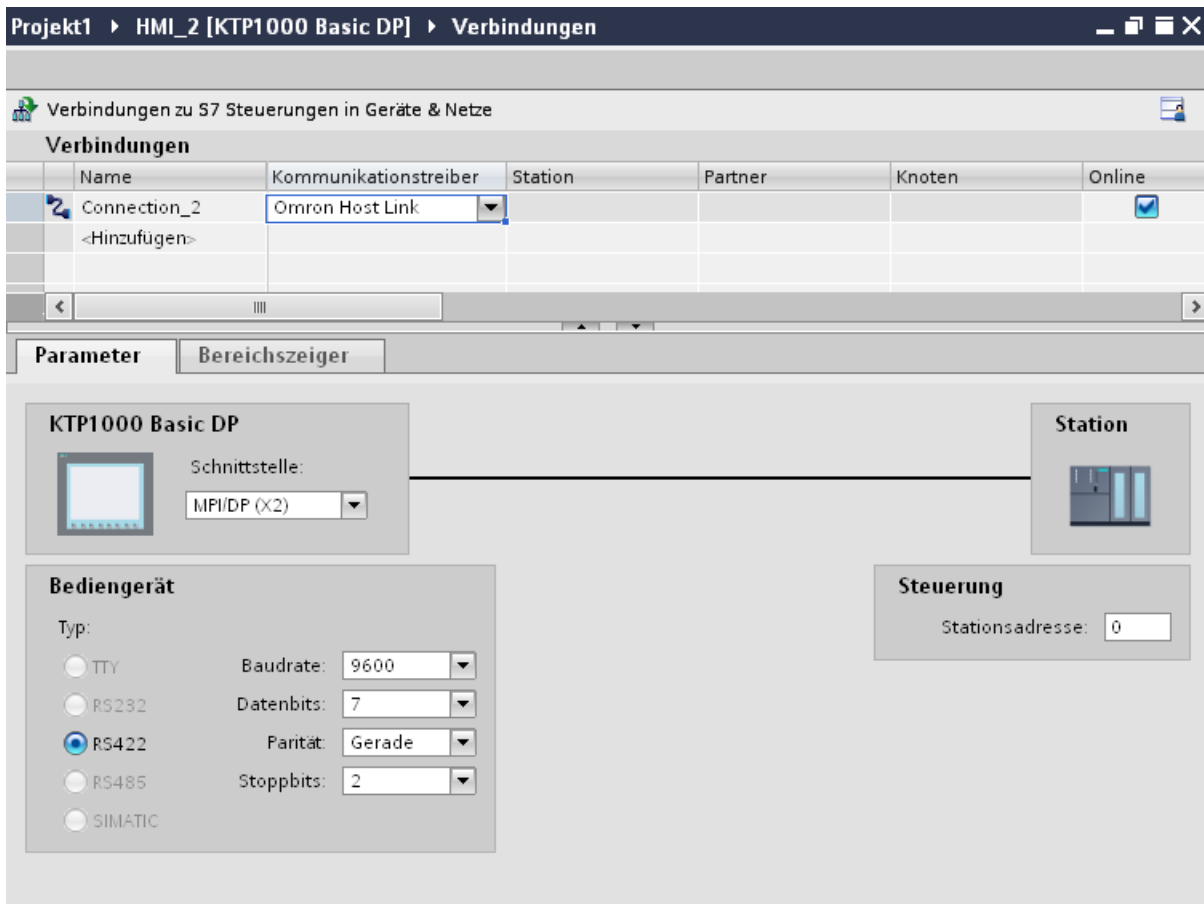
- Ein Projekt ist geöffnet.
- Ein Bediengerät ist angelegt.

### Vorgehensweise

1. Doppelklicken Sie in der Projektnavigation unter "Geräte" auf das Bediengerät.
2. Doppelklicken Sie auf den Eintrag "Verbindungen".
3. Doppelklicken Sie im Editor "Verbindungen" auf "<Hinzufügen>".



4. Wählen Sie in der Spalte "Kommunikationstreiber" den Treiber "Omron Host Link" aus.



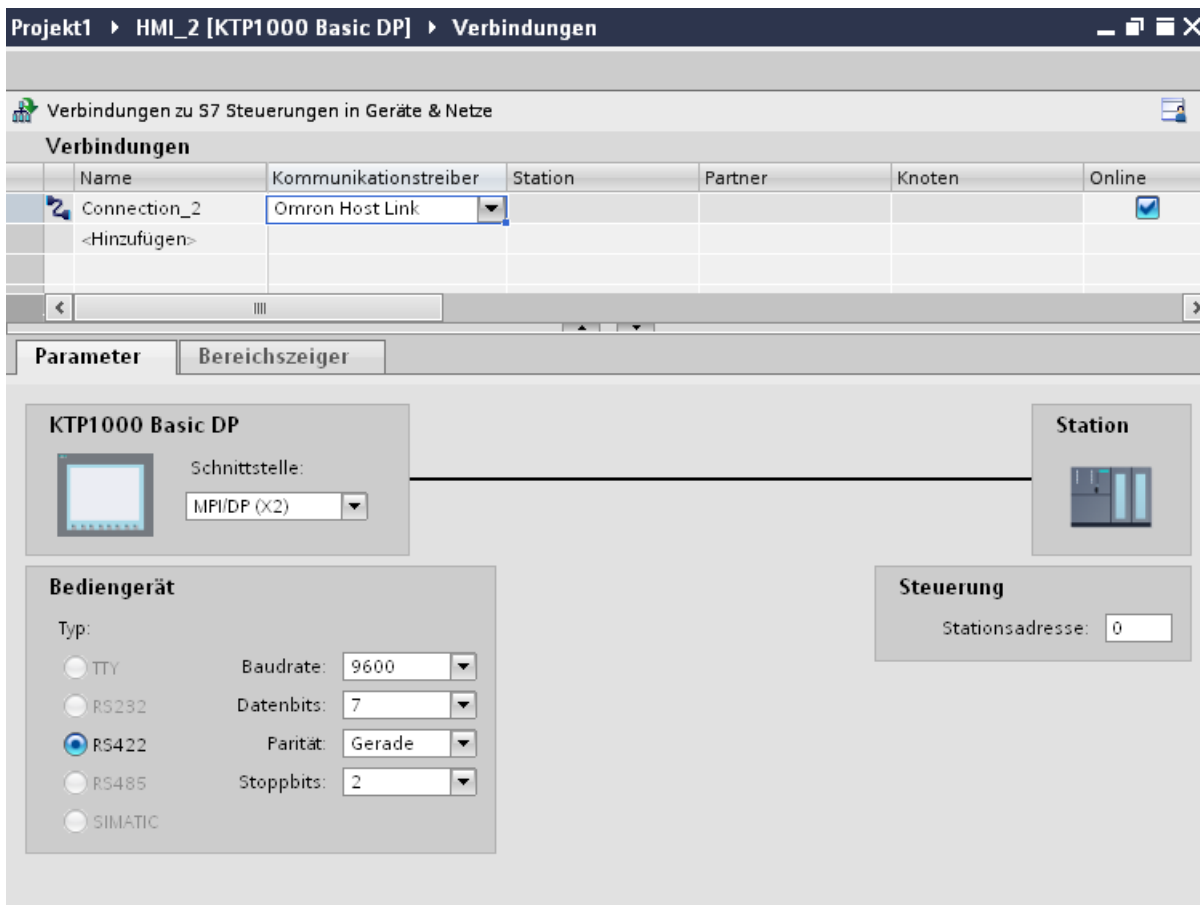
5. Wählen Sie im Inspektorfenster unter "Parameter" alle notwendigen Verbindungsparameter für die Schnittstelle aus.

## Parameter für die Verbindung (Omron Hostlink)

### Einzustellende Parameter

Um die Parameter für die Verbindung, wie z. B. Adressen und Profile einzustellen, klicken Sie im Verbindungseditor auf die angelegte Verbindung.

Im Inspektorfenster unter "Parameter" werden die Kommunikationspartner schematisch dargestellt. Zur Parametereinstellung gibt es abhängig von der verwendeten Schnittstelle die Bereiche "Bediengerät" und "Steuerung".



### Parameter für das Bediengerät

Sie können im Inspektorfenster "Parameter" für das Bediengerät eine Schnittstelle auswählen. Abhängig vom Bediengerät stehen auch mehrere Schnittstellen zur Verfügung.

- Typ  
Legt den benutzten physikalischen Anschluss fest.
- Baudrate  
Unter "Baudrate" stellen Sie die Übertragungsgeschwindigkeit Bediengerät zu OMRON ein. Die Kommunikation ist mit einer Baudrate von 19200, 9600 Baud möglich.
- Datenbits  
Unter "Datenbits" können Sie zwischen "7 Bit" oder "8 Bit" wählen.
- Parität  
Unter "Parität" können Sie zwischen "keine", "gerade" und "ungerade" wählen.
- Stoppbits  
Unter "Stoppbits" können Sie zwischen 1 und 2 Bit wählen.

## Parameter für die Steuerung

- Stationsadresse  
Unter "Stationsadresse" geben Sie die Stationsnummer der angeschlossenen Steuerung ein.

## Bediengerät mit Steuerung verbinden

### Verbindungen über Omron Host Link

## Anschluss

Die Kopplung des Bediengeräts an eine OMRON-Steuerung beschränkt sich hauptsächlich auf den physikalischen Anschluss des Bediengeräts. Spezielle Bausteine für die Kopplung sind in der Steuerung nicht erforderlich.

## Verbindungskabel

Für den Anschluss des Bediengeräts an eine Omron-Steuerung stehen die folgenden Anschlusskabel zur Verfügung:

| Schnittstelle am Bediengerät | Omron-Steuerung |   |                |                        |
|------------------------------|-----------------|---|----------------|------------------------|
|                              | RS232, 9-polig  | RS232 Peripherieport                              | RS422, 9-polig | RS422, Klemmen/Stecker |
| RS232, 9-polig               | PP1             | Programmierkabel<br>(Standardkabel der Fa. Omron) | —              | —                      |
| RS232 über Konverter         | —               | —   | —              | Mehrpunktkabel 1       |
| RS422, 9-polig               | —               | —   | PP2            | Mehrpunktkabel 2       |

Welche Schnittstelle am Bediengerät Sie verwenden, entnehmen Sie dem entsprechenden Gerätehandbuch.

## Kommunikationsarten

### Freigegebene Kommunikationsarten

Die Kopplung von einem Bediengerät zu einer OMRON-CPU dem Omron Host Link-Protokoll über RS232 und über RS 422 ist von der Fa. Siemens AG systemgetestet und freigegeben.

Das betrifft folgende CPU-Typen:

- CP1x (CP1L, CP1H, CP1E)
- CJ1x(CJ1M, CJ1H, CJ1G)
- CJ2H

- CS1x(CS1G, CS1H, CS1D)
  - CPM2C
- 

**Hinweis**

Für Basic Panels, TP 177A und OP 77A sind nur folgende CPU-Typen getestet und freigegeben:

- CP1x(CP1L, CP1H, CP1E)
  - CJ1x(CJ1M, CJ1H, CJ1G)
- 

### Mehrpunkt-Ankopplung

Eine Mehrpunkt-Ankopplung an bis zu 4 freigegebene OMRON-Steuerungen in einen RS422-Vierdraht-Multidrop-Verbund kann mit Kommunikations-Modulen auf den Steuerungen realisiert werden und ist von der Fa. Siemens AG systemgetestet und freigegeben.

---

**Hinweis**

Das Bediengerät kann nur als Master betrieben werden. Es ist genau ein Master im RS422-Vierdraht-Multidrop-Verbund möglich.

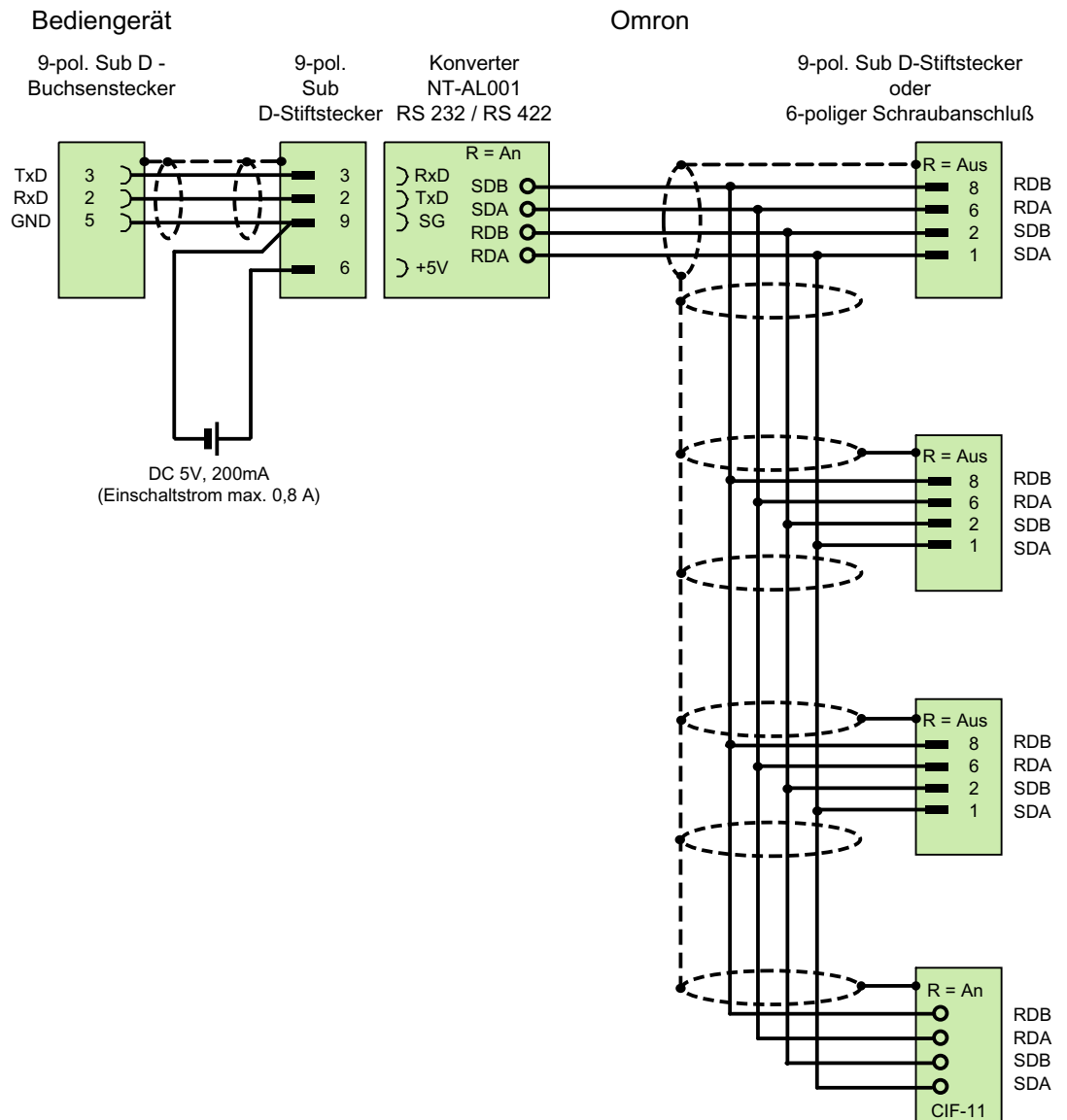
---



## Verbindungskabel

### Verbindungskabel MP1, RS 232 über Konverter, für Omron

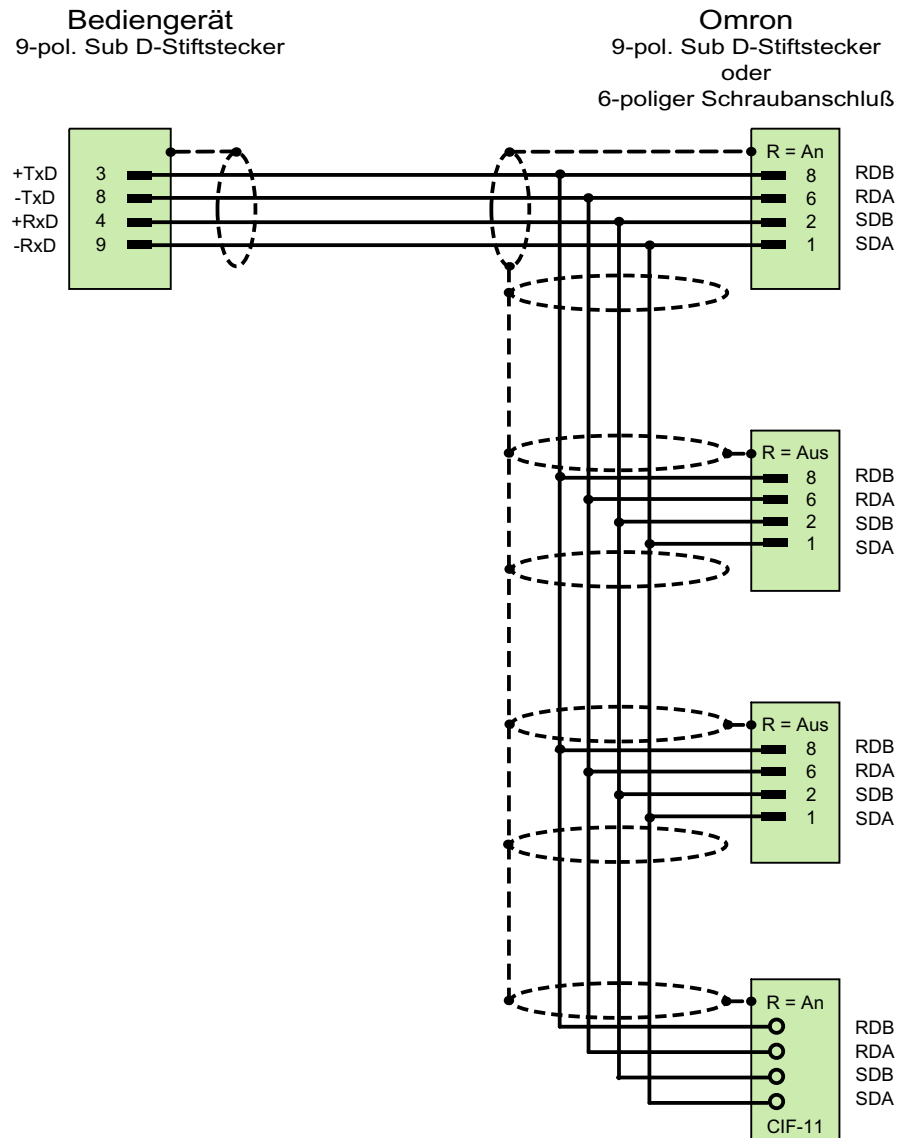
#### Mehrpunkt-Kabel 1: MP/TP/PC > SPS



<sup>1)</sup> Einschaltstrom max. 0,8 A  
 geschirmt, max. Länge 500 m

Verbindungskabel MP2, RS 422, für Omron

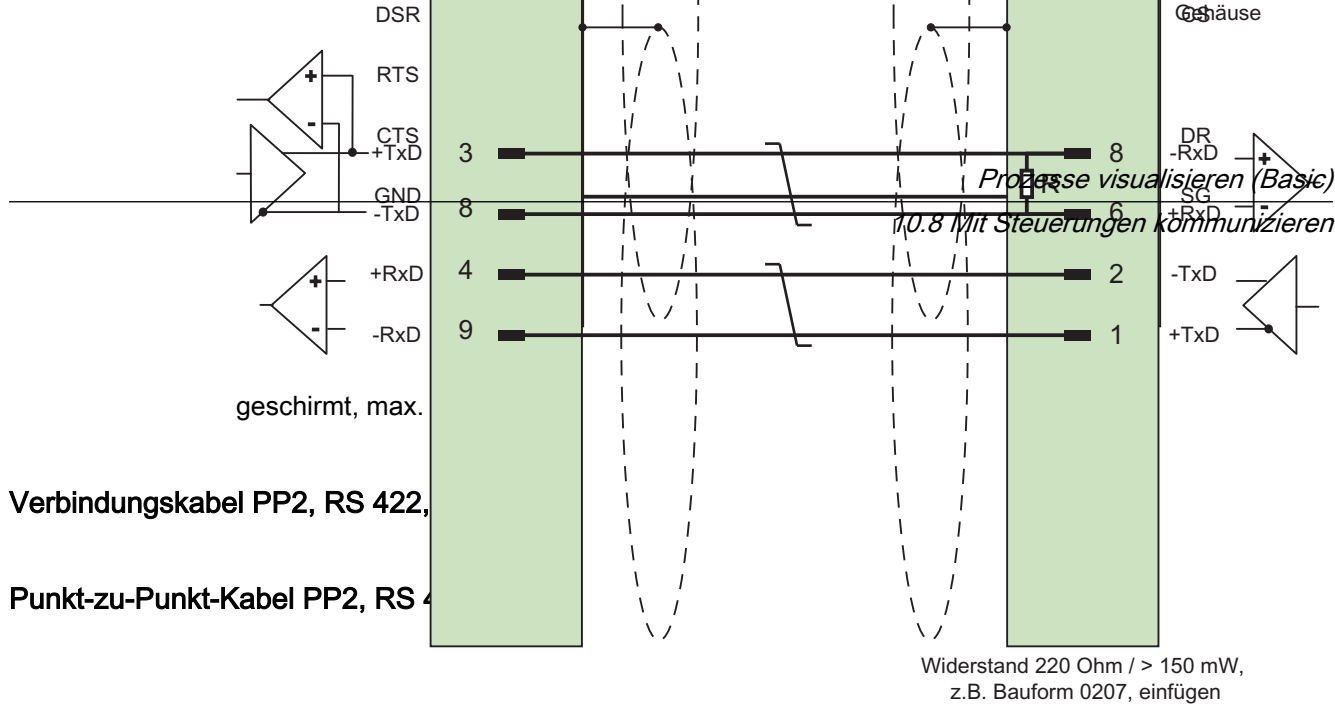
Mehrpunkt-Kabel 2: RS422, MP/TP/PC > SPS\_



geschirmt, max. Länge 500 m

Verbindungskabel PP1, RS 232, für Omron

Punkt-zu-Punkt-Kabel PP1, PC/TP/OP - SPS



geschirmt, max. Länge 500 m

## Leistungsmerkmale der Kommunikation

## Zulässige Datentypen für Omron Host Link

### Zulässige Datentypen

In der Tabelle sind die Datentypen aufgelistet, die bei Projektierung von Variablen und Bereichszeigern verwendet werden können.

| Datentyp | Operandentyp                                 | Länge            |
|----------|--|------------------|
| Bool     | I/O, HR, AR, LR, DM, T/<br>C Bit, CPU Status | 1 Bit            |
| Byte     | CPU Type                                     | 1 Byte           |
| DInt     | HR, AR, LR, DM                               | 4 Byte           |
| Int      | I/O, HR, AR, LR, DM, T/<br>C Val             | 2 Byte           |
| Real     | HR, DM                                       | 4 Byte           |
| String   | HR, AR, LR, DM                               | 0 bis 80 Zeichen |
| UDInt    | HR, AR, LR, DM                               | 4 Byte           |
| UInt     | I/O, HR, AR, LR, DM, T/<br>C Val             | 2 Byte           |

### Hinweis

Nur in der Betriebsart "STOP" oder "MONITOR" können alle Datenbereiche zuverlässig in der OMRON-Steuerung gelesen oder geschrieben werden.

Mit "I/O" wird je nach Steuerungsserie entweder der IR/SR- oder der CIO-Bereich bezeichnet. Die Operandentypen "LR", "HR" und "AR" sind nicht bei allen Steuerungsserien verfügbar.

**Hinweis**

Beachten Sie bei schreibenden Zugriffen:

Bei Datentyp "Bool" mit den Operandentypen "I/O", "HR", "AR", "LR" und "DM" wird nach der Änderung des angegebenen Bits das gesamte Wort wieder in die Steuerung zurückgeschrieben. Dabei wird nicht geprüft, ob sich zwischenzeitlich andere Bits im Wort geändert haben. Deshalb darf die Steuerung auf das angegebene Wort nur lesend zugreifen.

| Operandentyp alte SPS | Operandentyp CS und CJ SPS |
|-----------------------|----------------------------|
| CPU Status            | CPU Status                 |
| I/O                   | CIO                        |
| HR                    | H<br>Range 0-511           |
| AR                    | A                          |
| LR                    | n/a 1)                     |
| DM                    | D                          |
| T/C                   | T/C                        |
| CPU Typ               | CPU Typ                    |

- 1) Wenn Sie den Bereich LR in den folgenden SPS lesen oder schreiben, dann bekommen Sie keine Fehlermeldung:
- CS
  - CJ
  - CP

**Unterstützte CPU-Typen für Omron Host Link**

**CPU-Typen**

Folgende CPU-Typen werden bei der Projektierung des Kommunikationstreibers Omron Host Link unterstützt.

- CP1
  - CP1L
  - CP1H
  - CP1E
- CJ1
  - CJ1M
  - CJ1H
  - CJ1G

- CJ2
  - CJ2H
- CS1
  - CS1G
  - CS1H
  - CS1D
- CPM
  - CPM2C

### Adressierung bei Omron Host Link

#### Adressierung von Steuerungen bei Omron Host Link

Bei Steuerungen der Serien CS, CP und CJ werden die Timer 0-4095 mit T/C 0-2047 adressiert.

Die Counter 0-4095 müssen mit einem Offset von 2048 adressiert werden (T/C 2048-4095 entsprechen den Countern 0-2047). Counter und Timer mit Adressen > 2047 können über Host Link nicht adressiert werden.

Counter und Timer mit Adressen > 2047 können über Host Link nicht adressiert werden.

Beispiel:

Wenn Sie den Counter C20 adressieren wollen, müssen Sie  $T/C\ 20+2048 = T/C\ 2068$  adressieren.

#### Adressbereiche für Omron Hostlink

##### Omron

| Adressbereiche | Datentypen                |      |                     |                     |                   |                   |                   |                   |
|----------------|---------------------------|------|---------------------|---------------------|-------------------|-------------------|-------------------|-------------------|
|                | Bool                      | Byte | UInt                | Int                 | UDInt             | DInt              | Real              | String            |
| I/O            | I/O 0.0 -<br>I/O 9999.15  | --   | I/O 0 -<br>I/O 9999 | I/O 0 -<br>I/O 9999 | --                | --                | --                | --                |
| HR             | HR 0.0 -<br>HR 9999.15    |      | HR 0 -<br>HR 9999   | HR 0 -<br>HR 9999   | HR 0 -<br>HR 9998 | HR 0 - HR<br>9998 | HR 0 -<br>HR 9999 | HR 0 -<br>HR 9999 |
| AR             | AR 0.0 -<br>AR 9999.15    |      | AR 0 -<br>AR 9999   | AR 0 -<br>AR 9999   | AR 0 -<br>AR 9998 | AR 0 -<br>AR 9998 |                   | AR 0 -<br>AR 9999 |
| LR             | LR 0.0 -<br>LR 9999.15    |      | LR 0 -<br>LR 9999   | LR 0 -<br>LR 9999   | LR 0 -<br>LR 9998 | LR 0 -<br>LR 9998 |                   | LR 0 -<br>LR 9999 |
| DM             | DM 0.0 -<br>DM<br>9999.15 |      | DM 0 -<br>DM 9999   | DM 0 -<br>DM 9999   | DM 0 -<br>DM 9998 | DM 0 -<br>DM 9998 | DM 0 -<br>DM 9999 | DM 0 -<br>DM 9999 |

| Adressbereich | Datentypen                     |      |                                |                                |       |      |      |        |
|---------------|--------------------------------|------|--------------------------------|--------------------------------|-------|------|------|--------|
|               | Bool                           | Byte | UInt                           | Int                            | UDInt | DInt | Real | String |
| T/C Bit       | T/C Bit 0 -<br>T/C Bit<br>4095 |      |                                |                                |       |      |      |        |
| T/C Val       |                                |      | T/C Val 0 -<br>T/C Val<br>4095 | T/C Val 0 -<br>T/C Val<br>4095 |       |      |      |        |
| CPU Status    | RUN,<br>MONITOR                |      |                                |                                |       |      |      |        |
| CPU Type      | CPU Type                       |      |                                |                                |       |      |      |        |

### Komponenten in Betrieb nehmen

#### Projekt auf das Bediengerät übertragen

1. Versetzen Sie das Bediengerät in den "Transfermodus".
2. Stellen Sie alle notwendigen Transferparameter ein:
  - Schnittstelle
  - Übertragungsparameter
  - Zielspeicher
3. Starten Sie die Übertragung des Projekts.  
Das Projekt wird automatisch generiert.  
In einem Fenster werden die einzelnen Schritte der Generierung und der Übertragung mitprotokolliert.

#### Steuerung und Bediengerät miteinander verbinden

1. Verbinden Sie die Steuerung und das Bediengerät mit einem geeigneten Anschlusskabel.
2. Am Bediengerät erscheint die Meldung "Verbindung zur Steuerung .... ist hergestellt".

#### Projektierung optimieren

##### Erfassungszyklus und Aktualisierungszeit

Die in der Projektierungssoftware angegebenen Erfassungszyklen für die "Bereichszeiger" und die Erfassungszyklen der Variablen sind wesentliche Faktoren für die tatsächlich erreichbaren Aktualisierungszeiten.

Die Aktualisierungszeit ist Erfassungszyklus plus Übertragungszeit plus Verarbeitungszeit.

Um optimale Aktualisierungszeiten zu erreichen, beachten Sie bei der Projektierung Folgendes:

- Richten Sie die einzelnen Datenbereiche so klein wie möglich und so groß wie nötig ein.
- Zu klein gewählte Erfassungszyklen beeinträchtigen unnötigerweise die Gesamtperformance. Stellen Sie den Erfassungszyklus entsprechend der Änderungsgeschwindigkeit der Prozesswerte ein. Der Temperaturverlauf eines Ofens ist z. B. wesentlich träger als der Drehzahlverlauf eines elektrischen Antriebs. Richtwert für den Erfassungszyklus ca. 1 Sekunde.
- Legen Sie die Variablen einer Meldung oder eines Bildes ohne Lücken in einen Datenbereich.
- Damit Änderungen in der Steuerung sicher erkannt werden, müssen diese mindestens während des tatsächlichen Erfassungszyklus anstehen.

## Bilder

Bei Bildern hängt die tatsächlich erreichbare Aktualisierungsrate von Art und Anzahl der darzustellenden Daten ab.

Beachten Sie bei der Projektierung, dass Sie nur für diejenigen Objekte kurze Erfassungszyklen projektieren, die tatsächlich schnell aktualisiert werden müssen. Damit werden die Aktualisierungszeiten kürzer.

## Kurven

Wenn bei bitgetriggerten Kurven das Sammelbit im "Kurvenübertragungsbereich" gesetzt wird, so aktualisiert das Bediengerät jedes Mal alle Kurven, deren Bit in diesem Bereich gesetzt ist. Danach setzt es die Bits wieder zurück.

Erst wenn vom Bediengerät alle Bits zurückgesetzt wurden, darf das Sammelbit im SPS-Programm wieder gesetzt werden.

## Steuerungsaufträge

Wenn viele Steuerungsaufträge in schneller Folge zum Bedienen gesendet werden, so kann dadurch die Kommunikation zwischen Bediengerät und Steuerung überlastet werden.

Wenn das Bediengerät den Wert 0 in das erste Datenwort des Auftragsfachs einträgt, hat das Bediengerät den Steuerungsauftrag entgegengenommen. Jetzt arbeitet das Bediengerät den Auftrag ab, wofür es noch Zeit benötigt. Wenn sofort wieder ein neuer Steuerungsauftrag in das Auftragsfach eingetragen wird, kann es eine Zeit dauern, bis das Bediengerät den nächsten Steuerungsauftrag ausführt. Der nächste Steuerungsauftrag wird erst wieder entgegengenommen, wenn Rechnerleistung zur Verfügung steht.

## Datenaustausch

### Bereichszeiger bei Omron

### Bereichszeiger bei Verbindungen über Omron Kommunikationstreiber

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu.

Nähere Hinweise zu Bereichszeigern und deren Projektierung finden Sie im Kapitel:  
"Datenaustausch über Bereichszeiger".

### **Besonderheiten bei Verbindungen über Omron Host Link**

Bereichszeiger können nur in den "Dateitypen" "DM", "I/O", "HR", "AR" und "LR" angelegt werden.

### **Siehe auch**

Datenaustausch über Bereichszeiger (Seite 3821)

### **Kurven**

### **Kurven**

Eine Kurve ist die grafische Darstellung eines oder mehrerer Werte aus der Steuerung. Das Auslesen des Werts erfolgt bei den Basic Panels zeitgetriggert.

Nähere Hinweise hierzu finden Sie unter:

Kurvenanzeige für Werte aus der Steuerung projektieren (Seite 3201)

### **Zeitgetriggerte Kurven**

In einem bei der Projektierung festgelegten Zeittakt liest das Bediengerät die Kurvenwerte zyklisch ein.

Zeitgetriggerte Kurven eignen sich für kontinuierliche Verläufe, wie z. B. die Betriebstemperatur eines Motors.

### **Meldungen**

### **Meldungen projektieren**

### **Meldungen projektieren bei nicht integrierten Verbindungen**

Um Meldungen wie Warnungen, Fehlermeldungen und Quittierung zu projektieren sind mehrere Schritte notwendig.

- Schritt 1: Variablen anlegen
- Schritt 2: Meldungen projektieren
- Schritt 3: Quittierung projektieren

Nähere Hinweise hierzu finden Sie im Kapitel

Arbeiten mit Meldungen (Seite 3213)



## Besonderheiten bei der Projektierung von Meldungen

Wenn Sie Verbindungen von Bediengeräten mit Steuerungen anderer Hersteller projektieren, beachten Sie folgende Besonderheiten bei der Projektierung:

- Datentypen der Variablen
- Adressierung der Variablen
- Zählweise der Bitstellen

## Datentypen

Für Verbindungen mit Omron Kommunikationstreiber, werden folgende Datentypen unterstützt:

| Steuerung               | Zulässige Datentypen |                        |
|-------------------------|----------------------|------------------------|
|                         | Bitmeldungen         | Analogmeldungen        |
| CP1, CJ1, CJ2, CS1, CPM | UInt, int            | UInt, Int, UDInt, DInt |

## Zählweise der Bitstellen

Für Verbindungen mit Omron Kommunikationstreiber, gilt folgende Zählweise:

| Zählweise der Bitstellen   | Linkes Byte |  |  |  |  |  |  |   | Rechtes Byte |  |  |  |  |  |  |  |   |
|----------------------------|-------------|--|--|--|--|--|--|---|--------------|--|--|--|--|--|--|--|---|
| In Omron-Steuerungen       | 15          |  |  |  |  |  |  | 8 | 7            |  |  |  |  |  |  |  | 0 |
| In WinCC projektieren Sie: | 15          |  |  |  |  |  |  | 8 | 7            |  |  |  |  |  |  |  | 0 |

Als Trigger-Variable für Bitmeldungen sind nur Variablen für die "Datentypen" "DM", "I/O", "HR", "AR" und "LR" zulässig.

## Bitmeldungen projektieren

Verwenden Sie für Bitmeldungen Arrays und hängen Sie die einzelnen Meldungen jeweils an ein Bit der Array-Variablen selbst und nicht an die einzelnen Unterelemente.

Für Bitmeldungen und Arrays sind nur Variablen der Bereiche "DM", "I/O", "HR", "AR" und "LR" und der Datentypen "Int" und "UInt" zulässig.

## Quittierung von Meldungen

### Vorgehensweise

Richten Sie in der Steuerung entsprechende Variablen ein, um eine Störmeldung zu quittieren. Diese Variablen müssen Sie im Editor "Bitmeldungen" einer Meldung zuordnen. Die Zuordnung erfolgt in "Eigenschaften > Quittierung".

Bei der Quittierung wird Folgendes unterschieden:

- Quittierung von der Steuerung
- Quittierung am Bediengerät

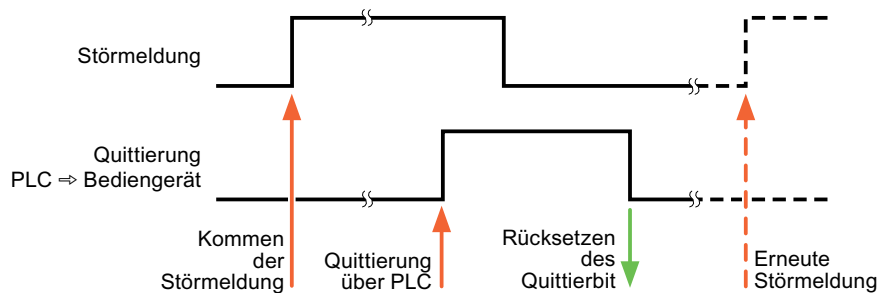
### Quittieren von der Steuerung

Unter "PLC-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, anhand derer das Bediengerät eine Quittierung durch die Steuerung erkennen kann.

Ein in der Variablen gesetztes Bit bewirkt die Quittierung des zugeordneten Störmeldebites am Bediengerät. Damit erfüllt ein in der Variablen gesetztes Bit die gleiche Funktion, wie das Quittieren vom Bediengerät, z. B. durch Drücken der Taste "ACK".

Das Quittierbit muss in der gleichen Variablen liegen wie das Bit für die Störmeldung.

Setzen Sie das Quittierbit wieder zurück, bevor Sie das Bit im Störmeldebereich erneut setzen. Das nachfolgende Bild zeigt das Impulsdiagramm.



### Quittierung am Bediengerät

Unter "HMI-Quittvariable" projektieren Sie die Variable oder die Arrayvariable und die Bitnummer, die nach der Quittierung vom Bediengerät in die Steuerung geschrieben wird. Achten Sie bei der Verwendung einer Arrayvariablen darauf, dass diese nicht länger als 6 Worte ist.

Damit beim Setzen des zugeordneten Quittierbits einer quittierpflichtigen Bitmeldung immer ein Signalwechsel erzeugt wird, setzt das Bediengerät, sobald eine quittierpflichtige Meldung erkannt wurde, das der Meldung zugeordnete Quittierbit zurück und schreibt die Quittvariable in die Steuerung. Da das Bediengerät die Vorgänge verarbeiten muss, liegt zwischen Erkennen der Meldung und Schreiben der Quittvariable in die Steuerung eine gewisse Zeitspanne.

Wenn eine quittierpflichtige Bitmeldung am Bediengerät quittiert wird, wird das entsprechende Bit in der zugeordneten Quittvariablen gesetzt. Die gesamte Quittvariable wird dann vom

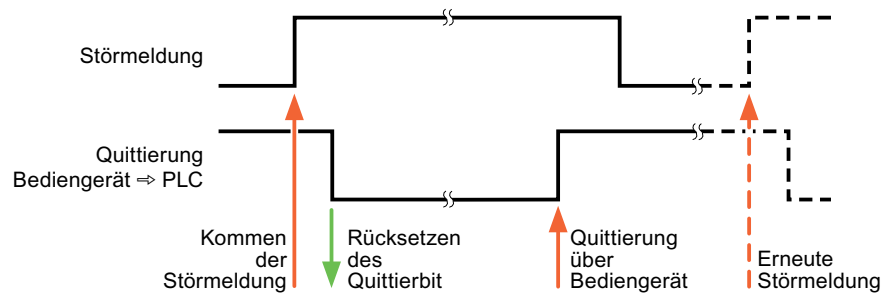
Bediengerät in die Steuerung geschrieben. Damit kann die Steuerung erkennen, dass eine bestimmte Störmeldung am Bediengerät quittiert wurde.

#### Hinweis

Alle seit dem letzten Runtime-Start quittierten Meldebites bleiben in der Quittvariablen erhalten, bis ein erneutes Kommen der entsprechenden Bitmeldungen erkannt wird.

Dieser Bereich sollte von der Steuerung nur gelesen werden, da mit dem nächsten Schreiben der Quittvariablen der gesamte Bereich vom Bediengerät überschrieben wird.

Das nachfolgende Bild zeigt das Impulssdiagramm.



#### 10.8.11.4 Datenaustausch über Bereichszeiger

##### Allgemeines zu Bereichszeigern

##### Einleitung

Über einen Bereichszeiger greifen Sie auf einen Datenbereich in der Steuerung zu. In diese Datenbereiche wird während der Kommunikation wechselseitig von der Steuerung und dem Bediengerät geschrieben und gelesen.

Durch die Auswertung der abgelegten Daten lösen Steuerung und Bediengerät gegenseitig fest definierte Aktionen aus.

### Projektierung von Bereichszeigern

Bevor Sie den Bereichszeiger verwenden, aktivieren Sie die Bereichszeiger unter "Verbindungen ► Bereichszeiger". Danach parametrieren Sie die Bereichszeiger.

| Parameter                               |                   | Bereichszeiger |                      |         |       |                       |                  |           |
|---|-------------------|----------------|----------------------|---------|-------|-----------------------|------------------|-----------|
| Aktiv                                   | Anzeigename       | PLC-Variable   | Zugriffsart          | Adresse | Länge | Erfassungsart         | Erfassungszyklus | Kommentar |
| <input type="checkbox"/>                | Koordinierung     | <Undefiniert>  | <Symbolischer Zug... |         | 1     | Zyklisch fortlaufe... | <Undefiniert>    |           |
| <input type="checkbox"/>                | Datum/Uhrzeit     | <Undefiniert>  | <Symbolischer Zug... |         | 6     | Zyklisch fortlaufe... | <Undefiniert>    |           |
| <input type="checkbox"/>                | Steuerungsauftrag | <Undefiniert>  | <Symbolischer Zug... |         | 4     | Zyklisch fortlaufe... | <Undefiniert>    |           |
| <input type="checkbox"/>                | Datensatz         | <Undefiniert>  | <Symbolischer Zug... |         | 5     | Zyklisch fortlaufe... | <Undefiniert>    |           |
| Globale Bereichszeiger des Bediengeräts |                   |                |                      |         |       |                       |                  |           |
| Verbindung                              | Anzeigename       | PLC-Variable   | Zugriffsart          | Adresse | Länge | Erfassungsart         | Erfassungszyklus | Kommentar |
| <Undefiniert> ...                       | Projektkennung    | <Undefiniert>  | <Symbolischer Zu...  |         | 1     | Zyklisch fortlaufe... | <Undefiniert>    |           |
| <Undefiniert>                           | Bildnummer        | <Undefiniert>  | <Symbolischer Zu...  |         | 5     | Zyklisch fortlaufe... | <Undefiniert>    |           |
| <Undefiniert>                           | Datum/Uhrzeit PLC | <Undefiniert>  | <Symbolischer Zu...  |         | 6     | Zyklisch fortlaufe... | <Undefiniert>    |           |

- **Aktiv**  
Aktiviert den Bereichszeiger.
- **Anzeigename**  
Name des Bereichszeigers, der von WinCC vorgegeben wird.
- **Steuerungsvariable**  
Hier wählen Sie die Steuerungsvariable oder das Variablenarray, das Sie als Datenbereich für den Bereichszeiger projiziert haben.
- **Adresse**  
Aufgrund der symbolischen Adressierung wird in diesem Feld keine Adresse ausgegeben.
- **Länge**  
Die Länge des Bereichszeigers wird von WinCC vorgegeben.
- **Erfassungszyklus**  
Für Bereichszeiger, die vom Bediengerät gelesen werden, legen Sie in diesem Feld den Erfassungszyklus fest. Berücksichtigen Sie, dass eine sehr kurze Erfassungszeit die Performance des Bediengeräts beeinflussen kann.
- **Kommentar**  
Hinterlegen Sie einen Kommentar, z. B. für die Verwendung des Bereichszeigers.

## Zugriff auf Datenbereiche

### Zugriff auf Datenbereiche

Die folgende Tabelle zeigt, wie Bediengerät und Steuerung auf einzelne Datenbereiche zugreifen, lesend (R) oder schreibend (W).

| Datenbereich      | Erforderlich für   | Bediengerät | Steuerung |
|-------------------|--|-------------|-----------|
| Bildnummer        | Auswertung von der Steuerung, welches Bild im Moment aufgeschlagen ist.                          | W           | R         |
| Datensatz         | Übertragung von Datensätzen mit Synchronisation  | R/W         | R/W       |
| Datum/Uhrzeit     | Übertragung von Datum und Uhrzeit vom Bediengerät zur Steuerung                                  | W           | R         |
| Datum/Uhrzeit PLC | Übertragung von Datum und Uhrzeit von der Steuerung zum Bediengerät                              | R           | W         |
| Koordinierung     | Status des Bediengeräts im Steuerungsprogramm abfragen   | W           | R         |
| Projektkennung    | Runtime überprüft, ob die WinCC Projektkennung und das Projekt in der Steuerung konsistent sind. | R           | W         |
| Steuerungsauftrag | Auslösen von Funktionen am Bediengerät durch das Steuerungsprogramm                              | R/W         | R/W       |

### Bereichszeiger "Bildnummer"

#### Funktion

Die Bediengeräte legen im Bereichszeiger "Bildnummer" Informationen über das am Bediengerät aufgerufene Bild ab.

Dadurch ist es möglich, Informationen zum aktuellen Bildinhalt des Bediengeräts zur Steuerung zu übertragen. In der Steuerung können bestimmte Reaktionen getriggert werden, z. B. den Aufruf eines anderen Bildes.

#### Verwendung

Bevor der Bereichszeiger "Bildnummer" verwendet werden kann, muss dieser unter "Kommunikation > Verbindungen" eingerichtet und aktiviert werden. Der Bereichszeiger "Bildnummer" kann nur in **einer** Steuerung und in dieser Steuerung nur **einmal** angelegt werden.

Die Bildnummer wird immer zur Steuerung übertragen, wenn ein neues Bild aktiviert wird oder der Fokus innerhalb eines Bildes von einem Bildobjekt zu einem anderen wechselt.

### Aufbau

Der Bereichszeiger ist ein Datenbereich im Speicher der Steuerung mit einer festen Länge von 5 Worten.

|         | 15                  | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------|---------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1. Wort | Aktueller Bildtyp   |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 2. Wort | Aktuelle Bildnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 3. Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 4. Wort | Aktuelle Feldnummer |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
| 5. Wort | Reserviert          |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

- Aktueller Bildtyp  
"1" für Grundbild oder  
"4" für Permanentfenster
- Aktuelle Bildnummer  
1 bis 32767
- Aktuelle Feldnummer  
1 bis 32767

### Bereichszeiger "Datum/Uhrzeit"

#### Funktion

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von dem Bediengerät zur Steuerung verwendet.

Die Steuerung schreibt den Steuerauftrag "41" in das Auftragsfach.

Mit der Auswertung des Steuerauftrags schreibt das Bediengerät sein aktuelles Datum und die Uhrzeit in den im Bereichszeiger "Datum/Uhrzeit" projektierten Datenbereich. Alle Angaben sind BCD-codiert.

Wenn in einem Projekt mehrere Verbindungen projektiert sind, und in einer der Verbindungen der Bereichszeiger "Datum / Uhrzeit" verwendet werden soll, so muss der Bereichszeiger für jede projektierte Verbindung aktiviert werden.

---

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit" projektiert haben, dann können Sie den Bereichszeiger "Datum/Uhrzeit PLC" nicht verwenden.

---

Der Datenbereich Datum/Uhrzeit hat folgenden Aufbau:

| Datenwort | Linkes Byte       |  |  |   | Rechtes Byte          |  |  |   |         |
|-----------|-------------------|--|--|---|-----------------------|--|--|---|---------|
|           | 15                |  |  | 8 | 7                     |  |  | 0 |         |
| n+0       | Reserviert        |  |  |   | Stunde (0-23)         |  |  |   | Uhrzeit |
| n+1       | Minute (0-59)     |  |  |   | Sekunde (0-59)        |  |  |   |         |
| n+2       | Reserviert        |  |  |   | Reserviert            |  |  |   |         |
| n+3       | Reserviert        |  |  |   | Wochentag (1-7, 1=So) |  |  |   | Datum   |
| n+4       | Tag (1-31)        |  |  |   | Monat (1-12)          |  |  |   |         |
| n+5       | Jahr (80-99/0-29) |  |  |   | Reserviert            |  |  |   |         |

---

#### Hinweis

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

---

## Bereichszeiger "Datum/Uhrzeit PLC"

### Funktion

Dieser Bereichszeiger wird für die Übertragung von Datum und Uhrzeit von der Steuerung zum Bediengerät verwendet. Sie setzen diesen Bereichszeiger ein, wenn die Steuerung Master für die Zeit ist.

Die Steuerung lädt den Datenbereich des Bereichszeigers. Alle Angaben sind BCD-codiert.

Das Bediengerät liest zyklisch die Daten über den projektierten Erfassungszyklus und synchronisiert sich.

---

#### Hinweis

Wählen Sie in der Projektierung den Erfassungszyklus für den Bereichszeiger Datum/Uhrzeit nicht zu klein, da dies die Performance des Bediengeräts beeinflusst.  
Empfehlung: Erfassungszyklus 1 Minute, wenn Ihr Prozess diesen Zyklus erlaubt.

---

"Datum/Uhrzeit PLC" ist ein globaler Bereichszeiger und kann in einem Projekt nur ein Mal projektiert werden.

---

#### Hinweis

Wenn Sie den Bereichszeiger "Datum/Uhrzeit PLC" projektiert haben, können Sie den Bereichszeiger "Datum/Uhrzeit" nicht verwenden.

---

Der Datenbereich "Datum/Uhrzeit PLC" hat folgenden Aufbau:

| Datenwort         | Linkes Byte       |       |   | Rechtes Byte   |                       |   |
|-------------------|-------------------|-------|---|----------------|-----------------------|---|
|                   | 15                | ..... | 8 | 7              | .....                 | 0 |
| n+0               | Jahr (80-99/0-29) |       |   | Monat (1-12)   |                       |   |
| n+1               | Tag (1-31)        |       |   | Stunde (0-23)  |                       |   |
| n+2               | Minute (0-59)     |       |   | Sekunde (0-59) |                       |   |
| n+3               | Reserviert        |       |   | Reserviert     | Wochentag (1-7, 1=So) |   |
| n+4 <sup>1)</sup> | Reserviert        |       |   | Reserviert     |                       |   |
| n+5 <sup>1)</sup> | Reserviert        |       |   | Reserviert     |                       |   |

- 1) Die beiden Datenwörter müssen im Datenbereich vorhanden sein, um eine Übereinstimmung des Datenformats mit WinCC sicherzustellen und das Lesen falscher Informationen zu vermeiden.

---

**Hinweis**

Beachten Sie bei der Eingabe in den Datenbereich "Jahr", dass die Werte 80-99 die Jahreszahlen 1980 bis 1999 und die Werte 0-29 die Jahreszahlen 2000 bis 2029 ergeben.

---

**Bereichszeiger "Koordination"**

**Funktion**

Der Bereichszeiger "Koordination" dient zur Realisierung der folgenden Funktionen:

- Anlauf des Bediengeräts im Steuerungsprogramm erkennen
- Aktuelle Betriebsart des Bediengeräts im Steuerungsprogramm erkennen
- Kommunikationsbereitschaft des Bediengeräts im Steuerungsprogramm erkennen

Der Bereichszeiger "Koordination" hat eine Länge von einem Wort.

**Verwendung**

---

**Hinweis**

Bei jeder Aktualisierung des Bereichszeigers durch das Bediengerät wird immer der komplette Koordinierungsbereich geschrieben.  
Das SPS-Programm darf deshalb im Koordinierungsbereich keine Änderungen vornehmen.

---





## Verwendung

Um diesen Bereichszeiger zu verwenden, richten Sie bei der Projektierung Folgendes ein:

- Angabe der Version, welche die Projektierung hat. Möglicher Wert zwischen 1 und 255. Sie geben die Version ein im Editor "Runtime-Einstellungen > Allgemein" im Bereich "Identifizierung".
- Datenadresse des Werts für die Version, die in der Steuerung hinterlegt ist: Sie geben die Datenadresse ein im Editor "Kommunikation > Verbindungen".

## Ausfall einer Verbindung

Ein Verbindungsausfall zu einem Gerät, auf dem der Bereichszeiger "Projektkennung" projiziert ist, hat zur Folge, dass auch alle anderen Verbindungen im Projekt "Offline" geschaltet werden.

Dieses Verhalten hat folgende Voraussetzungen:

- Sie haben in einem Projekt mehrere Verbindungen projiziert.
- Sie verwenden in mindestens einer Verbindung den Bereichszeiger "Projektkennung".

Folgende Ursachen können Verbindungen in den Zustand "Offline" setzen:

- Die Steuerung ist nicht erreichbar.
- Die Verbindung wurde im Engineering System offline geschaltet.

## Bereichszeiger "Steuerungsauftrag"

### Funktion

Über das Steuerungsauftrags-Fach können dem Bediengerät Steuerungsaufträge gegeben und damit Aktionen am Bediengerät getriggert werden. Zu diesen Funktionen gehören z. B.:

- Bild anzeigen
- Datum und Uhrzeit stellen

### Datenstruktur

Im ersten Wort des Steuerungsauftrags-Fachs steht die Auftragsnummer. Je nach Steuerungsauftrag können dann bis zu drei Parameter übergeben werden.

| Wort | Linkes Byte | Rechtes Byte   |
|------|-------------|----------------|
| n+0  | 0           | Auftragsnummer |
| n+1  |             | Parameter 1    |
| n+2  |             | Parameter 2    |
| n+3  |             | Parameter 3    |

Wenn das erste Wort des Steuerungsauftrags-Fachs ungleich 0 ist, wertet das Bediengerät den Steuerungsauftrag aus. Aus diesem Grund müssen zuerst die Parameter in das Steuerungsauftrags-Fach eingetragen werden und dann erst die Auftragsnummer.

Wenn das Bediengerät den Steuerungsauftrag angenommen hat, wird das erste Wort wieder auf 0 gesetzt. Die Ausführung des Steuerungsauftrags ist zu diesem Zeitpunkt im Allgemeinen noch nicht abgeschlossen.

## Steuerungsaufträge

Nachfolgend sind alle Steuerungsaufträge und deren Parameter aufgelistet. Die Spalte "Nr." enthält die Auftragsnummer des Steuerungsauftrags. Generell können Steuerungsaufträge nur dann von der Steuerung getriggert werden, wenn das Bediengerät im Online-Betrieb ist.

### Hinweis

Beachten Sie, dass nicht alle Bediengeräte Steuerungsaufträge unterstützen.

| Nr.       | Funktion   |   |
|-----------|--|---|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>   |   |
|           | Parameter 1  | Linkes Byte: -<br>Rechtes Byte: Stunden (0-23)                      |
|           | Parameter 2  | Linkes Byte: Minuten (0-59)<br>Rechtes Byte: Sekunden (0-59)        |
|           | Parameter 3  | -   |
| <b>15</b> | <b>Datum stellen (BCD-codiert)</b>   |   |
|           | Parameter 1  | Linkes Byte: -<br>Rechtes Byte: Wochentag<br>(1-7: Sonntag-Samstag) |
|           | Parameter 2  | Linkes Byte: Tag (1-31)<br>Rechtes Byte: Monat (1-12)               |
|           | Parameter 3  | Linkes Byte: Jahr   |
| <b>23</b> | <b>Benutzer anmelden</b>   |   |
|           | Meldet den Benutzer mit dem Namen "PLC User" mit der im Parameter 1 übergebenen Gruppennummer am Bediengerät an.<br>Voraussetzung für die Anmeldung ist, dass die übergebene Gruppennummer im Projekt vorhanden ist. |   |
|           | Parameter 1  | Gruppennummer 1 - 255   |
|           | Parameter 2, 3   | -   |
| <b>24</b> | <b>Benutzer abmelden</b>   |   |
|           | Meldet den aktuell angemeldeten Benutzer ab.<br>(Funktion entspricht der Systemfunktion "Abmelden")  |   |
|           | Parameter 1, 2, 3  | -   |
| <b>40</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |   |
|           | (Im S7-Format DATE_AND_TIME)<br>Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.   |   |
|           | Parameter 1, 2, 3  | -   |

| Nr.       | Funktion   |  |
|-----------|--|--|
| <b>14</b> | <b>Uhrzeit stellen (BCD-codiert)</b>   |  |
| <b>41</b> | <b>Datum/Uhrzeit zur Steuerung übertragen</b>  |  |
|           | (Im Format OP/MP)<br>Zwischen zwei Aufträgen müssen mindestens 5 Sekunden liegen, da sonst das Bediengerät überlastet wird.  |  |
|           | Parameter 1, 2, 3  | -  |
| <b>46</b> | <b>Variable aktualisieren</b>  |  |
|           | Veranlasst das Bediengerät den aktuellen Wert der Variablen aus der Steuerung zu lesen, deren Aktualisierungskennung mit dem im Parameter 1 übergebenen Wert übereinstimmt.<br>(Funktion entspricht der Systemfunktion "AktualisiereVariable") |  |
|           | Parameter 1  | 1 - 100  |
| <b>49</b> | <b>Meldepuffer löschen</b>   |  |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Warnings" aus dem Meldepuffer.  |  |
|           | Parameter 1, 2, 3  | -  |
| <b>50</b> | <b>Meldepuffer löschen</b>   |  |
|           | Löscht alle Analogmeldungen und alle Bitmeldungen der Meldeklasse "Errors" aus dem Meldepuffer.  |  |
|           | Parameter 1, 2, 3  | -  |
| <b>51</b> | <b>Bildanwahl <sup>1)</sup></b>  |  |
|           | Parameter 1  | Bildnummer   |
|           | Parameter 2  | -  |
|           | Parameter 3  | Feldnummer   |
| <b>69</b> | <b>Datensatz aus Steuerung lesen</b>   |  |
|           | Parameter 1  | Rezeptnummer (1-999)   |
|           | Parameter 2  | Datensatznummer (1-65535)  |
|           | Parameter 3  | 0: Vorhandenen Datensatz nicht überschreiben<br>1: Vorhandenen Datensatz überschreiben |
| <b>70</b> | <b>Datensatz in Steuerung schreiben</b>  |  |
|           | Parameter 1  | Rezeptnummer (1-999)   |
|           | Parameter 2  | Datensatznummer (1-65535)  |
|           | Parameter 3  | -  |

<sup>1)</sup> Auf den Bediengeräten OP 73, OP 77A und TP 177A wird der Steuerungsauftrag "Bildanwahl" auch dann ausgeführt, wenn die Bildschirmtastatur geöffnet ist.

## Bereichszeiger "Datensatz"

## Bereichszeiger "Datensatz"

### Funktion

Bei der Übertragung von Datensätzen zwischen Bediengerät und Steuerung greifen beide Kommunikationspartner wechselseitig auf gemeinsame Kommunikationsbereiche in der Steuerung zu.

### Übertragungsarten

Für die Übertragung von Datensätzen zwischen Bediengerät und Steuerung gibt es zwei Möglichkeiten:

- Übertragung ohne Synchronisation
- Übertragung mit Synchronisation über das Datenfach

Datensätze werden immer direkt übertragen. D. h., die Variablenwerte werden direkt, ohne den Umweg über eine Zwischenablage, aus der Adresse gelesen oder in die Adresse geschrieben, die für die Variable projektiert ist.

### Übertragung von Datensätzen anstoßen

Für den Anstoß der Übertragung gibt es drei Möglichkeiten:

- Bedienung in der Rezepturanzeige
- Steuerungsaufträge  
Die Übertragung der Datensätze kann auch durch die Steuerung getriggert werden.
- Auslösen projektiierter Funktionen

Wenn die Übertragung von Datensätzen durch eine projektierte Funktion oder einen Steuerungsauftrag getriggert wird, ist die Rezepturanzeige am Bediengerät weiterhin ungehindert bedienbar. Die Datensätze werden im Hintergrund übertragen.

Das gleichzeitige Abarbeiten mehrerer Übertragungsanforderungen ist jedoch nicht möglich. In diesem Fall lehnt das Bediengerät eine weitere Übertragung mit einer Systemmeldung ab.

### Übertragung ohne Synchronisation

Bei der asynchronen Übertragung von Datensätzen zwischen Bediengerät und Steuerung findet keine Koordination über gemeinsam benutzte Datenbereiche statt. Die Einrichtung eines Datenbereichs beim Projektieren ist daher nicht nötig.

Die asynchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Ein unkontrolliertes Überschreiben der Daten durch die Kommunikationspartner kann systembedingt ausgeschlossen werden.
- Die Steuerung braucht keine Informationen über die Rezepturnummer und die Datensatznummer.
- Die Übertragung von Datensätzen wird durch Bedienung am Bediengerät getriggert.

### Werte lesen

Beim Anstoß der Übertragung zum Lesen werden die Werte aus den Steuerungsadressen gelesen und zum Bediengerät übertragen.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die Werte werden in das Bediengerät geladen. Im Bediengerät können Sie diese weiterverarbeiten, z. B. Werte ändern, speichern etc.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte werden sofort auf dem Datenträger gespeichert.

### Werte schreiben

Beim Anstoß der Übertragung zum Schreiben werden die Werte in die Steuerungsadressen geschrieben.

- Anstoß über Bedienung in der Rezepturanzeige:  
Die aktuellen Werte werden in die Steuerung geschrieben.
- Anstoß über Funktion oder Steuerungsauftrag:  
Die Werte vom Datenträger werden in die Steuerung geschrieben.

### Übertragung mit Synchronisation

Bei der synchronen Übertragung setzen beide Kommunikationspartner Status-Bits im gemeinsam benutzten Datenbereich. Dadurch können Sie in Ihrem Steuerungsprogramm ein unkontrolliertes gegenseitiges Überschreiben der Daten verhindern.

### Anwendung

Die synchrone Datensatz-Übertragung bietet sich z. B. in folgenden Fällen an:

- Die Steuerung ist der "aktive Partner" bei der Übertragung von Datensätzen.
- In der Steuerung werden Informationen über die Rezepturnummer und die Datensatznummer ausgewertet.
- Die Übertragung von Datensätzen wird per Steuerungsauftrag getriggert.

## Voraussetzungen

Damit Datensätze synchronisiert zwischen Bediengerät und Steuerung übertragen werden, müssen beim Projektieren folgende Voraussetzungen erfüllt sein:

- Ein Bereichszeiger ist eingerichtet: Editor "Kommunikation > Verbindungen" unter "Bereichszeiger".
- In der Rezeptur ist die Steuerung angegeben, mit der das Bediengerät die Übertragung der Datensätze synchronisiert:  
Editor "Rezepturen" im Inspektorfenster unter "Allgemein > Synchronisation > Einstellungen" die Auswahl "Koordinierte Übertragung der Datensätze"

## Aufbau des Datenbereichs

Der Datenbereich hat eine feste Länge von 5 Worten. Der Datenbereich ist wie folgt aufgebaut:

|         |                                       |          |
|---------|---------------------------------------|----------|
|         | <b>15</b>                             | <b>0</b> |
| 1. Wort | Aktuelle Rezepturnummer (1 - 999)     |          |
| 2. Wort | Aktuelle Datensatznummer (0 - 65.535) |          |
| 3. Wort | Reserviert                            |          |
| 4. Wort | Status (0, 2, 4, 12)                  |          |
| 5. Wort | Reserviert                            |          |

- Status  
Das Statuswort (Wort 4) kann folgende Werte annehmen:

| Wert    |           | Bedeutung                            |
|---------|-----------|--------------------------------------|
| Dezimal | Binär     |                                      |
| 0       | 0000 0000 | Übertragung zulässig, Datenfach frei |
| 2       | 0000 0010 | Übertragung läuft.                   |
| 4       | 0000 0100 | Übertragung fehlerfrei beendet       |
| 12      | 0000 1100 | Übertragung mit Fehler beendet       |

## Ablauf der Übertragung bei Bedienung in der Rezepturanzeige

### Lesen aus der Steuerung durch Bedienung in der Rezepturanzeige

| Schritt | Aktion   |                            |
|---------|--|----------------------------|
| 1       | Prüfung: Statuswort = 0?   |                            |
|         | Ja   | Nein                       |
| 2       | Das Bediengerät trägt die zu lesende Rezepturnummer sowie den Status "Übertragung läuft" im Datenfach ein und setzt die Datensatznummer auf 0. | Abbruch mit Systemmeldung. |

| Schritt | Aktion  |
|---------|---|
| 3       | Das Bediengerät liest die Werte aus der Steuerung und zeigt diese in der Rezepturanzeige an.<br>Bei Rezepturen mit synchronisierten Variablen werden die Werte aus der Steuerung auch in die Variablen geschrieben. |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".   |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |

### Schreiben in die Steuerung durch Bedienung in der Rezepturanzeige

| Schritt   | Aktion  |    |      |   |                            |
|---|---|----|------|---|----------------------------|
|   | Prüfung: Statuswort = 0?  |    |      |   |                            |
| 1   | <table border="1" style="width: 100%;"> <thead> <tr> <th>Ja</th> <th>Nein</th> </tr> </thead> <tbody> <tr> <td>Das Bediengerät trägt die zu schreibende Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.</td> <td>Abbruch mit Systemmeldung.</td> </tr> </tbody> </table> | Ja | Nein | Das Bediengerät trägt die zu schreibende Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein. | Abbruch mit Systemmeldung. |
| Ja  | Nein  |    |      |   |                            |
| Das Bediengerät trägt die zu schreibende Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein. | Abbruch mit Systemmeldung.  |    |      |   |                            |
| 2   | Das Bediengerät schreibt die aktuellen Werte in die Steuerung.<br>Bei Rezepturen mit synchronisierten Variablen werden die geänderten Werte zwischen Rezepturanzeige und Variablen abgeglichen und dann in die Steuerung geschrieben.   |    |      |   |                            |
| 3   | Das Bediengerät setzt den Status "Übertragung beendet".   |    |      |   |                            |
| 4   | Das Steuerungsprogramm kann jetzt ggf. die übertragenen Daten auswerten.  |    |      |   |                            |
| 5   | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |    |      |   |                            |

#### Hinweis

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

#### Hinweis

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach ist auf "Übertragung beendet" gesetzt.
- Der Status im Datenfach ist auf "Übertragung mit Fehler beendet" gesetzt.

### Ablauf der Übertragung bei Steuerungsauftrag

Die Übertragung von Datensätzen zwischen Bediengerät und Steuerung kann vom Bediengerät oder von der Steuerung initiiert werden.

Für diese Art der Übertragung stehen die beiden Steuerungsaufträge Nr. 69 und Nr. 70 zur Verfügung.



**Nr. 69: Datensatz aus Steuerung lesen ("SPS → DAT")**

Der Steuerungsauftrag Nr. 69 überträgt Datensätze von der Steuerung zum Bediengerät. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Linkes Byte (LB)   | Rechtes Byte (RB) |
|--------|--|-------------------|
| Wort 1 | 0  | 69                |
| Wort 2 | Rezepturnummer (1-999)   |                   |
| Wort 3 | Datensatznummer (1-65.535)   |                   |
| Wort 4 | Vorhandenen Datensatz nicht überschreiben: 0<br>Vorhandenen Datensatz überschreiben: 1 |                   |

**Nr. 70: Datensatz in Steuerung schreiben ("DAT → SPS")**

Der Steuerungsauftrag Nr. 70 überträgt Datensätze vom Bediengerät zur Steuerung. Der Steuerungsauftrag ist wie folgt aufgebaut:

|        | Linkes Byte (LB)           | Rechtes Byte (RB) |
|--------|----------------------------|-------------------|
| Wort 1 | 0                          | 70                |
| Wort 2 | Rezepturnummer (1-999)     |                   |
| Wort 3 | Datensatznummer (1-65.535) |                   |
| Wort 4 | —                          |                   |

**Ablauf bei Lesen aus der Steuerung durch Steuerungsauftrag "SPS → DAT" (Nr. 69)**

| Schritt | Aktion  |                           |
|---------|---|---------------------------|
| 1       | Prüfung: Statuswort = 0?  |                           |
|         | Ja  | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese Werte in dem Datensatz, der im Steuerungsauftrag angegeben ist.   |                           |
| 4       | <ul style="list-style-type: none"> <li>• Falls im Auftrag "Überschreiben" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben. Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>• Falls im Auftrag "Nicht überschreiben" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                           |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                           |

**Ablauf bei Schreiben in die Steuerung durch Steuerungsauftrag "DAT → SPS" (Nr. 70)**

| Schritt | Aktion   |                           |
|---------|--|---------------------------|
| 1       | Prüfung: Statuswort = 0?   |                           |
|         | Ja   | Nein                      |
| 2       | Das Bediengerät trägt die im Auftrag angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch ohne Rückmeldung. |
| 3       | Das Bediengerät holt die Werte des im Auftrag angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.   |                           |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                           |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                           |

**Ablauf der Übertragung bei Anstoß durch projizierte Funktion**

**Lesen aus der Steuerung durch projizierte Funktion**

| Schritt | Aktion  |                            |
|---------|---|----------------------------|
| 1       | Prüfung: Statuswort = 0?  |                            |
|         | Ja  | Nein                       |
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.   | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät liest die Werte aus der Steuerung und speichert diese in dem Datensatz ab, der in der Funktion angegeben ist.   |                            |
| 4       | <ul style="list-style-type: none"> <li>Falls in der Funktion bei "Überschreiben" "Ja" gewählt wurde, wird ein vorhandener Datensatz ohne Rückfrage überschrieben. Das Bediengerät setzt den Status "Übertragung beendet".</li> <li>Falls in der Funktion bei "Überschreiben" "Nein" gewählt wurde und der Datensatz bereits existiert, bricht das Bediengerät den Vorgang ab und trägt 0000 1100 in das Statuswort des Datenfachs ein.</li> </ul> |                            |
| 5       | Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen.   |                            |

**Schreiben in die Steuerung durch projizierte Funktion**

| Schritt | Aktion                   |      |
|---------|--------------------------|------|
| 1       | Prüfung: Statuswort = 0? |      |
|         | Ja                       | Nein |

| Schritt | Aktion   |                            |
|---------|--|----------------------------|
| 2       | Das Bediengerät trägt die in der Funktion angegebene Rezeptur- und Datensatznummer sowie den Status "Übertragung läuft" im Datenfach ein.  | Abbruch mit Systemmeldung. |
| 3       | Das Bediengerät holt die Werte des in der Funktion angegebenen Datensatzes vom Datenträger und schreibt diese Werte in die Steuerung.  |                            |
| 4       | Das Bediengerät setzt den Status "Übertragung beendet".  |                            |
| 5       | Das Steuerungsprogramm kann jetzt die übertragenen Daten auswerten.<br>Um eine erneute Übertragung zu ermöglichen, muss das Steuerungsprogramm das Statuswort wieder auf 0 setzen. |                            |

## Mögliche Fehlerursachen bei der Übertragung von Datensätzen

### Mögliche Fehlerursachen

Falls die Übertragung von Datensätzen mit Fehler beendet wird, so kann dies u. a. folgende Ursachen haben:

- Variablen-Adresse in der Steuerung nicht eingerichtet
- Überschreiben von Datensätzen nicht möglich
- Rezepturnummer nicht vorhanden
- Datensatznummer nicht vorhanden

---

#### Hinweis

Das Setzen des Statusworts darf immer nur durch das Bediengerät erfolgen. Die Steuerung darf das Statuswort nur wieder auf null rücksetzen.

---

#### Hinweis

Die Auswertung der Rezeptur- und Datensatznummer in der Steuerung darf wegen der Datenkonsistenz erst erfolgen, wenn eine der folgenden Bedingungen erfüllt ist:

- Der Status im Datenfach wurde auf "Übertragung beendet" gesetzt.
  - Der Status im Datenfach wurde auf "Übertragung mit Fehler beendet" gesetzt.
- 

## Reaktion auf fehlerbedingten Abbruch

Das Bediengerät reagiert auf einen fehlerbedingten Abbruch der Übertragung von Datensätzen wie folgt:

- Anstoß über Bedienung in der Rezepturanzeige  
Hinweise in der Statuszeile der Rezepturanzeige und Ausgabe von Systemmeldungen
- Anstoß über Funktion  
Ausgabe von Systemmeldungen
- Anstoß über Steuerungsauftrag  
Keine Rückmeldung am Bediengerät

Unabhängig davon können Sie den Status der Übertragung durch Abfragen des Statusworts im Datenfach auswerten.

## 10.9 Übergreifende Funktionen nutzen

### 10.9.1 Grundlagen zum Bediengeräte-Assistenten

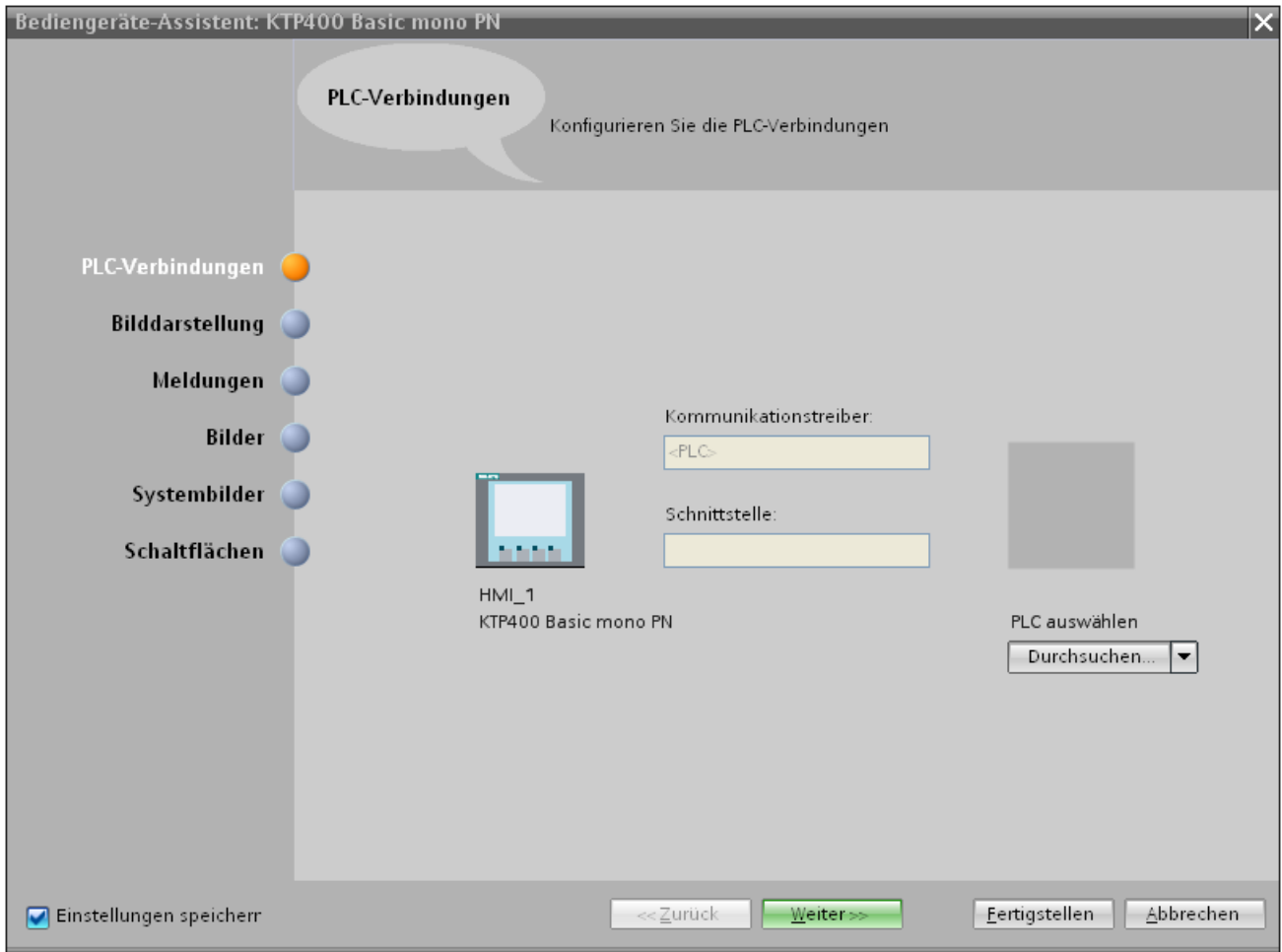
#### Einleitung

Wenn Sie in Ihrem Projekt ein neues Bediengerät anlegen, startet automatisch der Bediengeräte-Assistent.

#### Bediengeräte-Assistent

Der Bediengeräte-Assistent führt Sie Schritt für Schritt durch die einzelnen Dialoge und unterstützt Sie beim Anlegen eines Geräts. Mit dem Bediengeräte-Assistenten legen Sie

grundsätzliche Einstellungen für ihr Bediengerät fest, z. B. die Bilddarstellung oder die Verbindung zu Ihrer PLC.



## 10.9.2 Arbeiten mit Bibliotheken

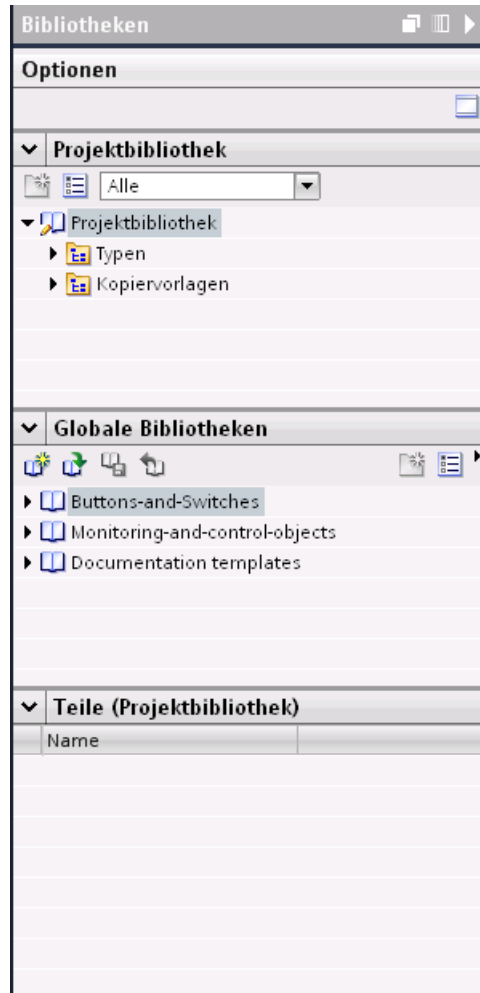
### 10.9.2.1 Grundlagen zu Bibliotheken

#### Einleitung

In den Bibliotheken legen Sie Objekte zentral ab, die Sie häufig benötigen. Ein in der Bibliothek abgelegtes Objekt müssen Sie nur einmal konfigurieren. Danach können Sie es beliebig oft wiederverwenden. Bibliotheksobjekte erweitern die Menge der verfügbaren Bildobjekte und steigern durch Mehrfachverwendung von Fertigobjekten die Effektivität beim Projektieren.

In der Task Card "Bibliotheken" werden die Bibliotheken verwaltet. Folgende Bibliotheken stehen Ihnen zur Verfügung:

- Projektbibliothek
- Globale Bibliotheken



---

**Hinweis**

Eine Symbolbibliothek befindet sich in der Task Card "Werkzeuge" in der Palette "Grafiken".

---

### Projektbibliothek

In jedem Projekt gibt es genau eine Projektbibliothek. Die Objekte der Projektbibliothek werden mit den Projektdaten gespeichert und stehen nur für das Projekt zur Verfügung, in dem die Bibliothek angelegt wurde. Wenn das Projekt auf einen anderen PC verschoben wird, wird eine darin angelegte Projektbibliothek mitverschoben.

Um das Bibliotheksobjekt der Projektbibliothek auch in anderen Projekten nutzen zu können, verschieben oder kopieren Sie das Objekt in eine Globale Bibliothek.

## **Globale Bibliotheken**

Neben den Objekten aus der Projektbibliothek können in ein Projekt auch Objekte aus Globalen Bibliotheken eingebunden werden. Eine Globale Bibliothek wird unabhängig von den Projektdaten in einer eigenen Datei mit der Erweiterung \*.al11 gespeichert.

Ein Projekt kann auf mehrere Globale Bibliotheken zugreifen. Dieselbe Globale Bibliothek kann von mehreren Projekten gleichzeitig verwendet werden.

Wenn ein Bibliotheksobjekt von einem Projekt geändert wurde, wird die Bibliothek danach auch in den Projekten geändert, die diese Bibliothek geöffnet haben.

## **Bibliotheksobjekte**

Eine Bibliothek kann alle WinCC-Objekte enthalten. Beispiele:

- Ganzes Bediengerät
- Bilder
- Anzeige- und Bedienobjekte einschließlich Variablen und Funktionen
- Grafiken
- Variablen
- Meldungen
- Text- und Grafiklisten
- Bildbausteine
- Strukturen

## **Siehe auch**

Kopiervorlagen und Typen (Seite 3841)

### **10.9.2.2 Kopiervorlagen und Typen**

#### **Einleitung**

Sowohl die "Projektbibliothek" als auch die "Globale Bibliothek" enthalten die beiden Ordner "Kopiervorlagen" und "Typen". Sie können die Bibliotheksobjekte entweder als Kopiervorlage oder als Typ erzeugen bzw. verwenden.

#### **Kopiervorlagen**

Um voneinander unabhängige Kopien des Bibliotheksobjekts zu erzeugen, verwenden Sie Kopiervorlagen.

## Typen

Von Objekten des Ordners "Typen" erstellen Sie Instanzen und verwenden diese in Ihrem Projekt. Die Instanzen sind an ihren jeweiligen Typ gebunden. Änderungen einer Instanz ändern auch alle anderen Instanzen. Typen werden in der Task Card "Bibliotheken" mit einem grünen Dreieck gekennzeichnet.

## Verwalten der Bibliotheksobjekte

Sie kopieren und verschieben Bibliotheksobjekte nur innerhalb der gleichen Bibliothek. Sie kopieren Kopiervorlagen nur in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen". Auch Typen können Sie nur in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen" einfügen.

## Siehe auch

Grundlagen zu Bibliotheken (Seite 3839)

### 10.9.2.3 Bibliotheken in WinCC

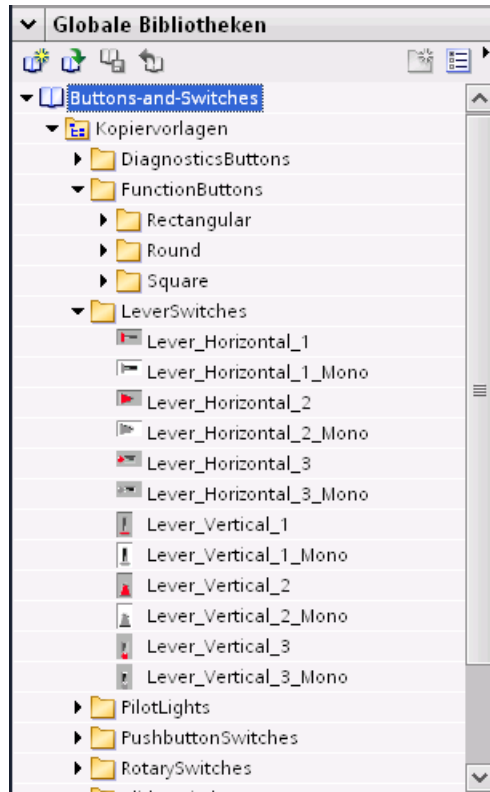
## Einleitung

Mit WinCC werden umfangreiche Bibliotheken mitgeliefert. Thematisch in Ordnern gegliedert, finden Sie vorgefertigte grafische Objekte, die Sie in Bildern für das Bedienen und Beobachten Ihrer Anlage verwenden können.



## Globale Bibliothek "Buttons and Switches"

Die Bibliothek "Buttons and Switches" bietet Ihnen eine große Auswahl an Schaltern und Schaltflächen.



Die Ordner gliedern Schalter bzw. Schaltflächen in Kategorien. Im Ordner "DiagnosticsButtons" finden Sie z. B. das Objekt "System-Diagnoseindikator". Das Objekt "System-Diagnoseindikator" verwenden Sie für die Systemdiagnose in Ihrer Anlage.

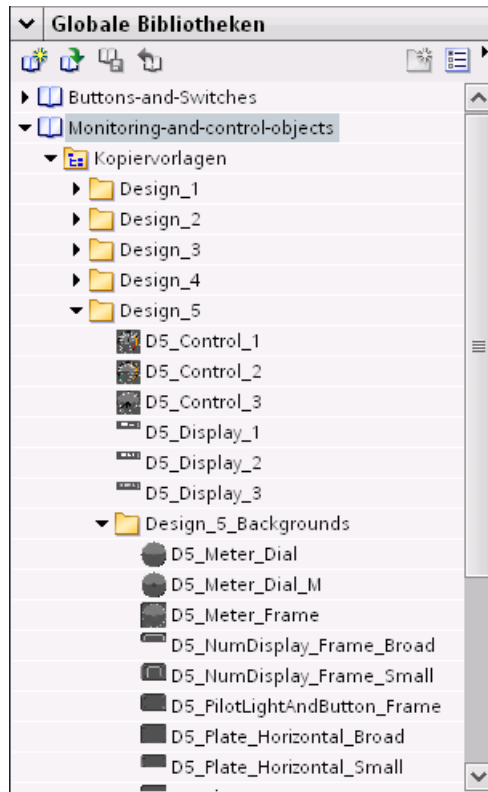
### Hinweis

Die Objekte im Ordner "DiagnosticsButtons" können Sie nur auf Comfort Panels nutzen.

Die Objekte mit der Bezeichnung "Switch" im Objektnamen bzw. im zugehörigen Ordernamen können Sie nicht in Runtime-Professional einsetzen.

### Globale Bibliothek "Monitoring and Control objects"

Die Bibliothek "Monitoring and Control objects" bietet Ihnen komplexere Bedien- und Anzeigeobjekte in mehreren Designs sowie dazu passende Kontrolllampen, Schaltflächen und Schalter.



Zusätzlich sind zu den Designs passende Grafikanzeigen in den Ordnern "Design\_Backgrounds" abgelegt, die zur kundenspezifischen Erweiterung des Bibliotheksumfangs als Objekthintergründe genutzt werden können.

---

#### Hinweis

Die Objekte mit der Bezeichnung "Switch" im Objektnamen können Sie nicht in Runtime Professional einsetzen. Das Gleiche gilt für das Objekt "D5\_Display\_3" mit dem darin enthaltenen Datum/Uhrzeit-Feld.

---

#### 10.9.2.4 Bibliotheksobjekte anzeigen

##### Einleitung

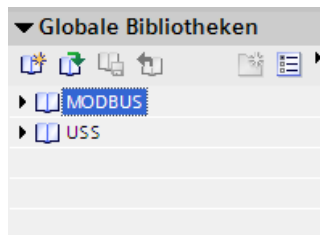
Die Bibliotheken werden als Dateiordner in der entsprechenden Palette angezeigt. Die in der Bibliothek enthaltenen Elemente werden im Dateiordner und in der Palette "Elemente" angezeigt.

## Voraussetzung

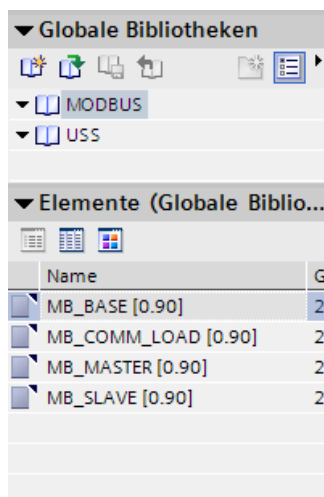
- Mindestens ein Bibliotheksobjekt ist in einer Bibliothek angelegt.
- Die Task Card "Bibliotheken" ist geöffnet.

## Vorgehen




1. Selektieren Sie die Bibliothek in der entsprechenden Palette, deren Bibliotheksobjekte Sie anzeigen wollen



2. Klicken Sie auf das Symbol . Die enthaltenen Bibliotheksobjekte werden in der Palette "Elemente" angezeigt.




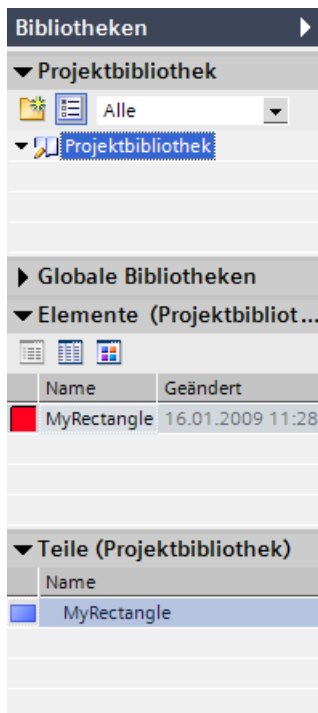
3. Klicken Sie auf eines der folgenden Symbole:

| Symbol  | Beschreibung                                   |
|---|--|
|  | Elementansicht im Detailmodus                  |
|  | Elementsansicht im Listenmodus                 |
|  | Elementansicht im Übersichtsmodus mit Symbolen |

Wenn der Bibliothek über Mehrfachauswahl mehreren Objekten zugeordnet werden, wird in der Palette "Elemente" nur eines der Objekte angezeigt. Die einzelnen Bestandteile dieses Elements werden in der Palette "Teile" angezeigt.

### Teile der Bibliotheksobjekte anzeigen

1. Selektieren Sie die Bibliothek in der entsprechenden Palette, aus der Sie die Bestandteile eines Elements sehen wollen.
2. Klicken Sie auf das Symbol .
3. Die enthaltenen Bibliotheksobjekte werden in der Palette "Elemente" angezeigt.
4. Selektieren Sie das Element.  
In der Palette "Teile" sehen Sie die Objekte, aus denen das Element besteht.



### Ergebnis

Die Bibliotheksobjekte werden entsprechend der vorgenommenen Konfiguration angezeigt. Die Bestandteile der Bildbausteine werden angezeigt.

## 10.9.2.5 Bibliotheksobjekte verwalten

### Einleitung

Sie können jederzeit Bibliotheksobjekte zwischen Kategorien innerhalb einer Bibliothek verschieben oder kopieren. Bibliotheksobjekte, die Sie nicht mehr benötigen, löschen Sie.

---

#### Hinweis

##### Kopiervorlagen und Typen

Sie kopieren und verschieben Bibliotheksobjekte nur innerhalb der gleichen Bibliothek. Sie kopieren Kopiervorlagen nur in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen". Auch Typen können Sie nur in den Ordner "Typen" oder einen beliebigen Unterordner von "Typen" einfügen.

---

### Voraussetzung

- Eine Bibliothek mit mehreren Kategorien und mindestens einem Objekt ist geöffnet.
- Das Bibliotheksobjekt wird angezeigt.

### Bibliotheksobjekt verschieben

1. Selektieren Sie das Bibliotheksobjekt.
2. Ziehen Sie das Objekt mit Drag&Drop in den gewünschten Ordner.

### Bibliotheksobjekt kopieren

1. Selektieren Sie das Bibliotheksobjekt.
2. Wählen Sie im Kontextmenü den Befehl "Kopieren".
3. Markieren Sie den Ordner, in dem Sie das Bibliotheksobjekt einfügen wollen.
4. Wählen Sie im Kontextmenü den Befehl "Einfügen".

### Bibliotheksobjekt löschen

1. Selektieren Sie das Bibliotheksobjekt.
2. Wählen Sie im Kontextmenü den Befehl "Löschen".

### Bibliotheksobjekt umbenennen

Um ein Bibliotheksobjekt umzubenennen, gehen Sie folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Objekt, das Sie umbenennen möchten.
2. Wählen Sie im Kontextmenü den Befehl "Umbenennen".
3. Geben Sie den neuen Namen ein.

### 10.9.2.6 Objekt in einer Bibliothek hinterlegen

#### Einleitung

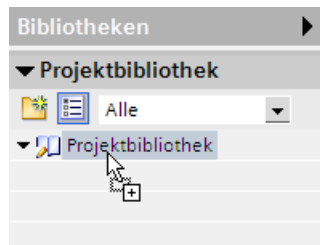
In Bibliotheken können Sie alle WinCC-Objekte, z. B. Bilder, Variablen, Grafikobjekte oder Meldungen hinterlegen. Sie ziehen dazu per Drag&Drop aus dem Arbeitsbereich, der Projektfenster oder der Detailansicht das entsprechende Objekt in die Bibliothek. Wenn Sie die Bibliothek in Kategorien eingeteilt haben, können Sie ein Objekt auch direkt in eine bestimmte Kategorie einfügen.

#### Voraussetzung

- Der Editor "Bilder" ist geöffnet.
- Ein Bildobjekt ist im Arbeitsbereich des Bildes angelegt.
- Die angelegten Bibliotheken werden angezeigt.

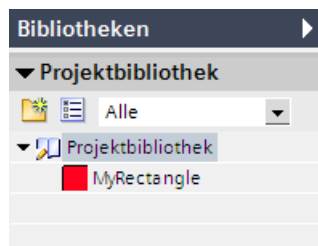
#### Vorgehen

1. Selektieren Sie das Objekt im Arbeitsbereich des Editors "Bilder".
2. Ziehen Sie das Objekt per Drag&Drop aus dem Arbeitsbereich in die gewünschte Bibliothek. Dem Mauszeiger verwandelt sich in ein Fadenkreuz mit angehängtem Objektsymbol.



#### Ergebnis

Das Objekt ist in der Bibliothek hinterlegt und kann beim Projektieren mehrfach verwendet werden.



### 10.9.2.7 Bibliotheksobjekt einfügen

#### Einleitung

Beim Einfügen eines Bibliotheksobjekts wird der Objektname standardmäßig aus der Bezeichnung des Objekttyps und einer fortlaufenden Nummer zusammengesetzt. Wenn das eingefügte Objekt bereits vorhanden ist, können Sie über ein Dialogfenster angeben, ob Sie das vorhandene Objekt ersetzen oder das neue unter einem anderen Namen einfügen wollen. Wenn das vorhandene Objekt nicht ersetzt werden soll, geben Sie einen neuen Namen an.

Bibliotheksobjekte, die vom Bediengerät nicht unterstützt werden, können nicht eingefügt werden.

---

#### Hinweis

Wenn Sie ein Bild mit verknüpfter Vorlage aus der Bibliothek einfügen, wird auch die Vorlage eingefügt. Eine eventuell vorhandene, passende Vorlage wird nicht genutzt.

---

#### Voraussetzung

- Die Task Card "Bibliotheken" ist geöffnet.
- Der Editor, in den Sie das Bibliotheksobjekt einfügen wollen, ist geöffnet.

#### Vorgehen

1. Selektieren Sie das Bibliotheksobjekt in der Bibliothek, das Sie einfügen wollen.
2. Ziehen Sie das Bibliotheksobjekt per Drag&Drop an die Stelle des Arbeitsbereichs, an der Sie das Objekt einfügen wollen.  
Das Bibliotheksobjekt wird eingefügt.
3. Selektieren Sie das Bibliotheksobjekt im Bild und passen Sie es an.

#### Ergebnis

Wenn das Objekt im Ordner "Kopiervorlagen" enthalten war, haben Sie eine unabhängige Kopie des Bibliotheksobjekts im Editor eingefügt.

Wenn das Objekt im Ordner "Typen" enthalten war, haben Sie eine Instanz des Bibliotheksobjekts im Editor eingefügt.

### 10.9.2.8 Globale Bibliothek anlegen


#### Einleitung

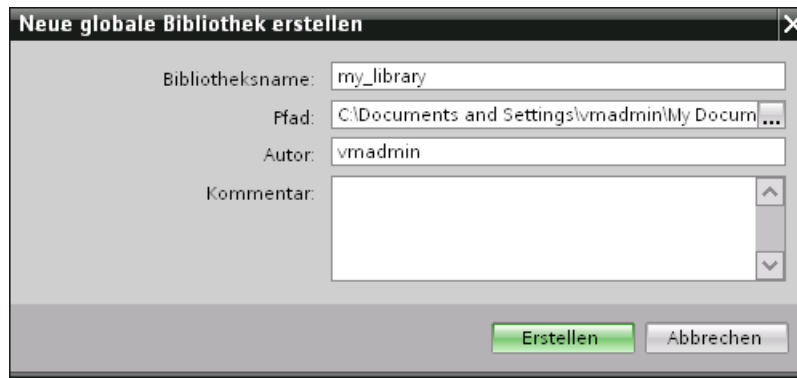
In Bibliotheken hinterlegen Sie konfigurierte Objekte, die Sie bei der Projektierung mehrfach verwenden wollen. Um Objekte in mehreren Projekten zu nutzen, legen Sie eine Globale Bibliothek an.

## Voraussetzung

- Das Projekt ist geöffnet.
- Die Task Card "Bibliotheken" ist geöffnet.

## Vorgehen

1. Klicken Sie in der Task Card "Bibliotheken > Globale Bibliotheken" auf das Symbol . Der Dialog "Neue globale Bibliothek erstellen" wird geöffnet.



2. Geben Sie einen Name ein.
3. Wählen Sie den Pfad, wo die neue Bibliothek gespeichert werden soll.
4. Klicken Sie auf "Erstellen".

## Ergebnis

Die neue Bibliothek wird in der Palette "Globale Bibliotheken" angezeigt. Die Globale Bibliothek enthält die Ordner "Typen" bzw. "Kopiervorlagen". Sie können Objekte in der Bibliothek hinterlegen.

### 10.9.2.9 Globale Bibliothek speichern

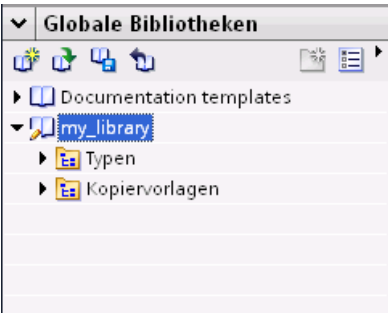
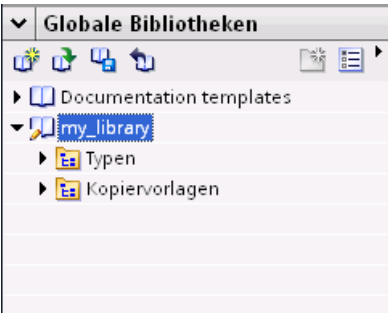
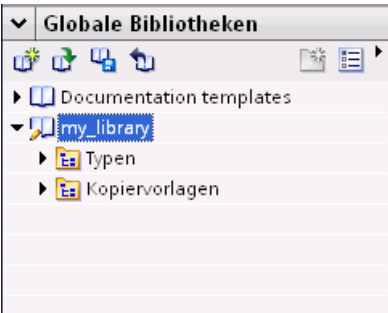
#### Einleitung

Eine Globale Bibliothek wird als separate Datei auf Ihrer Festplatte gespeichert. Die Datei enthält die Objekte der Globalen Bibliothek einschließlich der referenzierten Objekte. Z. B. wird die Referenz einer Variablen, die an ein E/A-Feld projiziert wurde, ebenfalls in der Bibliothek gespeichert.

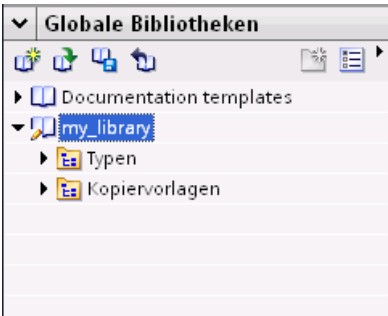
Wenn Sie Ihr Projekt oder WinCC schließen, ohne zu speichern, fordert WinCC Sie auf, die Globalen Bibliotheken zu speichern. Sie können auch während der Projektierung die Globale Bibliothek speichern, ohne dafür das gesamte Projekt zu speichern.




## Voraussetzung

-  Bibliothek ist geöffnet.
-  geöffnet.
- 

## Vorgehen

1.  othek, die Sie speichern möchten.

2. Klicken Sie in der Task Card "Bibliotheken" in der Palette "Globale Bibliothek" auf das Symbol .

Alternativ verwenden Sie im Kontextmenü "Bibliothek speichern".

Wenn Sie die Globale Bibliothek in einem anderen Ordner speichern wollen, wählen Sie im Kontextmenü "Speichern unter". Wählen Sie den Pfad, in dem Sie die neue Bibliothek speichern wollen, und geben Sie einen Dateinamen an.

## Ergebnis

Die Globalen Bibliotheken werden unter dem aktuellen oder dem neu vergebenen Dateinamen gespeichert.

### 10.9.2.10 Globale Bibliothek öffnen


## Einleitung

In WinCC sind Globale Bibliotheken als separate Dateien abgelegt. Sie können eine Globale Bibliothek in jedem Projekt verwenden.

## Voraussetzung

- Eine Globale Bibliothek ist gespeichert.
- Ein Projekt ist geöffnet.
- Die Task Card "Bibliotheken" ist geöffnet.

## Vorgehen

1. Klicken Sie in der Palette "Globale Bibliotheken" auf das Symbol . Der Dialog "Globale Bibliothek öffnen" wird geöffnet.
2. Wählen Sie den Pfad, in dem die Bibliothek gespeichert ist.
3. Klicken Sie auf "Öffnen".

---

### Hinweis

Wenn auf eine globale Bibliothek von mehreren Projekten aus zugegriffen werden soll, muss die globale Bibliothek schreibgeschützt geöffnet werden. Sobald eine globale Bibliothek nicht schreibgeschützt geöffnet ist, wird der Zugriff von den Projekten blockiert.

---

## Ergebnis

WinCC zeigt die geöffnete Globale Bibliothek in der Palette "Globale Bibliotheken" an.

## 10.9.3 Projektdaten importieren und exportieren

### 10.9.3.1 Import und Export von Projektdaten

## Einleitung

WinCC bietet Ihnen die Möglichkeit, Projektdaten zwischen den Projekten auszutauschen oder in externe Anwendungen zu kopieren.

## Export und Import zwischen Projekten

Sie können folgende Projektdaten aus einem Projekt exportieren und in ein anderes Projekt importieren.

- Rezepturdatensätze
- Meldungen
- Variablen
- Textlisten
- Projekttexte

Durch den Export und Import reduzieren Sie den Arbeitsaufwand. Anstatt neue Datensätze anzulegen, verwenden Sie Daten, die Sie bereits in früheren Projekten erstellt haben.

## Bearbeitung der Exportdatei

Je nach Editor stehen Ihnen folgende Dateiformate für den Export bzw. Import zur Verfügung:

- \*.xlsx für Meldungen, Variablen, Projekttexte und Textlisten
- \*.csv für Rezepturdatensätze

Zur Bearbeitung der Importdatei verwenden Sie z. B. Excel.

## Dateiformat XLSX

Das Format XLSX ist ein Dateiformat für Excel-Tabellen auf der Grundlage des Open-XML-Formats. XLSX-Dateien sind für Microsoft Excel 2007 optimiert.

In der XLSX-Datei können Sie die Spalten beliebig sortieren.

## Dateiformat CSV

CSV steht für Comma Separated Value. In diesem Format werden die Tabellenspalten, die den Namen und den Wert des Eintrags enthalten, durch Semikola getrennt. Jede Tabellenzeile wird durch einen Zeilenumbruch abgeschlossen. Zur Bearbeitung können Sie die CSV-Datei auch in Excel öffnen.

## Import von Projektdaten

Bei einem Import der Projektdaten werden die Objekte im Projekt erzeugt.

Beim Import von Dateien wird die Syntax der Importdatei geprüft. Die Korrektheit der importierten Werte oder Abhängigkeiten zwischen den importierten Werten werden nicht geprüft.

Wenn die importierten Daten Fehler aufweisen, so werden diese beim Übersetzen des Projekts gemeldet.

## Kopieren im Excel-Format

In allen Tabelleneditoren können Sie die Inhalte im Excel-Format in den Zwischenspeicher Ihres PCs kopieren. Anschließend fügen Sie die Projektdaten im Excel-Format direkt in eine beliebige Anwendung außerhalb des TIA-Portals ein. Sie nutzen dafür den entsprechenden Befehl im Kontextmenü des Arbeitsbereichs:

- Wenn Sie den Befehl im Kontextmenü des Zeilenkopfs wählen, wird die ganze Zeile in den Zwischenspeicher kopiert.
- Wenn Sie den Befehl im Kontextmenü einer Zelle wählen, wird nur der Zelleninhalt in den Zwischenspeicher kopiert.
- Wenn Sie mehrere Zeilen markieren und den Befehl wählen, werden immer alle markierten Daten in den Zwischenspeicher kopiert.

Dieser Datenaustausch ist nur als Export möglich.

### Siehe auch

Meldungen exportieren (Seite 3858)

Variablen exportieren (Seite 3865)

Textlisten exportieren (Seite 3871)

## 10.9.3.2 Rezepturen importieren und exportieren

### Rezepturen exportieren


#### Einleitung

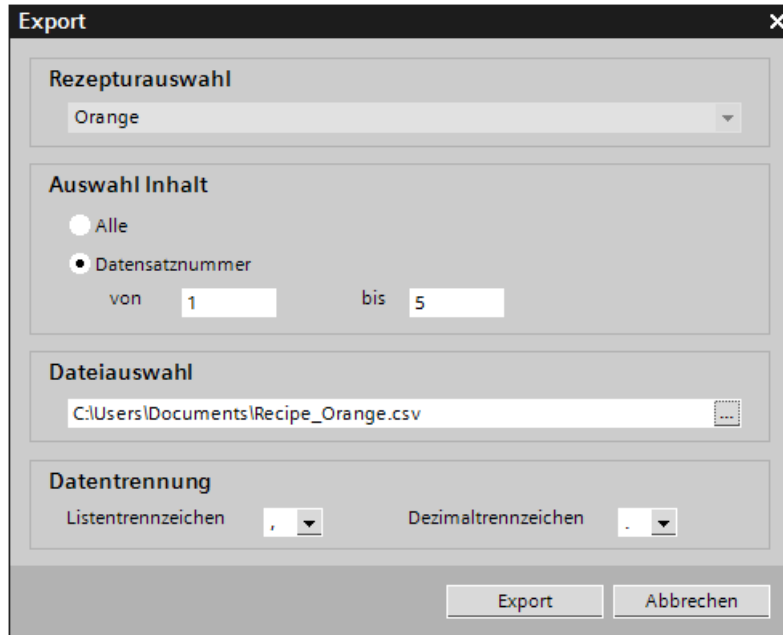
Für den Export von Datensätzen aus Rezepturen steht Ihnen in WinCC eine Exportfunktion zur Verfügung.

#### Voraussetzungen

- Das WinCC Projekt für den Export ist geöffnet.
- In dem Projekt sind Rezepturen angelegt.
- Der Editor "Rezepturen" ist geöffnet.

## Rezepturen exportieren

1. Markieren Sie im Editor "Rezepturen" die Rezeptur, deren Datensätze Sie exportieren wollen.
2. Klicken Sie auf das Symbol .  
Der Dialog "Export" wird geöffnet.



Unter "Rezepturauswahl" wird die markierte Rezeptur angezeigt.

3. Legen Sie unter "Auswahl Inhalt" fest, ob Sie alle oder nur ausgewählte Datensätze exportieren.
4. Legen Sie unter "Dateiauswahl" fest, in welche Datei die Rezepturdaten gespeichert werden.
5. Legen Sie unter "Datentrennung" Listentrennzeichen und Dezimaltrennzeichen fest.
6. Klicken Sie auf "Export".  
Der Export wird gestartet.

## Ergebnis

Die exportierten Daten wurden in eine CSV-Datei geschrieben. Die CSV-Datei wird im angegebenen Verzeichnis abgelegt.

## Siehe auch

Meldungen exportieren (Seite 3858)

Import und Export von Projektdaten (Seite 3852)

Variablen exportieren (Seite 3865)

Textlisten exportieren (Seite 3871)

## Rezepturen importieren

### Einleitung


Rezepturen werden über ihre Namen identifiziert. Die Rezepturnamen müssen daher eindeutig sein. Zur Kontrolle der Datenstruktur in der Importdatei öffnen Sie die Datei in einem einfachen Texteditor.

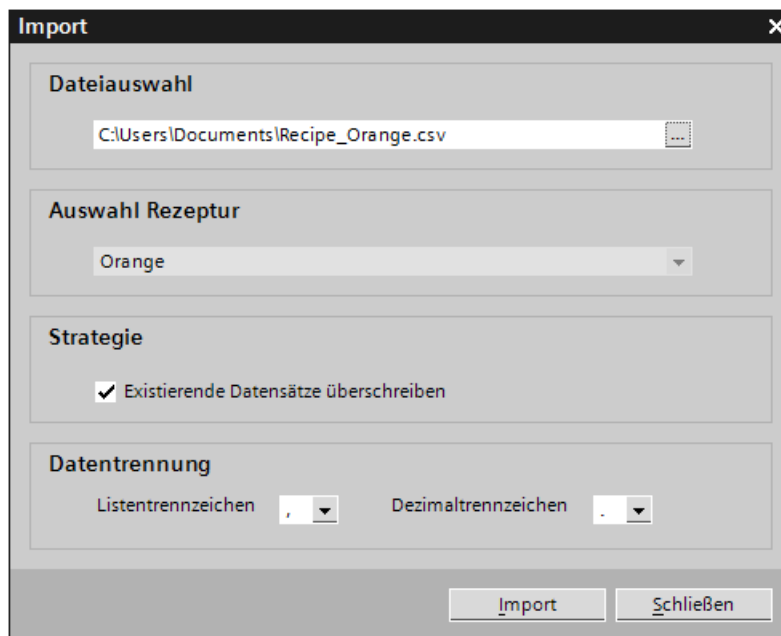
Beim Import wählen Sie, ob bereits bestehende Datensätze mit gleichen Namen überschrieben werden.

### Voraussetzungen

- Eine CSV-Datei mit mindestens einer Rezeptur ist angelegt.
- Das WinCC Projekt für den Import ist geöffnet.
- Der Editor "Rezepturen" mit mindestens einer Rezeptur ist geöffnet.

### Rezeptur importieren

1. Markieren Sie im Editor "Rezepturen" die Rezeptur, in die Sie Datensätze importieren wollen.
2. Klicken Sie auf das Symbol . Der Dialog "Import" wird geöffnet.



Unter "Auswahl Rezeptur" wird die markierte Rezeptur angezeigt.

3. Wählen Sie unter "Dateiauswahl" die Datei, die Sie importieren wollen.
4. Legen Sie unter "Strategie" fest, ob Sie vorhandene gleichnamige Datensätze überschreiben wollen.

5. Wählen Sie unter "Datentrennung" das Listentrennzeichen und das Dezimaltrennzeichen, das in der CSV-Datei verwendet wurde.
6. Klicken Sie auf "Import".  
Der Import wird gestartet.

## Ergebnis

Die Datensätze werden in der markierten Rezeptur angelegt. Abhängig von der Einstellung unter "Strategie" werden vorhandene Datensätze mit gleichnamigen Datensätzen aus der CSV-Datei überschrieben.

Wenn Sie die Option "Existierende Datensätze überschreiben" deaktivieren, werden gleichnamigen Datensätzen aus der CSV-Datei zusätzlich importiert.

## Format der Daten von Rezepturen

### Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei für den Import von Rezepturen. Die Datei mit den Daten der Rezepturen muss im Dateiformat "\*.csv" vorliegen. :

### Struktur der Rezepturdaten

Der Aufbau der Importdatei ist zwingend vorgeschrieben. Das folgende Beispiel zeigt die Struktur für eine Rezeptur, die zwei Rezepturelemente mit je zwei Datensätzen enthält:

```
List separator=<Listentrennzeichen>Decimal
symbol=<Dezimaltrennzeichen><Listentrennzeichen><Zeilenumbruch>
<Name der
Rezeptur><Listentrennzeichen><Listentrennzeichen><Zeilenumbruch>
LANGID_<ID der Sprache><Listentrennzeichen>
<Anzeigename Rezepturelement 1><Listentrennzeichen>
<Anzeigename Rezepturelement 2><Zeilenumbruch>
<Nummer Rezeptur><Listentrennzeichen>
<Rezepturdatensatznummer 1><Listentrennzeichen>
<Rezepturdatensatznummer 2><Zeilenumbruch>
<Variable Rezepturelement 1><Listentrennzeichen>
<Rezepturdatensatz 1 Wert 1><Listentrennzeichen>
<Rezepturdatensatz 2 Wert 1><Zeilenumbruch>
<Variable Rezepturelement 2><Listentrennzeichen>
<Rezepturdatensatz 1 Wert 2><Listentrennzeichen>
<Rezepturdatensatz 2 Wert 2><Zeilenumbruch>
```

### ID der Sprache

Verwenden Sie die "Windows-Sprachen-ID" in dezimaler Schreibweise, z. B. "1033" für Englisch. Weitere Informationen finden Sie in der Dokumentation zum Betriebssystem Windows.

## Siehe auch

Rezepturen exportieren (Seite 3854)

### 10.9.3.3 Meldungen importieren und exportieren

#### Meldungen exportieren


##### Einleitung

WinCC stellt eine Exportfunktion für Meldungen zur Verfügung.

##### Voraussetzungen

- Das WinCC Projekt für den Export ist geöffnet.
- In dem Projekt sind Meldungen angelegt.
- Der Editor "HMI-Meldungen" ist geöffnet.

#### Meldungen exportieren

1. Klicken Sie in "Bitmeldungen" oder "Analogmeldungen" auf die Schaltfläche . Der Dialog "Export" wird geöffnet.



2. Klicken Sie auf die Schaltfläche "..." und legen Sie fest, in welche Datei die Daten gespeichert werden.
3. Legen Sie fest, ob Sie "Bitmeldungen" oder "Analogmeldungen" exportieren wollen.
4. Klicken Sie auf "Export". Der Export wird gestartet.

##### Ergebnis

Die exportierten Daten wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.



## Siehe auch

- Meldungen importieren (Seite 3859)
- Format der Daten von Analogmeldungen (Seite 3860)
- Format der Daten von Bitmeldungen (Seite 3863)
- Rezepturen exportieren (Seite 3854)
- Variablen exportieren (Seite 3865)
- Import und Export von Projektdaten (Seite 3852)

## Meldungen importieren

### Einleitung


Meldungen werden über ihre Meldenummer identifiziert. Die Meldenummern müssen innerhalb der Meldetypen Analogmeldung und Bitmeldung eindeutig sein. Meldungen mit identischen Meldenummern werden daher überschrieben. Eine Meldung mit nicht existenter Meldenummer wird neu angelegt.

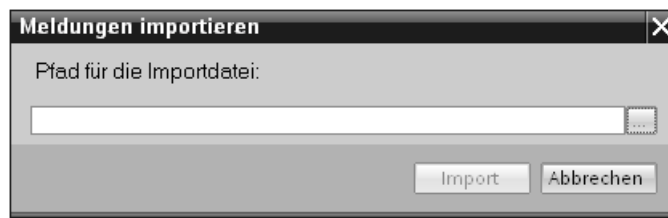
Wenn eine xlsx-Datei leere Listeneinträge für bereits bestehende Meldungen enthält, werden diese beim Import ignoriert. Die Einträge der bestehenden Meldung bleiben erhalten und werden nicht durch leere ersetzt.

### Voraussetzungen

- Eine xlsx-Datei mit Meldungen ist angelegt.
- Die xlsx-Datei ist den Anforderungen entsprechend aufgebaut.
- Das WinCC Projekt für den Import ist geöffnet.
- Der Editor "HMI-Meldungen" ist geöffnet.

## Meldungen importieren

1. Klicken Sie in "Bitmeldungen" oder "Analogmeldungen" auf die Schaltfläche . Der Dialog "Import" wird geöffnet.



2. Klicken Sie auf die Schaltfläche "...".
3. Wählen Sie die Datei, die Sie importieren wollen.
3. Klicken Sie auf die Schaltfläche "Importieren". Der Import wird gestartet. Ein Laufbalken zeigt den Fortschritt des Importvorgangs an.

## Ergebnis

Mit den importierten Daten wurden die entsprechenden Meldungen einschließlich der Meldetexte in WinCC angelegt. Im Ausgabefenster werden Meldungen zum Importvorgang ausgegeben. Im Quellverzeichnis der Importdateien wird eine Log-Datei abgelegt. Die Log-Datei erhält den gleichen Namen wie die jeweilige Importdatei und hat die Dateierweiterung "\*.xml".

Achten Sie beim Import der Daten darauf, ob Verknüpfungen zu Objekten vorhanden, sind z. B. dynamische Parameter wie Variablen.

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, legen Sie ein Objekt mit dem entsprechenden Namen an oder legen Sie eine neue Verknüpfung an.

---

### Hinweis

Beim Import von xlsx-Dateien wird die Syntax der Importdatei geprüft. Die Bedeutung der Eigenschaften oder Abhängigkeiten zwischen den Eigenschaften werden jedoch nicht geprüft. Es ist möglich, eine Triggervariable vom falschen Typ z. B. String einer Meldung zuzuweisen. Beim Übersetzen wird ein Fehler gemeldet.

---

## Siehe auch

Meldungen exportieren (Seite 3858)

## Format der Daten von Analogmeldungen

### Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei für den Import von Analogmeldungen. Die Datei mit den Daten der Analogmeldungen muss im Dateiformat "\*.xlsx" vorliegen.

### Struktur der Meldungsdaten

Die Importdatei besteht in Microsoft Excel aus Tabellenblättern:

- Analog alarms (Analogmeldungen)
- Limits (Grenzen)

Jede Meldung erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Analogmeldungen muss folgendermaßen aufgebaut sein:

**Beispiel für das Tabellenblatt "Analog alarms"**

|    | A  | B               | C  | D                               | E            |
|----|----|-----------------|--|---------------------------------|--------------|
| 1  | ID | Name            | Event text [en-US], Alarm text   | FieldInfo [Alarm text]          | Class        |
| 2  | 1  | Analog_alarm_1  | AA1 Error-AC with maximum text length: abcdefghijklmnopqrstuvwxyz          |                                 | Errors       |
| 3  | 2  | Analog_alarm_2  | AA2 Warning-AC <b>this text should be bold</b>                             |                                 | Warnings     |
| 4  | 3  | Analog_alarm_3  | AA3 SDm-AC <i>this text should be italic</i>                               |                                 | SDm          |
| 5  | 4  | Analog_alarm_4  | AA4 SDo-AC <u>this text should be underlined</u>                           |                                 | SDo          |
| 6  | 5  | Analog_alarm_5  | AA5 SystemAcknowledgement-AC <blink>this text should be flashing<          |                                 | System_Ackn  |
| 7  | 25 | Analog_alarm_25 | Internal AA23 switchDT: Deadband mode in case of violation - value HL      |                                 | AA2T-Interna |
| 8  | 26 | Analog_alarm_26 | Internal AA23 switchDT: Deadband mode in case of violation - percent       |                                 | AA2T-Interna |
| 9  | 31 | Analog_alarm_31 | Internal AA23 switchDT: Low limit violation static                         |                                 | AA2T-Interna |
| 10 | 32 | Analog_alarm_32 | Internal AA23 switchDT: High limit violation static                        |                                 | AA2T-Interna |
| 11 | 33 | Analog_alarm_33 | Internal AA23 switchDT: Low limit violation dynamic                        |                                 | AA2T-Interna |
| 12 | 34 | Analog_alarm_34 | Internal AA23 switchDT: High limit violation dynamic                       |                                 | AA2T-Interna |
| 13 | 35 | Analog_alarm_35 | Internal AA23 switchDT: delay 3 seconds High limit violation               |                                 | AA2T-Interna |
| 14 | 42 | Analog_alarm_40 | Internal AA23 switchDT: delay 3 seconds Low limit violation LLV            |                                 | AA2T-Interna |
| 15 | 23 | Analog_alarm_41 | AA23 DT in event text: <field ref="0" /> Bool, <field ref="1" /> Byte, <fi | <ref id = 0; type = AlarmTag; T | ACAT         |
| 16 | 24 | Analog_alarm_42 | AA24 DT in event text: <field ref="0" /> Timer, <field ref="1" /> Counte   | <ref id = 0; type = AlarmTag; T | ACAT         |

Tabelle 10-14 Bedeutung der Einträge

| Listeneintrag                  | Bedeutung  |
|--------------------------------|--|
| ID                             | Die Meldenummer dient zum Referenzieren einer Meldung. Die Meldenummer ist eindeutig. Meldungen mit identischen Meldenummern werden daher beim Import überschrieben. Eine Meldung mit nicht existenter Meldenummer wird neu angelegt.  |
| Name                           | Name der Analogmeldung   |
| Event text [de-DE], Alarm text | Zeigt den Meldetext der Meldung. Die Bezeichnung des Feldes enthält eine Sprachkennung. Meldetexte müssen für den Import mit einer Sprachkennung versehen sein<br><br>Wenn ein dynamischer Parameter im Meldetext vorhanden ist, wird dem Text ein Ausdruck mit einer Referenz-ID hinzugefügt. Beispiel: text <field ref="0" />. Über die ID wird der dynamische Parameter einem Meldetext zugeordnet. |
| FeldInfo                       | Gibt an, ob dynamische Parameter im Meldetext enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt.<br><br>Beispiel für dynamische Parameter:<br>Variable: <ref id = 0; type = AlarmTag; Tag = Tag1; DisplayType = Decimal; Length = 5;><br>Textliste: <ref id = 1; type = CommonTextList; TextList = Textlist1; Tag = tag 2; Length = 5;>  |
| Class                          | Mit der Zugehörigkeit zu einer Meldeklasse wird festgelegt, ob die Meldung quittiert werden muss oder nicht. Ferner kann damit die Erscheinung der Meldung bei der Anzeige am Bediengerät gesteuert werden. Die Meldeklasse bestimmt auch, ob und wo die zugehörige Meldung archiviert wird.   |
| Group                          | Gibt die Zugehörigkeit zu einer Meldegruppe an. Durch die Zugehörigkeit kann die Meldung gemeinsam mit anderen Meldungen derselben Gruppe durch nur eine Bedienung quittiert werden.   |
| Trigger tag                    | Gibt die Variable an, die auf Grenzwert-Verletzung überwacht wird.   |

| Listeneintrag                | Bedeutung   |
|------------------------------|---|
| Delay time value             | Gibt die Verzögerungszeit an. Die Meldung wird erst dann ausgelöst, wenn die Grenzwert-Verletzung die angegebene Verzögerungszeit andauert.   |
| Delay time unit              | Gibt die Zeiteinheit für die Verzögerung an.  |
| Report                       | Aktiviert die Protokollierung der jeweiligen Meldung auf einem Drucker.<br>True oder "1" = Protokollierung aktiviert<br>False oder "0" = Protokollierung deaktiviert.<br>Die Protokollierung muss zusätzlich im Projekt global aktiviert werden.  |
| Info text [de-DE], Info text | Der Tooltip ist eine optionale Eigenschaft einer Meldung. Der Tooltip kann ergänzende Informationen zur Meldung enthalten. Der Tooltip wird in einem separaten Fenster am Bediengerät angezeigt, wenn der Bediener die <HELP>-Taste drückt.<br>Die Bezeichnung des Feldes enthält eine Sprachkennung. |

Beispiel für das Tabellenblatt "Limits"

|    | A        | B          | C           | D           | E           | F             | G                   |
|----|----------|------------|-------------|-------------|-------------|---------------|---------------------|
| 1  | Alarm ID | Limit type | Limit value | Limit mode  | Deadband mo | Deadband valu | Deadband in percent |
| 2  | 1        | Constant   | 0           | Upper limit | Off         | 0             | False               |
| 3  | 2        | Constant   | 1           | Upper limit | Off         | 0             | False               |
| 4  | 3        | Constant   | 2           | Upper limit | Off         | 0             | False               |
| 5  | 4        | Constant   | 3           | Upper limit | Off         | 0             | False               |
| 6  | 5        | Constant   | 4           | Upper limit | Off         | 0             | False               |
| 7  | 25       | Constant   | 50          | Upper limit | On both     | 5             | False               |
| 8  | 26       | Constant   | 50          | Upper limit | On both     | 10            | True                |
| 9  | 31       | Constant   | 50          | Lower limit | Off         | 0             | False               |
| 10 | 32       | Constant   | 50          | Upper limit | Off         | 0             | False               |
| 11 | 33       | Tag        | AASDTdyn    | Lower limit | Off         | 0             | False               |
| 12 | 34       | Tag        | AASDT1dyn   | Upper limit | Off         | 0             | False               |
| 13 | 35       | Constant   | 50          | Upper limit | Off         | 0             | False               |
| 14 | 36       | Constant   | 50          | Lower limit | On both     | 5             | False               |

Tabelle 10-15 Bedeutung der Einträge

| Listeneintrag | Bedeutung  |
|---------------|--|
| Alarm ID      | Meldenummer<br>Die Meldenummer dient zum Referenzieren einer Meldung. Die Meldenummer ist eindeutig. Meldungen mit identischen Meldenummern werden daher beim Import überschrieben. Eine Meldung mit nicht existenter Meldenummer wird neu angelegt. |
| Limit mode    | Triggermodus<br>Gibt die Methode an, mit welcher der Grenzwert überwacht wird.   |
| Limit type    | Gibt die Grenze an, die überwacht wird. Als Grenzwert ist sowohl eine Variable als auch eine Konstante möglich.  |

| Listeneintrag       | Bedeutung  |
|---------------------|--|
| Limit value         | Grenzwert<br>Gibt die Variable bzw. Konstante an, die auf Grenzwert-Verletzung überwacht wird.   |
| Deadband mode       | Hysteresemodus<br>Gibt an, ob und in welchen Fällen eine Hysterese angewendet wird.<br>Bei "Gehen"<br>Bei "Kommen"<br>Bei "Kommen" und "Gehen"                 |
| Deadband in percent | 0 = Der bei "Hysterese" angegebene Wert wird als Absolutwert betrachtet.<br>1 = Der bei "Hysterese" angegebene Wert wird prozentual auf den Grenzwert bezogen. |
| Deadband mode       | Hysterese<br>Gibt eine Konstante als Wert der Hysterese an.  |

**Hinweis****"No Value" in der Tabelle**

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Meldung.

**Siehe auch**

Meldungen exportieren (Seite 3858)

**Format der Daten von Bitmeldungen****Einleitung**

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei für den Import von Bitmeldungen. Die Datei mit den Daten der Bitmeldungen muss im Dateiformat "\*.xlsx" vorliegen.

**Struktur der Meldungsdaten**

Die Importdatei besteht in Microsoft Excel aus dem Tabellenblättern "Discrete alarms" (Bitmeldungen). Jede Meldung erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Bitmeldungen muss folgendermaßen aufgebaut sein:

Beispiel für das Tabellenblatt "Discrete alarms"

|    | A  | B                 | C  | D                                      | E           | F       |
|----|----|-------------------|--|--|-------------|---------|
| 1  | ID | Name              | Event text [en-US], Alarm text   | FieldInfo [Alarm text]                 | Class       | Trigger |
| 2  | 1  | Discrete_alarm_1  | DA1 Error-AC with maximum text length: abcdefghijklmnopqrstuvwxyz äöü\na           |  | Errors      | HMI_AC  |
| 3  | 2  | Discrete_alarm_2  | DA2 Warning-AC <b>this text should be bold</b>                                     |  | Warnings    | HMI_AC  |
| 4  | 3  | Discrete_alarm_3  | DA3 SDm-AC <i>this text should be italic</i>                                       |  | SDm         | HMI_AC  |
| 5  | 4  | Discrete_alarm_4  | DA4 SDo-AC <u>this text should be underlined</u>                                   |  | SDo         | HMI_AC  |
| 6  | 5  | Discrete_alarm_5  | DA5 SystemAcknowledgement-AC <blink>this text should be flashing</blink>           |  | System_Ackn | HMI_AC  |
| 7  | 6  | Discrete_alarm_6  | DA6 SystemNoAcknowledgement-AC mixed test: <b>Bold,</b> <i>Italic,</i> <           |  | System_No_A | HMI_AC  |
| 8  | 7  | Discrete_alarm_7  | DA7 DT in event text: <field ref="0" /> Integer, <field ref="1" /> Real, <field re | <ref id = 0; type = AlarmTag; Tag = Pl | ACAT        | HMI_Tri |
| 9  | 8  | Discrete_alarm_8  | DA8 DT in event text: <field ref="0" /> S5Time, <field ref="1" /> Timer, <field r  | <ref id = 0; type = AlarmTag; Tag = Pl | ACAT        | HMI_Tri |
| 10 | 11 | Discrete_alarm_9  | DA11 DT in event text: <field ref="0" /> Int, <field ref="1" /> Real, <field ref=" | <ref id = 0; type = AlarmTag; Tag = In | ACAT        | HMI_Tri |
| 11 | 12 | Discrete_alarm_10 | DA12 DT in event text: <field ref="0" /> UDInt, <field ref="1" /> UInt,            | <ref id = 0; type = AlarmTag; Tag = U  | ACAT        | HMI_Tri |
| 12 | 13 | Discrete_alarm_11 | DA13 Textformat: Integer: <field ref="0" /> decimal, <field ref="1" /> binary, <   | <ref id = 0; type = AlarmTag; Tag = H  | ACAT        | HMI_TF  |

Tabelle 10-16 Bedeutung der Einträge

| Listeneintrag                  | Bedeutung  |
|--------------------------------|--|
| ID                             | Die Meldenummer dient zum Referenzieren einer Meldung. Die Meldenummer ist eindeutig. Meldungen mit identischen Meldenummern werden daher beim Import überschrieben. Eine Meldung mit nicht existenter Meldenummer wird neu angelegt.  |
| Name                           | Name der Analogmeldung   |
| Event text [de-DE], Alarm text | Zeigt den Meldetext der Meldung. Die Bezeichnung des Feldes enthält eine Sprachkennung. Meldetexte müssen für den Import mit einer Sprachkennung versehen sein.<br><br>Wenn ein dynamischer Parameter im Meldetext vorhanden ist, wird dem Text ein Ausdruck mit einer Refrenz-ID hinzugefügt. Beispiel: text <field ref="0" />. Über die ID wird der dynamische Parameter einem Meldetext zugeordnet. |
| FeldInfo                       | Gibt an, ob dynamische Parameter im Meldetext enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt.<br><br>Beispiel für dynamische Parameter:<br>Variable: <ref id = 0; type = AlarmTag; Tag = Tag1; DisplayType = Decimal; Length = 5;><br>Textliste: <ref id = 1; type = CommonTextList; TextList = Textlist1; Tag = tag 2; Length = 5;>  |
| Class                          | Mit der Zugehörigkeit zu einer Meldeklasse wird festgelegt, ob die Meldung quittiert werden muss oder nicht. Ferner kann damit die Erscheinung der Meldung bei der Anzeige am Bediengerät gesteuert werden. Die Meldeklasse bestimmt auch, ob und wo die zugehörige Meldung archiviert wird.   |
| Group                          | Gibt die Zugehörigkeit zu einer Meldegruppe an. Durch die Zugehörigkeit kann die Meldung gemeinsam mit anderen Meldungen derselben Gruppe durch nur eine Bedienung quittiert werden.   |
| Trigger tag                    | Gibt die Variable an, die das Bit enthält durch das die Meldung ausgelöst wird.  |
| Trigger bit                    | Gibt die Nummer des Bits an, durch das die Meldung ausgelöst wird.   |
| Acknowledge tag                | Gibt die Variable an, die das Bit enthält, das beim Quittieren durch den Bediener gesetzt wird. Nur verfügbar, wenn die gewählte Meldeklasse ein Quittieren der Meldung erfordert.   |
| Acknowledgment bit             | Gibt die Nummer des Bits an, das gesetzt wird, wenn der Bediener die Meldung quittiert.  |

| Listeneintrag                | Bedeutung   |
|------------------------------|---|
| PLC acknowledgement tag      | Gibt die Variable an, die das Bit enthält, durch das die Meldung vom Steuerungsprogramm quittiert wird. Nur verfügbar, wenn die gewählte Meldeklasse ein Quittieren der Meldung erfordert.  |
| PLC acknowledgment bit       | Gibt die Nummer des Bits an, durch das die Meldung vom Steuerungsprogramm quittiert wird.   |
| Delay time value             | Gibt die Verzögerungszeit an. Die Meldung wird erst dann ausgelöst, wenn die Grenzwert-Verletzung die angegebene Verzögerungszeit andauert.   |
| Delay time unit              | Gibt die Zeiteinheit für die Verzögerung an.  |
| Report                       | Aktiviert die Protokollierung der jeweiligen Meldung auf einem Drucker.<br>True oder "1" = Protokollierung aktiviert<br>False oder "0" = Protokollierung deaktiviert.<br>Die Protokollierung muss zusätzlich im Projekt global aktiviert werden.  |
| Info text [de-DE], Info text | Der Tooltip ist eine optionale Eigenschaft einer Meldung. Der Tooltip kann ergänzende Informationen zur Meldung enthalten. Der Tooltip wird in einem separaten Fenster am Bediengerät angezeigt, wenn der Bediener die <HELP>-Taste drückt.<br>Die Bezeichnung des Feldes enthält eine Sprachkennung. |

**Hinweis****"No Value" in der Tabelle**

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Meldung.

**Siehe auch**

Meldungen exportieren (Seite 3858)


**10.9.3.4 Variablen importieren und exportieren****Variablen exportieren****Einleitung**

WinCC stellt eine Exportfunktion für Variablen zur Verfügung.

**Voraussetzungen**

- Das WinCC Projekt für den Export ist geöffnet.
- In dem Projekt sind Variablen angelegt.
- Der Editor "HMI-Variablen" ist geöffnet.

## Variablen exportieren

1. Klicken Sie in der Registerkarte "HMI-Variablen" auf die Schaltfläche . Der Dialog "Export" wird geöffnet.



2. Klicken Sie auf die Schaltfläche "..." und legen Sie fest, in welche Datei die Daten gespeichert werden.
3. Klicken Sie auf "Export". Der Export wird gestartet.

## Ergebnis

Die exportierten Daten wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

## Siehe auch

- Variablen importieren (Seite 3866)
- Meldungen exportieren (Seite 3858)
- Import und Export von Projektdaten (Seite 3852)
- Textlisten exportieren (Seite 3871)
- Rezepturen exportieren (Seite 3854)
- Format der Variablendaten (Seite 3868)

## Variablen importieren

### Einleitung


Variablen werden über den Namen der Variablen identifiziert. Wenn der Name der Variablen im Projekt bereits besteht, so wird die bestehende Variablen mit den Daten aus der xlsx-Datei überschrieben. Wenn die Variable noch nicht besteht, wird eine neue Variable angelegt.

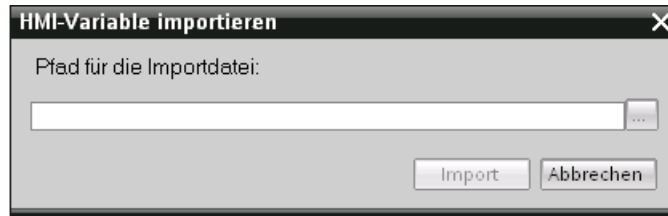
### Voraussetzungen

- Eine xlsx-Datei mit Variablen ist angelegt.
- Die xlsx-Datei ist den Anforderungen entsprechend aufgebaut.
- Das WinCC Projekt für den Import ist geöffnet.



## Variablen importieren

1. Klicken Sie in der Projektnavigation auf "HMI-Variablen".
2. Doppelklicken Sie "Alle Variablen anzeigen". Der Editor "HMI-Variablen" wird geöffnet.
3. Klicken Sie auf die Schaltfläche . Der Dialog "Import" wird geöffnet.



4. Klicken Sie auf die Schaltfläche "..." und wählen Sie die Datei, die Sie importieren wollen.
5. Klicken Sie auf die Schaltfläche "Importieren". Der Import wird gestartet.

## Ergebnis

Die entsprechenden Variablen wurden in WinCC angelegt. Im Ausgabefenster werden Meldungen zum Importvorgang ausgegeben. Im Quellverzeichnis der Importdateien wird eine Log-Datei abgelegt. Die Log-Datei erhält den gleichen Namen wie die jeweilige Importdatei und hat die Dateierweiterung "\*.xml".

Achten Sie beim Import der Daten darauf, ob Verknüpfungen zu Objekten vorhanden, sind z. B. dynamische Parameter wie Variablen.

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, legen Sie ein Objekt mit dem entsprechenden Namen an oder legen Sie eine neue Verknüpfung an.

---

### Hinweis

Beim Import von xlsx-Dateien wird die Syntax der Importdatei geprüft. Die Bedeutung der Eigenschaften oder Abhängigkeiten zwischen den Eigenschaften werden jedoch nicht geprüft. Es ist möglich, eine Triggervariable vom falschen Typ z. B. String einer Variable zuzuweisen. Beim Übersetzen wird ein Fehler gemeldet.

---

## Siehe auch

Variablen exportieren (Seite 3865)

## Format der Variablendaten

### Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei mit den Variablendaten für den Variablenimport. Die Datei mit den Variablendaten muss im Dateiformat "\*.xlsx" vorliegen.

### Struktur der Variablendaten

Die Importdatei besteht in Microsoft Excel aus Tabellenblättern:

- HMI Tags (HMI-Variablen)
- Multiplexing (Multiplex Variablen)

Jede Variable erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Variablendaten muss folgendermaßen aufgebaut sein:

### Beispiel für das Tabellenblatt "HMI Tags"

|    | A                           | B                 | C                | D                | E             | F      | G              | H          |
|----|-----------------------------|-------------------|------------------|------------------|---------------|--------|----------------|------------|
| 1  | Name                        | Path              | Connection       | PLC tag          | DataType      | Length | Address        | Access Me  |
| 2  | HMI_Int                     | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 3  | Mux_Tag_1                   | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 4  | Mux_Tag_2                   | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 5  | Mux_11                      | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 6  | Mux_21                      | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 7  | Mux_13                      | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 8  | Mux_12                      | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 9  | Mux_23                      | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 10 | Mux_22                      | Internal Tags     | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 11 | Mux_Tag_1_Index             | Internal Tags     | <No Value>       | <No Value>       | UInt          | 2      | <No Value>     | <No Value> |
| 12 | Mux_Tag_12                  | Internal Tags     | <No Value>       | <No Value>       | USInt         | 1      | <No Value>     | <No Value> |
| 13 | Mux_Tag_11                  | Internal Tags     | <No Value>       | <No Value>       | USInt         | 1      | <No Value>     | <No Value> |
| 14 | Mux_Tag_13                  | Internal Tags     | <No Value>       | <No Value>       | USInt         | 1      | <No Value>     | <No Value> |
| 15 | HMI_UDInt                   | Internal Tags     | <No Value>       | <No Value>       | UDInt         | 4      | <No Value>     | <No Value> |
| 16 | Gauge_Process               | Default tag table | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 17 | Button_Tag_4                | Default tag table | <No Value>       | <No Value>       | Int           | 2      | <No Value>     | <No Value> |
| 18 | HMI_USInt                   | Internal Tags     | <No Value>       | <No Value>       | USInt         | 1      | <No Value>     | <No Value> |
| 19 | Data_block_2_PLC_DateTime_2 | Default tag table | HMI_connection_1 | Data_block_2.PLC | Date_And_Time | 8      | %DB28.DBX598.C | <absolute  |

Tabelle 10-17 Bedeutung der Einträge

| Listeneintrag | Bedeutung   |
|---------------|---|
| Name          | Gibt den projektierten Namen einer HMI-Variable an.   |
| Path          | Gibt an, in welchem Ordnen in der Projektnavigation die Variable abgelegt wird. Die Ordnerstruktur wird durch "\" dargestellt: "Ordnername1\Ordnername2\Variablenname". |
| PLC Tag       | Gibt an, ob die Variable mit einer PLC-Variable verbunden ist.  |

| Listeneintrag        | Bedeutung   |
|----------------------|---|
| Connection           | Gibt den Namen der Verbindung zur PLC an.   |
| Data type            | Gibt den Datentyp der Variable an. Die erlaubten Datentypen sind abhängig vom verwendeten Kommunikationstreiber. Weitere Informationen zu den erlaubten Datentypen der verschiedenen Kommunikationstreiber finden Sie in der Dokumentation im Kapitel "Kommunikation".  |
| Length               | Gibt die Länge der Variable an. Der Eintrag ist nur für Datentypen mit dynamischer Länge hilfreich z.B. String. Bei anderen Datentypen bleibt der Eintrag leer.   |
| Address              | Gibt die Variablenadresse in der PLC an. Die Variablenadresse muss genau so geschrieben sein, wie sie in WinCC dargestellt wird, z. B. "%DB1.DBW0". Für interne Variablen bleibt die Variablenadresse leer.   |
| Multiplexing         | Gibt an, ob Multiplexing verwendet wird.  |
| Index tag            | Zeigt den Namen der Indexvariable für das Multiplexen an.<br>In Runtime wird zunächst der Wert der Indexvariable ausgelesen. Dann wird auf die Variable zugegriffen, die an der entsprechenden Stelle der Variablenliste steht.   |
| StartValue           | Gibt den Startwert der Variable an.   |
| ID tag               | Über die Aktualisierungskennung aktualisieren Sie den Wert einer Variablen mit Hilfe einer Funktion oder eines PLC-Auftrags. Innerhalb eines Bediengeräts muss die Aktualisierungskennung eindeutig sein.   |
| Coding               | Zeigt die Methode zur Codierung an.   |
| DisplayName [de_DE]  | Zeigt den Anzeigenamen einer HMI-Variable. Die Bezeichnung des Feldes enthält eine Sprachkennung. Die Bezeichnung des Feldes enthält eine Sprachkennung. Anzeigenamen müssen für den Import mit einer Sprachkennung versehen sein. Beim Import wird der Text in die zugehörige Projektsprache importiert.   |
| Acquisition mode     | Gibt die Erfassungsart der Variable an.   |
| Acquisition cycle    | Gibt den Erfassungszyklus der Variable an. Der Erfassungszyklus muss genau so geschrieben werden, wie die Darstellung in WinCC. Der Wert ist sprachunabhängig und daher in jeder Sprache gleich anzugeben. Der Standardwert ist "1 s". Ist die Erfassungsart der Variable "auf Anforderung", dann ist der Erfassungszyklus undefiniert.<br>Benutzerdefinierte Erfassungszyklen müssen vorher angelegt sein, da die Datei sonst nicht importiert wird. |
| High High Limit type | Gibt an, ob der Grenzwert "High high" mit einer Konstanten, Variablen oder gar nicht überwacht wird.  |
| High High Limit      | Zeigt den Grenzwert "High High" an.   |
| High Limit type      | Gibt an, ob der Grenzwert "High" mit einer Konstanten, Variablen oder gar nicht überwacht wird.   |
| High Limit           | Zeigt den Grenzwert "High" an.  |
| Low Limit type       | Gibt an, ob der Grenzwert "Low" mit einer Konstanten, Variablen oder gar nicht überwacht wird.  |
| Low Limit            | Zeigt den Grenzwert "Low" an.   |
| Low Low Limit type   | Gibt an, ob der Grenzwert "Low Low" mit einer Konstanten, Variablen oder gar nicht überwacht wird.  |
| Low Low Limit        | Zeigt den Grenzwert "Low Low" an.   |
| Linear scaling       | Gibt an, ob die lineare Skalierung aktiviert ist. Der Eintrag kann nur für externe Variablen verwendet werden.  |
| End value PLC        | Gibt den Endwert der PLC-Variable an.   |
| Start value PLC      | Gibt den Startwert der PLC-Variable an.   |

| Listeneintrag   | Bedeutung                               |
|-----------------|---|
| End value HMI   | Gibt den Endwert der HMI-Variable an.   |
| Start value HMI | Gibt den Startwert der HMI-Variable an. |

**Beispiel für das Tabellenblatt "Multiplexing"**

|    | A            | B              | C     |
|----|--------------|----------------|-------|
| 1  | HMI Tag name | Multiplex Tag  | Index |
| 2  | Mux_Tag_1    | Mux_11         | 0     |
| 3  | Mux_Tag_1    | Mux_12         | 1     |
| 4  | Mux_Tag_1    | Mux_13         | 2     |
| 5  | Mux_Tag_2    | Mux_21         | 0     |
| 6  | Mux_Tag_2    | Mux_22         | 1     |
| 7  | Mux_Tag_2    | Mux_23         | 2     |
| 8  | Mux_Tag_12   | HMI_Array_Mux2 | -1    |
| 9  | Mux_Tag_11   | HMI_Array_Mux1 | -1    |
| 10 | Mux_Tag_13   | HMI_Array_Mux3 | -1    |

Tabelle 10-18 Bedeutung der Einträge

| Listeneintrag | Bedeutung  |
|---------------|--|
| Name          | Gibt den projektierten Namen einer HMI-Variable an, die eine indirekte Adressierung verwendet. Die HMI-Variable muss im Tabellenblatt "HMI Tags" vorhanden sein. |
| Index         | Zeigt den Wert an, von dem die Auswahl der Variable abhängt.   |
| Multiplex Tag | Zeigt die zum Indexwert zugehörige Variable aus der Variablenliste.  |

**Hinweis**

**"No Value" in der Tabelle**

Einträge in der Tabelle, die den Wert "No Value" haben, löschen die entsprechenden Werte einer bestehenden, namensgleichen Variablen.

**Siehe auch**

Variablen exportieren (Seite 3865)

### 10.9.3.5 Textlisten importieren und exportieren

#### Textlisten exportieren


##### Einleitung

WinCC stellt eine Exportfunktion für Textlisten zur Verfügung.

##### Voraussetzungen

- Das WinCC Projekt für den Export ist geöffnet.
- In dem Projekt sind Textlisten angelegt.
- Der Editor "Text- & Grafiklisten" ist geöffnet.

##### Textlisten exportieren

1. Klicken Sie in der Registerkarte "Textlisten" auf die Schaltfläche . Der Dialog "Export" wird geöffnet.



2. Klicken Sie auf die Schaltfläche "..." und legen Sie fest, in welche Datei die Daten gespeichert werden.
3. Klicken Sie auf "Export". Der Export wird gestartet.

##### Ergebnis

Die exportierten Daten wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

##### Siehe auch

- Textlisten importieren (Seite 3872)
- Import und Export von Projektdaten (Seite 3852)
- Variablen exportieren (Seite 3865)
- Rezepturen exportieren (Seite 3854)
- Format der Daten für Textlisten (Seite 3873)

## Textlisten importieren


### Einleitung

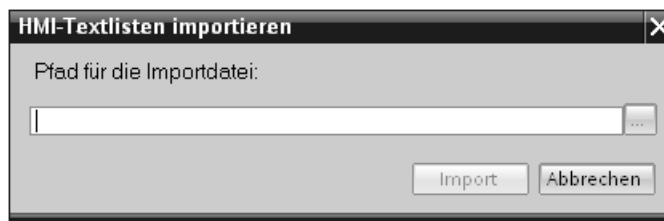
Im Folgenden importieren Sie Textlisten aus einer xlsx-Datei in WinCC.

### Voraussetzungen

- Eine xlsx-Datei mit Textlisten ist angelegt.
- Die xlsx-Datei ist den Anforderungen entsprechend aufgebaut.
- Das WinCC Projekt für den Import ist geöffnet.
- Der Editor "Text- & Grafiklisten" ist geöffnet.

### Textlisten importieren

1. Klicken Sie in der Registerkarte "Textlisten" auf die Schaltfläche . Der Dialog "Import" wird geöffnet.



2. Wählen Sie unter "Dateiauswahl" die Datei, die Sie importieren wollen.
3. Klicken Sie auf die Schaltfläche "Importieren". Der Import wird gestartet.

### Ergebnis

Sie haben die Textlisten importiert. Die entsprechenden Textlisten wurden in WinCC angelegt. Im Ausgabefenster werden Meldungen zum Importvorgang ausgegeben. Im Quellverzeichnis der Importdateien wird eine Log-Datei abgelegt. Die Log-Datei erhält den gleichen Namen wie die jeweilige Importdatei und hat die Dateierweiterung "\*.xml".

Achten Sie beim Import der Daten darauf, ob Verknüpfungen zu Objekten vorhanden, sind z. B. dynamische Parameter wie Variablen.

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, legen Sie ein Objekt mit dem entsprechenden Namen an oder legen Sie eine neue Verknüpfung an.

### Siehe auch

Textlisten exportieren (Seite 3871)

## Format der Daten für Textlisten

### Einleitung

Dieses Kapitel beschreibt den erforderlichen Aufbau der Datei mit den Textlisten für den Import. Die Datei mit den Textlistendaten muss im Dateiformat "\*.xlsx" vorliegen.

### Struktur der Variablendaten

Die Importdatei besteht in Microsoft Excel aus zwei Tabellenblättern:

- TextList (Textlisten)
- TextListEntry (Textlisteneintrag)

Jede Textliste erhält eine separate Zeile in der Importdatei. Die Importdatei mit den Daten muss folgendermaßen aufgebaut sein:

### Beispiel für das Tabellenblatt "TextList"

|   | A             | B         | C       |
|---|---------------|-----------|---------|
| 1 | Name          | ListRange | Comment |
| 2 | TLValue/Range | Decimal   |         |
| 3 | TLBit         | Bit       |         |
| 4 | TLBitnumber   | Binary    |         |
| 5 | TLall_1       | Decimal   |         |
| 6 | TLall_2       | Decimal   |         |

Tabelle 10-19 Bedeutung der Einträge

| Listeneintrag | Bedeutung  |
|---------------|--|
| Name          | Zeigt den Namen der Textliste.   |
| ListRange     | Gibt den Bereich der Textliste an:<br>Nummer = Bitnummer (0-31)<br>Range = Wert/Bereich<br>Bit = Bit (0;1) |
| Comment       | Beliebiger Kommentar zu der Textliste. Sie können maximal 500 Zeichen verwenden                            |

**Beispiel für das Tabellenblatt "TextListEntry"**

|    | A                 | B             | C            | D     | E   | F   |
|----|-------------------|---------------|--------------|-------|---|---|
| 1  | Name              | Parent        | DefaultEntry | Value | Text  | FieldInfos  |
| 2  | Text_list_entry_1 | TLValue/Range | TRUE         | 0 - 1 | Default entry TLValue/Range ->                      |   |
| 3  | Text_list_entry_2 | TLValue/Range |              | 2 - 3 | TLValue/Range = 2-3 ->                              |   |
| 4  | Text_list_entry_3 | TLValue/Range |              | 1     | TLValue Range - single value = 1 ->                 |   |
| 5  | Text_list_entry_1 | TLBit         |              | 0     | TLBit = 0 ->  |   |
| 6  | Text_list_entry_2 | TLBit         |              | 1     | TLBit = 1 ->  |   |
| 7  | Text_list_entry_1 | TLBitnumber   | TRUE         | 0     | Default entry TLBitnumber; ->                       |   |
| 8  | Text_list_entry_2 | TLBitnumber   |              | 0     | TLBitnumber - Bitnumber 0 is set ->                 |   |
| 9  | Text_list_entry_3 | TLBitnumber   |              | 1     | TLBitnumber - Bitnumber 1 is set ->                 |   |
| 10 | Text_list_entry_4 | TLBitnumber   |              | 2     | TLBitnumber - Bitnumber 2 is set ->                 |   |
| 11 | Text_list_entry_5 | TLBitnumber   |              | 3     | TLBitnumber - Bitnumber 3 is set ->                 |   |
| 12 | Text_list_entry_1 | TL1           | TRUE         | 0 - 1 | Default entry TL1                                   |   |
| 13 | Text_list_entry_2 | TL1           |              | 1 - 3 | TL1 Value between 1 - 3 ->                          |   |
| 14 | Text_list_entry_3 | TL1           |              | 4 - 6 | TL1 Value between 4 - 6 ->                          |   |
| 15 | Text_list_entry_4 | TL1           |              | 7     | TL1 Single value = 7 ->                             |   |
| 16 | Text_list_entry_1 | TL2           | TRUE         | 0 - 1 | <field ref="0" />->                                 | <ref id = 0; type = CommonTextList;<br>TextList = TL1; Tag = HMI_TL1control; Length = 1 |
| 17 | Text_list_entry_2 | TL2           |              | 1     | TL2 Single value = 1 ->                             |   |
| 18 | Text_list_entry_3 | TL2           |              | 2 - 3 | TL2 Range between 2 - 3 ->                          |   |
| 19 | Text_list_entry_1 | TLMultilined  | TRUE         | 0 - 1 | Default<br />entry<br />TLMultilined;<br />last row |   |
| 20 | Text_list_entry_2 | TLMultilined  |              | 0 - 3 | TLMultilined Value between 0-3\nwith test of"\n"    |   |
| 21 | Text_list_entry_1 | TLall_1       | TRUE         | 0 - 1 | Default entry TLall_1                               |   |
| 22 | Text_list_entry_1 | TLall_2       | TRUE         | 0 - 1 | Default entry TLall_2                               |   |

Tabelle 10-20 Bedeutung der Einträge

| Listeneintrag | Bedeutung   |
|---------------|---|
| Name          | Zeigt den Namen des Textlisteneintrags.   |
| Parent        | Zeigt den Namen der zugehörigen Textliste.  |
| DefaultEntry  | Gibt an, ob der Textlisteneintrag ein Standardeintrag ist. Der Standardeintrag erscheint immer dann, wenn die Variable einen nicht definierten Wert annimmt.  |
| Value         | Gibt die ganzzahligen Werte bzw. Wertebereiche einer Variablen an, die den Texteinträge aus der Textliste zugeordnet sind.  |
| Text          | <p>Zeigt den Textlisteneintrag. Die Bezeichnung des Feldes enthält eine Sprachkennung. Textlisteneinträge müssen für den Import mit einer Sprachkennung versehen sein.</p> <p>Wenn ein dynamischer Parameter im Textlisteneintrag vorhanden ist, wird dem Text ein Ausdruck mit einer Refrenz-ID hinzugefügt. Beispiel: text &lt;field ref="0" /&gt;. Über die ID wird der dynamische Parameter einem Textlisteneintrag zugeordnet.</p>   |
| FeldInfo      | <p>Gibt an, ob dynamische Parameter in der Textliste enthalten sind. Die Einstellungen sind durch ein Semikolon ";" getrennt.</p> <p>Beispiel für dynamische Parameter:</p> <p>Variable: &lt;ref id = 0; type = CommonTagDisplayFormat; Tag = tag 1; DisplayType = Decimal; DisplayFormat = 9;&gt;</p> <p>Textliste: &lt;ref id = 1; type = CommonTextList; TextList = Textliste_1; Tag = tag 2; Length = 5;&gt;</p> <p>Steuerungsvariable: &lt;ref id = 0; type = CommonControlTagDisplayFormat; DisplayType = Decimal; DisplayFormat = 9;&gt;</p> |



## Siehe auch

Textlisten exportieren (Seite 3871)

### 10.9.3.6 Projekttexte importieren und exportieren

#### Projekttexte exportieren

Sie exportieren Projekttexte zur Übersetzung. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann z. B. in Microsoft Excel bearbeitet werden.

Sie können die Datei mit den Übersetzern austauschen und nach der Übersetzung direkt in das Projekt wieder importieren.

#### Voraussetzungen

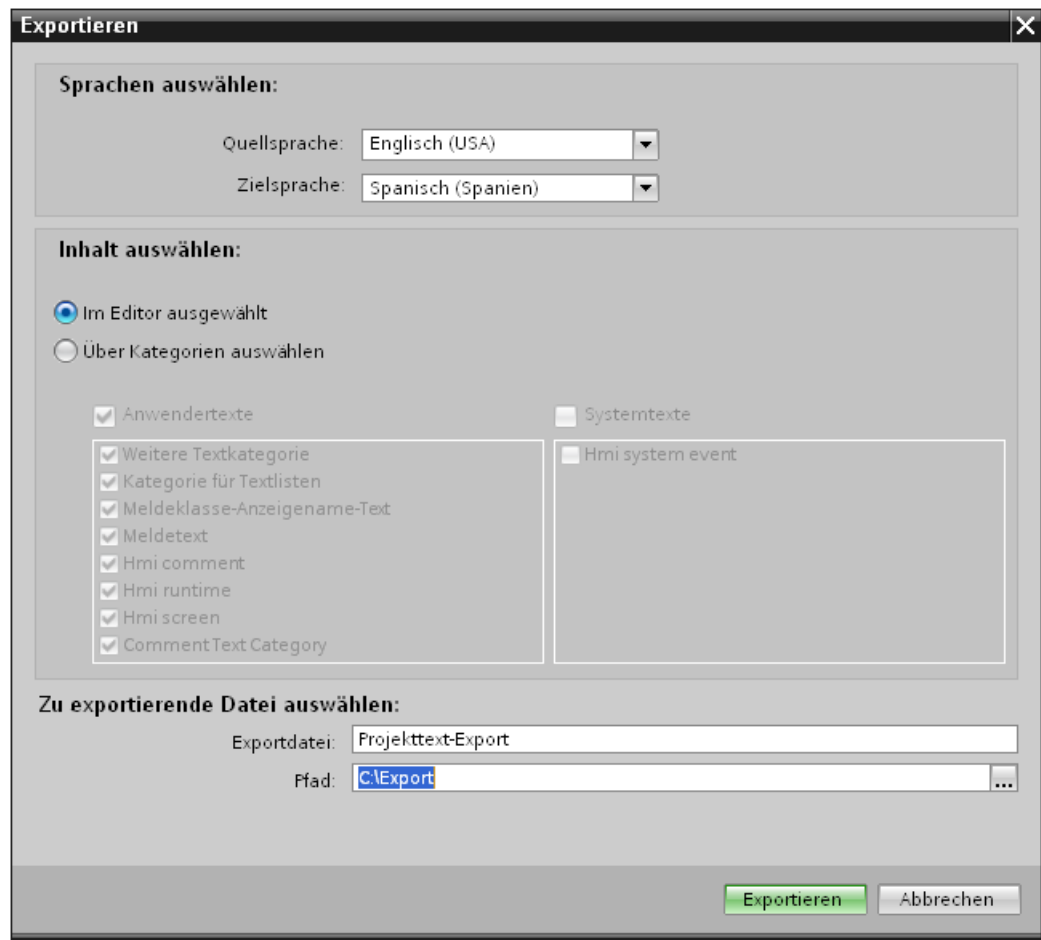
- Im Editor "Projektsprachen" sind mindestens zwei Sprachen aktiviert, z.B. Italienisch und Französisch.

#### Projekttexte exportieren

Um einzelne Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte". Der Editor "Projekttexte" wird geöffnet.
3. Selektieren Sie die Texte, die Sie exportieren möchten.

4. Klicken Sie auf die Schaltfläche . Der Dialog "Export" wird geöffnet.



5. Wählen Sie aus der Klappliste "Quellsprache" die Sprache, aus der Sie die Übersetzung vornehmen möchten, z. B. Italienisch.
6. Wählen Sie aus der Klappliste "Zielsprache" die Sprache, in die Sie die Texte übersetzen möchten, z. B. Französisch.
7. Geben Sie im Eingabefeld "Exportdatei" einen Speicherpfad und einen Dateinamen für die Exportdatei an.
8. Klicken Sie auf "Export".

## Ergebnis

Die im Editor "Projekttexte" selektierten Texte wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

Alternativ können Sie auch alle Projekttexte über Kategorien auswählen und exportieren. Aktivieren Sie im Dialog "Export" die Optionen "Anwendertexte" oder "Systemtexte", abhängig

davon, welche Texte Sie exportieren möchten. Der Export kann in diesem Fall zusätzlich nach Kategorien eingegrenzt werden.

---

#### Hinweis

Projekttexte in Bibliotheksobjekten können nicht exportiert werden.

---

#### Siehe auch

Projekttexte importieren (Seite 3877)

#### Projekttexte importieren


Sie bearbeiten die xlsx-Datei oder geben Sie an einen Übersetzer weiter. Nach abgeschlossener Übersetzung importieren Sie die übersetzten Texte. Die Fremdsprachen werden im Projekt an das entsprechende Objekt importiert.

#### Voraussetzungen

- Im Editor "Projektsprachen" sind mindestens zwei Sprachen aktiviert, z.B. Italienisch und Französisch.

#### Projekttexte importieren

Um eine Datei mit Projekttexten zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte". Der Editor "Projekttexte" wird geöffnet.
3. Klicken Sie auf die Schaltfläche . Der Dialog "Import" wird geöffnet.
4. Wählen Sie aus dem Feld "Importdatei" den Pfad und den Dateinamen der Importdatei aus.
5. Aktivieren Sie das Optionskästchen "Quellsprache importieren", wenn Sie in der Exportdatei Änderungen an der Quellsprache vorgenommen haben und die Einträge im Projekt mit den Änderungen überschreiben möchten.
6. Klicken Sie auf "Importieren".

#### Ergebnis

Sie haben die Projekttexte importiert.

#### Siehe auch

Projekttexte exportieren (Seite 3875)

## 10.9.4 Querverweise nutzen

### 10.9.4.1 Allgemeine Informationen zu Querverweisen

#### Einführung

Die Querverweisliste bietet einen Überblick über die Verwendung von Objekten innerhalb des Projekts.

#### Nutzen der Querverweise

Die Querverweisliste bietet Ihnen folgende Vorteile:

- Bei der Projekterstellung und bei Änderungen behalten Sie die Übersicht über verwendete Objekte, Variablen, Meldungen usw.
- Von den Querverweisen aus können Sie direkt an die jeweilige Verwendungsstelle von Objekten springen.
- Bei der Fehlersuche erfahren Sie z. B. Folgendes:
  - Welche Objekte werden in welchem Bild verwendet.
  - Welche Meldungen und Rezepturen werden in welcher Anzeige angezeigt.
  - Welche Variable wird in welcher Meldung oder welchem Objekt verwendet.
- Als Teil der Projektdokumentation liefern die Querverweise einen umfassenden Überblick über alle verwendeten Objekte, Meldungen, Rezepturen, Variablen, Bilder usw.

### 10.9.4.2 Querverweisliste anzeigen

#### Einführung

Über die Querverweisliste erhalten Sie Informationen über die Verwendung von Objekten. In der Projektnavigation zeigen Sie Querverweise für Bediengeräte, Ordner und alle Editoren an. Über die Detailansicht können Sie auch einzelne Objekte der Editoren wählen.

#### Voraussetzung

Ein Projekt ist angelegt.

Mehrere Objekte sind angelegt.

#### Vorgehen

1. Selektieren Sie in der Projektnavigation oder in der Detailansicht den gewünschten Eintrag.
2. Wählen Sie im Kontextmenü den Befehl "Querverweise". Die Querverweisliste wird im Arbeitsbereich geöffnet.

3. Um anzuzeigen, an welchen Stellen die in der Querverweisliste angezeigten Objekte verwendet werden, öffnen Sie die Registerkarte "Verwendet von".
4. Um die Verwender der in der Querverweisliste angezeigten Objekte zu sehen, öffnen Sie die Registerkarte "Verwendet".
5. Um die Einträge in der Spalte "Objekt" auf- bzw. absteigend zu sortieren, klicken Sie auf die jeweilige Spaltenüberschrift.
6. Um an die Verwendungsstelle des jeweiligen Objekts zu gelangen, klicken Sie auf den angezeigten Link.

## Ergebnis

Im Arbeitsbereich wird die Querverweisliste zum gewählten Objekt angezeigt.

### 10.9.4.3 Aufbau der Querverweisliste

#### Ansichten der Querverweisliste

Es gibt zwei Ansichten der Querverweisliste, die sich darin unterscheiden, welche Objekte jeweils in der ersten Spalte angezeigt werden:

- **Verwendet von:**  
Anzeige der referenzierten Objekte. Hier wird angezeigt, an welchen Stellen das Objekt verwendet wird.
- **Verwendet:**  
Anzeige der referenzierenden Objekte. Hier werden die Verwender des Objekts angezeigt.

Die zugeordneten Tooltips geben weitere Informationen zu den jeweiligen Objekten.

#### Aufbau der Querverweisliste

| Spalte            | Inhalt/Bedeutung   |
|-------------------|--|
| Objekt            | Name des Objekts, das die unterlagerten Objekte verwendet, bzw. von diesen verwendet wird. |
| Anzahl            | Anzahl der Verwendungen  |
| Verwendungsstelle | Jeweilige Verwendungsstelle, z. B. Objekt oder Ereignis                                    |
| Eigenschaft       | Funktion der referenzierten Objekte, z. B. Variable für Datensatz oder Prozesswert         |
| Verbunden mit     | PLC-Variable, mit der das Objekt verbunden ist.  |
| Typ               | Typ des Objekts  |
| Pfad              | Pfad des Objekts   |

Abhängig von den installierten Produkten werden zusätzliche oder andere Spalten für die Querverweise angezeigt.

## Einstellungen in der Querverweisliste

Folgende Einstellungen legen Sie über die Symbole in der Funktionsleiste für die Querverweisliste fest:

- **Querverweisliste aktualisieren**  
Aktualisiert die aktuelle Querverweisliste.
- **Einstellungen für die Querverweisliste festlegen**  
Hier legen Sie fest, ob alle verwendeten, alle nicht verwendeten, alle definierten bzw. alle nicht definierten Objekte angezeigt werden. Wenn die Option "nicht definierte Objekte" aktiviert ist, werden auch Referenzen zu bereits gelöschten Objekten angezeigt.
- **Einträge reduzieren**  
Reduziert die Einträge in der aktuellen Querverweisliste, indem die unterlagerten Objekte geschlossen werden.
- **Einträge erweitern**  
Erweitert die Einträge in der aktuellen Querverweisliste, indem die unterlagerten Objekte geöffnet werden.

## Sortieren in der Querverweisliste

Die Einträge in der Spalte "Objekt" können auf- bzw. absteigend sortiert werden. Klicken Sie dazu auf die Spaltenüberschrift.

### 10.9.4.4 Querverweise im Inspektorfenster anzeigen

#### Einführung

Im Inspektorfenster "Info > Querverweise" werden die Querverweisinformationen zu einem selektierten Objekt angezeigt. Die Querverweisinformationen werden im Inspektorfenster tabellarisch dargestellt.

#### Voraussetzung

Ein Projekt ist angelegt.

Mehrere Objekte sind angelegt.

## Vorgehen

1. Selektieren Sie in einem Bild oder in einem tabellarischen Editor ein Objekt.
2. Wählen Sie im Kontextmenü "Querverweis-Information". Im Inspektorfenster werden die Querverweise geöffnet.



## Ergebnis

Für das selektierte Objekt wird gezeigt, an welchen Stellen und von welchen anderen Objekten es verwendet wird.

Die folgende Tabelle zeigt, welche Detailinformationen im Register "Info > Querverweis" aufgelistet werden:

| Spalte            | Bedeutung  |
|-------------------|--|
| Objekt            | Name des Objekts, das die unterlagerten Objekte verwendet, bzw. von diesen verwendet wird. |
| Anzahl            | Anzahl der Verwendungen  |
| Verwendungsstelle | Jeweilige Verwendungsstelle, z. B. Objekt oder Ereignis                                    |
| Eigenschaft       | Funktion der referenzierten Objekte, z. B. Variable für Datensatz oder Prozesswert         |
| Verbunden mit     | PLC-Variable, mit der das Objekt verbunden ist.  |
| Typ               | Typ des Objekts  |
| Pfad              | Pfad des Objekts   |

Abhängig von den installierten Produkten werden zusätzliche oder andere Spalten für die Querverweise angezeigt.

## 10.9.5 Sprachen verwalten

### 10.9.5.1 Sprachbegriffe in WinCC

#### Oberflächensprache und Projektsprachen

In WinCC wird zwischen zwei Sprachebenen unterschieden:

- **Oberflächensprache**  
In der Oberflächensprache werden beim Projektieren die Texte in den Menüs und Dialogen von WinCC angezeigt. Die Oberflächensprache beeinflusst auch die Beschriftung der Bedienobjekte, die Parameter der Systemfunktionen, die Onlinehilfe usw.
- **Projektsprachen**  
Als Projektsprachen werden alle Sprachen bezeichnet, in denen ein Projekt später bearbeitet werden soll. Projektsprachen werden verwendet, um ein Projekt in mehreren Sprachen zu erstellen.

Beide Sprachebenen sind vollkommen unabhängig voneinander. Zum Beispiel können Sie mit einer deutschen Oberflächensprache jederzeit englische Projekte erstellen und umgekehrt.

#### Projektsprachen

Innerhalb der Projektsprachen werden folgende Sprachen unterschieden:

- **Referenzsprache**  
Die Referenzsprache ist die Sprache, in der Sie das Projekt zunächst projektieren. Während der Projektierung wählen Sie eine der Projektsprachen als Referenzsprache. Die Referenzsprache verwenden Sie als Vorlage für Übersetzungen. Sie erstellen zunächst alle Texte des Projekts in der Referenzsprache und übersetzen diese anschließend. Während Sie übersetzen, lassen Sie sich gleichzeitig die Texte der Referenzsprache anzeigen.
- **Editiersprache**  
In der Editiersprache erstellen Sie die Übersetzungen der Texte. Nachdem Sie Ihr Projekt in der Referenzsprache erstellt haben, übersetzen Sie die Texte in die übrigen Projektsprachen. Wählen Sie jeweils eine Projektsprache als Editiersprache und bearbeiten Sie die Texte für die entsprechende Sprachvariante. Sie können die Editiersprache jederzeit wechseln.

---

#### Hinweis

Beim Umschalten der Projektsprache wird auch die Tastenbelegung der Tastatur geändert. Bei einigen Sprachen (z. B. Spanisch) ist die Umstellung der Tastenbelegung betriebssystembedingt nicht möglich. In diesem Fall wird die Tastenbelegung auf Englisch umgeschaltet.

---

- **Runtime-Sprachen**  
Die Runtime-Sprachen sind die Projektsprachen, die auf das Bediengerät übertragen werden. Abhängig von den Erfordernissen Ihres Projekts entscheiden Sie, welche der Projektsprachen auf das Bediengerät übertragen werden. Damit der Bediener in Runtime zwischen diesen Sprachen umschalten kann, stellen Sie ihm entsprechende Bedienelemente zur Verfügung.



## Siehe auch

- Einstellungen für Sprachen im Betriebssystem (Seite 3883)
- Einstellungen im Betriebssystem für asiatische Sprachen (Seite 3884)
- Oberflächensprache wählen (Seite 3885)
- Editor "Grafiken" (Seite 3896)
- Sprachen in Runtime (Seite 3900)
- Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 3907)

### 10.9.5.2 Einstellungen für Sprachen im Betriebssystem

#### Einleitung

Die Einstellungen im Betriebssystem des Projektierungs-PCs haben in folgenden Bereichen Einfluss auf die Sprachverwaltung von WinCC:

- Auswahl der Projektsprachen
- Sprachabhängige Darstellung von Datumsangaben, Uhrzeiten, Geldbeträgen und Zahlen
- Darstellung von ASCII-Zeichen

#### Auswahl der Projektsprache

Eine Sprache steht nur dann als Projektsprache zur Verfügung, wenn sie im Betriebssystem installiert ist.

#### Sprachabhängige Darstellung von Datumsangaben, Uhrzeiten, Geldbeträgen und Zahlen

Die Darstellung von Datumsangaben und Uhrzeiten wird im Datum-Uhrzeit-Feld durch WinCC für die eingestellte Projektsprache und Runtime-Sprache fest vorgegeben.

Damit Datumsangaben, Uhrzeiten und Zahlen in der gewählten Bearbeitungssprache formell korrekt dargestellt werden, müssen Sie diese Sprache in der Systemsteuerung über die Ländereinstellungen einstellen.

#### Darstellung von ASCII-Zeichen

Bei Textausgabefeldern ist die Darstellung der ASCII-Zeichen ab 128 abhängig von der eingestellten Sprache und von dem verwendeten Betriebssystem.

Wenn auf verschiedenen PCs die gleichen Sonderzeichen dargestellt werden sollen, dann müssen auf den PCs das verwendete Betriebssystem und die verwendeten Ländereinstellung gleich sein.

## Siehe auch

Sprachbegriffe in WinCC (Seite 3882)

Einstellungen im Betriebssystem für asiatische Sprachen (Seite 3884)

### 10.9.5.3 Einstellungen im Betriebssystem für asiatische Sprachen

#### Einstellungen auf westlichen Betriebssystemen

Wenn Sie asiatische Zeichen eingeben wollen, müssen Sie im Betriebssystem die Unterstützung für diese Sprache aktivieren.

Zur Projektierung von asiatischen Texten steht Ihnen unter Windows der Input Method Editor (IME) zur Verfügung. Ohne diesen Editor können Sie asiatische Texte nur anzeigen, aber nicht bearbeiten. Weitere Informationen zum Input Method Editor finden Sie in der Windows-Dokumentation. Zur Eingabe von asiatischen Zeichen bei der Projektierung wechseln Sie im "Input Methode Editor" zu der asiatischen Eingabemethode.

Um sprachabhängige Projekttexte wie Meldetexte im Simulator in asiatischen Zeichen angezeigt zu lassen, stellen Sie das Betriebssystem auf die gewünschte Sprache um.

#### Einstellungen auf asiatischen Betriebssystemen

Wenn Sie auf einem asiatischen Betriebssystem projektieren, müssen Sie für die Eingabe von ASCII-Zeichen, z. B. für Objektnamen, auf das englische Eingabegebietsschema wechseln. Da das englische Eingabegebietsschema in der Grundinstallation des Betriebssystems enthalten ist, brauchen Sie kein zusätzliches Eingabegebietsschema installieren.

#### Sprachunterstützung aktivieren

1. Öffnen Sie die Systemsteuerung.
2. Wählen Sie "Regions- und Sprachoptionen".
3. Aktivieren Sie auf der Registerkarte "Sprachen" das Kontrollkästchen "Dateien für ostasiatische Sprachen installieren".
4. Klicken Sie anschließend im Bereich "Textdienste und Eingabesprachen" auf die Schaltfläche "Details". Der Dialog "Textdienste und Eingabesprachen" wird geöffnet.
5. Auf der Registerkarte "Einstellungen" fügen Sie im Bereich "Installierte Dienste" das benötigte Eingabegebietsschema hinzu.
6. Auf der Registerkarte "Erweitert" wählen Sie im Bereich "Sprache für Programme, die Unicode nicht unterstützen" die Sprache des Betriebssystems aus.

## Siehe auch

Sprachbegriffe in WinCC (Seite 3882)

Einstellungen für Sprachen im Betriebssystem (Seite 3883)

## **10.9.5.4 Projektsprachen einstellen**

### **Oberflächensprache wählen**

#### **Einleitung**

Die Oberflächensprache wird für die Anzeige von Menüeinträgen, Titelleisten, Hilfetexten, Texten in Dialogfenstern und weiteren Bezeichnungen in der Benutzeroberfläche von WinCC verwendet.

Während der Projektierung können Sie zwischen den installierten Oberflächensprachen wechseln. Die Bezeichnung der Bedienobjekte bleibt auch nach dem Wechsel der Oberflächensprache in der Sprache erhalten, die Sie beim Einfügen des Objekts eingestellt hatten.

#### **Vorgehen**

1. Wählen Sie im Menü "Extras > Einstellungen".  
Der Dialog "Einstellungen" wird geöffnet.
2. Wählen Sie unter "Allgemein > Allgemeine Einstellungen" die gewünschte Oberflächensprache aus.

#### **Ergebnis**

Die gewählte Sprache wird von WinCC als Oberflächensprache verwendet.

#### **Siehe auch**

- Projektsprachen aktivieren (Seite 3885)
- Referenzsprache und Editiersprache wählen (Seite 3886)
- Sprachbegriffe in WinCC (Seite 3882)

### **Projektsprachen aktivieren**

#### **Einleitung**

Im Editor "Projektsprachen" stellen Sie die Projektsprachen ein. Sie legen fest, welche der Projektsprachen als Referenzsprache und welche als Editiersprache dient.

### Projektsprachen aktivieren

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projektsprachen".  
Im Arbeitsbereich werden die möglichen Projektsprachen angezeigt.
3. Aktivieren Sie die Sprachen, die als Projektsprachen zur Verfügung stehen sollen.

---

#### Hinweis

##### Kopieren mehrsprachig projektierter Objekte

Beim Kopieren eines mehrsprachig projizierten Objekts in ein anderes Projekt werden nur Texte in den Projektsprachen mitkopiert, die im Zielprojekt aktiviert sind. Aktivieren Sie im Zielprojekt alle Projektsprachen, deren Texte Sie beim Kopieren mit in das Zielprojekt übernehmen wollen.

---

### Projektsprachen deaktivieren

1. Deaktivieren Sie die Sprachen, die nicht als Projektsprachen zur Verfügung stehen sollen.

|   |
|---|
| <b>ACHTUNG</b>  |
| Wenn Sie eine Projektsprache deaktivieren, werden im aktuellen Projekt alle in dieser Sprache bereits erstellten Texte und Grafiken gelöscht. |

### Siehe auch

Oberflächensprache wählen (Seite 3885)

Referenzsprache und Editiersprache wählen (Seite 3886)

### Referenzsprache und Editiersprache wählen

### Einleitung

Im Editor "Projektsprachen" stellen Sie die Projektsprachen ein. Sie legen fest, welche der Projektsprachen als Referenzsprache und welche als Editiersprache dient. Die Editiersprache können Sie jederzeit ändern.

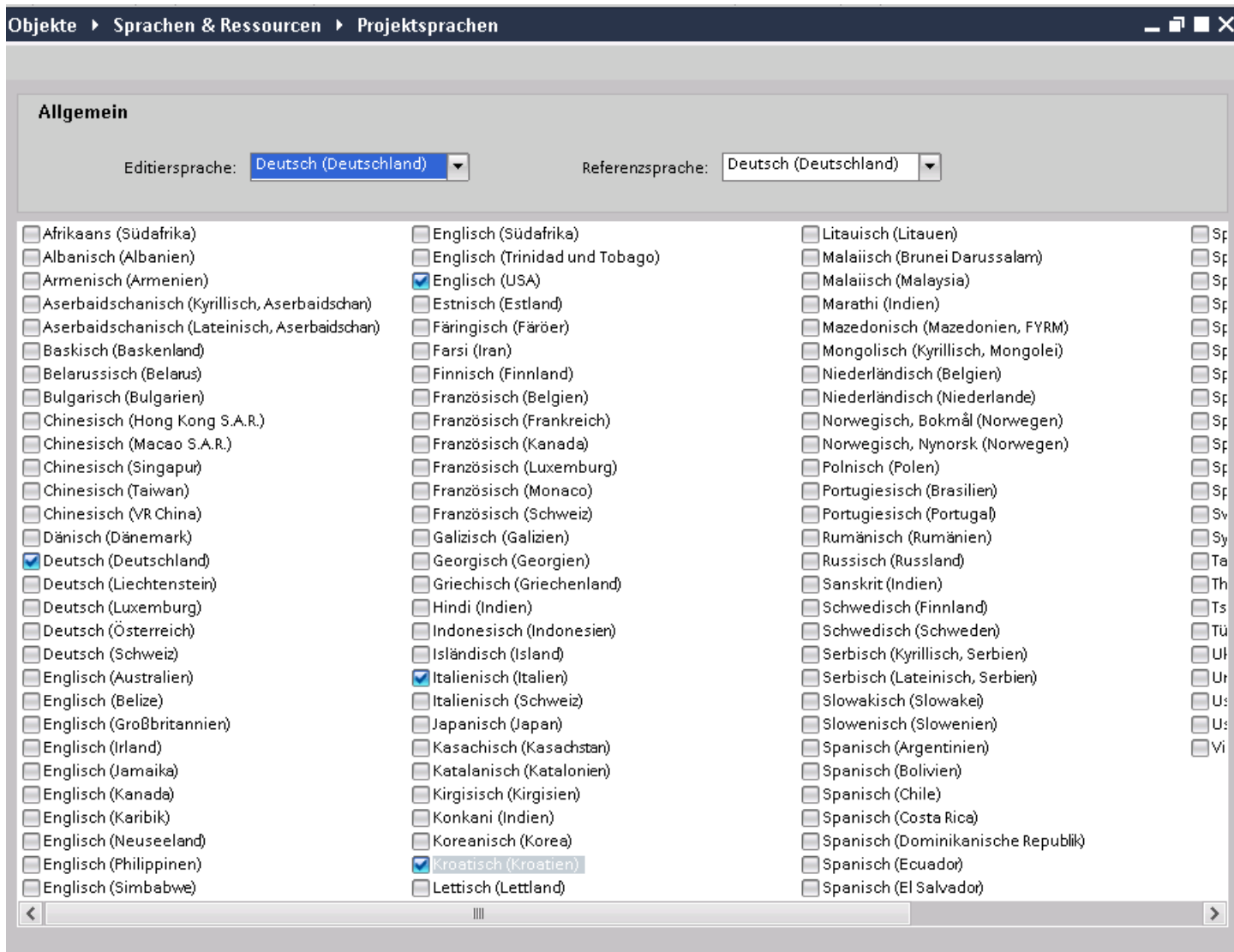
### Voraussetzungen

Der Editor "Projektsprachen" ist geöffnet.

Mehrere Projektsprachen sind aktiviert.

## Referenzsprache und Editiersprache wählen

1. Klicken Sie im Bereich "Allgemein > Editiersprache" in der Auswahlliste auf das Pfeilsymbol.
  2. Klicken Sie in der Auswahlliste auf die gewünschte Sprache z. B. Deutsch.
  3. Klicken Sie im Bereich "Allgemein > Referenzsprache" in der Auswahlliste auf das Pfeilsymbol.
  4. Klicken Sie in der Auswahlliste auf die gewünschte Sprache z. B. Englisch.
- Die ausgewählte Sprache wird im Listenfeld angezeigt.



## Ergebnis

Sie haben die Editiersprache und die Referenzsprache gewählt.

Wenn Sie die Editiersprache ändern, werden alle zukünftigen Texteingaben in der neuen Editiersprache gespeichert.

### Siehe auch

Oberflächensprache wählen (Seite 3885)

Projektsprachen aktivieren (Seite 3885)

## 10.9.5.5 Ein Projekt in mehreren Sprachen erstellen

### Arbeiten mit mehreren Sprachen

#### Mehrsprachige Projektierung in WinCC

Mit WinCC projektieren Sie Ihre Projekte auch in mehreren Sprachen. Die Erstellung eines Projekts in mehreren Sprachen kann verschiedene Ursachen haben:

- Sie setzen ein Projekt in mehreren Ländern ein.  
Sie erstellen das Projekt in mehreren Sprachen; bei der Inbetriebnahme des Bediengeräts wird nur die Sprache auf das Bediengerät übertragen, die von den Bedienern am jeweiligen Aufstellungsort gesprochen wird.
- Die Bediener einer Anlage sprechen verschiedene Sprache.  
Beispiel: Ein Bediengerät wird in China eingesetzt, das Servicepersonal versteht jedoch nur Englisch.

#### Projekttexte übersetzen

Mit WinCC können Sie Projekttexte in mehreren Sprachen in verschiedenen Editoren direkt eingeben, z.B. im Editor "Projekttexte". Weiterhin stellt Ihnen WinCC zum Übersetzen Ihrer Projektierung Export- und Importmöglichkeiten zur Verfügung. Dies ist besonders von Vorteil, wenn Sie Projekte mit einem hohen Textanteil projektieren und übersetzen lassen möchten.

#### Sprachverwaltung und Übersetzung in WinCC

Folgende Editoren dienen in WinCC der Verwaltung von Sprachen und zur Übersetzung von Texten:

| Editor                 | Kurzbeschreibung  |
|------------------------|---|
| Projektsprachen        | Auswahl der Projektsprachen, der Editiersprache und der Referenzsprache.            |
| Sprachen und Schriften | Auswahl der Runtime-Sprachen und der am Bediengerät verwendeten Schriftarten.       |
| Projekttexte           | Zentrale Verwaltung der projektierten Texte in allen Projektsprachen.               |
| Grafiken               | Grafiksammlung für die Verwaltung der Grafiken und deren sprachabhängige Varianten. |

## Siehe auch

- Grundlagen zu Projekttexten (Seite 3889)
- Texte direkt übersetzen (Seite 3890)
- Texte mithilfe von Referenztexten übersetzen (Seite 3892)
- Projekttexte exportieren (Seite 3893)
- Projekttexte importieren (Seite 3895)

## Grundlagen zu Projekttexten

### Texte in unterschiedlichen Sprachen im Projekt

Texte, die während der Prozessbearbeitung auf Anzeigegeräten ausgegeben werden, werden üblicherweise in der Sprache eingegeben, in der die Automatisierungslösung programmiert ist. Ebenso werden Kommentare und Bezeichnungen von Objekten nur in dieser Sprache eingegeben.

Beherrscht ein Bediener die Sprache nicht, benötigt er eine Übersetzung aller bedienerrelevanten Texte in seine Muttersprache. Sie können daher sämtliche Texte in jede beliebige Sprache übersetzen. So können Sie sicherstellen, dass jeder, der zu einem späteren Zeitpunkt mit den Texten im Projekt konfrontiert wird, diese Texte in der von ihm gewünschten Sprache angezeigt bekommt.

### Anwendertexte und Systemtexte

Um die Übersichtlichkeit zu verbessern, wird zwischen Anwendertexten und Systemtexten unterschieden:

- Anwendertexte sind Texte, die vom Nutzer angelegt worden sind.
- Systemtexte sind Texte, die automatisch in Abhängigkeit von der Konfiguration im Projekt angelegt werden.

Die Projekttexte werden im Projekttexte-Editor verwaltet. Dieser ist in der Projektnavigation unter "Sprachen & Ressourcen > Projekttexte" zu finden.

### Beispiele für mehrsprachige Projekttexte

Sie können z. B. die folgenden Textarten mehrsprachig verwalten:

- Anzeigetexte
- Meldungstexte
- Kommentare in Tabellen
- Beschriftungen von Bildobjekten
- Textlisten

## Übersetzen von Texten

Um Texte zu übersetzen, gibt es zwei unterschiedliche Vorgehensweisen.

- **Texte direkt übersetzen**  
Sie können die Übersetzungen für die einzelnen Projektsprachen direkt im Editor "Projekttexte" eingeben.
- **Texte mithilfe von Referenztexten übersetzen**  
Für kleinere Textmengen können Sie die Editiersprache wechseln. Während die Texte der Referenzsprache eingeblendet bleiben, geben Sie in die neuen Texte in der Editiersprache ein.

## Siehe auch

Arbeiten mit mehreren Sprachen (Seite 3888)

Texte direkt übersetzen (Seite 3890)

Texte mithilfe von Referenztexten übersetzen (Seite 3892)

Projekttexte exportieren (Seite 3893)

Projekttexte importieren (Seite 3895)

## Texte direkt übersetzen

### Texte übersetzen

Wenn Sie in Ihrem Projekt mehrere Sprachen verwenden, können Sie einzelne Texte direkt übersetzen. Sobald Sie die Sprache der Softwareoberfläche ändern, stehen die übersetzten Texte in der jeweiligen Sprache zur Verfügung.

### Voraussetzungen

- Sie befinden sich in der Projektansicht.
- Ein Projekt ist geöffnet.
- Sie haben mindestens zwei weitere Projektsprache gewählt.





## Vorgehen

Um einzelne Texte zu übersetzen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die darunter liegenden Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte".  
Im Arbeitsbereich erscheint eine Liste mit den Texten im Projekt. Für jede Projektsprache existiert eine eigene Spalte.

| Chinesisch (VR China) | Französisch ( ) | Englisch (USA) | Italienisch (Italien) | Deutsch (Deutschland) | Kategorie | Referenz                              |
|-----------------------|-----------------|----------------|-----------------------|-----------------------|-----------|---------------------------------------|
| !                     | !               | !              | !                     | !                     | Alarm     | \\Objekte\\HMI_1\\HMI-Meldungen\\F... |
| \$                    | \$              | \$             | \$                    | \$                    | Alarm     | \\Objekte\\HMI_1\\HMI-Meldungen\\S... |
| 0                     | 0               | 0              | 0                     | 0                     | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 0                     | 0               | 0              | 0                     | 0                     | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 0                     | 0               | 0              | 0                     | 0                     | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 1                     | 1               | 1              | 1                     | 1                     | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 1                     | 1               | 1              | 1                     | 1                     | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 1                     | 1               | 1              | 1                     | 1                     | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 10                    | 10              | 10             | 10                    | 10                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 10                    | 10              | 10             | 10                    | 10                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 10                    | 10              | 10             | 10                    | 10                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 11                    | 11              | 11             | 11                    | 11                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 11                    | 11              | 11             | 11                    | 11                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 11                    | 11              | 11             | 11                    | 11                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 12                    | 12              | 12             | 12                    | 12                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 12                    | 12              | 12             | 12                    | 12                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 12                    | 12              | 12             | 12                    | 12                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 13                    | 13              | 13             | 13                    | 13                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 13                    | 13              | 13             | 13                    | 13                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |
| 13                    | 13              | 13             | 13                    | 13                    | TextList  | \\Objekte\\SYSTEM_AlarmServices_...   |

3. Um identische Texte zu gruppieren und sie auf einmal zu übersetzen, klicken Sie in der Funktionsleiste auf die Schaltfläche .
4. Um Texte mit fehlenden Übersetzungen auszublenden, klicken Sie in der Funktionsleiste auf die Schaltfläche .
5. Klicken Sie in eine leere Spalte und geben Sie die Übersetzung ein.

## Ergebnis

Sie haben einzelne Texte im Editor "Projekttexte" übersetzt. In Runtime werden die die Texte in der jeweiligen Runtime-Sprache angezeigt.

## Siehe auch

Arbeiten mit mehreren Sprachen (Seite 3888)

Projekttexte exportieren (Seite 3893)

Grundlagen zu Projekttexten (Seite 3889)

Projekttexte importieren (Seite 3895)

## Texte mithilfe von Referenztexten übersetzen

### Einführung

Nach dem Wechseln der Editiersprache werden alle Texte in Eingabefeldern in der neuen Editiersprache angezeigt. Sind für diese Sprache noch keine Übersetzungen vorhanden, sind die Eingabefelder leer oder mit den Standardwerten befüllt.

Wenn Sie in ein entsprechendes Eingabefeld erneut Text eingeben, wird dieser in der aktuellen Editiersprache gespeichert. Im Anschluss existieren für dieses Eingabefeld Texte in zwei Projektsprachen - in der ehemaligen Editiersprache und in der aktuellen Editiersprache. Auf diese Weise ist es möglich, Texte in mehreren Projektsprachen anzulegen.

Sie können sich bei einem Eingabefeld bereits bestehende Übersetzungen in anderen Projektsprachen anzeigen lassen. Diese dienen als Vergleich für die Texteingabe in der aktuellen Editiersprache und werden als Referenzsprache bezeichnet.

### Voraussetzung

Für ein Eingabefeld existiert mindestens eine Übersetzung in einer weiteren Projektsprache.

### Vorgehen

Um sich für ein Eingabefeld die Übersetzung in einer Referenzsprache anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Task Card "Aufgaben > Sprachen & Ressourcen".
2. Wählen Sie aus der Klappliste "Referenzsprache" eine Referenzsprache aus.

### Ergebnis

Die Referenzsprache ist voreingestellt. Wenn Sie in ein Textfeld klicken, werden in der Task Card "Aufgaben > Referenztext" Übersetzungen angezeigt, die bereits in anderen Projektsprachen vorhanden sind.

## Siehe auch

- Arbeiten mit mehreren Sprachen (Seite 3888)
- Projekttexte exportieren (Seite 3893)
- Grundlagen zu Projekttexten (Seite 3889)
- Projekttexte importieren (Seite 3895)

## Projekttexte exportieren

Sie exportieren Projekttexte zur Übersetzung. Der Export erfolgt in eine Office Open XML Datei mit der Endung ".xlsx". Diese kann z. B. in Microsoft Excel bearbeitet werden.

Sie können die Datei mit den Übersetzern austauschen und nach der Übersetzung direkt in das Projekt wieder importieren.

## Voraussetzungen

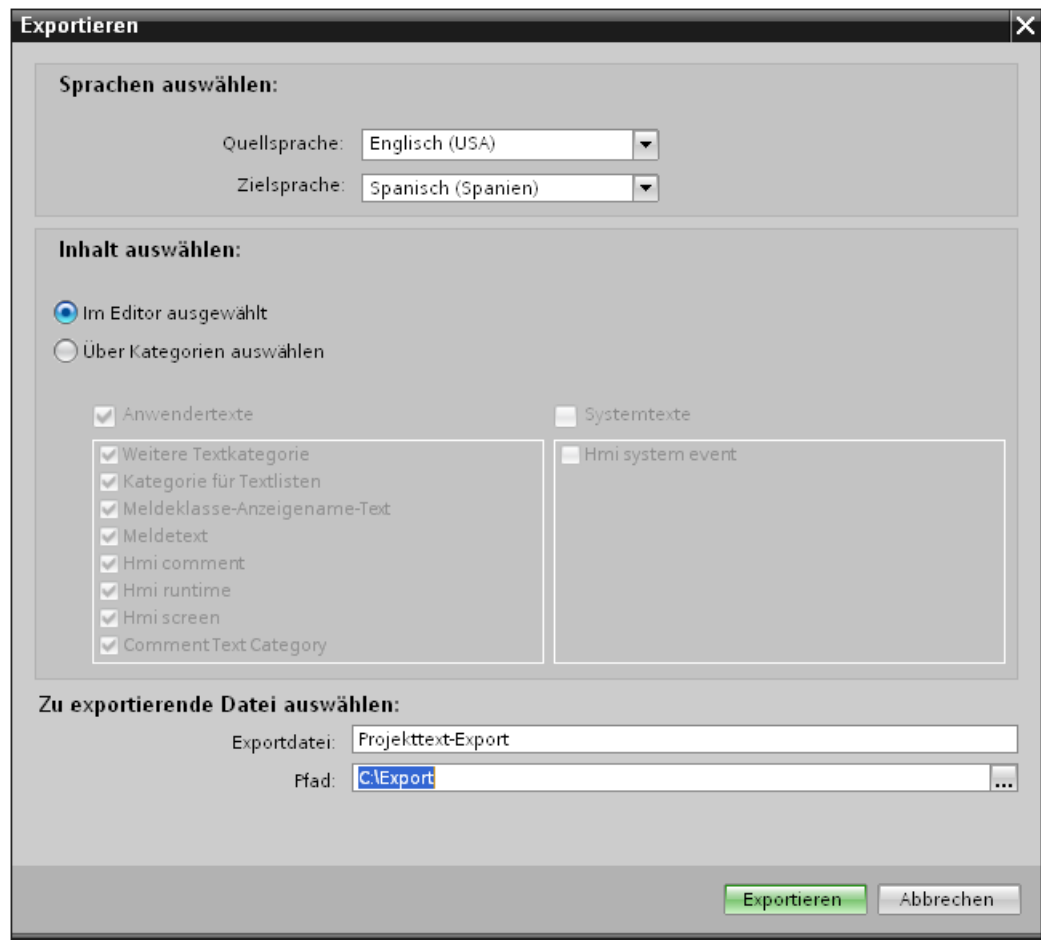
- Im Editor "Projektsprachen" sind mindestens zwei Sprachen aktiviert, z.B. Italienisch und Französisch.

## Projekttexte exportieren

Um einzelne Projekttexte zu exportieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die untergeordneten Elemente werden angezeigt.
2. Doppelklicken Sie auf "Projekttexte". Der Editor "Projekttexte" wird geöffnet.
3. Selektieren Sie die Texte, die Sie exportieren möchten.

4. Klicken Sie auf die Schaltfläche . Der Dialog "Export" wird geöffnet.



5. Wählen Sie aus der Klappliste "Quellsprache" die Sprache, aus der Sie die Übersetzung vornehmen möchten, z. B. Italienisch.
6. Wählen Sie aus der Klappliste "Zielsprache" die Sprache, in die Sie die Texte übersetzen möchten, z. B. Französisch.
7. Geben Sie im Eingabefeld "Exportdatei" einen Speicherpfad und einen Dateinamen für die Exportdatei an.
8. Klicken Sie auf "Export".

## Ergebnis

Die im Editor "Projekttexte" selektierten Texte wurden in eine xlsx-Datei geschrieben. Die xlsx-Datei wird im angegebenen Ordner abgelegt.

Alternativ können Sie auch alle Projekttexte über Kategorien auswählen und exportieren. Aktivieren Sie im Dialog "Export" die Optionen "Anwendertexte" oder "Systemtexte", abhängig

davon, welche Texte Sie exportieren möchten. Der Export kann in diesem Fall zusätzlich nach Kategorien eingegrenzt werden.

---

#### Hinweis

Projekttexte in Bibliotheksobjekten können nicht exportiert werden.

---

#### Siehe auch

- Arbeiten mit mehreren Sprachen (Seite 3888)
- Texte mithilfe von Referenztexten übersetzen (Seite 3892)
- Texte direkt übersetzen (Seite 3890)
- Grundlagen zu Projekttexten (Seite 3889)
- Projekttexte importieren (Seite 3895)

#### Projekttexte importieren


Sie bearbeiten die xlsx-Datei oder geben Sie an einen Übersetzer weiter. Nach abgeschlossener Übersetzung importieren Sie die übersetzten Texte. Die Fremdsprachen werden im Projekt an das entsprechende Objekt importiert.

#### Voraussetzungen

- Im Editor "Projektsprachen" sind mindestens zwei Sprachen aktiviert, z.B. Italienisch und Französisch.

#### Projekttexte importieren

Um eine Datei mit Projekttexten zu importieren, gehen Sie folgendermaßen vor:

- Klicken Sie in der Projektnavigation auf das Pfeilsymbol links von "Sprachen & Ressourcen".  
Die untergeordneten Elemente werden angezeigt.
- Doppelklicken Sie auf "Projekttexte". Der Editor "Projekttexte" wird geöffnet.
- Klicken Sie auf die Schaltfläche . Der Dialog "Import" wird geöffnet.
- Wählen Sie aus dem Feld "Importdatei" den Pfad und den Dateinamen der Importdatei aus.
- Aktivieren Sie das Optionskästchen "Quellsprache importieren", wenn Sie in der Exportdatei Änderungen an der Quellsprache vorgenommen haben und die Einträge im Projekt mit den Änderungen überschreiben möchten.
- Klicken Sie auf "Importieren".

#### Ergebnis

Sie haben die Projekttexte importiert.

## Siehe auch

- Projekttexte exportieren (Seite 3893)
- Arbeiten mit mehreren Sprachen (Seite 3888)
- Grundlagen zu Projekttexten (Seite 3889)
- Texte direkt übersetzen (Seite 3890)
- Texte mithilfe von Referenztexten übersetzen (Seite 3892)

### 10.9.5.6 Sprachabhängigen Grafiken einsetzen

## Editor "Grafiken"

### Einleitung

Im Editor "Grafiken" verwalten Sie die projektierten Grafikobjekte in mehreren Sprachversionen. Mehrsprachige Projekte erfordern teilweise auch sprachabhängige Varianten der Grafiken, z. B. wenn:

- Die Grafiken Text enthalten,
- In den Grafiken kulturelle Gegebenheiten eine Rolle spielen.

### Öffnen des Editors "Grafiken"

Doppelklicken Sie in der Projektnavigation auf "Sprachen und Ressourcen > Grafiken".

### Arbeitsbereich

Der Arbeitsbereich zeigt Ihnen alle projektierten Grafikobjekte in einer Tabelle. Für jede Projektsprache ist eine eigene Tabellenspalte angelegt. Die Tabellenspalten enthalten die Grafikvarianten einer Sprache.

Zusätzlich können Sie für jede Grafik eine Standard-Grafik festlegen, die immer dann angezeigt wird, wenn es für eine Projektsprache keine sprachspezifische Grafik gibt.

### Vorschau

In der Vorschau lassen Sie sich die Darstellungen der Grafiken auf verschiedenen Bediengeräten in einer Vorschau anzeigen.

## Siehe auch

- Externe Grafik in der Grafiksammlung hinterlegen (Seite 3898)
- Grafik in der Grafiksammlung hinterlegen (Seite 3897)
- Sprachbegriffe in WinCC (Seite 3882)

## Grafik in der Grafiksammlung hinterlegen

### Einleitung

Mit dem Editor "Grafiken" importieren Sie Grafiken, die Sie in Bildobjekten im Editor "Bilder" verwenden. Zusätzlich verwalten Sie die für sprachabhängige Grafikvarianten. Eine Vorschau zeigt Ihnen die Darstellungen einer Grafik auf verschiedenen Bediengeräten an.

### Voraussetzung

- Die sprachabhängigen Varianten einer Grafik stehen zur Verfügung.
- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Der Editor "Grafiken" ist geöffnet.

### Grafik einfügen

1. Klicken Sie in der Tabelle "Grafiksammlung" auf "Hinzufügen". Ein Dialog wird geöffnet.
  2. Wählen Sie die gewünschte Grafikdatei.
  3. Klicken Sie im Dialogfenster auf "Öffnen".  
Die Grafik wird in das Projekt eingefügt und im Editor "Grafiken" in allen Zellen dieser Zeile angezeigt.
  4. Klicken Sie in die entsprechende Zelle einer Sprache, für die eine sprachabhängige Variante dieser Grafik existiert.
  5. Wählen Sie im Kontextmenü den Eintrag "Grafik einfügen". Ein Dialog wird geöffnet
  6. Wählen Sie die gewünschte Grafikdatei und klicken Sie auf "Öffnen".  
An Stelle der Grafik der Referenzsprache wird die sprachabhängige Variante in die Tabelle eingefügt.
  7. Fügen Sie abschließend in die Spalte "Standardgrafik" eine Grafik ein, die in Runtime für diejenigen Sprachen angezeigt wird, für die keine sprachspezifische Grafik vorhanden ist.
- Alternativ können Sie auch per Drag&Drop eine Grafik aus dem Windows Explorer, an die gewünschte Stelle in der Tabelle "Grafiksammlung" ziehen.

### Grafiken in der Bediengerätevorschau anzeigen

1. Klicken Sie in der Tabelle auf eine Grafik.
2. Wählen Sie im Inspektorfenster "Eigenschaften > Grafikeinstellungen > Vorschau Bediengerät" das gewünschte Bediengerät aus.  
Die Grafik wird Ihnen in der Vorschau so angezeigt, wie es in Runtime am ausgewählten Bediengerät dargestellt wird.

### Ergebnis

Die eingefügten Grafiken stehen im Editor "Grafiken" zur Verfügung. Beim Editieren wird die Grafik angezeigt, die der jeweiligen Editiersprache zugeordnet ist. In allen Editiersprachen, für die kein Bild importiert wurde, wird das Standardbild angezeigt.

In Runtime werden die der jeweiligen Runtime-Sprache zugeordneten Bilder angezeigt. In allen Runtime-Sprachen, für die kein Bild importiert wurde, wird das Standardbild angezeigt.

---

**Hinweis**

Wenn Sie eine Projektsprache deaktivieren, werden im aktuellen Projekt alle in dieser Sprache bereits erstellten Grafiken gelöscht.

---

**Siehe auch**

Editor "Grafiken" (Seite 3896)

**Externe Grafik in der Grafiksammlung hinterlegen**

**Einleitung**

Um in einem externen Grafikprogramm erstellte Grafiken in Ihren Bildern anzuzeigen, hinterlegen Sie diese vorher in der Grafiksammlung des WinCC Projekts.

**Voraussetzung**

- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Der Editor "Grafiken" ist geöffnet.
- Eine Grafik ist im Editor "Grafiken" vorhanden.

**Neue Grafik als OLE-Objekt erstellen und einfügen**

1. Klicken Sie in der Tabelle "Grafiksammlung" auf "Hinzufügen". Ein Dialog wird geöffnet.
2. Navigieren Sie zum Ordner, in dem die Grafik gespeichert ist.
3. Klicken Sie im Dialogfenster auf "Öffnen".  
Die Grafik wird in das Projekt eingefügt und im Editor "Grafiken" in allen Zellen dieser Zeile angezeigt.
4. Klicken Sie in die entsprechende Zelle einer Sprache, für die eine sprachabhängige Variante dieser Grafik existiert.
5. Wählen Sie im Kontextmenü den Eintrag "Objekt einfügen". Der Dialog "Objekt einfügen" wird geöffnet.

---

**Hinweis**

Zusätzlich wird der Dialog "Externe Applikation läuft..." geöffnet. Der Dialog wird erst geschlossen, wenn die externe Applikation beendet wurde.

---

6. Wählen Sie im Dialog "Objekt einfügen > Neu erstellen" und einen Objekttyp.



7. Klicken Sie auf "OK". Das dazugehörige Grafikprogramm wird geöffnet.
8. Wenn Sie die Grafik erstellt haben, schließen Sie das Grafikprogramm. Die Grafik wird im Standardformat des Grafikprogramms abgespeichert und in der Grafiksammlung angezeigt.

### Erstellte Grafik in WinCC einfügen

1. Klicken Sie in die entsprechende Zelle einer Sprache, für die eine sprachabhängige Variante dieser Grafik existiert.
2. Wählen Sie im Kontextmenü den Eintrag "Objekt einfügen". Der Dialog "Objekt einfügen" wird geöffnet.

---

**Hinweis**

Zusätzlich wird der Dialog "Externe Applikation läuft..." geöffnet. Der Dialog wird erst geschlossen, wenn die externe Applikation beendet wurde.

---

3. Wählen Sie im Dialog "Objekt einfügen" die Option "Aus Datei erstellen".
4. Klicken Sie auf die Schaltfläche "Durchsuchen".
5. Navigieren Sie zu der erstellten Grafik und wählen Sie diese aus.

---

**Hinweis**

Um Grafikdateien zu importieren, beachten Sie folgende Größenbeschränkungen:

\*.bmp, \*.tif, \*.emf, \*.wmf ≤4 MByte

\*.jpg, \*.jpeg, \*.ico, \*.gif "\*"≤1 MByte

---

### Ergebnis

Die eingefügten OLE-Objekt stehen im Editor "Grafiken" zur Verfügung.

Im Editor "Bilder" wird die Grafikvariante der aktuell eingestellten Editiersprache angezeigt. In allen Editiersprachen, für die kein Bild importiert wurde, wird die Standardgrafik angezeigt.

In Runtime wird jeweils die Grafikvariante der aktuellen Runtime-Sprache angezeigt. In allen Runtime-Sprachen, für die keine Grafik eingefügt wurde, wird die Standardgrafik angezeigt.

Die in der Grafiksammlung hinterlegten OLE-Objekte öffnen Sie durch Doppelklicken mit dem dazugehörigen Grafikprogramm, um sie zu bearbeiten.

### Siehe auch

Editor "Grafiken" (Seite 3896)

## 10.9.5.7 Sprachen in Runtime

### Sprachen in Runtime

#### Mehrere Runtime-Sprachen einsetzen

Sie bestimmen, welche Projektsprachen in Runtime auf einem bestimmten Bediengerät zum Einsatz kommen. Die Anzahl der Runtime-Sprachen, die gleichzeitig auf dem Bediengerät zur Verfügung stehen, ist geräteabhängig. Um dem Bediener die Sprachumschaltung in Runtime zu ermöglichen, müssen Sie ein entsprechendes Bedienelement projektieren.

Bei Runtime-Start wird das Projekt in der zuletzt eingestellten Sprache angezeigt. Beim ersten Runtime-Start erscheint die Sprache mit der niedrigsten Nummer in "Reihenfolge für Sprachumschaltung".

#### Einstellung bei der Projektierung

Im Editor "Sprachen und Schriften" bestimmen Sie,

- welche der Projektsprachen als Runtime-Sprachen für den jeweiligen Bediengerätetyp zur Verfügung stehen.
- in welcher Reihenfolge die Sprachen bei einem Sprachwechsel umgeschaltet werden.

#### Siehe auch

Methoden der Sprachumschaltung (Seite 3900)

Runtime-Sprache aktivieren (Seite 3901)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 3903)

Standardschrift für eine Runtime-Sprache einstellen (Seite 3905)

Archivsprache wählen (Seite 3906)

Sprachbegriffe in WinCC (Seite 3882)

### Methoden der Sprachumschaltung

#### Einleitung

Damit auf dem Bediengerät mehrere Runtime-Sprachen zur Verfügung stehen, projektieren Sie eine Sprachumschaltung. Nur so ist es dem Bediener möglich, zwischen verschiedenen Runtime-Sprachen zu wählen.

## Methoden der Sprachumschaltung

Sie können folgende Methoden der Sprachumschaltung projektieren:

- **Direkte Sprachwahl**  
Jede Sprache wird über eine eigene Schaltfläche eingestellt. In diesem Fall erstellen Sie für jede Runtime-Sprache eine Schaltfläche.
- **Sprachumschaltung**  
Der Bediener schaltet die Sprachen über eine Schaltfläche um.

Die Beschriftungen der Schaltflächen müssen bei beiden Methoden in jede verwendete Sprache übersetzt werden. Sie können zusätzlich ein Ausgabefeld projektieren, das die gerade gewählte Sprache anzeigt.

## Siehe auch

Sprachen in Runtime (Seite 3900)

Archivsprache wählen (Seite 3906)

Runtime-Sprache aktivieren (Seite 3901)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 3903)

Standardschrift für eine Runtime-Sprache einstellen (Seite 3905)

## Runtime-Sprache aktivieren

### Einleitung

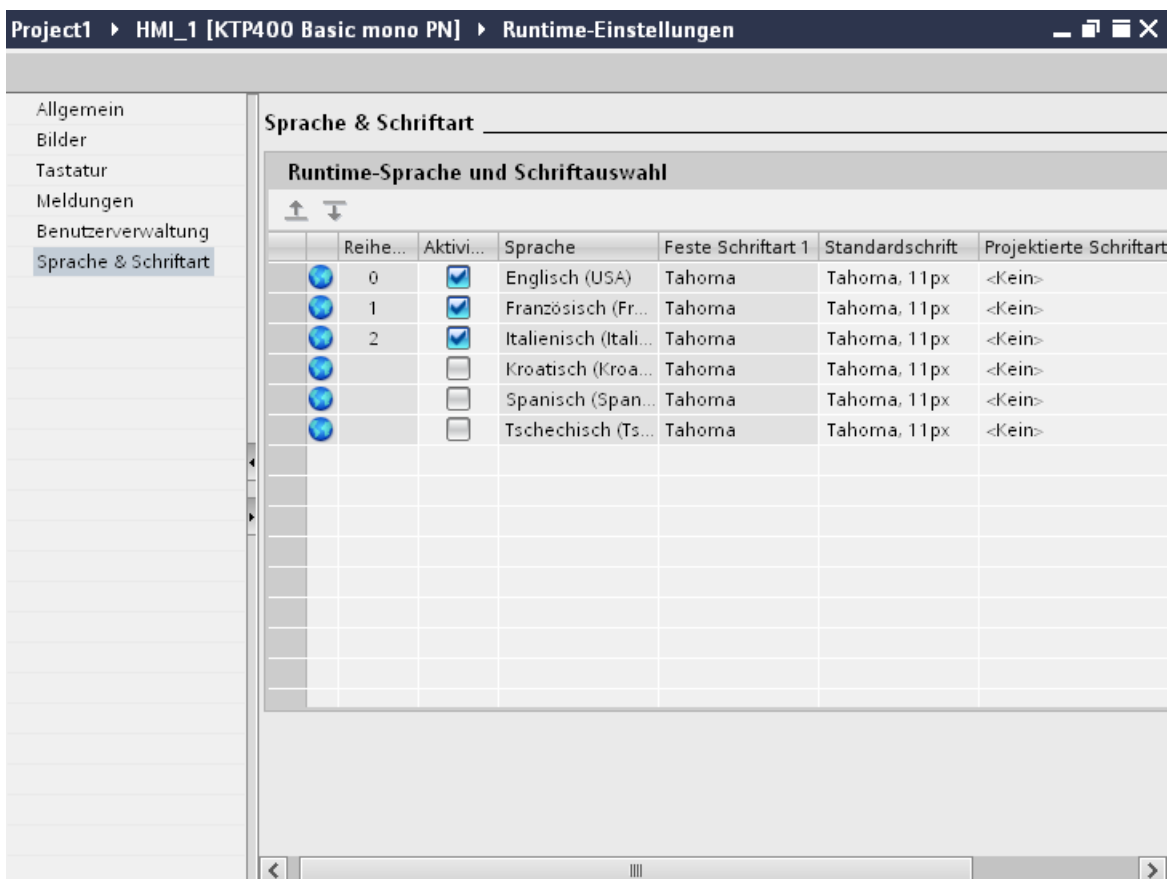
Der Editor "Sprache & Schriftart" zeigt Ihnen alle im Projekt verfügbaren Projektsprachen an. Sie wählen, welche Projektsprachen als Runtime-Sprachen auf dem Bediengerät zur Verfügung stehen sollen.

### Voraussetzungen

Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert..

## Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Klicken Sie auf "Sprache & Schriftart".
3. Aktivieren Sie die folgenden Sprachen:
  - Deutsch
  - Chinesisch
  - Französisch



## Ergebnis

Sie haben drei Runtime-Sprachen aktiviert. In der Spalte "Reihenfolge" wird für jede Sprache automatisch eine Nummer vergeben. Die aktivierten Runtime-Sprachen werden mit dem übersetzten Projekt auf das Bediengerät geladen.

Wenn Sie mehr Sprachen auswählen, als auf das Bediengerät übertragen werden können, werden Sie durch eine farbige Hinterlegung in der Tabelle darauf aufmerksam gemacht.

## **Siehe auch**

- Sprachen in Runtime (Seite 3900)
- Archivsprache wählen (Seite 3906)
- Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 3903)
- Methoden der Sprachumschaltung (Seite 3900)

## **Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen**


### **Einleitung**

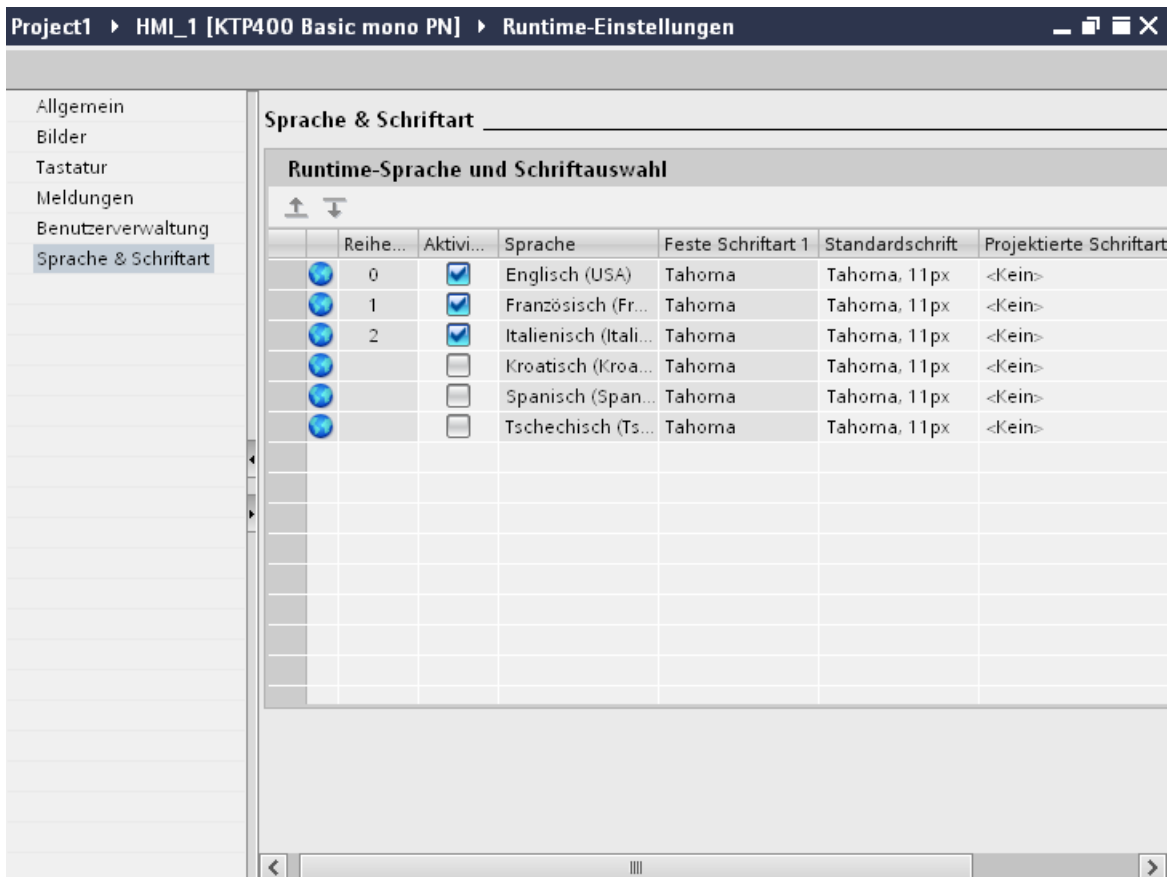
Sie legen fest, in welcher Reihenfolge die Runtime-Sprachen bei einem Sprachwechsel umgeschaltet werden. Beim ersten Runtime-Start erscheint das Projekt in der Sprache mit der niedrigsten Nummer in der Spalte "Reihenfolge".

### **Voraussetzungen**

- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Im Editor "Sprache & Schriftart" ist geöffnet und drei Runtime-Sprachen sind aktiviert in folgender Reihenfolge:
  1. Deutsch
  2. Chinesisch
  3. Französisch

## Vorgehen

1. Selektieren Sie die Runtime-Sprache "Deutsch".
2. Klicken Sie auf das Schaltfläche . Die Runtime-Sprache "Deutsch" wird nach unten verschoben. In der Spalte "Reihenfolge" wird die Nummer automatisch auf "2" geändert.



## Ergebnis

Sie haben die Reihenfolge der Runtime-Sprachen geändert. Beim ersten Start von Runtime erscheint das Projekt in der Sprache mit der niedrigsten Nummer, also Chinesisch. Bei einer Sprachumschaltung wechselt die Sprache in der Reihenfolge der vergebenen Nummern.

## Siehe auch

- Sprachen in Runtime (Seite 3900)
- Archivsprache wählen (Seite 3906)
- Runtime-Sprache aktivieren (Seite 3901)
- Standardschrift für eine Runtime-Sprache einstellen (Seite 3905)
- Methoden der Sprachumschaltung (Seite 3900)

## Standardschrift für eine Runtime-Sprache einstellen

### Einleitung

Im Editor "Sprache & Schriftart" legen Sie für jede Runtime-Sprache fest, in welcher Schriftart die Texte auf dem Bediengerät angezeigt werden. Die Standardschriftart wird für alle Texte verwendet, bei denen Sie keine Schriftart festlegen können, z. B. Dialogtexte.

WinCC bietet hierbei nur die Schriftarten zur Auswahl an, die vom Bediengerät unterstützt werden.

### Voraussetzungen

- Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.
- Im Editor "Sprache & Schriftart" sind drei Runtime-Sprachen aktiviert.
  1. Chinesisch
  2. Deutsch
  3. Französisch

### Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Klicken Sie auf "Sprache & Schriftart". Die Tabelle zeigt die aktivierten Runtime-Sprachen und die Schriftarten.
3. Klicken Sie in der Zeile für "Französisch" auf die Spalte "Standardschrift".
4. Wählen Sie die Schriftart aus, die standardmäßig verwendet wird, wenn Sie für einen Text keine Schriftart wählen können.

### Ergebnis

Die Projekttexte der Runtime-Sprache "Französisch" werden auf dem Bediengerät in der gewählten Schriftart angezeigt.

Diese Schriftarten werden beim Transfer zusätzlich zum Bediengerät übertragen.

Die Standardschriftart wird auch für die Darstellung der Dialoge aus dem Betriebssystem des Bediengeräts verwendet. Wenn Dialogtexte oder Dialogüberschriften nicht vollständig dargestellt werden, wählen Sie eine kleinere Schriftgröße für die Standardschriftart.

### Siehe auch

Sprachen in Runtime (Seite 3900)

Archivsprache wählen (Seite 3906)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 3903)

Methoden der Sprachumschaltung (Seite 3900)

## Archivsprache wählen

### Einleitung

Im Editor "Runtime-Einstellungen > Allgemein" wählen Sie die Sprache, die verwendet wird, um Archive in Runtime zu schreiben.

### Voraussetzungen

- Im Editor "Projektsprachen" sind die Sprachen aktiviert, die Sie in Ihrem Projekt verwenden z. B. "Deutsch" und "Englisch".

### Vorgehen

1. Doppelklicken Sie in der Projektnavigation auf "Runtime-Einstellungen".
2. Klicken Sie auf "Sprache & Schriftart".
3. Aktivieren Sie die Runtime-Sprachen z. B. "Deutsch" und "Englisch".
4. Legen Sie die "Reihenfolge" fest:
  - 1 Deutsch
  - 2 Englisch
5. Klicken Sie "Runtime-Einstellungen > Allgemein".
6. Wählen Sie für "Archive > Archivierungssprache" die Sprache "Deutsch".

### Ergebnis

Nach dem Laden startet das Projekt in der Runtime-Sprache "Deutsch". Die Archive werden in Deutsch geschrieben. Während Runtime schaltet der Bediener die Runtime-Sprache auf Englisch. Die Archive werden jedoch weiterhin in Deutsch geschrieben.

### Siehe auch

Sprachen in Runtime (Seite 3900)

Standardschrift für eine Runtime-Sprache einstellen (Seite 3905)

Reihenfolge der Runtime-Sprache bei Sprachumschaltung festlegen (Seite 3903)

Methoden der Sprachumschaltung (Seite 3900)

Runtime-Sprache aktivieren (Seite 3901)



### 10.9.5.8 Beispiel zur mehrsprachigen Projektierung

#### Beispiel: Schaltfläche zur Sprachumschaltung projektieren

##### Einleitung

In diesem Beispiel projektieren Sie eine Schaltfläche, mit der Sie in Runtime zwischen mehreren Runtime-Sprachen umschalten.

##### Voraussetzungen

- Sie haben das Beispiel "Mehrsprachige Beschriftung einer Schaltfläche projektieren" ausgeführt.
- Das Bild "Screen\_1" ist geöffnet.
- Die Schaltfläche ist im Bild selektiert.

##### Vorgehen

1. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Drücken".
2. Klicken Sie in der Tabelle auf "Funktion hinzufügen".
3. Wählen Sie die Systemfunktion "SetzeSprache".

##### Ergebnis

Sie haben der Schaltfläche die Funktion "SetzeSprache" zugewiesen. Wenn Sie in Runtime die Schaltfläche drücken, dann schaltet die Runtime-Sprache um. Die Reihenfolge der Runtime-Sprachen wird so umgesetzt, wie sie im Editor "Sprache & Schriftart" durch die Nummerierung der Sprachen vorgegeben ist.

##### Siehe auch

Beispiel: Mehrsprachige Beschriftung einer Schaltfläche projektieren (Seite 3907)

Beispiel: Je Runtime-Sprache eine Schaltfläche zur Sprachumschaltung projektieren (Seite 3908)

Sprachbegriffe in WinCC (Seite 3882)

#### Beispiel: Mehrsprachige Beschriftung einer Schaltfläche projektieren

##### Einleitung

In diesem Beispiel projektieren Sie eine Schaltfläche, die in Deutsch mit "Sprache umschalten" und in Englisch mit "Switch Language" beschriftet ist.

## Voraussetzungen

- Im Editor "Projektsprachen" sind die Sprachen "Deutsch" und "Englisch" aktiviert.
- Deutsch ist als Editiersprache und Referenzsprache gewählt.
- Das Bild "Screen\_1" ist angelegt und geöffnet.
- Das Inspektorfenster ist geöffnet.

## Vorgehen

1. Ziehen Sie eine Schaltfläche per Drag&Drop aus der Task Card "Werkzeuge" in das Bild. Die Schaltfläche wird im Bild eingefügt.
2. Öffnen Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein".
3. Geben Sie unter "Text > Aus" die Beschriftung "Sprache umschalten" ein.
4. Bestätigen Sie die Eingabe mit <Return>. Die Schaltfläche ist beschriftet.
5. Öffnen Sie die Task Card "Aufgaben".
6. Wählen Sie "Sprachen & Ressourcen > Editiersprache" die Sprache "Englisch" aus.
7. Geben Sie im Inspektorfenster "Eigenschaften > Eigenschaften > Allgemein > Text > Aus" die Beschriftung "Switch Language" ein.

## Ergebnis

Die Beschriftung der Schaltfläche ist in deutscher und englischer Sprache projiziert. In Runtime wird die der aktuellen Runtime-Sprache entsprechende Beschriftung angezeigt.

## Siehe auch

Beispiel: Schaltfläche zur Sprachumschaltung projektieren (Seite 3907)

Beispiel: Je Runtime-Sprache eine Schaltfläche zur Sprachumschaltung projektieren (Seite 3908)

## Beispiel: Je Runtime-Sprache eine Schaltfläche zur Sprachumschaltung projektieren


### Einleitung

In diesem Beispiel projektieren Sie eine Schaltfläche, die in Deutsch mit "Sprache umschalten" und in Englisch mit "Switch Language" beschriftet ist.

## Voraussetzungen

- Im Editor "Projektsprachen" sind folgende Sprachen aktiviert
  - Deutsch
  - Englisch
  - Italienisch
- Alle Sprachen sind im Editor "Runtime-Einstellungen > Sprache & Schriftart" als Runtime-Sprache aktiviert.
- Das Bild "Screen\_1" ist angelegt und geöffnet.
- Im Bild sind drei Schaltflächen angelegt:
  - Schaltfläche\_1 mit der Beschriftung "Deutsch"
  - Schaltfläche\_2 mit der Beschriftung "English"
  - Schaltfläche\_3 mit der Beschriftung "Italiano"
- Das Inspektorfenster ist geöffnet.

## Vorgehen

1. Selektieren Sie "Schaltfläche\_1".
2. Klicken Sie im Inspektorfenster "Eigenschaften > Ereignisse > Drücken".
3. Klicken Sie in der Tabelle auf "<Funktion hinzufügen>".
4. Wählen Sie die Systemfunktion "SetzeSprache".
5. Klicken Sie auf das Feld "Umschalten".
6. Klicken Sie auf die Schaltfläche .
7. Wählen Sie "Runtime-Sprache". Das Feld wird rot markiert.
8. Wählen Sie in der Aufklappliste "Deutsch".
9. Wiederholen Sie die Schritte 1 - 8 für die anderen beiden Schaltflächen und wählen Sie die entsprechende Runtime-Sprache aus.

## Ergebnis

Sie haben drei Schaltflächen für die Sprachumschaltung in Runtime projiziert. Jede Schaltfläche schaltet in eine andere Runtime-Sprache um. Wenn Sie in Runtime z. B. auf die Schaltfläche "English" klicken, dann schalten Sie Runtime-Sprache auf Englisch um.

## Siehe auch

Beispiel: Schaltfläche zur Sprachumschaltung projizieren (Seite 3907)

Beispiel: Mehrsprachige Beschriftung einer Schaltfläche projizieren (Seite 3907)

## 10.9.6 Gerät tauschen

### 10.9.6.1 Grundlagen

#### Einleitung

Mit dem Tausch der Geräte können Sie bereits vorhandene Projektierungen für Ihre neuen Geräte verwenden und mit geringem manuellen Aufwand optimieren.

Alle von Ihnen projektierten Daten bleiben in den Projektierungsdaten enthalten. Daher ist es nicht erforderlich, einzelne Objekte eines Geräts zu kopieren und in ein anderes Gerät einzufügen.

#### Prinzip

Beim Tausch eines Geräts gelten folgende Grundsätze:

- Nur vom neuen Gerät unterstützte Funktionen sind verfügbar. Nur Projektierungsdaten, die vom neuen Gerät unterstützt werden, werden angezeigt.  
Betroffen sind z. B.
  - Rezepturen,
  - verfügbare Objekte in Bildern,
  - verfügbare Systemfunktionen,
  - verfügbare Kommunikationsprotokolle.
- Auf dem neuen Gerät kann die Anzahl der unterstützten Objekte z. B. Bilder oder Variablen begrenzt sein. Wenn die vorhandenen Objekte die Beschränkungen auf dem neuen Gerät überschreiten, werden die Objekte vollständig angezeigt. Die Objekte werden allerdings in den einzelnen Editoren farblich gekennzeichnet. Beim Übersetzen der Projektierungsdaten wird ein Fehler ausgegeben.  
Daher ist es erforderlich, dass Sie bei einem Tausch auf ein Gerät mit geringerem Leistungsumfang manuell nachbearbeiten.  
Beispiel: Mengenbeschränkung bei Verbindungen  
Wenn auf dem neuen Gerät weniger Verbindungen unterstützt werden als projektiert wurden, werden alle Verbindungen farbig als ungültig markiert. Löschen Sie überzählige Verbindungen.

---

#### Hinweis

Wenn Sie ein Gerät z. B. ein Panel tauschen und als neues Gerät eine PC Station wählen, wird in der Projektnavigation WinCC Runtime Advanced automatisch unter die PC Station verschoben.

---

## Siehe auch

Geräteabhängige Funktionen (Seite 3911)  
Möglichkeiten der Bildanpassung (Seite 3914)  
Tastenbelegung beim Tausch eines Geräts (Seite 3912)  
Engineering System (Seite 4019)  
Basic Panel (Seite 4020)

### 10.9.6.2 Geräteabhängige Funktionen

#### Geräteabhängige Funktionen

##### Vom Gerät abhängige Funktionen

Vom Gerät abhängige Funktionen werden folgendermaßen umgesetzt:

- **Farben**  
Wenn Sie von einem Gerät mit Farbanzeige gegen ein Gerät tauschen, das einen geringeren Farbumfang besitzt, wird die Farbe automatisch geändert.  
Wenn Sie die Farbe manuell ändern und wieder zu einem Gerät mit größerem Farbumfang zurückwechseln, bleibt der reduzierte Farbumfang erhalten.
- **Schriftarten**  
Wenn eine projektierte Schriftart auf einem Gerät nicht vorhanden ist, wird sie durch eine ähnliche Schriftart oder die projektierte Standardschriftart ersetzt. Die Standardschriftart hängt vom ausgewählten Gerät ab.
- **Zeichensätze mit unterschiedlichen Schriftgrößen**  
Vermeiden Sie beim Projektieren für folgende Geräte die Verwendung von vielen unterschiedlichen Schriftgrößen:
  - OP 73
  - OP 77A
  - TP 177A

Für jede Schriftgröße wird ein Zeichensatz auf das Gerät geladen. Beim Übersetzen überprüfen Sie im Inspektorfenster, wie viel Speicher im Gerät durch Zeichensätze belegt wird.

- **Schriftgröße**  
Um den Text auf Geräten anzuzeigen, verwenden Sie kleine Windows-Schriftarten. Wenn Sie große Windows-Schriftart verwenden, dann wird, abhängig von der Größe des Displays, der Text nicht vollständig angezeigt.  
Wenn Sie für die Geräte OP 77A und TP 177A Schriftgrößen größer oder gleich 28 Pixel verwenden, dann wirkt sich das auf die Performance aus.  
In asiatische Sprachen ist der Zeichenumfang deutlich höher. Daher wirkt sich die Verwendung unterschiedlicher Schriftgrößen massiv auf den Speicherbedarf auf allen Geräten aus.  
Für eine performante Projektierung verwenden Sie für alle großen Schriften im gesamten Projekt die gleiche Schriftart.
- **Bilder und Bildobjekte**  
Wenn beim Tausch eines Geräts, das neue Gerät eine andere Auflösung unterstützt als das vorherige Gerät, haben Sie mehrere Möglichkeiten um die Bilder anzupassen.  
Im Menü "Extras > Einstellungen > Visualisierung > Bilder Größenanpassung" passen Sie die Größe der Bilder an das neue Gerät an.

### Siehe auch

Grundlagen (Seite 3910)

### Tastenbelegung beim Tausch eines Geräts

#### Einleitung

Da die verfügbaren Geräte mit unterschiedlichen Funktionstasten ausgestattet sind, werden die darauf projizierten Funktionen bei dem Tausch des Gerätes auf die verfügbaren Funktionstasten des neuen Geräts abgebildet.

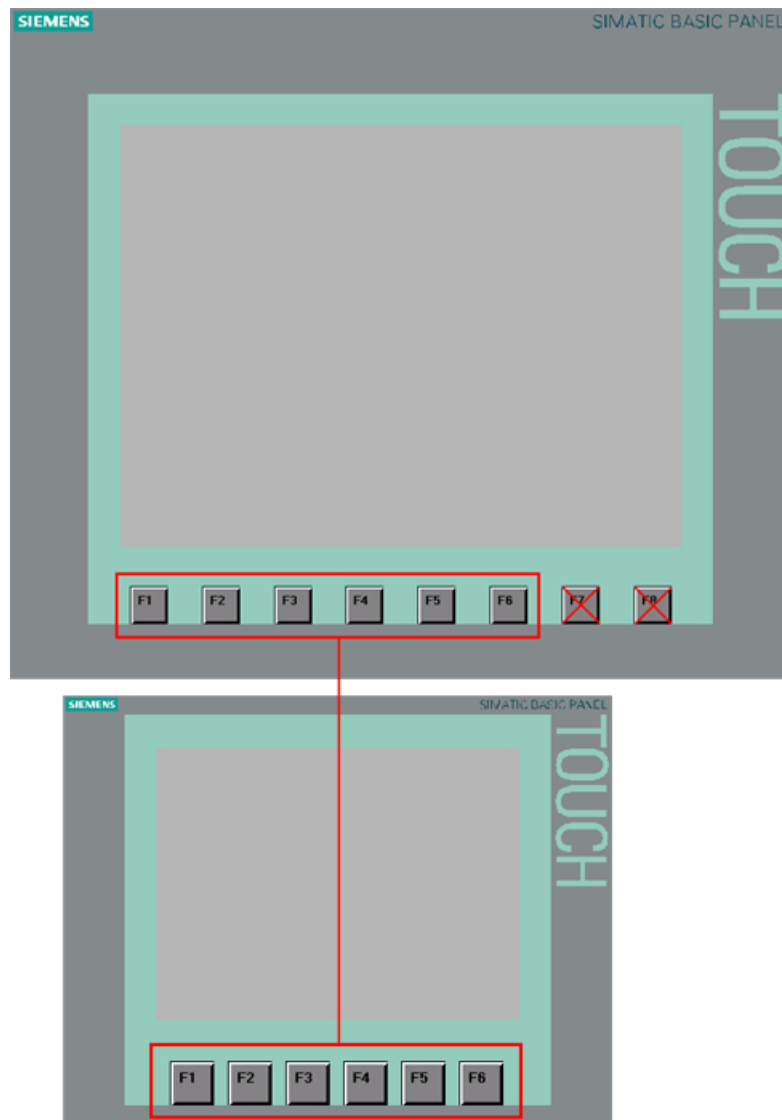
#### Abbildung der Funktionstasten

Die Funktionstasten unterhalb des Displays werden von links beginnend nach rechts auf das neue Gerät abgebildet. Hat das neue Gerät weniger Tasten, so werden die fehlenden Tasten nicht abgebildet.

#### Beispiel: Tausch KTP1000 Basic gegen KTP600 Basic

Sie haben am KTP1000 Basic auf die Taste F2 eine Funktion projiziert. Die Funktion wird nach dem Tausch auf ein KTP600 Basic mit der Taste F2 ausgelöst.

Haben Sie bei einem KTP1000 Basic die Taste F7 verwendet, steht diese Funktion nach dem Tausch auf ein KTP600 Basic nicht mehr zur Verfügung.



### Abbildung der Steuertasten und Cursortasten

Die folgenden Tasten werden nur auf die gleichen Tasten des neuen Geräts abgebildet:

- HELP
- ESC
- ACK
- ENTER
- PAGE UP
- PAGE DOWN
- CURSOR UP
- CURSOR DOWN

## Siehe auch

Grundlagen (Seite 3910)

Möglichkeiten der Bildanpassung (Seite 3914)

### 10.9.6.3 Bilder an das neue Gerät anpassen

## Möglichkeiten der Bildanpassung

### Einleitung

Vor dem Tausch eines Geräts wählen Sie die Größenanpassung der Bilder. Die Größenanpassung ist vor allem bei einem Tausch von Geräten mit verschiedenen Auflösungen des Displays von Bedeutung.

Bei Objekten wie Grafikanzeigen oder Textfeldern kann die Objektanpassung an den Inhalt unterbunden werden.

---

#### Hinweis

Wenn Sie ein Gerät mit einem Display im Querformat gegen ein Gerät mit Hochformat tauschen, werden die Objekte verzerrt dargestellt. Durch das veränderte Displayformat werden z. B. die Beschriftungen der Objekte abgeschnitten oder der Inhalt nicht an das Objekt angepasst. Passen Sie daher nach dem Tausch des Geräts die Bilder auf dem neuen Gerät an.

---

## Bildanpassung beim Tausch eines Geräts

Im Menü "Extras > Einstellungen > Visualisierung > Bilder Größenanpassung" passen Sie die Größe der Bilder an das neue Gerät an.

Anpassungen Bilder und Bildobjekte

**Anpassungen Bilder und Bildobjekte**

Keine

Breite und Höhe an neues Bild anpassen

Höhe an neues Gerät anpassen

Breite an neues Gerät anpassen

Freie Skalierung

Seitenverhältnis beibehalten

X:

Y:



Wählen Sie eine der folgenden Einstellungen.

### **Keine (Standardeinstellung)**

Die Bilder werden nicht skaliert. Die Objekte behalten ihre Position und Größe im Bild. Verwenden Sie diese Einstellung als ersten Test zum Prüfen eines möglichen Tauschergebnisses, da beim Hin- und Rücktausch keine Rundungsverluste auftreten.

Wenn das Display des neuen Geräts kleiner ist, kann diese Option dazu führen, dass sich Objekte außerhalb des projektierbaren Bereichs befinden.

### **Breite und Höhe an neues Gerät anpassen**

Position und Objektgröße werden an die neue Displaygröße angepasst. Die Anpassung findet auf der x- und y-Achse statt. Grafiken und Schriftgrößen werden entsprechend angepasst.

### **Höhe an neues Gerät anpassen**

Das Seitenverhältnis wird beibehalten und die Bilder an die Höhe des neuen Geräts angepasst.

Verwenden Sie diese Option, wenn Sie z. B. ein Gerät mit Displayformat 4:3 gegen ein Gerät mit Widescreen tauschen.

### **Breite an neues Gerät anpassen**

Das Seitenverhältnis wird beibehalten und die Bilder an die Breite des neuen Geräts angepasst.

Verwenden Sie diese Option, wenn Sie z. B. ein Gerät mit Widescreen gegen ein Gerät mit Displayformat 4:3 tauschen.

### **Freie Skalierung**

Sie bestimmen einen freien Skalierungsfaktor für die Bildanpassung. Sie können sowohl einen Faktor für die x-Achse als auch für die y-Achse bestimmen.

Wenn Sie bei der freien Skalierung einen Faktor  $< 1$  verwenden, kann es zu Verzerrungen der Objekte führen. Z. B. wird die Beschriftung der Objekte abgeschnitten oder der Inhalt nicht an das Objekt angepasst.

Passen Sie daher nach dem Tausch des Geräts die Bilder auf dem neuen Gerät an.

---

#### **Hinweis**

Bei Objekten mit einem festen Seitenverhältnis z. B. Zeigerinstrument, Kreis, findet keine Anpassung des Seitenverhältnisses statt. Die Objekte werden auf dem neuen Gerät mit gleichem Seitenverhältnis dargestellt, wie vor dem Tausch.

---

## Siehe auch

- Position der Bildobjekte festlegen (Seite 3919)
- Objekte an Inhalt anpassen (Seite 3916)
- Grundlagen (Seite 3910)
- Tastenbelegung beim Tausch eines Geräts (Seite 3912)

## Objekte an Inhalt anpassen

### Einleitung

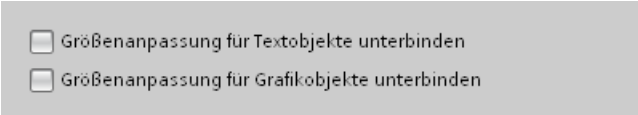
Für einige Objekte können Sie im Inspektorfenster festlegen, dass sich ihre Größe an den jeweiligen Inhalt anpasst, z. B.:

- Textfeld: Größe an den enthaltenen Text anpassen.
- E/A-Feld: Größe an den enthaltenen Text anpassen.
- Symbolisches E/A-Feld: Größe an den enthaltenen Text oder die Textliste anpassen.
- Grafikanzeige: Größe an die enthaltene Grafik anpassen.

### Größenanpassung für Textobjekte und Grafikobjekte

Im Menü "Extras > Einstellungen > Visualisierung > Anpassungen Bilder und Bildobjekte > Objekte an Inhalt anpassen" unterbinden Sie die automatische Größenanpassung der einzelnen Objekte. Damit werden die Objekte so skaliert, wie Sie es unter "Extras > Einstellungen > Visualisierung > Anpassungen Bilder und Bildobjekte" festgelegt haben.

Wählen Sie, für welche Objekte die automatische Größenanpassung unterbunden wird.

- 
- Größenanpassung für Textobjekte unterbinden
  - Größenanpassung für Grafikobjekte unterbinden

- Wenn "Größenanpassung für Textobjekte unterbinden" aktiviert ist, wird in den Eigenschaften der Textobjekte die automatische Größenanpassung ignoriert. Wenn Sie z. B. "Höhe an neues Gerät anpassen" aktiviert haben, wird das Textfeld ebenso wie die anderen Objekte, entsprechend der Höhe des neuen Geräts skaliert.
- Wenn "Größenanpassung für Grafikobjekte unterbinden" aktiviert ist, wird in den Eigenschaften der Grafikobjekte die automatische Größenanpassung ignoriert. Wenn Sie z. B. "Breite an neues Gerät anpassen" aktiviert haben, wird die Grafikanzeige ebenso wie die anderen Objekte, entsprechend der Breite des neuen Geräts skaliert.

---

#### Hinweis

Die Einstellungen haben keine Auswirkung auf Bildobjekte, deren Größe nicht geändert werden kann z. B. Meldeindikator oder Bildobjekte mit festem Seitenverhältnis.

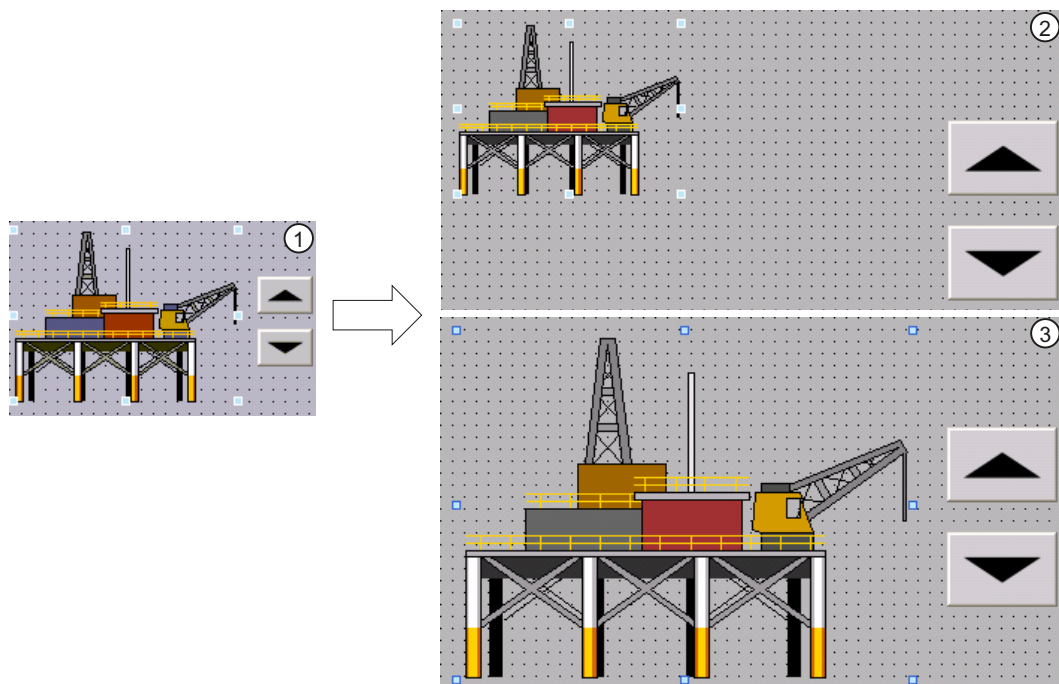
---

"Größenanpassung für Textobjekte unterbinden" und "Größenanpassung für Textobjekte unterbinden" haben keine Auswirkung, wenn:

- Sie "Anpassung Bilder und Bildobjekte > Keine" aktiviert haben.
- Sie "Breite und Höhe an neues Gerät anpassen" aktiviert haben und das neue Gerät hat die gleiche Auflösung wie das aktuelle Gerät.
- Sie "Höhe an neues Gerät anpassen" aktiviert haben und das neue Gerät hat die gleiche Auflösung wie das aktuelle Gerät.
- Sie "Breite an neues Gerät anpassen" aktiviert haben und das neue Gerät hat die gleiche Auflösung wie das aktuelle Gerät.

### Beispiel

Die folgende Abbildung zeigt die Auswirkungen der automatischen Größenanpassung am Beispiel eines Grafikobjekts, an dem zwei Schaltflächen ausgerichtet sind:



- ① Ausgangslage:
  - Zwei Schaltflächen sind an einem Grafikobjekt ausgerichtet.
  - In den Objekteigenschaften des Grafikobjekts ist unter "Darstellung > Größenanpassung" die Option "Objektgröße an Grafik anpassen" oder "Objektgröße an Grafik angleichen" aktiviert.
- ② Möglichkeit 1: Nach der Bediengerätumschaltung sollen die ursprünglichen Eigenschaften des Grafikobjekts beibehalten werden.
  - Deaktivieren Sie in den Einstellungen unter "Größenanpassung der Objekte" die Option "Größenanpassung für Grafikobjekte unterbinden".

Auswirkung: Das Grafikobjekt wird nach der Bediengerätumschaltung in seiner ursprünglichen Größe beibehalten. Die Ausrichtung zu den Schaltflächen geht verloren.
- ③ Möglichkeit 2: Nach der Bediengerätumschaltung soll das Grafikobjekt relativ zur neuen Bildschirmauflösung platziert werden.
  - Aktivieren Sie in den Einstellungen unter "Größenanpassung der Objekte" die Option "Größenanpassung für Grafikobjekte unterbinden".

In den Objekteigenschaften des Grafikobjekts wird die Option "Grafik an Objektgröße anpassen" automatisch aktiviert. Die beiden Schaltflächen sind auch nach der Bediengerätumschaltung korrekt am Grafikobjekt ausgerichtet.

## Siehe auch

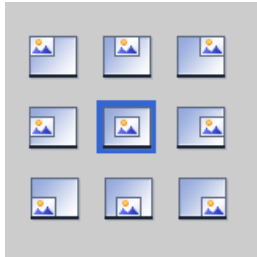
Position der Bildobjekte festlegen (Seite 3919)

Möglichkeiten der Bildanpassung (Seite 3914)

## Position der Bildobjekte festlegen

### Einleitung

Um die Position der Bildobjekte an das neue Gerät anzupassen, haben Sie verschiedene Möglichkeiten.



### Position wählen

Im Menü "Extras > Einstellungen > Visualisierung > Bilder Größenanpassung > Position wählen" passen Sie die Position der Bildobjekte an das neue Gerät an.

### Beispiel

Die folgende Option richtet die Objekte an der oberen linken Kante aus.



Die folgende Option zentriert die Objekte in der Mitte des Bildschirms.



### Siehe auch

Objekte an Inhalt anpassen (Seite 3916)

Möglichkeiten der Bildanpassung (Seite 3914)

## 10.9.7 Zwischen Bediengeräten und Editoren kopieren

### 10.9.7.1 Grundlagen

#### Grundlagen

##### Kopieren und Einfügen innerhalb eines Bediengeräts

Innerhalb eines Bediengeräts kopieren und fügen Sie Objekte ein, z. B. ein Anzeigeobjekt. Wenn das Objekt bereits in dem Editor angelegt ist, dann wird beim Einfügen dem Objektamen automatisch eine Nummerierung nach folgendem Prinzip angehängt:

- "<Objektnamen>\_1" wird umbenannt zu "<Objektnamen>\_2".
- "<Objektnamen>\_2" wird umbenannt zu "<Objektnamen>\_3".

##### Kopieren und Einfügen zwischen Bediengeräten

Sie können auch zwischen Bediengeräten kopieren und einfügen. Wenn bereits ein Objekt mit demselben Namen existiert, haben Sie folgende Möglichkeiten:



---

#### Hinweis

##### Ausnahme von dieser Grundregel

Das Kopieren und Einfügen einer Meldeklasse, die aus einer projektweiten Meldeklasse erzeugt wurde, wird abweichend von dieser Grundregel behandelt. Wenn die kopierte Meldeklasse im Ziel-Bediengerät innerhalb des gleichen Projekts bereits vorhanden ist, wird der Befehl "Einfügen" nicht ausgeführt.

---

## Benutzerdefinierte Ordner kopieren

In der Projektnavigation legen Sie benutzerdefinierte Ordner für Editoren z. B. für HMI-Variablen, Bilder usw. an.

Sie kopieren benutzerdefinierte Ordner und fügen sie in ein anderes Bediengerät ein. Die in einem benutzerdefinierten Ordner enthaltenen Objekte können die für ein anderes Bediengerät bestehenden Einschränkungen überschreiten, z. B. bei der Anzahl der unterstützten Bilder. Nach dem Einfügen werden alle Objekte angezeigt. Beim Übersetzen der Projektdaten wird ein Fehler ausgegeben.

Systemordner sind grundsätzlich nicht kopierbar.

## Siehe auch

Nicht unterstützte Objekte und Funktionalitäten (Seite 3921)

## Nicht unterstützte Objekte und Funktionalitäten

### Einleitung

Beim Kopieren eines Objekts werden alle seine Eigenschaften und Einstellungen vom Ziel-Bediengerät übernommen.

### Nicht unterstützte Objekte

Objekte, die im Ziel-Bediengerät nicht unterstützt werden, können nicht eingefügt werden.

---

#### Hinweis

Wenn Sie ein Bild kopieren, das Objekte enthält, die vom Ziel-Bediengerät nicht unterstützt werden, bleiben die Objekte im Hintergrund erhalten. Wenn Sie das Bild erneut kopieren und das neue Gerät die Objekte unterstützt, werden Sie wieder angezeigt.

---

### Ungültige Objekte

Folgende Objekte werden nach dem Einfügen im Ziel-Bediengerät ungültig:

- Referenzierte Objekte, die im Ziel-Bediengerät nicht vorhanden sind.
- Objekte mit Einstellungen, die im Ziel-Bediengerät nicht unterstützt werden.
- An Objekte projizierte Systemfunktionen, die im Ziel-Bediengerät nicht unterstützt werden.

Ungültige Objekte werden durch eine farbige Markierung gekennzeichnet. Wählen Sie ein unterstütztes Objekt oder legen Sie ein neues Objekt an. Wenn Sie ein ungültiges Objekt beibehalten, wird beim Übersetzen der Projektdaten ein Fehler ausgegeben.

## Farben und Schriftarten

Farben und Schriftarten werden von Bediengeräten in unterschiedlichem Umfang unterstützt. Beim Einfügen werden nicht unterstützte Farben und Schriftarten durch unterstützte Farben und Schriftarten ersetzt. Wenn Sie dasselbe Objekt zurück in das Quell-Bediengerät einfügen, sind die ursprünglich vorhandenen Einstellungen wieder aktiv.

### Siehe auch

Grundlagen (Seite 3920)

## 10.9.7.2 Kopieren und Einfügen

### Kopieren von Bildern

#### Einleitung

Sie kopieren ein oder mehrere Bilder aus dem Ordner "Bilder" und fügen sie in den Ordner "Bilder" eines anderen Bediengeräts ein.

### Art und Größe des Displays

Bei Bediengeräten mit Tasten werden die zur Verfügung stehenden Tasten automatisch im Bild angezeigt. Beim Kopieren eines Bildes zwischen den Bediengeräten werden die Tasten ein- bzw. ausgeblendet. An Funktionstasten projizierte Funktionen werden nicht übernommen.

Wenn im Ziel-Bediengerät weniger Platz für das Bild zur Verfügung steht als im Quell-Bediengerät, können Sie die Größe der vorhandenen Objekte und die Abstände zwischen den vorhandenen Objekten anpassen.

### Automatische Größenanpassung der Objekte

1. Wählen Sie im Menü "Extra > Einstellungen > Visualisierung > Anpassung Bilder und Bildobjekte".
2. Aktivieren Sie z. B. "Breite und Höhe an neues Gerät anpassen".

### Siehe auch

Kopieren von Rezepturen innerhalb eines Bediengeräts (Seite 3923)

Kopieren von Objekten mit verknüpften Objekten (Seite 3923)

Automatisch kopierte verknüpfte Objekte (Seite 3924)



## Kopieren von Rezepturen innerhalb eines Bediengeräts

### Editor "Rezepturen"

Sie kopieren Rezepturen, Rezepturelemente und Rezepturdatensätze innerhalb der jeweiligen Tabelle. Sie kopieren ein Rezepturelement in eine andere Rezeptur.

Nur WinCC Runtime Professional: Sie können ein Rezeptursicht-Element in eine andere Rezeptursicht kopieren. Wenn ein Rezeptursicht-Element mit dem gleichen Namen bereits existiert, wird ein Konflikt-Dialog angezeigt. Sie können wählen, das Rezepturelement zu ersetzen oder umzubenennen. Sie können Rezepturelemente kopieren in die erste leere Zeile des Editors "Rezeptursichten", Register "Elemente".

Sie können einen Rezepturdatensatz in eine andere Rezeptur kopieren, falls die andere Rezeptur die gleiche Anzahl Rezepturelemente enthält. Wenn die Datentypen unterschiedlich sind, wird der Wert zwar in den Ziel-Datensatz kopiert, erhält jedoch eine Fehlermarkierung.

### Editor "Variablen"

Sie können eine Variable per Drag&Drop auf ein Rezepturelement in die Spalte "Variable" ziehen. Die Variable wird mit dem Rezepturelement verbunden. Wenn schon eine Variable verbunden ist, wird eine Fehlermeldung ausgegeben.

### Editor "Bilder"

Wenn Sie eine Rezeptur per Drag&Drop auf ein Bild ziehen, wird eine neue Rezepturanzeige erstellt und mit der Rezeptur verbunden.

### Siehe auch

Kopieren von Bildern (Seite 3922)

Kopieren von Objekten mit verknüpften Objekten (Seite 3923)

Automatisch kopierte verknüpfte Objekte (Seite 3924)

## Kopieren von Objekten mit verknüpften Objekten

### Einleitung

Ein Objekt wird z. B. in folgenden Fällen mit einem anderen Objekt verknüpft:

- Sie legen für eine Meldung eine Variable als Triggervariable fest.  
Die Meldung ist das Objekt. Die Variable ist das verknüpfte Objekt.
- Sie legen für eine externe Variable eine Verbindung fest.  
Die Variable ist das Objekt. Die Verbindung ist das verknüpfte Objekt.

Das Objekt wird beim Kopieren und Einfügen immer vollständig eingefügt. Ob das verknüpfte Objekt eingefügt wird, hängt vom benutzten Befehl für das Einfügen ab.

### Einfaches Einfügen

Das verknüpfte Objekt wird nicht kopiert. Die Verknüpfung zum Objekt wird übernommen und im Ziel-Bediengerät wie folgt umgesetzt:

- Wenn ein gleichnamiges Objekt vorhanden ist, wird das vorhandene Objekt mit seinen Einstellungen genutzt.
- Wenn kein gleichnamiges Objekt vorhanden ist, wird der Objektname angezeigt. Das Objekt wird ungültig.

Bei einigen Objekten werden verknüpfte Objekte beim einfachen Einfügen automatisch mit kopiert und eingefügt.

### Erweitertes Einfügen

Wenn Sie im Kontextmenü den Befehl "Erweitertes Einfügen" wählen, legen Sie fest, dass verknüpfte Objekte mit eingefügt werden. Wenn im Ziel-Bediengerät gleichnamige Objekte vorhanden sind, legen Sie für jedes Objekt fest, ob das vorhandene Objekt überschrieben wird.

### Siehe auch

Kopieren von Bildern (Seite 3922)

Kopieren von Rezepturen innerhalb eines Bediengeräts (Seite 3923)

### Automatisch kopierte verknüpfte Objekte

#### Kopieren verknüpfter Objekte

Die folgende Tabelle zeigt die Objekte, bei denen verknüpfte Objekte beim einfachen Einfügen automatisch mit eingefügt werden.

| Objekt                | Verknüpftes Objekt |
|-----------------------|--------------------|
| Bild                  | Vorlage            |
| Symbolisches E/A-Feld | Textliste          |
| Grafisches E/A-Feld   | Grafikliste        |
| Grafikanzeige         | Grafik             |
| Variable              | Meldung            |
|                       | Zyklus             |
| Rezepturelement       | Textliste          |
| Aufgabenplaner        | Trigger            |

### Siehe auch

Kopieren von Bildern (Seite 3922)

Kopieren von Rezepturen innerhalb eines Bediengeräts (Seite 3923)

## Drag&Drop aus der Detailansicht

### Einleitung

Mit einfachen Handgriffen erhöhen Sie die Effizienz Ihrer Projektierung. Im Folgenden sehen Sie einige Beispiele für eine effiziente Projektierung.

### Objekte aus der Detailansicht in ein Bild einfügen

Sie ziehen aus der Detailansicht Objekte verschiedener Editoren in andere Editoren.

### Symbolische E/A-Feld einfügen

1. Öffnen Sie ein Bild.
2. Klicken Sie in der Projektnavigation auf den Editor "Text - und Grafiklisten". In der Detailansicht werden alle verfügbaren Text- und Grafiklisten angezeigt.
3. Klicken Sie in der Detailansicht auf eine Textliste z. B. "Textlist1".
4. Ziehen Sie per Drag&Drop eine Textliste aus der Detailansicht in ein Bild. Ein Symbolisches E/A-Feld wurde angelegt und mit der Textliste "Textlist1" verschaltet.

### Grafisches E/A - Feld einfügen

1. Öffnen Sie ein Bild.
2. Klicken Sie in der Projektnavigation auf den Editor "Text - und Grafiklisten". In der Detailansicht werden alle verfügbaren Text- und Grafiklisten angezeigt.
3. Klicken Sie in der Detailansicht auf eine Grafikliste z. B. "Graficlist1".
4. Ziehen Sie per Drag&Drop eine Grafikliste aus der Detailansicht in ein Bild. Ein Grafisches E/A-Feld wurde angelegt und mit der Grafikliste "Graficlist1" verschaltet.

### E/A - Feld einfügen

1. Öffnen Sie ein Bild.
2. Klicken Sie in der Projektnavigation auf den Editor "HMI-Variablen". In der Detailansicht werden alle verfügbaren HMI-Variablen angezeigt.
3. Klicken Sie in der Detailansicht auf HMI-Variable z. B. "Tag1".
4. Ziehen Sie per Drag&Drop die HMI-Variable aus der Detailansicht in ein Bild. Ein E/A-Feld wurde angelegt und mit der HMI-Variable "Tag1" verschaltet.

### 10.9.7.3 Kopieren zwischen unterschiedlichen RT- und ES-Versionen

#### Einleitung

Sie können Projektdaten, z. B. Bilder, Objekte oder Variablen, zwischen Projekten mit unterschiedlichen WinCC-Versionen kopieren und einfügen.

Wenn Sie zwischen verschiedenen WinCC-Versionen kopieren, bleiben alle Projektierungen erhalten, die auch in der Ziel-Version unterstützt werden. Projektierungen, welche die Ziel-Version nicht unterstützt, werden farbig als ungültig markiert.

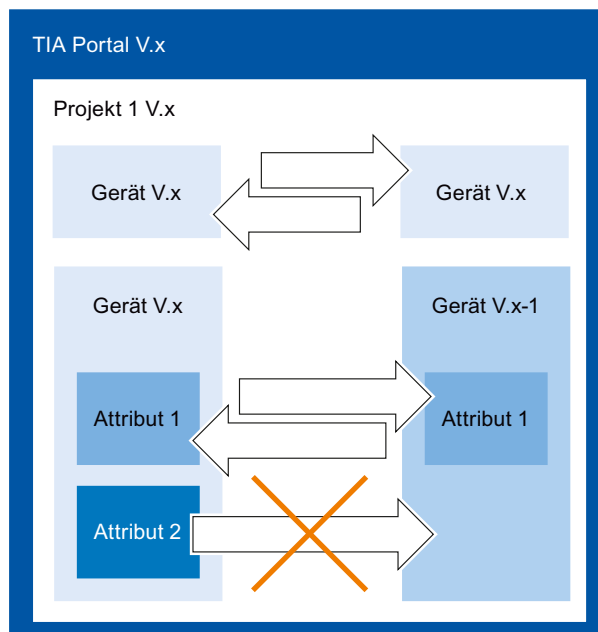
#### Kopieren zwischen von Bediengeräten unterschiedlicher Geräteversionen

WinCC unterstützt alle Projektierungen aus einer vorhergehenden WinCC-Version.

Beim Kopieren innerhalb unterschiedlicher ES-Versionen gilt Folgendes:

- Alle Projektierungen bleiben erhalten, die auch in der jeweiligen RT-Version unterstützt werden.
- Für Projektierungen die ausschließlich in der WinCC-Version des Ziel-Projekts unterstützt werden, werden Standardeinstellungen festgelegt.
- Projektierungen, welche die jeweiligen RT-Version nicht unterstützt, werden farbig als ungültig markiert oder nicht angezeigt.

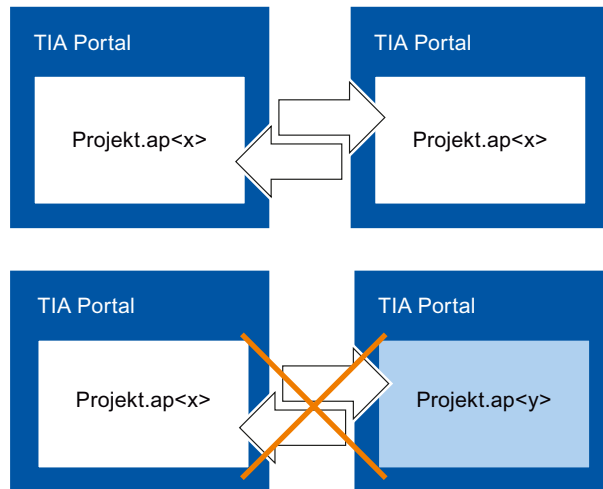
Wenn Sie ein Objekt unverändert zurück in das Quell-Bediengerät kopieren, sind alle ursprünglich im Quell-Bediengerät festgelegten Eigenschaften und Einstellungen wieder aktiv.



- Das Bediengerät muss für die aktuelle Runtime-Version gültig sein.

## Kopieren zwischen unterschiedlichen ES-Versionen

Um zwischen zwei TIA-Projekten zu kopieren, öffnen Sie eine zweite Instanz Ihres TIA-Portals. Sie können nur zwischen Projekten der gleichen ES-Version kopieren. Die ES-Version eines Projekts erkennen Sie an der Dateiendung \*.ap<Versionsnummer>.



## 10.9.8 WinCC-Versionskompatibilität nutzen

### 10.9.8.1 Grundlagen zur Versionskompatibilität

#### Einleitung

Bestehende Projekte bearbeiten Sie wie folgt mit WinCC:

- Sie bearbeiten, übersetzen und laden bestehende Projekte mit dem Funktionsumfang der WinCC-Vorgängerversion. Diese Projekte können Sie anschließend mit der WinCC-Vorgängerversion weiter bearbeiten.
- Sie rüsten bestehende Projekte hoch und nutzen die Funktionen der aktuellen WinCC-Version.

---

#### Hinweis

##### WinCC-Funktionen

Wenn Sie ein Projekt einer vorhergehenden WinCC-Version bearbeiten, stehen Ihnen nur die Funktionen und Bediengeräte dieser vorhergehenden WinCC-Version zur Verfügung.

---

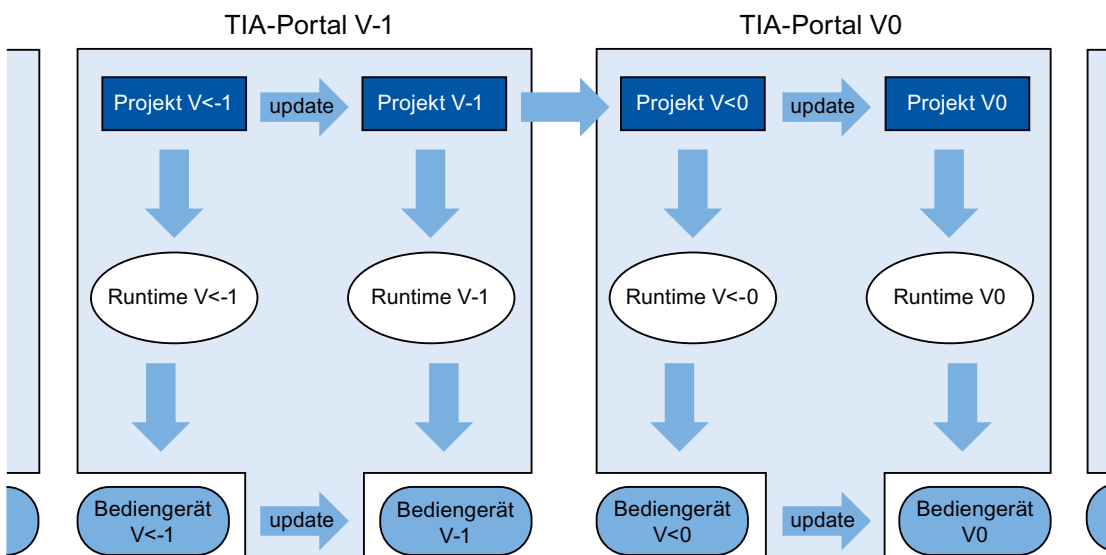
## Versionen in WinCC

In WinCC arbeiten Sie mit unterschiedlichen Arten von Versionen:

- **WinCC-Version**  
Die am Projektierungs-PC installierte WinCC-Version, z. B. WinCC V12.
- **WinCC-Version eines Projekts**  
Projekte werden mit der WinCC-Version erstellt, die auf dem Projektierungs-PC installiert ist.  
Wenn Sie z. B. mit der aktuellen WinCC-Version ein WinCC-Projekt aus der Vorgängerversion bearbeiten, wird in der Projektnavigation die Versionsbezeichnung hinter dem Projektnamen angezeigt.
- **Runtime-Version**  
In WinCC können Sie Bediengeräte mit unterschiedlichen Runtime-Versionen projektieren. Die Runtime-Version legen Sie einmal für ein Bediengerät fest. Die Bediengeräteversion muss zur Runtime-Version passen.
- **Bediengeräteversionen**  
Abhängig vom verwendeten Bediengerät setzt sich das Image aus Betriebssystem und / oder Runtime-Software zusammen. Für jedes Bediengerät stellt WinCC verschiedene Images zur Verfügung, die entsprechend der Projektierung bei Bedarf auf das Bediengerät geladen werden. Die Bediengeräteversion entspricht einem bestimmten Image. Die Bediengeräteversion muss zur Projektierung passen.

## Kompatibilität von WinCC-Versionen, Runtime-Versionen und Bediengeräteversionen

Die folgende Abbildung zeigt das Zusammenwirken der Versionen im TIA-Portal:



## Anlegen von Projekten

Wenn Sie ein Projekt in WinCC neu anlegen, öffnen und bearbeiten Sie es mit der WinCC-Version, in der Sie es angelegt haben.

## Speichern

Um ein Projekt einer vorhergehenden WinCC-Version für diese Version wieder abzuspeichern, speichern Sie wie gewohnt. Erst wenn Sie manuell das Projekt auf Ihre WinCC-Version hochrüsten, können Sie das Projekt in der vorhergehenden WinCC-Version nicht mehr öffnen.

Um ein Projekt einer vorhergehenden WinCC-Version in der aktuellen Version zu speichern, rüsten Sie das Projekt auf Ihre WinCC-Version hoch. Sie können dieses Projekt dann nicht mehr mit einer vorhergehenden WinCC-Version bearbeiten.

## Übersetzen, Simulieren und Laden

Wenn Sie ein Projekt einer WinCC-Vorgängerversion verwenden, können Sie mit Ihrer aktuellen Version Runtime-Daten dieser Vorgängerversion erzeugen. Dadurch laden Sie auch Bediengeräte, die mit Ihrer WinCC-Version nicht mehr kompatibel sind.

## Kopieren innerhalb von Projekten mit verschiedenen WinCC-Versionen

Wenn Objekte und Projektierungen in der Ziel-Version auch zur Verfügung stehen, kopieren Sie diese über die Zwischenablage oder Drag&Drop nach Bedarf.

## Öffnen, Bearbeiten und Speichern von Projekten einer vorhergehenden WinCC-Version

Sie öffnen und bearbeiten Projekte vorhergehender WinCC-Versionen nach Bedarf. Sie nutzen dabei nur die Funktionen der vorhergehenden WinCC-Version. Nach Ihrer Bearbeitung kann das Projekt wieder in der vorhergehenden WinCC-Version bearbeitet und gespeichert werden.

## Übersetzen, Laden und Simulieren von Projekten mit vorhergehenden WinCC-Versionen

Sie übersetzen, laden und simulieren Projekte vorhergehender WinCC-Versionen nach Bedarf. Dabei stellt Ihre aktuelle WinCC-Version die Runtimes und Bediengeräteversionen der entsprechenden WinCC-Version zur Verfügung.

### 10.9.8.2 Projekt einer WinCC-Vorgängerversion bearbeiten

#### Einleitung

WinCC bietet die Möglichkeit, Projekte einer vorhergehenden WinCC-Version zu bearbeiten. Während Sie ein Projekt einer vorhergehenden WinCC-Version bearbeiten, stehen Ihnen nur die Funktionen dieser Version zur Verfügung. Um die Funktionen Ihrer aktuellen WinCC-Version für dieses Projekt zu nutzen, rüsten Sie das Projekt auf Ihre WinCC-Version hoch.

---

#### Hinweis

Wenn Sie ein Projekt auf Ihre WinCC-Version hochrüsten, können Sie es nicht mehr in der vorhergehenden WinCC-Version öffnen und bearbeiten.

---

### Voraussetzung

- Ein Projekt einer vorhergehenden WinCC-Version ist angelegt.
- Auf dem Projektierungs-PC ist die aktuelle WinCC-Version installiert.

### Vorgehen

Um ein Projekt einer vorhergehenden WinCC-Version zu bearbeiten, gehen Sie folgendermaßen vor:

1. Öffnen Sie das Projekt.
2. Bearbeiten Sie das Projekt mit den Funktionen der vorhergehenden WinCC-Version.
3. Speichern Sie das Projekt.
4. Übersetzen Sie das Projekt.
5. Laden und Simulieren Sie das Projekt.
6. Öffnen Sie das Projekt bei Bedarf mit der vorhergehenden WinCC-Version und bearbeiten Sie es dort weiter.

### Ergebnis

Die geänderten Projektdaten können an einem anderen Projektierungs-PC mit der vorhergehenden WinCC-Version weiterbearbeitet werden. Das Runtime-Projekt wurde in der entsprechenden Runtime-Version erzeugt und geladen.

#### 10.9.8.3 Projekt hochrüsten

### Einleitung

Wenn die Projektversion älter als die WinCC-Version ist, wird in der Projektnavigation die Versionsbezeichnung angezeigt. Ihre WinCC-Version enthält auch die Vorgängerversion, mit der Sie bei Bedarf Projekte bearbeiten. Um in einem Projekt die Funktionen und Möglichkeiten Ihrer WinCC-Version zu nutzen, rüsten Sie das Projekt auf Ihre WinCC-Version hoch.

---

#### Hinweis

#### WinCC-Versionskompatibilität

Wenn Sie ein Projekt auf Ihre WinCC-Version hochrüsten, können Sie es nicht mehr mit der vorhergehenden Version bearbeiten.

---

### Voraussetzungen

- Die WinCC-Version des Projekts ist eine vorhergehende zu Ihrer WinCC-Version.
- Sie haben Schreibrechte auf Ihrem Projektlaufwerk.
- Die Speicherkapazität auf Ihrem Projektlaufwerk ist ausreichend für ein weiteres Projekt dieser Größe.



## Vorgehen

Um ein Projekt auf Ihre WinCC-Version hochzurüsten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation das Projekt.
2. Wählen Sie im Kontextmenü des Projekts den Befehl "Projekt hochrüsten".  
Ein Dialog wird geöffnet.
3. Klicken Sie "Bestätigen".  
Das Projekt wird geschlossen und ein Fortschrittsbalken angezeigt.

Wenn das Projekt hochgerüstet ist, wird eine Meldung ausgegeben.

## Ergebnis

- Im Projektlaufwerk wurde das Projekt in der vorhergehenden WinCC-Version mit der entsprechenden Dateierweiterung gespeichert.
- Im Projektlaufwerk wird das Projekt mit der aktuellen WinCC-Version und der entsprechenden Dateierweiterung angezeigt.
- Das Projekt wird in der Projektnavigation ohne WinCC-Versionsbezeichnung angezeigt.

### 10.9.8.4 Wechsel zwischen Bediengeräteversionen

#### Auswahl der Bediengeräteversion

Wenn Sie ein neues Bediengerät projektieren, wählt WinCC automatisch die neueste Bediengeräteversion aus.

Wenn Sie auf einem Bediengerät eine andere Bediengeräteversion verwenden wollen als in WinCC eingestellt wurde, übertragen Sie ein Image auf das Bediengerät. WinCC liefert die benötigten Images der unterstützten Bediengeräte mit.

Informationen zu den in WinCC verwendeten Bediengeräteversionen finden Sie als FAQ im Internet unter der Beitrags-ID 21742389.

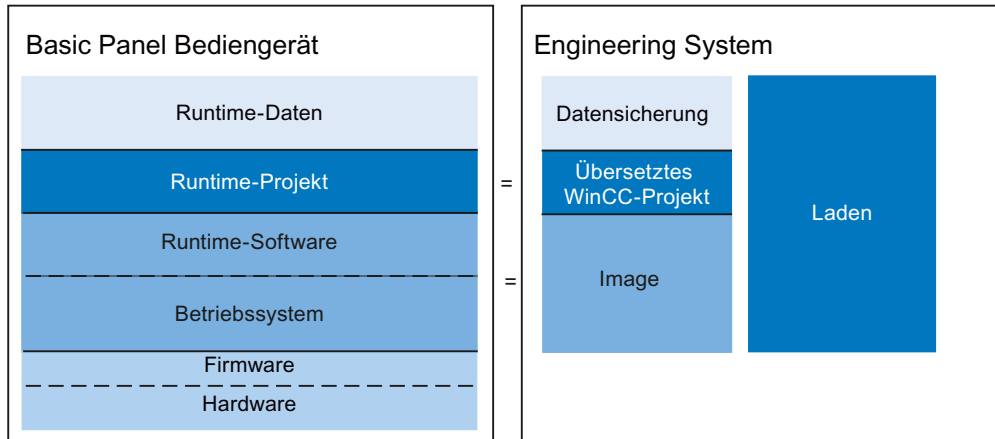
#### **ACHTUNG**

**Das Ändern der Bediengeräteversion löscht alle Daten auf dem Bediengerät.**

Wenn Sie die Bediengeräteversion ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb vorhandene Runtime-Daten und ändern Sie die Bediengeräteversion nachträglich.

### Aufbau Bediengerät

Die folgende Abbildung zeigt die Software-Komponenten eines Bediengeräts:



#### 10.9.8.5 Bediengeräteversion ändern

##### Einleitung

Abhängig von der gewünschten Runtime-Version wählen Sie für Ihre Projektierung die passende Bediengeräteversion.

##### Hinweis

##### Auswahl Bediengeräteversionen

Die bereitgestellte Auswahl von Bediengeräteversionen ist abhängig von der WinCC-Version Ihres Projekts.

##### Bediengeräteversion prüfen

Um die Bediengeräteversion eines vorhandenen Bediengerätes zu prüfen, verbinden sie den Projektierungs-PC online mit dem Bediengerät. Doppelklicken Sie in der Projektnavigation unter Ihrem Projekt "Online & Diagnose > Diagnose" und prüfen Sie dort die Bediengeräteversion.

##### Voraussetzungen

- Ein Projekt ist angelegt und geöffnet.
- Das Projekt enthält ein Bediengerät.

## Vorgehen

Um die Bediengeräteversion zu ändern, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf "Geräte & Netze".  
Der Editor wird geöffnet.
2. Selektieren Sie das gewünschte Bediengerät in der Gerätesicht.
3. Wählen Sie im Kontextmenü des Bediengeräts den Eintrag "Gerät / Version ändern".  
Ein Dialog wird geöffnet.
4. Selektieren Sie das gewünschte Bediengerät.
5. Wählen Sie unter "Version" die gewünschte Bediengeräteversion.
6. Betätigen Sie Ihre Auswahl mit "OK".

## Ergebnis

Sie haben die Bediengeräteversion des Gerätes im WinCC-Projekt geändert.

### ACHTUNG

**Das Ändern der Bediengeräteversion löscht beim Laden alle Daten auf dem Bediengerät.**

Wenn Sie die Bediengeräteversion ändern und das Projekt übersetzen und Laden, werden auf dem Bediengerät Daten gelöscht. Sichern Sie deshalb vor dem Laden vorhandene Runtime-Daten.

## 10.9.9 Daten einer Speicherkarte anzeigen

### 10.9.9.1 Grundlagen

#### Einleitung

WinCC bietet Ihnen die Möglichkeit Daten Ihrer Speicherkarte anzeigen zu lassen. Dabei können Sie sowohl Speicherkarten des Bediengeräts als auch der CPU verwenden.

Sie haben folgende Möglichkeiten:

Backup anzeigen (Seite 3934)

Backup umbenennen und löschen (Seite 3935)

Auto-Hotspot

Auto-Hotspot

Auto-Hotspot

## Siehe auch

Backup anzeigen (Seite 3934)

Backup umbenennen und löschen (Seite 3935)

### 10.9.9.2 Arbeiten mit Backups

#### Backup anzeigen

##### Einleitung

Wenn Sie auf einer Speicherkarte das Backup eines Basic Panels speichern, können Sie sich das Backup im TIA-Portal anzeigen lassen.

##### Voraussetzungen

- WinCC ist installiert.
- Eine Speicherkarte mit einem Backup ist vorhanden.
- Der Kartenleser ist mit dem Projektierungs-PC verbunden.
- Die Projektansicht ist geöffnet.

##### Backup auf der Speicherkarte im Kartenleser

1. Legen Sie die Speicherkarte in den Kartenleser.
2. Öffnen Sie in der Projektnavigation "SIMATIC Card Reader".
3. Wählen Sie das Laufwerk des Kartenlesers.  
Der Ordner "Online Card Data" wird angezeigt
4. Öffnen Sie den Ordner "Online Card Data "
5. Klicken Sie auf das Backup und öffnen Sie das Kontextmenü.
6. Wählen Sie "Eigenschaften".

##### Backup auf der Speicherkarte der PLC

Wenn das Backup auf der Speicherkarte der PLC abgelegt wurde, gehen Sie wie folgt vor:

1. Verbinden Sie die PLC mit dem Projektierungs-PC.
2. Klicken Sie in der Projektnavigation auf die PLC.
3. Wählen Sie im Kontextmenü "Online verbinden".  
Eine Verbindung zur PLC wird hergestellt.  
Wenn die PLC verbunden ist, wird der Ordner "Online Card Data" angezeigt.

4. Öffnen Sie den Ordner "Online Card Data".

---

#### Hinweis

#### Zugriff auf eine kennwortgeschützte PLC

Wenn Sie auf eine PLC zugreifen, die durch ein Kennwort geschützt ist, werden Sie aufgefordert, ein Kennwort einzugeben.

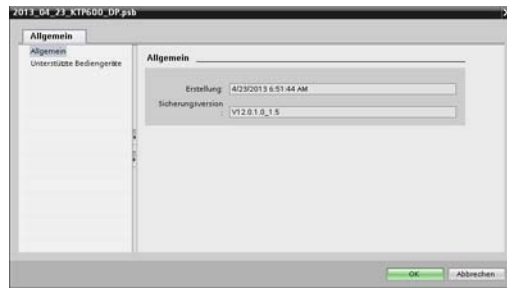
Um die Daten der Speicherkarte anzeigen zu lassen, benötigen Sie mindestens einen Lesezugriff.

---

5. Klicken Sie auf das Backup und öffnen Sie das Kontextmenü.
6. Wählen Sie "Eigenschaften".

## Ergebnis

Die Eigenschaften des Backups werden in einem separaten Dialog angezeigt.



## Siehe auch

- Grundlagen (Seite 3933)
- Backup umbenennen und löschen (Seite 3935)

## Backup umbenennen und löschen

## Einleitung

Sie haben die Möglichkeit die Backups einer Speicherkarte in der Projektnavigation des TIA-Portals umzubenedenennen und zu löschen.

## Voraussetzungen

- WinCC ist installiert.
- Der Kartenleser ist mit dem Projektierungs-PC verbunden.  
Bzw. Die PLC ist mit dem Projektierungs-PC online verbunden.
- Eine Speicherkarte mit einem Backup ist vorhanden.

- Die Projektansicht ist geöffnet.
- Das Backup wird in der Projektnavigation angezeigt.

---

**Hinweis**

**Zugriff auf eine kennwortgeschützte PLC**

Wenn Sie auf eine PLC zugreifen, die durch ein Kennwort geschützt ist, werden Sie aufgefordert, ein Kennwort einzugeben.

Um die Daten der Speicherkarte zu löschen bzw. umzubenennen, benötigen Sie einen Schreibzugriff.

---

## Vorgehen

1. Klicken Sie in der Projektnavigation auf das Backup.
2. Öffnen Sie das Kontextmenü.
3. Um die Datei umzubenennen, wählen Sie "Umbenennen".
4. Geben Sie einen neuen Namen ein.
5. Um die Datei zu löschen, wählen Sie "Löschen".

## Ergebnis

Die Backup-Datei wurde umbenannt bzw. gelöscht.

## Siehe auch

Grundlagen (Seite 3933)

Backup anzeigen (Seite 3934)

## 10.10 Übersetzen und Laden

### 10.10.1 Übersetzen und Laden von Projekten

#### 10.10.1.1 Übersicht zum Übersetzen und Laden von Projekten

## Übersicht

Bereits während Sie ein Projekt in WinCC projektieren, wird das Projekt laufend im Hintergrund übersetzt. Dadurch werden die Zeiten für das endgültige Übersetzen verkürzt. Wenn Sie das Übersetzen starten, entsteht eine Datei, die am jeweiligen Bediengerät ablauffähig ist.

Wenn beim Übersetzen Fehler auftreten, unterstützt WinCC Sie dabei, die Fehler zu finden und zu beheben.

Nach der Fehlerbehebung laden Sie das übersetzte Projekt auf die Bediengeräte, auf denen Sie das Projekt einsetzen wollen.

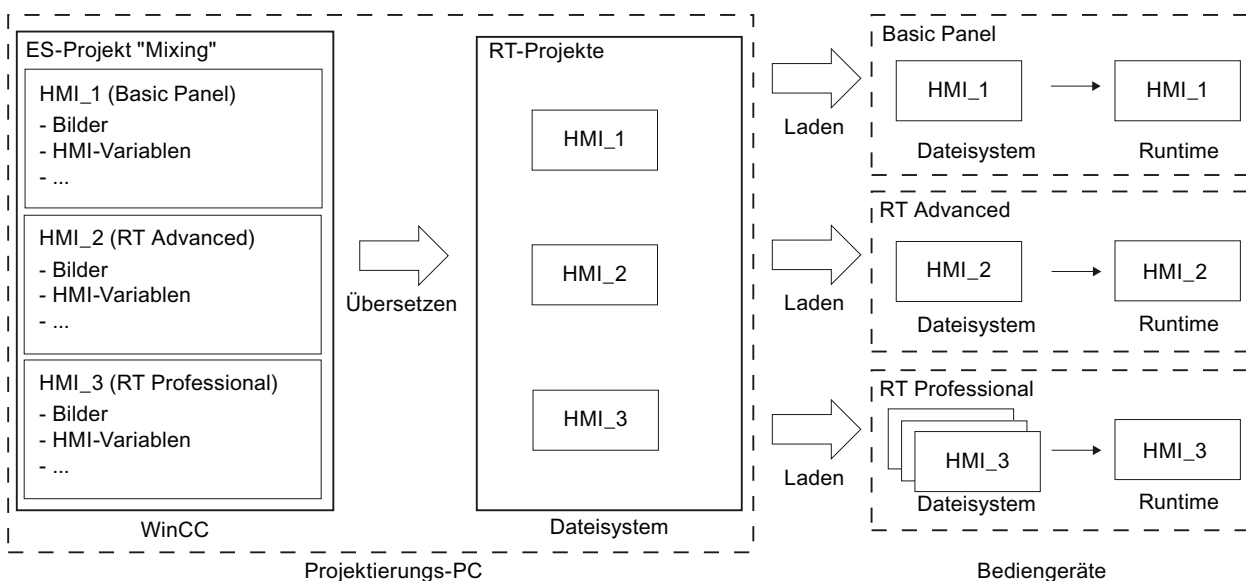
Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch alle geänderten S7-Bausteine über den Befehl "Übersetzen > Software" im Kontextmenü.

## Begriffsklärung

Im Kontext vom Übersetzen und Laden wird der Begriff "Projekt" synonym verwendet. Zum einen bezeichnet "Projekt" das WinCC-Projekt auf dem Projektierungs-PC. Zum anderen bezeichnet "Projekt" das Runtime-Projekt, das Sie aus den Projektierungsdaten eines Bediengeräts durch Übersetzen erzeugen und auf das Bediengerät laden.

- WinCC-Projekt: Enthält die Projektierungsdaten von einem oder mehreren Bediengeräten
- Runtime-Projekt: Enthält die übersetzten Projektierungsdaten eines Bediengeräts

Das folgende Bild zeigt den Zusammenhang zwischen WinCC-Projekt und Runtime-Projekt am Beispiel des Vorgangs von "Übersetzen und Laden":



## Runtime-Version

Die Runtime-Version ist abhängig vom Image des projektierten Bediengeräts. Die Runtime-Version des übersetzten Projekts wird im Inspektorfenster unter "Info" angezeigt.

### 10.10.1.2 Projekt übersetzen

#### Einleitung

Bereits während Sie ein Projekt in WinCC projektieren, werden die Änderungen am Projekt laufend im Hintergrund übersetzt. Wenn Sie ein Projekt laden, wird das Projekt automatisch übersetzt. Auf diese Weise wird sichergestellt, dass immer der neueste Stand des Projekts geladen wird.

Während des Übersetzens überprüft WinCC das Projekt auf Konsistenz. Im Inspektorfenster werden alle fehlerhaften Stellen im Projekt aufgelistet. Über den Eintrag im Inspektorfenster springen Sie direkt zur Fehlerursache. Sie überprüfen die festgestellten Fehler und beheben diese.

#### Umfang des Übersetzens

Sobald Sie mit der Projektierung eines Bediengeräts beginnen, werden die Projektierungsdaten laufend im Hintergrund übersetzt. Wenn Sie ein Projekt manuell übersetzen, werden nur die Änderungen in der Projektierung seit dem letzten Übersetzen im Hintergrund übersetzt.

Um z. B. die Konsistenz der projektierten Daten zu testen, können Sie ein komplettes Übersetzen des Projekts jederzeit manuell starten.

#### Voraussetzung

- Ein Projekt ist geöffnet.

#### Vorgehen

Um ein Projekt zu übersetzen, gehen Sie folgendermaßen vor:

1. Wenn Sie mehrere Bediengeräte gleichzeitig übersetzen wollen, wählen Sie alle gewünschten Bediengeräte mit der Mehrfachauswahl in der Projektnavigation aus.
2. Übersetzen Sie das Projekt:
  - Um nur Änderungen im Projekt zu übersetzen, wählen Sie im Kontextmenü des Bediengeräts den Befehl "Übersetzen > Software".
  - Um das Projekt komplett zu übersetzen, wählen Sie im Kontextmenü den Befehl "Übersetzen > Software (komplett übersetzen)".

#### Ergebnis

Die Projektierungsdaten aller ausgewählten Bediengeräte werden übersetzt. Wenn beim Übersetzen Fehler auftreten, werden diese im Inspektorfenster angezeigt.



### 10.10.1.3 Laden von Projekten

## Übersicht zum Laden von Projekten

### Übersicht

Wenn Sie ein Projekt auf ein oder mehrere Bediengeräte übertragen, wird der seit dem letzten Übertragen geänderte Teil des Projekts automatisch vor dem Laden übersetzt. Auf diese Weise wird sichergestellt, dass immer der aktuelle Stand des Projekts übertragen wird.

### Laden eines Projekts auf ein Bediengerät

Vor dem Laden werden folgende Schritte ausgeführt:

1. Die Einstellungen für das Laden werden geprüft. Wenn Sie ein Projekt zum 1. Mal auf ein Bediengerät laden, wird automatisch der Dialog "Erweitertes Laden" geöffnet. In diesem Dialog legen Sie abhängig von der verwendeten Runtime des Bediengeräts Protokoll und Schnittstelle oder den Zielpfad für das Projekt fest.  
Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät..." aufrufen.  
Der Dialog "Vorschau laden" wird geöffnet.
2. Das Projekt wird übersetzt. Warnungen und Fehler beim Übersetzen werden im Inspektorfenster und im Dialog "Vorschau laden" angezeigt.
3. Im Dialog "Vorschau laden" werden Ihnen nach Bediengeräten getrennt folgende Informationen angezeigt:
  - Die Einzelschritte des Ladens
  - Wenn das Image des Ziel-Bediengeräts nicht dem Image aus der Projektierung entspricht, wird abgefragt, ob Sie das Image jetzt ändern wollen.

#### **ACHTUNG**

##### **Das Ändern des Images löscht alle Daten auf dem Bediengerät.**

Wenn Sie das Image ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb bei Bedarf zuerst folgende Daten:

- Benutzerverwaltung
- Rezepturen

- Voreinstellungen, die beim Laden wirksam werden. Passen Sie die Voreinstellungen für diesen Ladevorgang bei Bedarf an.
- Auftretende Warnungen (optional). Sie können ein Projekt trotz "Warnungen" laden. Die Funktionalität in Runtime kann jedoch eingeschränkt sein.
- Auftretende Fehler (optional). Sie können das Projekt nicht laden. Beheben Sie zunächst die Fehler und laden Sie das Projekt erneut.  
Wenn Sie im Inspektorfenster auf den angezeigten Fehler doppelklicken, öffnet WinCC die fehlerhafte Projektierung im entsprechenden Editor. Korrigieren Sie die Fehler und laden Sie das Projekt erneut.

Wenn Sie in Ihrem Projekt HMI-Variablen verwenden, die an Steuerungsvariablen angebunden sind, übersetzen Sie vor dem HMI-Bediengerät auch alle geänderten S7-Bausteine über den Befehl "Übersetzen > Software" im Kontextmenü.

### Laden über S7-Routing

Die Einstellungen zum S7 Routing projektieren Sie in der jeweiligen Steuerung im Editor "Geräte & Netze". Die Einstellungen sind abhängig vom projektierten Gerät.

S7 Routing unterstützt folgende Protokolle:

- MPI / PROFIBUS
- Ethernet

### Projekt laden

#### Einleitung

Damit ein Projekt auf einem Bediengerät abläuft, laden Sie das Projekt auf das Bediengerät. Beim Laden legen Sie insbesondere fest, ob auf dem Bediengerät vorhandene Daten wie "Benutzerverwaltung" oder "Rezepturdaten" überschrieben werden.

Wenn das Bediengerät PROFINET unterstützt, wird der in der Projektnavigation eingetragene Name des Bediengeräts als Gerätenamen für die PROFINET-Kommunikation verwendet. Der Name wird beim Laden in das Bediengerät geschrieben. Wenn im Bediengerät bereits ein Gerätenamen für die PROFINET-Kommunikation eingetragen war, wird dieser überschrieben.

Generell kann auf einem Bediengerät immer nur ein Projekt in Runtime sein. Standardmäßig ist ein Bediengerät so konfiguriert, dass eine laufende Runtime beim Laden automatisch beendet wird. Ansonsten müssen Sie Runtime am Bediengerät manuell beenden.

Wenn das Image des Ziel-Bediengeräts nicht dem Image aus der Projektierung entspricht, wird abgefragt, ob Sie das Image jetzt ändern wollen.

|  |
|--|
| <b>ACHTUNG</b>   |
| <b>Das Ändern des Images löscht alle Daten auf dem Bediengerät.</b>  |
| Wenn Sie das Image ändern, werden Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb bei Bedarf zuerst folgende Daten: |
| <ul style="list-style-type: none"><li>• Benutzerverwaltung</li><li>• Rezepturen</li></ul>                                  |

Weiterführende Informationen zu den Transfereinstellungen finden Sie in der Dokumentation des verwendeten Bediengeräts.

---

### Hinweis

#### Automatisches Beenden der Runtime

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im Control Panel.

---

### Voraussetzung

- Im Projekt ist ein Bediengerät angelegt.
- Das Bediengerät ist mit dem Projektierungs-PC verbunden.
- Am Bediengerät ist der Transfermodus eingestellt.

### Vorgehen

Um ein Projekt zu laden, gehen Sie folgendermaßen vor:

1. Wenn Sie ein Projekt auf mehrere Bediengeräte gleichzeitig laden wollen, wählen Sie in der Projektnavigation alle gewünschten Bediengeräte mit der Mehrfachauswahl aus.
2. Wählen Sie im Kontextmenü eines Bediengeräts den Befehl "Laden in Gerät > Software".
3. Wenn der Dialog "Erweitertes Laden" geöffnet wird, konfigurieren Sie die "Einstellungen für das Laden". Achten Sie dabei darauf, dass die "Einstellungen für das Laden" den "Transfereinstellungen am Bediengerät" entsprechen:
  - Wählen Sie das verwendete Protokoll, z. B. Ethernet oder HTTP.
  - Konfigurieren Sie die entsprechenden Schnittstellenparameter am Projektierungs-PC.
  - Stellen Sie am Bediengerät bei Bedarf schnittstellen- oder protokollspezifische Einstellungen ein.
  - Klicken Sie auf "Laden".

Sie können den Dialog "Erweitertes Laden" jederzeit über den Menübefehl "Online > Erweitertes Laden in Gerät..." aufrufen.  
Der Dialog "Vorschau laden" wird geöffnet. Gleichzeitig wird das Projekt übersetzt. Das Ergebnis wird im Dialog "Vorschau laden" angezeigt.
4. Überprüfen Sie die angezeigten Voreinstellungen und ändern Sie diese bei Bedarf.
5. Klicken Sie auf "Laden".

## Ergebnis

Das Projekt wird auf die ausgewählten Bediengeräte geladen. Ein vorhandenes Projekt wird ersetzt. Abhängig von den Einstellungen im Dialog "Vorschau laden" werden die Daten für die Benutzerverwaltung und / oder Rezepturen ersetzt. Wenn Fehler oder Warnungen beim Laden auftreten, werden im Inspektorfenster unter "Info > Laden" entsprechende Meldungen ausgegeben.

Nach dem erfolgreichen Laden ist das Projekt auf dem Bediengerät ablauffähig.

## Siehe auch

Daten des Bediengeräts sichern und wiederherstellen (Seite 3951)

Betriebssystem des Bediengeräts aktualisieren (Seite 3953)

Fehlermeldungen beim Laden von Projekten (Seite 3955)

Anpassen des Projekts für ein anderes Bediengerät (Seite 3956)

Herstellen der Verbindung zum Bediengerät (Seite 3958)

### 10.10.1.4 Runtime-Start

#### Runtime auf dem Bediengerät starten

##### Einleitung

Sobald Sie das Projekt auf Bediengerät geladen haben, können Sie das Projekt in Runtime starten. Das Projekt wird auf dem Bediengerät in einer Datei mit folgender Endung gespeichert:

- Basic Panels sowie OP 73, OP 77A und TP 177A: "\*.srt"

Beim Start des Projekts in Runtime werden die Einstellungen wirksam, die Sie in Ihrem Projekt in den "Runtime-Einstellungen" des Bediengeräts eingestellt haben.

Die Programme zum Starten von Projekten am Bediengerät finden Sie im Installationsverzeichnis von Runtime.

---

#### **Hinweis**

##### **Automatisches Beenden der Runtime**

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im Control Panel.

---

#### **Voraussetzung**

Auf dem Bediengerät ist die WinCC Runtime installiert.

#### **Vorgehen**

Auf einem Panel wird die Anwendung "RT Loader" gestartet. Abhängig von der Projektierung startet nach einer Verzögerungszeit das geladene Projekt automatisch.

Wenn das Projekt nicht automatisch startet:

1. Um die Projektdatei zu wählen, klicken Sie auf "Einstellungen" und tragen Sie unter "Projektierungsdatei" den Pfad zur Projektdatei ein.
2. Klicken Sie auf "OK" und anschließend auf "Start".

## **10.10.2 Simulieren von Projekten**

### **10.10.2.1 Grundlagen zur Simulation**

#### **Einleitung**

Mit dem Simulator testen Sie das Verhalten Ihrer Projektierung am Projektierungs-PC. Auf diese Weise finden Sie frühzeitig und vor dem Produktivbetrieb logische Projektierungsfehler.

Sie können den Simulator auf folgende Arten starten:

- Im Kontextmenü des Bediengeräts oder eines Bildes: "Simulation starten"
- Menübefehl "Online > Simulation > [Starten|Mit Variablen-Simulator|Mit Skript-Debugger]"
- In der Portalanzeige unter "Visualisierung > Gerät simulieren"

## Voraussetzung

Auf dem Projektierungs-PC ist die Komponente Simulation/Runtime installiert.

## Einsatzgebiet

Sie können mit dem Simulator z. B. folgende Funktionen des Bedien- und Beobachtungssystems testen:

- Überprüfung von Grenzwertstufen und Meldeausgaben
- Durchgängigkeit von Alarmen
- Projektierte Alarmsimulationen
- Projektierte Warnungen
- Projektierte Fehlermeldungen
- Kontrolle von Statusanzeigen
- Verschaltung und Layout von Bildern

## Siehe auch

Projekt simulieren (Seite 3944)

### 10.10.2.2 Projekt simulieren

## Einleitung

Sie simulieren Ihr Projekt auf eine der folgenden zwei Arten:

- Ohne angeschlossene Steuerung  
Die Werte von Bereichszeigern und Variablen verändern Sie in einem Variablen-Simulator, der bei der Simulation von WinCC Runtime gelesen wird.
- Mit angeschlossener Steuerung ohne laufenden Prozess  
Sie simulieren Ihr Projekt, indem Sie es direkt in Runtime laufen lassen. Die Variablen und Bereichszeiger werden aktiv. So erreichen Sie in Runtime eine authentische Simulation Ihres projektierten Bediengeräts.

---

### Hinweis

#### Einschränkungen bei der Simulation

Folgende Systemfunktionen können nicht simuliert werden:

- Kalibrierte Touchscreen

Eine Simulation des Media Players ist nicht möglich. Im Simulationsfenster wird anstelle des Media Players ein statisches Bild angezeigt.

Bei Bediengeräten mit Windows CE sind Dateizugriffe über Skripte nicht möglich.

---

## Voraussetzung

- Simulation ohne angeschlossene Steuerung: Variablen sind erstellt
- Simulation mit angeschlossener Steuerung ohne laufenden Prozess: Ein Projekt mit Variablen und Bereichszeigern ist erstellt

## Vorgehen

Um ein Projekt über den Variablen-Simulator zu simulieren, gehen Sie folgendermaßen vor:

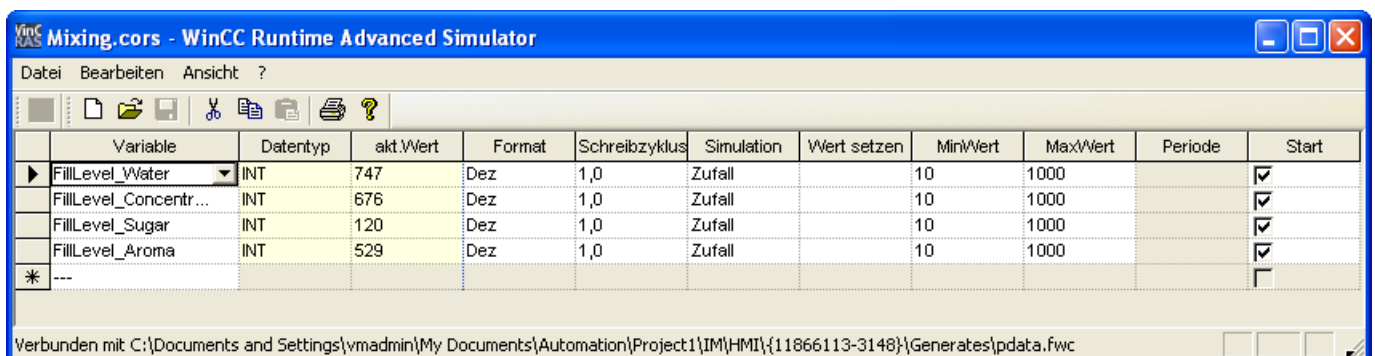
1. Öffnen Sie am Projektierungs-PC das Projekt.
2. Wählen Sie den Menübefehl "Online > Simulation > Mit Variablen-Simulator".  
Wenn Sie das Projekt zum 1. Mal simulieren, wird der Simulator mit einer neuen, leeren Tabelle gestartet. Gleichzeitig wird das Projekt in Runtime geöffnet.  
Zwischen Variablen-Simulator und Runtime schalten Sie mit <Alt+Tab> um.
3. Um einen Prozesswert zu simulieren, wählen Sie im Variablen-Simulator die gewünschte "Variable" aus.  
Die Auswahl enthält alle projektierten Variablen. Sie können maximal 300 Variablen gleichzeitig simulieren.
4. Wählen Sie in der Spalte "Simulation" die gewünschte Simulationsart aus.
5. Verändern Sie in den entsprechenden Spalten die Werte für die Variablen und die Bereichszeiger.
6. Um die Simulation für diese Variable zu starten, aktivieren Sie das Kontrollkästchen "Start".
7. Um die Simulation zu speichern, wählen Sie den Menübefehl "Datei > Speichern" und geben sie einen sprechenden Namen ein, z. B. "Mixing".  
Die Datei erhält die Endung "\*.cors".

## Ergebnis

Die Prozesswerte werden in Runtime simuliert. Abhängig von der Simulationsart werden die Variablenwerte z. B. zufällig erzeugt oder inkrementiert.

Um Variablenwerte vorzugeben, ändern Sie die Simulationsart auf "<Anzeige" und geben Sie unter "Wert setzen" den gewünschten Wert ein.

Das folgende Bild zeigt einen Variablen-Simulator mit vier Variablen, deren Werte im Wertebereich von 10 bis 1000 zufällig ermittelt werden:



|   | Variable              | Datentyp | akt.Wert | Format | Schreibzyklus | Simulation | Wert setzen | MinWert | MaxWert | Periode | Start                               |
|---|-----------------------|----------|----------|--------|---------------|------------|-------------|---------|---------|---------|-------------------------------------|
| ▶ | FillLevel_Water       | INT      | 747      | Dez    | 1,0           | Zufall     |             | 10      | 1000    |         | <input checked="" type="checkbox"/> |
|   | FillLevel_Concentr... | INT      | 676      | Dez    | 1,0           | Zufall     |             | 10      | 1000    |         | <input checked="" type="checkbox"/> |
|   | FillLevel_Sugar       | INT      | 120      | Dez    | 1,0           | Zufall     |             | 10      | 1000    |         | <input checked="" type="checkbox"/> |
|   | FillLevel_Aroma       | INT      | 529      | Dez    | 1,0           | Zufall     |             | 10      | 1000    |         | <input checked="" type="checkbox"/> |
| * | --                    |          |          |        |               |            |             |         |         |         | <input type="checkbox"/>            |

Verbunden mit C:\Documents and Settings\vmadmin\My Documents\Automation\Project1\IM\HMI\{11866113-3148}\Generates\pdata.fwc

### Simulationsdaten verwalten

Wenn Sie die Daten einer früheren Simulation gespeichert haben, können Sie die Datei zu einem späteren Zeitpunkt öffnen und Ihr Projekt erneut simulieren. Voraussetzung dafür ist, dass die im Variablen-Simulator enthaltenen Variablen und Bereichszeiger noch im Projekt enthalten sind.

Um eine Simulationsdatei zu öffnen, gehen Sie folgendermaßen vor:

1. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator".
2. Wählen Sie im Variablen-Simulator den Menübefehl "Datei > Öffnen".
3. Wählen Sie die entsprechende Simulationsdatei aus und klicken Sie auf "Öffnen".  
Der Simulator übernimmt die gespeicherten Daten.

### Variablen aktivieren und deaktivieren

Damit der Übergang von der Offline-Projektierung zur Online-Projektierung reibungslos möglich ist, starten und stoppen Sie die Simulation jeder Variable einzeln. Aktivieren Sie dazu in der entsprechenden Zeile "Start".

Wenn eine Variable aktiviert ist, werden die Simulationswerte berechnet und an den WinCC-Simulator übergeben.

### Variable löschen

Um eine Variable aus dem Variablen-Simulator zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Zelle mit dem Variablennamen.
2. Wählen Sie den Menübefehl "Bearbeiten > Ausschneiden".  
Die Variable wird aus der Tabelle entfernt.

### Siehe auch

Grundlagen zur Simulation (Seite 3943)

Arbeiten mit dem Variablen-Simulator (Seite 3946)

### 10.10.2.3 Arbeiten mit dem Variablen-Simulator

#### Erläuterungen zum Variablen-Simulator

Der Variablen-Simulator besteht aus folgenden Spalten:

| Spalte    | Beschreibung  |
|-----------|---|
| Variable  | Legt die Variablen für die Simulation fest.             |
| Datentyp  | Zeigt den Datentyp der ausgewählten Variable.           |
| akt. Wert | Zeigt den simulierten Wert der eingetragenen Variablen. |



| Spalte             | Beschreibung   |
|--------------------|--|
| Format             | Legt das gewählte Format fest, in welchem die Variablenwerte simuliert werden: <ul style="list-style-type: none"> <li>• Dezimal (1, 2, 3, 4, ...)</li> <li>• Hexadezimal (03CE, 01F3, ...)</li> <li>• Binär (0 und 1)</li> </ul> |
| Schreibzyklus      | Legt den gewählten Zeitabstand fest, in dem die aktuellen Variablenwerte simuliert werden. Wenn Sie z. B. den Wert "2" eintragen, wird alle 2 Sekunden der aktuelle Variablenwert angezeigt.                                     |
| Simulation         | Zeigt, auf welche Art die Variablenwerte während der Simulation verarbeitet werden.  |
| Wert setzen        | Legt den gewählten Wert für die entsprechende Variable fest. Die Simulation startet bei dem eingesetzten Wert.   |
| minWert<br>maxWert | Legt den Wertebereich der Variable fest. Sie legen dazu einen Minimal- und einen Maximalwert an. Standardmäßig liegen die Voreinstellungen bei -32768 für den Minimalwert und 32767 für den Maximalwert.                         |
| Periode            | Enthält die Periode, in der sich der Variablenwert bei den Simulationsarten "Inkrement" und "Dekrement" wiederholt.  |
| Start              | Startet die Simulation der Variable nach den vorangegangenen Eingaben.   |

## Simulationsarten

Der Simulator stellt sechs verschiedene Simulationsarten zur Verfügung. Während der Simulation werden die projizierten Variablen mit realitätsnahen Werten versorgt.

| Simulationsart | Beschreibung  |
|----------------|---|
| Sinus          | Verändert den Variablenwert in Form einer Sinuskurve. Die Darstellung erfolgt als periodische, nicht lineare Funktion.  |
| Zufall         | Stellt zufällig erzeugte Werte zur Verfügung. Der Variablenwert wird über eine Zufallsfunktion verändert.   |
| Inkrement      | Erhöht den Variablenwert kontinuierlich bis zu einem maximal angegebenen Wert. Beginnt nach Erreichen des Maximums wieder beim Minimum. Der Verlauf der Werte entspricht einer positiven Sägezahnkurve.     |
| Dekrement      | Verringert den Variablenwert kontinuierlich bis zu einem minimal angegebenen Wert. Beginnt nach Erreichen des Minimums wieder beim Maximum. Der Verlauf der Werte entspricht einer negativen Sägezahnkurve. |
| Bit schieben   | Schiebt ein gesetztes Bit kontinuierlich um eine Position weiter. Die vorherige Position wird jeweils zurückgesetzt. Damit können Sie z. B. die Meldungen für ein Bediengerät testen.                       |
| <Anzeige>      | Der aktuelle Variablenwert wird statisch angezeigt.   |

### Beispiel: Variablen simulieren mit der Simulationsart "Bit schieben"

Um die Variablen mit der Simulationsart "Bit schieben" zu simulieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie das zu simulierende Projekt.
2. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator". Der Variablen-Simulator wird geöffnet.
3. Wählen Sie in der Spalte "Variable" eine Variable aus Ihrem Projekt aus.

4. Wählen Sie in der Spalte "Format" "Bin" aus.
5. Geben Sie in der Spalte "Schreibzyklus" den Wert "1" ein.
6. Wählen Sie in der Spalte "Simulation" die Simulationsart "Bit schieben" aus.
7. Geben Sie in der Spalte "Wert setzen" den Wert "1" ein.
8. Aktivieren Sie die Variable mit dem Kontrollkästchen "Start".

## Ergebnis

Der Simulator testet die ausgewählte Variable bitweise folgendermaßen:

| Simulationswerte      | Byte für Meldungen |
|-----------------------|--------------------|
| Gesetzter Anfangswert | 00000001           |
| 1. Simulationswert    | 00000010           |
| 2. Simulationswert    | 00000100           |
| 3. Simulationswert    | 00001000           |
| ....                  | ...                |

Sie sehen in Runtime, ob bei dem jeweiligen Wert z. B. die gewünschte Meldung ausgegeben wird.

## Siehe auch

Projekt simulieren (Seite 3944)

### 10.10.2.4 Einschränkungen bei der Simulation

#### Meldungen mit dynamischen Paramtern

Wenn Sie Variablen oder Textlisten als externe Variablen für Meldungen verwenden, dann werden die Meldungen nicht angezeigt.

Für die Simulation von Meldungen im Variablen-Simulator sind nur interne Variablen freigegeben.

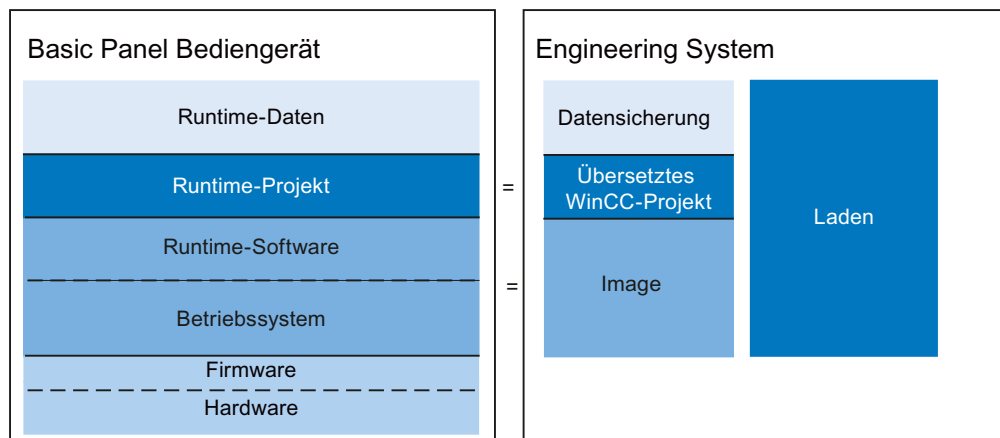
Verwenden Sie PLCSim um dynamische Paramter zu simulieren.

## 10.10.3 Wartung des Bediengeräts

### 10.10.3.1 Übersicht zur Wartung des Bediengeräts (Basic Panels)

#### Aufbau

Die folgende Abbildung zeigt die Software-Komponenten eines Bediengeräts und ihren Bezug zum Engineering System.



#### Runtime-Daten

Die Runtime-Daten werden während des laufenden Betriebs der Anlage erzeugt und auf dem Bediengerät gespeichert. Zu Ihnen gehören z. B. Rezepturen und Daten zur Benutzerverwaltung. Diese Daten werden beim Laden überschrieben. Sichern Sie bei Bedarf diese Daten vor den Laden eines Runtime-Projekts.

#### Runtime-Projekt

Das Runtime-Projekt enthält die übersetzten Projektierungsdaten für ein Bediengerät. Sie laden das Runtime-Projekt aus WinCC auf das Bediengerät.

#### Runtime-Software und Betriebssystem

Runtime-Software und Betriebssystem eines Bediengeräts bilden gemeinsam das Image. Für ein Bediengerät gibt es verschiedenen Images. Alle Images eines Bediengeräts sind in WinCC verfügbar. Abhängig von der Projektierung laden Sie bei Bedarf das passende Image gemeinsam mit dem Runtime-Projekt auf das Bediengerät.

#### Firmware und Hardware

Im Auslieferungszustand des Bediengeräts sind Firmware und Hardware fertig konfiguriert.

### 10.10.3.2 ProSave

#### Einleitung

Bei der WinCC-Installation wird das Service-Tool "ProSave" standardmäßig installiert. Die ProSave-Funktionen rufen Sie in WinCC über das Menü "Online > Bediengeräte Wartung" auf.

#### Funktionsumfang

ProSave stellt alle Funktionen zur Verfügung, die zur Übertragung von Dateien auf das Bediengerät notwendig sind:

- Datensicherung und Wiederherstellung von gesicherten Daten
- Betriebssystemaktualisierung
- Kommunikationseinstellungen

#### Siehe auch

Datensicherung des Bediengeräts (Seite 3950)

Aktualisierung des Betriebssystems (Seite 3952)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 3949)

### 10.10.3.3 Datensicherung des Bediengeräts

#### Einleitung

Wenn Sie die Daten eines Bediengeräts regelmäßig sichern, verkürzen Sie die Ausfallzeiten z. B. bei einem Gerätetausch auf ein Minimum. Sie übertragen die gesicherten Daten auf das Bediengerät und stellen damit den ursprünglichen Zustand wieder her.

#### Datensicherung mit WinCC

Wenn ein Bediengerät mit dem Projektierungs-PC verbunden ist, können Sie die Daten des Bediengeräts von dem Projektierungs-PC aus mit WinCC sichern und wiederherstellen.

#### Umfang der Datensicherung

Welche Daten gesichert und wiederhergestellt werden, ist vom Bediengerättyp abhängig:

- Vollständige Sicherung.  
Je nach Bediengerät: Runtime, Firmware, Betriebssystem, Projektierung, Rezepturen, Benutzerverwaltung, Einstelldaten
- Nur Rezepturen
- Nur Benutzerverwaltung
- Rezepturen als CSV-Datei

Bei der Datensicherung von einem Bediengerät wird eine komprimierte Datei mit der Endung "\*.psb" erstellt.

Grundsätzlich können Sie die Daten auf ein beliebiges Speichermedium sichern. Wenn das Bediengerät in einem Netzwerk integriert ist, können Sie die Daten auch auf einem Server sichern.

---

#### **Hinweis**

##### **Umfang der Datensicherung**

Bei der Datensicherung wird der Inhalt des Flash-Speichers gesichert. Beachten Sie Folgendes bei einer vollständigen Sicherung und Wiederherstellung des Datenbestandes für Bediengeräte:

- Bei einer vollständigen Wiederherstellung aller Daten werden alle vorher auf dem Gerät befindlichen Daten inklusive Betriebssystem unwiderruflich gelöscht.
  - Wenn die Wiederherstellung der Daten unterbrochen wurde, führen Sie den Befehl "Auf Werkseinstellungen zurücksetzen" aus. Starten Sie danach die Wiederherstellung der Daten erneut.
- 

#### **Siehe auch**

Daten des Bediengeräts sichern und wiederherstellen (Seite 3951)

ProSave (Seite 3950)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 3949)

#### **10.10.3.4 Daten des Bediengeräts sichern und wiederherstellen**

---

#### **Hinweis**

Verwenden Sie die Wiederherstellung von Projektdaten nur auf Bediengeräten, die mit derselben Projektierungs-Software projiziert wurden.

---

#### **Voraussetzung**

- Das Bediengerät ist mit dem Projektierungs-PC verbunden
- Das Bediengerät ist in der Projektnavigation ausgewählt
- Wenn ein Server zur Datensicherung verwendet wird: Der Projektierungs-PC hat Zugriff auf den Server

### Daten des Bediengeräts sichern

Um die Daten des Bediengeräts zu sichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online > Bediengeräte Wartung" den Befehl "Sichern".  
Der Dialog "SIMATIC ProSave" wird geöffnet.
2. Wählen Sie unter "Datenart" aus, welche Daten des Bediengeräts gesichert werden.
3. Geben Sie unter "Speichern unter" den Dateinamen der Sicherungsdatei ein.
4. Klicken Sie auf "Start Backup".

Die Datensicherung wird gestartet. Je nach gewählter Verbindung nimmt die Sicherung einige Zeit in Anspruch.

### Daten des Bediengeräts wiederherstellen

Um die Daten des Bediengeräts wiederherzustellen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online > Bediengeräte Wartung" den Befehl "Wiederherstellen".
2. Geben Sie unter "Öffnen von ..." den Dateinamen der Sicherungsdatei ein.  
Unter "Inhalt" werden Informationen zur gewählten Sicherungsdatei angezeigt.
3. Klicken Sie auf "Start Restore".

Die Wiederherstellung wird gestartet. Je nach gewählter Verbindung nimmt dieser Vorgang einige Zeit in Anspruch.

### Backup / Restore über Dialog "Backup/Restore" im Control Panel des Bediengeräts

Die Funktion über "Backup / Restore" ist für MMC, SD-Speicherkarten sowie USB-Massenspeicher frei gegeben.

### Siehe auch

Datensicherung des Bediengeräts (Seite 3950)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 3949)

### 10.10.3.5 Aktualisierung des Betriebssystems

#### Einleitung

Wenn das Image eines Bediengeräts einen nicht zur Projektierung passenden Versionsstand hat, wird es beim Laden des Projekts auf Nachfrage automatisch aktualisiert. Danach wird das Laden fortgesetzt. Wenn Sie die Aktualisierung des Images ablehnen, wird das Laden abgebrochen. Mit der Aktualisierung des Images wird auch das Betriebssystem des Bediengeräts aktualisiert.

## Aktualisierung des Images

Um ein Image zu aktualisieren, verbinden Sie das Bediengerät mit dem Projektierungs-PC. Verwenden Sie für diese Verbindung nach Möglichkeit die Schnittstelle mit der höchsten Bandbreite, z. B. Ethernet. Über eine serielle Verbindung kann das Aktualisieren des Images bis zu einer Stunde dauern.

## "Rücksetzen auf Werkseinstellungen"

Wenn das Betriebssystem auf dem Bediengerät nicht mehr funktionsfähig ist, aktualisieren Sie das Betriebssystem und setzen das Bediengerät auf die Werkseinstellungen zurück.

---

### Hinweis

Um die Werkseinstellungen über Ethernet zurückzusetzen, benötigen Sie Folgendes:

- MAC-Adresse des Bediengeräts
- Verfügbare IP-Adresse
- Auf Ethernet TCP/IP eingestellte PG/PC-Schnittstelle des Projektierungs-PC

Sie konfigurieren die PG/PC-Schnittstelle über die Systemsteuerung des Projektierungs-PC. Wählen Sie unter "Zugangspunkt der Applikation" den Eintrag "S7ONLINE (STEP7) -> TCP/IP".

---

## Siehe auch

Betriebssystem des Bediengeräts aktualisieren (Seite 3953)

ProSave (Seite 3950)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 3949)

### 10.10.3.6 Betriebssystem des Bediengeräts aktualisieren

Wenn Sie das Betriebssystem aktualisieren, aktualisieren Sie damit auch die Runtime-Software auf dem Bediengerät und ändern die Geräteversion.

| ACHTUNG  |
|--|
| <p><b>Betriebssystemaktualisierung löscht alle Daten auf dem Bediengerät</b></p> <p>Wenn Sie das Betriebssystem aktualisieren, werden alle vorhandenen Daten auf dem Zielsystem gelöscht. Sichern Sie deshalb zuerst folgende Daten:</p> <ul style="list-style-type: none"><li>• Benutzerverwaltung</li><li>• Rezepturen</li></ul> |



## Voraussetzung

- Das Bediengerät ist mit dem Projektierungs-PC oder dem PC mit ProSave verbunden
- Das Bediengerät ist in der Projektnavigation ausgewählt

## Betriebssystem aktualisieren

Um das Betriebssystem zu aktualisieren, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online > Bediengeräte Wartung " den Befehl "Betriebssystem aktualisieren".
2. Wählen Sie am Projektierungs-PC in WinCC im Menü "Online > Bediengeräte Wartung" den Befehl "Betriebssystem aktualisieren".  
Der Dialog "SIMATIC ProSave [OS-Update]" wird geöffnet. Der Pfad mit dem Image ist bereits voreingestellt.
3. Wählen Sie bei Bedarf einen anderen Pfad für das Image aus, das Sie auf das Bediengerät übertragen möchten.
4. Klicken Sie auf "Update OS".

Die Aktualisierung wird gestartet. Je nach gewählter Verbindung kann dieser Vorgang einige Zeit in Anspruch nehmen.

## Bediengerät auf Werkseinstellungen zurücksetzen

Um das Bediengerät wieder auf Werkseinstellungen zurückzusetzen, gehen Sie folgendermaßen vor:

1. Schalten Sie die Stromversorgung für das Bediengerät aus.
2. Wählen Sie am Projektierungs-PC in WinCC im Menü "Online > Bediengeräte Wartung" den Befehl "Betriebssystem aktualisieren".  
Der Dialog "SIMATIC ProSave [OS-Update]" wird geöffnet. Der Pfad mit dem Image ist bereits voreingestellt.
3. Wählen Sie bei Bedarf einen anderen Pfad für das Image aus, das Sie auf das Bediengerät übertragen möchten.
4. Aktivieren Sie "Rücksetzen auf Werkseinstellungen".
5. Klicken Sie auf "Update OS".
6. Um das "Rücksetzen auf Werkseinstellungen zu starten", schalten Sie die Stromversorgung des Bediengeräts wieder ein.

Dieser Vorgang kann einige Zeit in Anspruch nehmen.

## Ergebnis

Das Betriebssystem des Bediengeräts ist jetzt auf dem aktuellen Stand.

## Siehe auch

Aktualisierung des Betriebssystems (Seite 3952)

Übersicht zur Wartung des Bediengeräts (Basic Panels) (Seite 3949)



## 10.10.4 Referenz

### 10.10.4.1 Fehlermeldungen beim Laden von Projekten

#### Mögliche Probleme beim Laden

Während ein Projekt auf ein Bediengerät geladen wird, werden im Ausgabefenster Statusmeldungen über den Fortschritt des Ladevorgangs ausgegeben.

Wenn beim Laden des Projekts auf das Bediengerät Probleme auftreten, ist die Ursache in den meisten Fällen eine der Folgenden:

- Falsche Einstellungen für Laden auf dem Bediengerät
- Falscher Bediengerättyp im Projekt
- Bediengerät ist nicht mit Projektierungs-PC verbunden.

Nachfolgend sind die häufigsten Fälle für einen Abbruch des Ladens mit möglicher Ursache und Abhilfe aufgeführt.

#### Das serielle Laden bricht ab

Mögliche Abhilfe: Wählen Sie eine kleinere Baudrate.

#### Das Laden wird wegen eines Kompatibilitätskonflikts abgebrochen

| Mögliche Ursache  | Abhilfe   |
|---|---|
| Der Projektierungs-PC ist mit einem falschen Gerät verbunden, z. B. einer Steuerung | Kontrollieren Sie die Verkabelung.<br>Korrigieren Sie die Kommunikationsparameter |

#### Projekt wird nicht geladen

| Mögliche Ursache  | Abhilfe   |
|---|---|
| Verbindung zum Bediengerät kann nicht aufgebaut werden (Meldung im Ausgabefenster)    | Prüfen Sie die physikalische Verbindung zwischen Projektierungs-PC und Bediengerät.<br>Prüfen Sie, ob sich das Bediengerät im Transfermodus befindet. Ausnahme: Fernsteuerung |
| Im Gerätemanager von Windows ist der Standard-Kommunikationstreiber nicht eingetragen | Kontrollieren Sie im Eigenschaftsfenster des Gerätemanagers den Gerätestatus des COM-Anschlusses.   |

**Laden über die MPI/DP-Schnittstelle funktioniert nicht**

| Mögliche Ursache  | Abhilfe  |
|---|--|
| Am CP ist der "projektierte Betrieb" eingestellt, z. B. bei Verwendung der SIMATIC NET CD.  | Stellen Sie mit der Anwendung "PC-Station einstellen" für den CP den "PG-Betrieb" ein.<br>Kontrollieren Sie die Netzparameter "Baudrate" und "MPI-Adresse".<br>Laden Sie das Projekt aus WinCC auf den CP.<br>Stellen Sie für den CP wieder den "projektierten Betrieb" ein.   |
| Im PG/PC-Panel steht der Zugangspunkt "S7ONLINE" nicht auf einem physikalischen Gerät wie CP5611 (MPI). Ursache kann die Installation von "SIMATIC NET CD 7/2001" sein. | Stellen Sie mit der Anwendung "PG/PC-Panel" oder "PC-Station einstellen" den Zugangspunkt "S7ONLINE" auf ihr gewünschtes Gerät ein.<br>Kontrollieren Sie die Netzparameter "Baudrate" und "MPI-Adresse".<br>Laden Sie das Projekt aus WinCC auf das Bediengerät.<br>Stellen Sie den Zugangspunkt "S7ONLINE" wieder zurück auf das ursprüngliche Gerät. |

**Komplexität der Projektierung ist zu groß**

| Mögliche Ursache  | Abhilfe  |
|---|--|
| Ihre Projektierung enthält zu viele verschiedene Objekte oder Optionen für das ausgewählte Bediengerät. | Entfernen Sie alle Objekte eines Typs, z. B. alle Grafikanzeigen |

**10.10.4.2 Anpassen des Projekts für ein anderes Bediengerät**

**Einleitung**

Wenn Sie ein WinCC Projekt auf ein Bediengerät laden, prüft WinCC, ob das Bediengerät mit dem im Projekt verwendeten Bediengerätetyp übereinstimmt. Wenn die Bediengerätetypen nicht übereinstimmen, erhalten Sie beim Ladevorgang eine Meldung.

Der Ladevorgang wird abgebrochen.

## Anpassen des Projekts für das Bediengerät

Um das Projekt auf das angeschlossene Bediengerät zu laden, müssen Sie das Projekt entsprechend anpassen.

- Fügen Sie in der Projektnavigation ein neues Bediengerät ein. Wählen Sie in der Bediengeräteauswahl den richtigen Bediengerätetyp aus.
- Kopieren Sie die projektierten Bestandteile aus dem bisherigen Bediengerät in das neu angelegte Bediengerät.  
Viele Bestandteile kopieren Sie direkt in der Projektnavigation und der Detailansicht. Kopieren Sie z. B. den kompletten Ordner "Bilder" mithilfe des Kontextmenüs in den Ordner Bilder des neuen Bediengeräts.
- Bei einigen Einträgen in der Projektnavigation, für die im Kontextmenü der Befehl "Kopieren" nicht verfügbar ist, kopieren Sie die Inhalte über die Detailansicht.
- Selektieren Sie z. B. den Eintrag "Rezepturen" in der Projektnavigation. Die Rezepturen werden in der Detailansicht angezeigt.
- Selektieren Sie die Rezepturen in der Detailansicht und ziehen Sie diese per Drag&Drop auf den Eintrag "Rezepturen" des neuen Bediengeräts. Die Rezepturen werden kopiert. Eine Mehrfachselektion in der Detailansicht ist möglich.
- Projektieren Sie nicht kopierbare Bestandteile neu, z. B. Verbindungen, Bereichszeiger und Meldungen.
- Speichern Sie das Projekt auch zwischendurch.
- Übersetzen Sie das Projekt komplett.
- Wenn die Übersetzung fehlerfrei durchläuft, laden Sie das Projekt auf das Bediengerät.

## Verknüpfung von Referenzen

Wenn Sie Objekte kopieren, werden die Referenzen zu verbundenen Objekten mitkopiert. Nach dem Kopieren der verbundenen Objekte werden die Referenzen wieder miteinander verknüpft.

Beispiel:

Sie kopieren ein Bild, in dem Objekte mit Variablen verknüpft sind. Nach dem Einfügen des Bilds in das neue Bediengerät sind die Variablennamen an den einzelnen Objekten eingetragen. Die Variablennamen sind rot markiert, da die Referenzen offen sind. Wenn Sie anschließend die Variablen kopieren und in das neue Bediengerät einfügen, werden die offenen Referenzen geschlossen. Die rote Markierung der Variablennamen wird aufgehoben.

Damit Referenzen zu verbundenen Objekten in der Steuerung vervollständigt werden, müssen Sie zuerst eine Verbindung zu der Steuerung projektieren.

## Verwendung des Infobereichs

Wenn Sie das Projekt für das neue Bediengerät übersetzen, werden im Inspektorfenster auf der Registerkarte "Info" Fehler und Warnungen ausgegeben. Über den Kontextmenübefehl "Gehe zu" gelangen Sie direkt zur betreffenden Stelle, um den Fehler oder die Warnung zu korrigieren.

Arbeiten Sie die Liste der Fehler und Warnungen nacheinander ab.

Wenn die Übersetzung fehlerfrei durchläuft, laden Sie das Projekt auf das Bediengerät.

### 10.10.4.3 Herstellen der Verbindung zum Bediengerät

#### Einleitung

Um ein WinCC Projekt auf ein Bediengerät zu laden, muss die Verbindung zwischen dem Projektierungs-PC und dem Bediengerät aufgebaut und richtig konfiguriert sein. Wenn die Verbindung während des Ladens nicht zu Stande kommt, wird das Laden abgebrochen.

#### Verbindung zwischen Projektierungs-PC und Bediengerät herstellen

1. Kontrollieren Sie die Kabelverbindung zwischen Bediengerät und Projektierungs-PC.
2. Öffnen Sie in WinCC den Editor "Geräte & Netze" und aktivieren Sie die Netzsicht.
3. Selektieren Sie in der Netzsicht das Subnetz und prüfen Sie die Einstellungen für das Subnetz.
4. Selektieren Sie in der Netzsicht oder in der Gerätesicht die Schnittstelle des Bediengeräts und prüfen Sie im Inspektorfenster die Verbindungsparameter.
5. Schalten Sie das Bediengerät ein und tippen Sie im Loader auf die Schaltfläche "Control Panel".  
Das Control Panel wird geöffnet.
6. Tippen Sie im Control Panel zwei Mal auf Transfer.  
Der Dialog "Transfer Settings" wird geöffnet.
7. Kontrollieren Sie die Einstellungen und tippen Sie anschließend auf "Advanced".  
Der Dialog "[Protokoll\*] Settings" wird geöffnet.  
\*: Der Titel des Dialoges ist abhängig vom verwendeten Protokoll, z. B. "PROFIBUS Settings".
8. Kontrollieren Sie die erweiterten Einstellungen und schließen Sie den Dialog mit "OK".

#### Wichtige Einstellungen

Wenn Sie die Verbindungseinstellungen kontrollieren, prüfen Sie insbesondere folgende Parameter:

- Netzwerkadressen und Stationsadressen
- Eingestellte Übertragungsgeschwindigkeit
- Master am Bus; generell ist nur ein Master zulässig.

Wenn Sie für die Verbindung einen einstellbaren Adapter verwenden, prüfen Sie am Adapter ebenfalls die Einstellungen, z. B. Übertragungsgeschwindigkeit, Master am Bus.

## 10.11 In Runtime bedienen

### 10.11.1 Grundlagen

#### 10.11.1.1 Überblick

#### Projektierungs- und Prozessführungsphase

Für das Bedienen und Beobachten von Aufgaben in der Prozess- und Fertigungsautomatisierung werden Bediengeräte eingesetzt. Mit den auf den Bediengeräten angezeigten Anlagenbildern lassen sich ablaufende Prozesse anschaulicher darstellen. Das Bediengeräte-Projekt, das unter anderem die Anlagenbilder enthält, wird in der Projektierungsphase erstellt.

Für Einsatz in der Prozessführungsphase transferieren Sie das Projekt auf das Bediengerät. Als eine weitere Voraussetzung für die Prozessführungsphase gilt, dass das Bediengerät online an eine Steuerung gekoppelt ist. Danach ist die Prozessführungsphase – das Bedienen und Beobachten – laufender Arbeitsprozesse möglich.

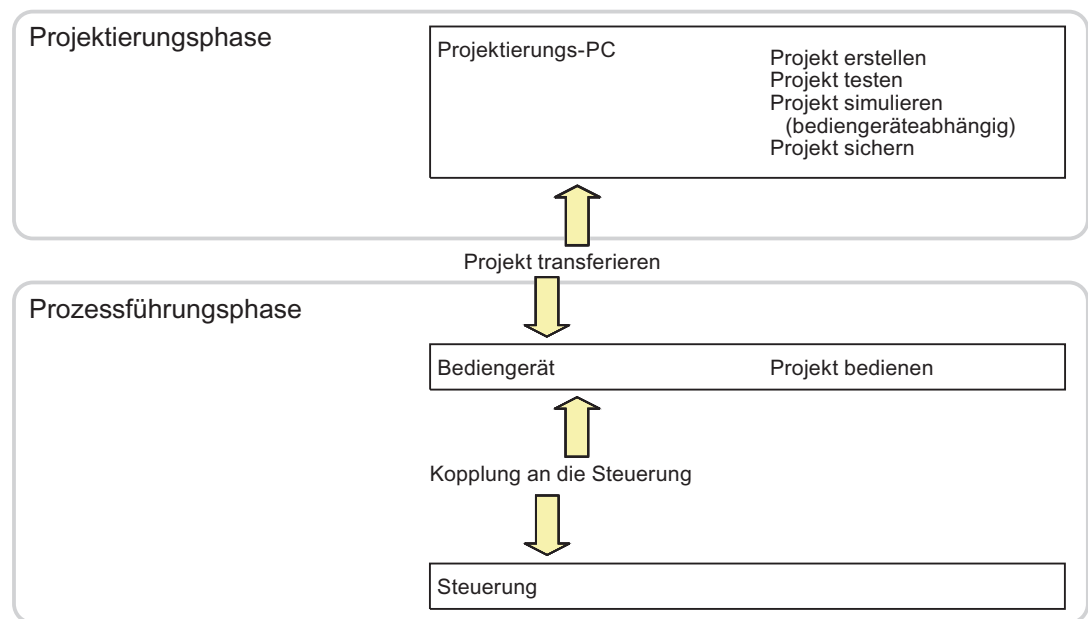


Bild 10-1 Projektierungs- und Prozessführungsphase

### Projekt auf das Bediengerät laden

Um ein Projekt auf ein Bediengerät zu laden, sind folgende Vorgehensweisen wählbar:

- Laden vom Projektierungs-PC
- Wiederherstellung über ProSave von einem PC  
Dabei wird ein gesichertes Projekt von einem PC auf das Bediengerät geladen. Auf diesem PC muss die Projektierungs-Software nicht installiert sein.

Diese Vorgehensweisen sind sowohl bei der Erstinbetriebnahme als auch bei der Wiederinbetriebnahme eines Projekts wählbar.

### Erstinbetriebnahme und Wiederinbetriebnahme

- Bei der Erstinbetriebnahme ist auf dem Bediengerät noch kein Projekt vorhanden. Diesen Zustand hat das Bediengerät auch nach dem Aktualisieren des Betriebssystems.
- Bei der Wiederinbetriebnahme wird ein bereits auf dem Bediengerät vorhandenes Projekt ersetzt.

#### 10.11.1.2 Variablen in Runtime

##### Definition

Variablen entsprechen festgelegten Speicherplätzen am Bediengerät, in die Werte geschrieben und/oder aus denen Werte gelesen werden. Dies kann von der Steuerung aus geschehen oder per Bedienung am Bediengerät.

#### 10.11.1.3 Systemfunktionen in Runtime

##### Verwendung

Die Systemfunktionen haben in Runtime folgende Zwecke:

- Den Prozess zu steuern
- Die Eigenschaften des Bediengeräts zu nutzen
- Die Systemeinstellungen online am Bediengerät zu konfigurieren

In WinCC ist jede Systemfunktion mit einem Objekt und einem Ereignis verknüpft. Sobald das Ereignis eintritt, wird die Systemfunktion ausgelöst.

##### Systemfunktionen

Systemfunktionen sind vordefinierte Funktionen, mit denen viele Aufgaben in Runtime realisiert werden, z. B.:

- Berechnungen, z. B. das Erhöhen eines Variablenwerts um einen bestimmten oder variablen Betrag
- Archivfunktionen, z. B. das Starten eines Prozesswertarchivs

- Einstellungen, z. B. das Wechseln der Steuerung oder das Setzen eines Bits in der Steuerung
- Meldungen, z. B. nach dem Ändern des Benutzers

## Ereignisse

Welches Ereignis als Auslöser für die Ausführung einer Systemfunktion definiert wird, hängt vom jeweiligen Objekt und von der gewählten Funktion ab.

So gehören z. B. zu dem Objekt "Variable" die Ereignisse "Wertänderung", "Untergrenze unterschritten" und "Obergrenze überschritten". Zu dem Objekt "Bild" gehören die Ereignisse "Aufgebaut" und "Abgebaut".

## 10.11.2 Projekte in Betrieb nehmen

### 10.11.2.1 Einstellungen der Runtime-Software

In der Projektierungs-Software WinCC nehmen Sie folgende Einstellungen für die Runtime-Software vor:

#### Anzeige am Zielsystem

In WinCC stellen Sie ein, wie das generierte Projekt in Runtime dargestellt wird. Bei den Basic Panels ist die Bildschirmauflösung fest. Wenn das Bild größer ist, als die eingestellte Bildschirmauflösung, erscheinen Bildlaufleisten.

Um die Task-Leiste auszuschalten, wählen Sie im Startmenü den Befehl "Einstellungen > Task-Leiste". Deaktivieren Sie im Dialog "Eigenschaften von Task-Leiste" "Immer im Vordergrund" und "Automatisch im Hintergrund".

#### Schriftart der Dialoge

Die Texte der Dialoge werden in der Standard-Schriftart dargestellt. Die Standard-Schriftart legen Sie im Editor "Geräteeinstellungen" unter "Sprache & Schriftart" fest.

#### Programmumschaltung sperren

Um zu verhindern, dass der Bediener in Runtime andere Aufwendungen aufruft, sperren Sie die Programmumschaltung.

Klicken Sie im Editor "Geräteeinstellungen" auf "Fenster" und aktivieren Sie "Programmumschaltung sperren". Schalten Sie zusätzlich unter Windows die Task-Leiste aus.

---

**Hinweis**

**Runtime beenden**

Wenn Sie die Programmumschaltung sperren, projektieren Sie in Ihrem Projekt unbedingt die Systemfunktion "StoppeRuntime" z. B. an eine Funktionstaste oder an eine Schaltfläche. Sonst wird weder Runtime noch Windows beendet.

---

## Bildschirmschoner

Für die meisten Bildschirme ist kein Bildschirmschoner mehr nötig, ja sogar schädlich. Diese Monitore schalten sich ab, sobald sich für eine einstellbare Zeit das Video-Signal nicht mehr ändert. Ein herkömmlicher Bildschirmschoner verhindert diese die Lebensdauer verlängernde Abschaltung.

---

**Hinweis**

**Frei gegebene Bildschirmschoner**

Wenn Sie einen Bildschirmschoner einsetzen wollen, so beachten Sie, dass für Runtime nur die im Lieferumfang von Windows enthaltenen Standard-Bildschirmschoner frei gegeben sind.

---

Stellen Sie sicher, dass für den PC, auf dem die Runtime-Software abläuft, die richtige Zeitzone eingestellt ist. Die Zeitzone stellen Sie in Windows im Startmenü unter "Einstellungen > Systemsteuerung > Datum und Uhrzeit" ein.

### 10.11.2.2 Projekt laden

## Überblick

Zum Laden des Projekts sind mehrere Szenarien möglich:

- Die Runtime-Software ist auf dem gleichen System installiert wie die Projektierungs-Software.
- Die Runtime-Software ist auf einem anderen System installiert als die Projektierungs-Software. Das Projekt wird vom Projektierungs-PC auf das Zielsystem geladen. Für den Transfer werden die Bediengeräte mit dem Projektierungs-PC verbunden. Als weitere Voraussetzung gilt, dass der Transfermodus an den Bediengeräten und in WinCC übereinstimmt.

---

**Hinweis**

Abhängig von der Projektierung gibt es beim Laden Sicherheitsabfragen. Die auf dem Bediengerät vorhandenen Rezepturdaten und die Kennwortliste werden auf Nachfrage überschrieben.

---



## Projektierungs-Software und Runtime-Software auf demselben System

Wenn sich die Projektierungs-Software und die Runtime-Software auf demselben System befinden, gehen Sie folgendermaßen vor:

1. Erstellen Sie Ihr Projekt und übersetzen Sie es anschließend.
2. Starten Sie Runtime direkt aus der laufenden Projektierungs-Software. Wählen Sie im Menü "Online" den Befehl " Runtime starten".
3. Wenn eine Kommunikation zur Steuerung eingerichtet ist, können Sie das Projekt mit der Steuerung testen oder betreiben.

## Projektierungs-Software und Runtime-Software auf verschiedenen Systemen

Wenn sich die Projektierungs-Software und die Runtime-Software auf zwei verschiedenen Systemen befinden, gehen Sie folgendermaßen vor:

1. Erstellen Sie Ihr Projekt und übersetzen Sie es anschließend. Nähere Hinweise hierzu finden Sie in "Projekt übersetzen".
2. Wenn Sie die Datei über Kabel laden wollen:  
Verbinden Sie das Bediengerät über ein geeignetes Standardkabel je nach gewünschter Übertragungsart mit dem Projektierungs-PC und schalten Sie das Bediengerät ein.
3. Schalten Sie das Bediengerät in den Transfermodus.  
Um den Transfermodus zu starten, betätigen Sie im Loader die Schaltfläche "Transfer".  
Alternativ projektieren Sie an ein Bedienobjekt die Systemfunktion "SetzeBetriebsartGeraet".
4. Laden Sie vom Projektierungs-PC das Projekt auf Ihr Zielsystem. Nähere Hinweise hierzu finden Sie unter "Projekt laden".

---

### Hinweis

Wenn es sich beim Bediengerät um einen PC handelt, können Sie die generierte Datei ohne den Loader z. B. über Ethernet übertragen. Am PC doppelklicken Sie dann auf die Datei und starten damit Runtime.

---

### 10.11.2.3 Runtime an der Engineering-Station starten

#### Einleitung

Während Sie in WinCC projektieren, können Sie gleichzeitig an der Engineering-Station Ihr Projekt in Runtime starten. Dieses so genannte Online-Projektieren unterliegt jedoch bestimmten Einschränkungen.

Während an der Engineering-Station die Runtime läuft, wird das Projekt nicht im Hintergrund übersetzt. Wenn Sie nach dem Beenden der Runtime das Projekt auf ein Bediengerät laden, wird der zwischenzeitlich geänderte Anteil des Projekts automatisch übersetzt. Alternativ können Sie das Übersetzen auch manuell starten.

Beim Start des Projekts in Runtime werden die Einstellungen wirksam, die Sie in Ihrem Projekt für das Bediengerät im Editor "Konfiguration" hinterlegt haben.

## Voraussetzung

Auf der Engineering-Station ist ein Projekt geöffnet.

## Vorgehen

Um die Runtime an der Engineering-Station zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie das gewünschte Bediengerät in der Projektnavigation aus.
2. Wählen Sie den Menübefehl "Online > Runtime starten".
3. Wenn Sie nach dem Starten von Runtime an der Engineering-Station das Projekt ändern, wählen Sie im Kontextmenü des Bediengeräts den Befehl "Übersetzen > Software".

In der Runtime an der Engineering-Station wird das aktualisierte Projekt angezeigt.

### 10.11.2.4 Runtime auf dem Bediengerät starten

## Einleitung

Sobald Sie das Projekt auf Bediengerät geladen haben, können Sie das Projekt in Runtime starten. Das Projekt wird auf dem Bediengerät in einer Datei mit folgender Endung gespeichert:

- Basic Panels sowie OP 73, OP 77A und TP 177A: "\*.srt"

Beim Start des Projekts in Runtime werden die Einstellungen wirksam, die Sie in Ihrem Projekt in den "Runtime-Einstellungen" des Bediengeräts eingestellt haben.

Die Programme zum Starten von Projekten am Bediengerät finden Sie im Installationsverzeichnis von Runtime.

---

### Hinweis

#### Automatisches Beenden der Runtime

Wenn auf dem Bediengerät der automatische Transfer aktiviert ist und wenn am Projektierungs-PC ein Transfer gestartet wird, wird das laufende Projekt automatisch beendet.

Das Bediengerät wechselt dann selbsttätig in die Betriebsart "Transfer".

Deaktivieren Sie nach der Inbetriebnahmephase den automatischen Transfer, damit das Bediengerät nicht versehentlich in den Transferbetrieb geht.

Der Transferbetrieb kann ungewollte Reaktionen in der Anlage auslösen.

Um den Zugriff auf die Transfereinstellungen zu sperren und damit ein unbefugtes Ändern zu vermeiden, vergeben Sie ein Kennwort im Control Panel.

---

## Voraussetzung

Auf dem Bediengerät ist die WinCC Runtime installiert.

## Vorgehen

Auf einem Panel wird die Anwendung "RT Loader" gestartet. Abhängig von der Projektierung startet nach einer Verzögerungszeit das geladene Projekt automatisch.

Wenn das Projekt nicht automatisch startet:

1. Um die Projektdatei zu wählen, klicken Sie auf "Einstellungen" und tragen Sie unter "Projektierungsdatei" den Pfad zur Projektdatei ein.
2. Klicken Sie auf "OK" und anschließend auf "Start".

### 10.11.2.5 Projekt testen

#### Einleitung

Für den Test eines WinCC-Projekts haben Sie folgende Möglichkeiten:

#### Projekt am Projektierungs-PC testen.

- Simulator  
Der Simulator wird zum Test von Projekten aus WinCC mit internen Variablen und Prozessvariablen eingesetzt. Nähere Hinweise hierzu finden Sie unter "Projekt simulieren". Mit dem Simulator lässt sich Folgendes testen:
  - Test einer Projektierung ohne angeschlossene Steuerung
  - Test einer Projektierung mit angeschlossener Steuerung aber ohne laufenden Prozess.
  - Realisierung eines Projekts für Demonstrationszwecke

#### Projekt am Bediengerät testen

- Projekt offline auf dem Bediengerät testen  
Offline testen bedeutet, dass während des Tests keine Kommunikationsverbindung zwischen dem Bediengerät und der Steuerung besteht. Sie bedienen das Bediengerät, es werden jedoch keine Daten zur Steuerung übertragen und umgekehrt. Um am Bediengerät die Betriebsart "Offline" einzustellen, projektieren Sie an ein Bedienobjekt die Systemfunktion "SetzeBetriebsartGeraet".
- Projekt online auf dem Bediengerät testen  
Online testen bedeutet, dass während des Tests eine Kommunikationsverbindung zwischen dem Bediengerät und der Steuerung besteht. Sie bedienen die Anlage mit dem Bediengerät entsprechend der Projektierung. Um am Bediengerät die Betriebsart "Online" einzustellen, projektieren Sie an ein Bedienobjekt die Systemfunktion "SetzeBetriebsartGeraet".

## Vorgehen

Um ein Projekt ohne Steuerungsanbindung am Projektierungs-PC zu simulieren, gehen Sie folgendermaßen vor:

1. Erstellen Sie ein Projekt, wie es auch später mit angekoppelter Steuerung ablaufen soll.
2. Speichern und generieren Sie das Projekt.
3. Starten Sie den Simulator direkt aus der laufenden Projektierungs-Software. Wählen Sie den Menübefehl "Online > Runtime simulieren > Mit Variablen-Simulator".  
Wenn Sie das Projekt zum ersten Mal simulieren, wird der Simulator mit einer neuen, leeren Simulationstabelle gestartet. Wenn Sie bereits eine Simulationstabelle zu Ihrem Projekt erstellt haben, wird diese geöffnet.  
In der Simulationstabelle "\*.six" sind alle Einstellungen gespeichert, die Sie für die Simulation benötigen. Nähere Hinweise hierzu finden Sie unter "Arbeiten mit dem Variablen-Simulator".
4. Manipulieren Sie nun in der Simulationstabelle die Variablen und Bereichszeiger Ihres Projekts.  
Zwischen Simulationstabelle und Runtime schalten Sie mit den Tasten <Alt + Tab> um.  
Speichern Sie die Einstellungen für die Simulation mit dem Menübefehl "Datei > Speichern". Geben Sie der Datei einen sprechenden Namen. Die Datei erhält automatisch die Endung "\*.six".

### 10.11.2.6 Projekt beenden

#### Einleitung

Über die Projektierung bestimmen Sie, mit welchen Bedienschritten Runtime beendet wird:

#### Vorgehen

Um Runtime zu beenden, gehen Sie folgendermaßen vor:

1. Wenn Runtime läuft, können Sie Runtime über das Schließen-Symbol oder über den Task-Manager beenden.
2. Wenn Runtime läuft, betätigen Sie die dafür vorgesehene Schaltfläche, um Runtime zu beenden. Das Beenden von Runtime wird gesondert projiziert.

### 10.11.2.7 Daten des Bediengeräts sichern und wiederherstellen

#### Einleitung

Sichern Sie die Daten eines Bediengeräts in regelmäßigen Abständen.

Wenn ein Bediengerät mit einer Engineering-Station verbunden ist, können Sie die Daten des Bediengeräts von der Engineering-Station aus mit WinCC sichern und wiederherstellen.

An einem Rechner ohne WinCC Installation haben Sie die Möglichkeit, eine zentrale Datensicherung komfortabel mit ProSave vorzunehmen.

## Voraussetzung

- Das Bediengerät und die Engineering-Station oder der Rechner mit ProSave sind miteinander verbunden.
- Das Bediengerät, dessen Daten gesichert oder wiederhergestellt werden sollen, ist im Projektnavigator ausgewählt.
- In den Eigenschaften des Bediengeräts sind die Einstellungen für Laden korrekt eingestellt.
- Bei Verwendung eines gesonderten Speichermediums, z. B. eines Datenservers: Das Bediengerät ist mit dem Speichermedium verbunden.

## Vorgehensweise

Um die Daten zu sichern, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online > Bediengeräte Wartung" den Befehl "Sicherung".
2. Wählen Sie den Umfang der Sicherung aus: "Vollständige Sicherung", "Rezepturen" oder "Benutzerverwaltung".
3. Klicken Sie auf die Schaltfläche "...", wählen Sie im Dialog "Sicherungsdatei auswählen" den Speicherort aus und geben Sie einen Dateinamen an.
4. Klicken Sie auf "OK".

Die Datensicherung wird gestartet. Die Sicherung kann einige Zeit in Anspruch nehmen.

## Vorgehensweise bei der Wiederherstellung der Daten

1. Wählen Sie im Menü "Online > Bediengeräte Wartung" den Befehl "Wiederherstellen...".
2. Klicken Sie auf "..." und wählen Sie im Dialog "Öffnen" den Speicherort und die Datei aus. Im Bereich "Inhalt" wird Ihnen angezeigt, von welchem Bediengerät die Datensicherung stammt und welchen Umfang sie besitzt.
3. Klicken Sie auf "OK".

Die Wiederherstellung wird gestartet. Dieser Vorgang kann einige Zeit in Anspruch nehmen.

## 10.11.3 Sprachen in Runtime

### 10.11.3.1 Sprachen in Runtime

#### Mehrere Runtime-Sprachen einsetzen

Sie bestimmen, welche Projektsprachen in Runtime auf einem bestimmten Bediengerät zum Einsatz kommen. Die Anzahl der Runtime-Sprachen, die gleichzeitig auf dem Bediengerät zur Verfügung stehen, ist geräteabhängig. Um dem Bediener die Sprachumschaltung in Runtime zu ermöglichen, müssen Sie ein entsprechendes Bedienelement projektieren.

Bei Runtime-Start wird das Projekt in der zuletzt eingestellten Sprache angezeigt. Beim ersten Runtime-Start erscheint die Sprache mit der niedrigsten Nummer in "Reihenfolge für Sprache".

### Einstellung bei der Projektierung

Im Editor "Sprache & Schriftart" bestimmen Sie Folgendes:

- Projektsprachen, die als Runtime-Sprachen für das Bediengerät zur Verfügung stehen.
- Reihenfolge, in der die Sprachen bei einem Sprachwechsel umgeschaltet werden.

### 10.11.3.2 Runtime-Sprache einstellen

#### Einleitung

Der Editor "Sprache & Schriftart" zeigt Ihnen alle im Projekt verfügbaren Projektsprachen an. Sie legen fest, welche Projektsprachen als Runtime-Sprachen auf dem Bediengerät zur Verfügung stehen. Außerdem legen Sie fest, in welcher Reihenfolge die Sprachen bei einem Sprachwechsel umgeschaltet werden.

#### Voraussetzung

Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.

#### Vorgehen

1. Öffnen Sie unter "Geräteeinstellungen" den Editor "Sprache & Schriftart".
2. Aktivieren Sie in der Spalte "Runtime-Sprache" die Sprache, die beim ersten Runtime-Start benutzt wird.  
In der Spalte "Reihenfolge für Sprache" wird für die gewählte Sprache die Nummer "0" vergeben.
3. Aktivieren Sie in der Spalte "Runtime-Sprache" die Sprache, die bei der Sprachumschaltung als nächste Sprache aktiviert wird.  
In der Spalte "Reihenfolge für Sprache" wird für die gewählte Sprache die Nummer "1" vergeben.

| Sprache & Schriftart               |                                     |                         |                           |                    |                 |                           |
|------------------------------------|-------------------------------------|-------------------------|---------------------------|--------------------|-----------------|---------------------------|
| Runtime-Sprache und Schriftauswahl |                                     |                         |                           |                    |                 |                           |
|                                    | Runtime-Sprache                     | Reihenfolge für Spra... | Name der Sprache          | Feste Schriftart 0 | Standardschrift | Projektierte Schriftart 0 |
|                                    | <input checked="" type="checkbox"/> | 0                       | Englisch (USA)            | Tahoma             | Tahoma, 11px    | <None>                    |
|                                    | <input checked="" type="checkbox"/> | 1                       | Deutsch (Deutschland)     | Tahoma             | Tahoma, 11px    | <None>                    |
|                                    | <input type="checkbox"/>            |                         | Chinesisch (Volksrepub... | SimSun             | SimSun, 16px    | <None>                    |
|                                    | <input type="checkbox"/>            |                         | Französisch (Frankreich)  | Tahoma             | Tahoma, 11px    | <None>                    |
|                                    | <input type="checkbox"/>            |                         | Italienisch (Italien)     | Tahoma             | Tahoma, 11px    | Arial                     |
|                                    | <input type="checkbox"/>            |                         | Spanisch (International)  | Tahoma             | Tahoma, 11px    | <None>                    |

4. Aktivieren Sie weitere Sprachen in der Reihenfolge, in der sie bei der Sprachumschaltung aktiviert werden.  
Wenn Sie mehr Sprachen auswählen, als auf das Bediengerät geladen werden können, werden Sie durch eine farbige Hinterlegung in der Tabelle darauf aufmerksam gemacht.
5. Wenn Sie eine Sprache in der Reihenfolge verschieben wollen, markieren Sie die gewünschte Zeile und wählen Sie im Kontextmenü den Befehl "Move up" oder "Move down".

| Runtime-Sprache und Schriftauswahl |                                     |                         |                          |
|------------------------------------|-------------------------------------|-------------------------|--------------------------|
|                                    | Runtime-Sprache                     | Reihenfolge für Sprache | Name der Sprache         |
|                                    | <input checked="" type="checkbox"/> | 0                       | Englisch (USA)           |
|                                    | <input checked="" type="checkbox"/> | 1                       | Spanisch (International) |
|                                    | <input checked="" type="checkbox"/> | 2                       | Französisch (Frankreich) |
|                                    | <input checked="" type="checkbox"/> | 3                       | Italienisch (Italien)    |

Move up  
 Move down

## Ergebnis

Die aktivierten Runtime-Sprachen werden mit dem übersetzten Projekt auf das Bediengerät geladen.

Beim ersten Runtime-Start erscheint das Projekt in der Sprache mit der niedrigsten Nummer in "Reihenfolge für Sprache".

Wenn eine Sprachumschaltung mit der Systemfunktion "SetzeSprache" projiziert wurde, erfolgt der Wechsel zwischen den Sprachen in der Reihenfolge der vergebenen Nummern.

### 10.11.3.3 Schriftart für eine Runtime-Sprache einstellen

#### Einleitung

Im Editor "Sprache & Schriftart" legen Sie für jede Runtime-Sprache fest, in welcher Schriftart die Texte dieser Sprache auf dem Bediengerät angezeigt werden. Die Standardschriftart wird für alle Texte verwendet, bei denen Sie keine Schriftart festlegen können.

WinCC bietet hierbei nur die Schriftarten zur Auswahl an, die vom Bediengerät unterstützt werden.

Die Standardschriftart wird auch für die Darstellung der Dialoge aus dem Betriebssystem des Bediengeräts verwendet. Wenn Dialogtexte oder Dialogüberschriften nicht vollständig dargestellt werden, wählen Sie eine kleinere Schriftgröße für die Standardschriftart.

#### Voraussetzung

Im Editor "Projektsprachen" sind mehrere Sprachen aktiviert.

## Vorgehensweise

1. Öffnen Sie unter "Geräteeinstellungen" den Editor "Sprache & Schriftart".
2. Aktivieren Sie in der Spalte "Runtime-Sprache" die Sprachen, die auf dem Bediengerät angezeigt werden sollen.  
In der Spalte "Feste Schriftart 0" zeigt WinCC die in Runtime standardmäßig verwendeten Schriftarten an.
3. Legen Sie zusätzlich in der Spalte "Projektierte Schriftart 0" eine andere Schriftart pro Sprache festlegen, die Ihnen während der Projektierung zur Verfügung steht.  
Diese Schriftarten werden beim Transfer zusätzlich zum Bediengerät übertragen.
4. Wählen Sie in der Spalte "Standardschrift" die Schriftart aus, die standardmäßig verwendet wird, wenn Sie für einen Text keine Schriftart wählen können.

## Ergebnis

Die Projekttexte der gewählten Sprache werden auf dem Bediengerät in der gewählten Schriftart angezeigt.

### 10.11.3.4 Sprachumschaltung projektieren

## Einleitung

Damit auf dem Bediengerät mehrere Runtime-Sprachen zur Verfügung stehen, projektieren Sie eine Sprachumschaltung. Nur so ist es dem Bediener möglich, zwischen verschiedenen Runtime-Sprachen zu wählen.

## Methoden der Sprachumschaltung

Sie können folgende Methoden der Sprachumschaltung projektieren:

- **Direkte Sprachwahl**  
Jede Sprache wird über eine eigene Schaltfläche eingestellt. In diesem Fall erstellen Sie für jede Runtime-Sprache eine Schaltfläche. An jede Schaltfläche projektieren Sie die Systemfunktion "SetzeSprache" mit der Nummer der Sprache oder dem Sprachkürzel als Parameter.
- **Sprachumschaltung**  
Der Bediener schaltet über eine Schaltfläche in alle Sprachen um.  
An die Schaltfläche projektieren Sie die Systemfunktion "SetzeSprache" mit dem Parameter "Umschalten". Die Sprache werde in der Reihenfolge aktiviert, die Sie im Editor "Sprache & Schriftart" festgelegt haben.

Die Beschriftungen der Schaltflächen müssen bei beiden Methoden in jede verwendete Sprache übersetzt werden. Sie können zusätzlich ein Ausgabefeld projektieren, das die gerade gewählte Sprache anzeigt.



### 10.11.3.5 Besonderheiten asiatischer und östlicher Sprachen in Runtime

#### Einleitung

Bei Projekten für asiatische Sprachen sind für den Betrieb in Runtime einige Besonderheiten zu beachten.

---

#### Hinweis

Bei der Projektierung können Sie ausschließlich die asiatischen Schriftarten verwenden, die von Ihrem Projektierungsrechner unterstützt werden.

---

#### Speicherbedarf für asiatische Zeichensätze

Der Speicherbedarf ist bei Verwendung von asiatischen Sprachen naturgemäß größer, daher muss beim Übersetzen des Projekts auf entsprechende Fehlermeldungen geachtet werden.

#### Eingabe östlicher und asiatischer Zeichen (nicht ANSI)

Die Eingabe östlicher und asiatischer Zeichen in Runtime ist auf Basic Panels nicht möglich.

#### Interpretation asiatischer Zeichen

Bei der Verwendung von Sm@rtAccess und Sm@rtService können nur Zeichen verwendet werden, die auf dem Bediengerät bekannt sind. Damit asiatische Zeichen verwenden können, müssen diese im Engineering-System projiziert werden. Zusätzlich projizierte Zeichen erfordern zusätzlichen Speicherplatz auf dem Bediengerät. Beachten Sie die Größe des verfügbaren Speichers auf dem Bediengerät.

#### Schriftgröße für asiatische Zeichensätze

Wenn Sie ein Projekt für asiatische Sprachen erstellen, verwenden Sie für die Darstellung in Runtime mindestens Schriftgröße 10 Punkt. Bei der Verwendung einer kleineren Schriftgröße werden asiatische Schriftzeichen unleserlich. Dies gilt auch für die Standardschriftart im Editor "Sprache & Schriftart".

#### Textfeldlänge bei asiatischen Sprachen

Wenn Sie mehrsprachige Projekte mit asiatischen Sprachen bearbeiten, achten Sie auf die Länge von Textfeldern. Abhängig von der Schriftart und -größe können Felder unvollständig angezeigt werden.

1. Öffnen Sie im Inspektorfenster des Textfeldes "Eigenschaften > Darstellung".
2. Deaktivieren Sie unter "Größenanpassung" die Option "Automatisch anpassen".
3. Prüfen Sie, ob die Anzeige in Runtime korrekt ist.

## 10.11.4 Projekt bedienen

### 10.11.4.1 Grundlagen

#### Überblick über die Bedienung eines Projekts

Alle Basic-Bediengeräte sind mit einem Touch-Screen ausgestattet. Einige Basic-Bediengeräte besitzen Funktionstasten. Über den Touch-Screen und die Funktionstasten bedienen Sie das Control Panel oder das Projekt, das auf Ihrem Bediengerät läuft.



#### **GEFAHR**

##### **Fehlbedienung**

Innerhalb des Projekts können Bedienhandlungen enthalten sein, die umfassende anlagenspezifische Kenntnisse des Bedieners voraussetzen.

Stellen Sie sicher, dass nur geschultes Fachpersonal die Anlage bedient.

#### Touch Screen bedienen

#### **ACHTUNG**

##### **Beschädigung des Touch-Screen**

Spitze oder scharfe Gegenstände können die Kunststoffoberfläche des Touch-Screen beschädigen.

Bedienen Sie den Touch-Screen ausschließlich mit dem Finger oder mit einem Touch-Stift.

##### **Auslösen unbeabsichtigter Aktionen**

Das gleichzeitige Berühren mehrere Bedienobjekte kann unbeabsichtigte Aktionen auslösen.

Berühren Sie immer nur ein Bedienobjekt am Bildschirm.

Bedienobjekte sind berührungssensitive Darstellungen am Bildschirm des Bediengeräts.

Die Bedienung unterscheidet sich grundsätzlich nicht vom Drücken mechanischer Tasten. Sie bedienen Bedienobjekte durch Berühren mit dem Finger.

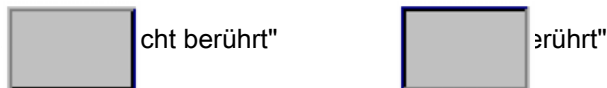
### Hinweis

Sobald das Bediengerät eine Berührung eines Bedienobjekts erkennt, reagiert es mit einer optischen Rückmeldung.

Die optische Rückmeldung ist unabhängig von einer Kommunikation mit der Steuerung. Die Rückmeldung ist deshalb kein Indiz dafür, dass die gewünschte Aktion tatsächlich ausgeführt wird.

Beispiele für Bedienobjekte:

- Schaltflächen  
Schaltflächen können folgende Zustände annehmen:



- Unsichtbare Schaltflächen  
Der Fokus unsichtbarer Schaltflächen ist standardmäßig nach dem Anwählen nicht gekennzeichnet. In diesem Fall folgt keine optische Bedienungsrückmeldung. Der Projektteur kann jedoch unsichtbare Schaltflächen auch so projektieren, dass deren Umrisse bei Berührung als Linien sichtbar sind. Die Umrisse bleiben so lange sichtbar, bis Sie ein anderes Bedienobjekt anwählen.
- EA-Felder  
Wenn Sie ein EA-Feld berühren, erscheint als optische Bedienungsrückmeldung eine Bildschirmtastatur, z. B. zur Eingabe eines Kennworts. Abhängig vom Bediengerät und vom projizierten Bedienobjekt werden unterschiedliche Bildschirmtastaturen für numerische oder alphanumerische Werte angezeigt. Beim Beenden der Eingabe wird die Bildschirmtastatur automatisch wieder ausgeblendet.

## Funktionstasten bedienen

Die Funktionstasten können global oder lokal belegt sein:

- Funktionstasten mit globaler Funktionsbelegung  
Eine Funktionstaste mit globaler Funktionsbelegung löst unabhängig vom angezeigten Bild immer dieselbe Aktion am Bediengerät bzw. in der Steuerung aus. Eine solche Aktion ist z. B. das Aktivieren eines Bildes oder das Schließen eines Meldefensters.
- Funktionstasten mit lokaler Funktionsbelegung  
Eine Funktionstaste mit lokaler Funktionsbelegung ist bildspezifisch und damit nur innerhalb des aktiven Bildes wirksam.  
Die Funktion einer Funktionstaste kann von Bild zu Bild unterschiedlich sein.

Innerhalb eines Bildes kann eine Funktionstaste nur eine Funktionsbelegung haben – entweder global oder lokal. Bei lokaler und globaler Funktionsbelegung hat die lokale Belegung Vorrang.

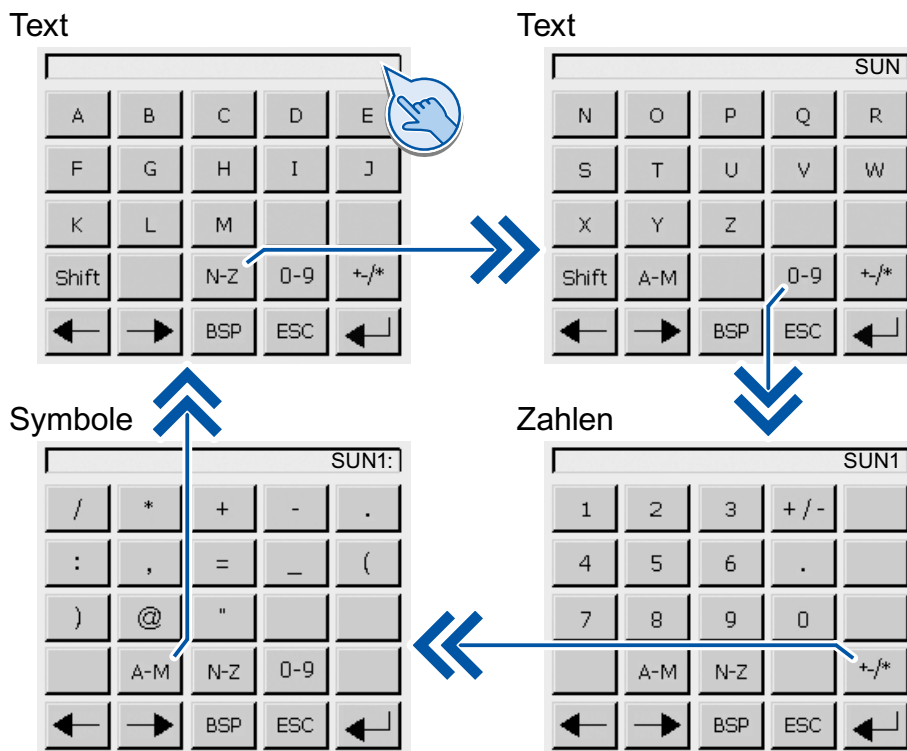
### Allgemeine Funktionen der Bildschirmtastatur

Folgende Tasten sind auf der Bildschirmtastatur aller Basic-Bediengeräte vorhanden:

|  |  |
|--|--|
|  | Cursor nach links  |
|  | Cursor nach rechts   |
|  | Ein Zeichen löschen  |
|  | Eingabe abbrechen  |
|  | Eingabe bestätigen   |
|  | Hilfetext anzeigen. Diese Taste erscheint nur, wenn für das Bedienobjekt ein Hilfetext projiziert wurde. |

### Daten am KTP400 Basic eingeben


Wegen des kleineren Displays unterscheidet sich die Bildschirmtastatur und das Eingabekonzept des KTP400 Basic von den übrigen Basic-Bediengeräten.



Wenn Sie am Touch-Screen des Bediengeräts ein Bedienobjekt berühren, das eine Eingabe erfordert, wird die Bildschirmtastatur angezeigt.

Die Bildschirmtastatur des KTP400 besitzt vier Ansichten. Über die Tasten in der vierten Zeile der Bildschirmtastatur können Sie während der Eingabe die Ansicht wechseln:


|  |                                       |
|--|---------------------------------------|
|  | <b>Wechselt zu Ansicht</b>            |
|  | Eingabe von Text, Zeichen "A" bis "M" |
|  | Eingabe von Text, Zeichen "N" bis "Z" |

|       |   |     |     |     |   |
|-------|---|-----|-----|-----|---|
| A     | B | C   | D   | E   |  |
| F     | G | H   | I   | J   |   |
| K     | L | M   |     |     |   |
| Shift |   | N-Z | 0-9 | +/* |   |
| ←     | → | BSP | ESC | ↵   |   |

0" bis "9", mit oder ohne Vorzeichen und Nachkommastellen  
 ichten  
 schaltung auf Kleinbuchstaben


**Steuerungswirkung**

Bei geöffneter Bildschirmtastatur hat der Steuerungsauftrag 51 "Bildanwahl" keine Wirkung.

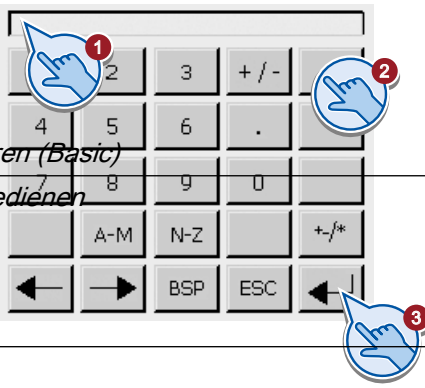
|       |     |     |     |     |   |
|-------|-----|-----|-----|-----|---|
| N     | O   | P   | Q   | R   |  |
| S     | T   | U   | V   | W   |   |
| X     | Y   | Z   |     |     |   |
| Shift | A-M |     | 0-9 | +/* |   |
| ←     | →   | BSP | ESC | ↵   |   |

umerischen Bildschirmtastatur ist einsprachig.  
 ekt hat keinen Einfluss auf die Belegung der  
 tastatur.

**Alphanumerische**

|  |  |  |  |  |
|--|--|--|--|--|
|   |  |  |  |  |
| <ol style="list-style-type: none"> <li>1. Berühren Sie innerhalb des Bildes das gewünschte Bedienobjekt.<br/>Die alphanumerische Bildschirmtastatur wird geöffnet.</li> <li>2. Geben Sie den Wert ein. Dabei gibt das Bediengerät je nach Einstellung einen Signalton aus.<br/>Wechseln Sie mit den Tasten &lt;N-Z&gt; und &lt;A-M&gt; die Ansicht der Bildschirmtastatur, falls erforderlich.<br/>Benutzen Sie die Taste &lt;Shift&gt; zur Eingabe von Kleinbuchstaben.</li> <li>3. Bestätigen Sie die Eingabe mit &lt;Return&gt; oder werfen Sie die Eingabe mit der Taste &lt;ESC&gt;.<br/>In beiden Fällen wird die Bildschirmtastatur geschlossen.</li> </ol> |  |  |  |  |

## Numerische Werte



1. Berühren Sie innerhalb des Bildes das gewünschte Bedienobjekt.  
Die numerische Bildschirmtastatur wird geöffnet.
2. Geben Sie den Wert ein. Dabei gibt das Bediengerät je nach Einstellung einen Signalton aus.  
Wechseln Sie zur Eingabe von Zahlen im Hexadezimalformat mit den Tasten <N-Z> und <A-M> die Ansicht der Bildschirmtastatur.
3. Bestätigen Sie die Eingabe mit <Return> oder verwerfen Sie die Eingabe mit der Taste <ESC>.  
In beiden Fällen wird die Bildschirmtastatur geschlossen.

## Grenzwertprüfung bei numerischen Werten

Für Variablen können Grenzwerte projektiert sein. Wenn Sie einen Wert eingeben, der außerhalb dieser Grenzen liegt, dann wird dieser nicht übernommen. Wenn ein Meldfenster projektiert ist, dann wird eine Systemmeldung ausgegeben und der ursprüngliche Wert wird wieder angezeigt.

## Nachkommastellen bei numerischen Werten

Der Projektteur kann für ein numerisches Eingabefeld die Anzahl der Nachkommastellen festlegen. Wenn Sie einen Wert in ein solches EA-Feld eingeben, wird die Anzahl der Nachkommastellen überprüft.

- Zu viele Nachkommastellen werden ignoriert.
- Zu wenige Nachkommastellen werden mit "0" aufgefüllt.

## Daten am KTP600, KTP1000, TP1500 Basic eingeben

### Alphanumerische Bildschirmtastatur

Wenn Sie am Touch-Screen des Bediengeräts ein Bedienobjekt berühren, das eine Eingabe erfordert, wird die Bildschirmtastatur angezeigt.

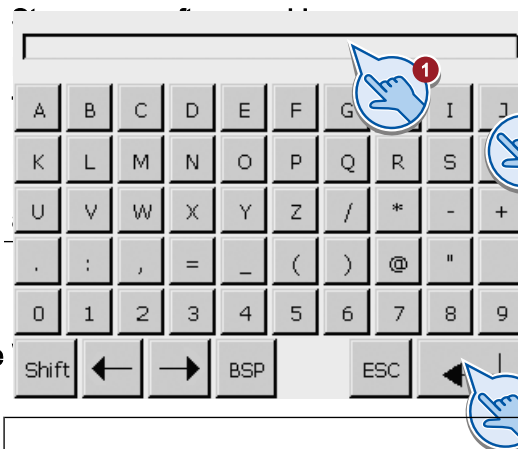
#### Text



#### Zahlen



#### Hinweis



...uerungsauftrag 51 "Bildanwahl" keine Wirkung.

...dschirmtastatur ist einsprachig.  
...n Einfluss auf die Belegung der

### Alphanumerische

1. Berühren Sie innerhalb des Bildes das gewünschte Bedienobjekt. Die alphanumerische Bildschirmtastatur wird geöffnet.
2. Geben Sie den Wert ein. Dabei gibt das Bediengerät je nach Einstellung einen Signalton aus. Benutzen Sie die Taste <Shift> zur Eingabe von Kleinbuchstaben.
3. Bestätigen Sie die Eingabe mit <Return> oder verwerfen Sie die Eingabe mit der Taste <ESC>. In beiden Fällen wird die Bildschirmtastatur geschlossen.

Prozesse visualisieren (Basic)  
10.11 In Runtime bedienen

Numerische Werte

|   |   |      |       |     |
|---|---|------|-------|-----|
| 1 | 2 | 3    | ESC   |     |
| B | 4 | 5    | 6     | BSP |
| 7 | 8 | 9    | + / - |     |
| D | E | F    | 0     | .   |
| ← | → | Help | ←     |     |

1. Berühren Sie innerhalb des Bildes das gewünschte Bedienobjekt. Die numerische Bildschirmtastatur wird geöffnet.
2. Geben Sie den Wert ein. Dabei gibt das Bediengerät je nach Einstellung einen Signalton aus.
3. Bestätigen Sie die Eingabe mit <Return> oder verwerfen Sie die Eingabe mit der Taste <ESC>. In beiden Fällen wird die Bildschirmtastatur geschlossen.

## Grenzwertprüfung bei numerischen Werten

Für Variablen können Grenzwerte projektiert sein. Wenn Sie einen Wert eingeben, der außerhalb dieser Grenzen liegt, dann wird dieser nicht übernommen. Wenn ein Meldfenster projektiert ist, dann wird eine Systemmeldung ausgegeben und der ursprüngliche Wert wird wieder angezeigt.

## Nachkommastellen bei numerischen Werten

Der Projekteur kann für ein numerisches Eingabefeld die Anzahl der Nachkommastellen festlegen. Wenn Sie einen Wert in ein solches EA-Feld eingeben, wird die Anzahl der Nachkommastellen überprüft.

- Zu viele Nachkommastellen werden ignoriert.
- Zu wenige Nachkommastellen werden mit "0" aufgefüllt.

## Tooltip anzeigen

### Verwendung

Mit Tooltips stellt Ihnen der Projekteur zusätzliche Informationen und Bedienhinweise zur Verfügung. Der Projekteur kann Tooltips zu Bildern und Bedienobjekten projektieren.





Der Tooltip eines EA-Felds kann z. B. Hinweise zum einzugebenden Wert enthalten.





## Vorgehen

Um den Tooltip für Bedienobjekte zu öffnen, gehen Sie folgendermaßen vor:

1. Berühren Sie das gewünschte Bedienobjekt.  
Die Bildschirmtastatur wird geöffnet. Sie erkennen an der Darstellung der Taste , ob für das Bedienobjekt oder das aktuelle Bild ein Tooltip projiziert wurde.
2. Berühren Sie auf der Bildschirmtastatur die Taste .  
Der Tooltip zum Bedienobjekt wird angezeigt. Wenn zum markierten Bildobjekt kein Tooltip existiert, wird der Tooltip für das aktuelle Bild angezeigt, falls dieser projiziert wurde. Bei langen Tooltips scrollen Sie den Inhalt mit den Schaltflächen  und .

---

### Hinweis

#### Wechsel zwischen den angezeigten Tooltips

Der Projektteur kann für ein EA-Feld und für das zugehörige Bild Tooltips projektieren. Sie schalten zwischen beiden Tooltips um, indem Sie das Tooltip-Fenster berühren.

---

3. Schließen Sie den angezeigten Tooltip mit der Schaltfläche .

## Alternative Vorgehensweise

Abhängig von der Projektierung können Sie Tooltips auch über ein davor vorgesehenes Bedienobjekt aufrufen.

## Projektsprache einstellen

### Einleitung

Das Projekt auf dem Bediengerät kann mehrsprachig sein. Zum Wechseln der am Bediengerät eingestellten Sprache während des laufenden Betriebs muss ein entsprechendes Bedienobjekt projiziert worden sein.

Nach dem Starten des Projekts ist immer die zuletzt aktive Projektsprache eingestellt.

## Voraussetzung

- Die gewünschte Sprache für das Projekt muss am Bediengerät verfügbar sein.
- Die Funktion für die Sprachumschaltung muss bei der Projektierung mit einem Bedienobjekt verknüpft sein, z. B. mit einer Schaltfläche.

## Sprache auswählen

Zwischen den Sprachen können Sie jederzeit umschalten. Unmittelbar nach dem Aufruf der Funktion für die Sprachumschaltung werden die sprachabhängigen Objekte in der neuen Sprache angezeigt.

Für die Sprachumschaltung gibt es folgende Möglichkeiten:

- Ein projektiertes Bedienobjekt schaltet in einer Auswahl von Sprachen weiter.
- Durch ein projektiertes Bedienobjekt wählen Sie die gewünschte Sprache direkt aus.

### 10.11.4.2 Objekte bedienen

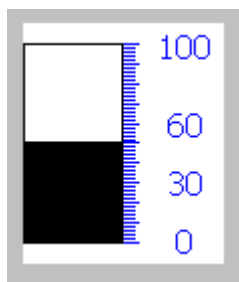
## Balken

## Verwendung

Der Balken ist ein dynamisches Anzeigeobjekt. Der Balken stellt einen Wert aus der Steuerung als rechteckige Fläche dar. Mit dem Balken erkennen Sie z. B. Folgendes auf einen Blick:

- Die Entfernung des aktuellen Werts von den projektierten Grenzwerten
- Das Erreichen eines vorgegebenen Sollwerts

Mit dem Balken werden z. B. Füllstände oder Stückzahlen dargestellt.



## Darstellung

Die Darstellung des Balkens ist abhängig von der Projektierung.

- Der Balken kann mit einer Werteskala beschriftet sein.
- Die projektierten Grenzwerte können durch Linien gekennzeichnet sein.
- Das Überschreiten oder Unterschreiten eines Grenzwerts kann durch einen Farbumschlag signalisiert werden.

## Datum-Uhrzeit-Feld

### Überblick

### Verwendung

Ein Datum-Uhrzeit-Feld kann in Runtime folgende Funktion haben:

- Ausgabe von Datum und Uhrzeit
- Kombinierte Eingabe und Ausgabe, in diesem Fall können Sie die ausgegebenen Werte editieren und so Datum und Uhrzeit neu setzen

31.12.2000 10:59:59

### Darstellung

Die Darstellung im Datum-Uhrzeit-Feld ist abhängig von der am Bediengerät eingestellten Sprache.

Das Datum kann in ausführlicher Form (z.B. Dienstag, 31. Dezember 2003) oder in Kurzform (31. 12. 2003) dargestellt werden.

### Bedienung

Abhängig von der Projektierung stehen Ihnen zur Bedienung des Datum-Uhrzeit-Felds folgende Möglichkeiten zur Verfügung:

- Standardbedienung: Datum und Uhrzeit ändern.

### Verhalten bei Bedienung

Wenn der Bediener in Runtime bei der Eingabe von Werten die Syntax nicht beachtet, oder unzulässige Werte eingibt, werden diese Eingaben nicht übernommen. Stattdessen wird im Datum-Uhrzeit-Feld wieder der ursprüngliche Wert (plus die inzwischen weitergelaufene Zeitspanne) angezeigt und eine Systemmeldung am Bediengerät ausgegeben.


## Touch- und Tasten-Bedienung

### Touch-Bedienung

Um das Datum-Uhrzeit-Feld zu bedienen, gehen Sie folgendermaßen vor:








1. Berühren Sie das Datum-Uhrzeit-Feld am Touch-Screen des Bediengeräts. Die Bildschirmtastatur wird automatisch eingeblendet.
2. Geben Sie über die Bildschirmtastatur den gewünschten Wert ein.
3. Bestätigen Sie die Eingabe am Bediengerät mit <Enter> oder verwerfen Sie die Eingabe mit <Esc>. Nachdem Sie die Eingabe bestätigt oder verworfen haben, wird die Bildschirmtastatur automatisch wieder ausgeblendet.

## Tasten-Bedienung

Aktivieren Sie das Datum-Uhrzeit-Feld z. B. mit einem oder mehreren  entsprechend der projektierten Tab-Reihenfolge. Die Anwahl wird durch einen Farbumschlag des Feldinhalts signalisiert.

## Vorgehen

Um das Datum-Uhrzeit-Feld zu bedienen, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Cursor mit den Cursor-Tasten und geben Sie den gewünschten Wert ein.
2. Drücken Sie . Das Objekt wechselt in den speziellen Editiermodus. Jetzt ist immer nur ein Zeichen des Feldes markiert.
  - Mit den Cursor-Tasten  /  scrollen Sie durch eine Zeichentabelle.
  - Mit den Cursor-Tasten  /  wechseln Sie zur nächsten oder vorherigen Stelle der Eingabe.
3. Bestätigen Sie Ihre Eingabe mit  oder verwerfen Sie die Eingabe mit .

## EA-Feld

## Überblick

## Verwendung

In einem EA-Feld geben Sie numerische oder alphanumerische Werte ein. Ein numerischer Wert ist z. B. die Zahl 80 als Sollwert für eine Temperatur. Ein alphanumerischer Wert ist z. B. der Text "Service" als Benutzername.



## Darstellung

Die Darstellung des EA-Feldes ist abhängig von der Projektierung:

- Numerisches EA-Feld  
Zur Eingabe von Zahlen in dezimaler, hexadezimaler oder binärer Darstellung.
- Alphanumerisches EA-Feld  
Zur Eingabe von Zeichenketten.
- EA-Feld für Datum und Uhrzeit  
Zur Eingabe von Kalenderdaten oder Zeitangaben. Das Format ist abhängig von der eingestellten Projektsprache.
- EA-Feld für Kennworteingabe  
Zur verdeckten Eingabe eines Kennworts. Die eingegebene Zeichenkette wird mit Platzhaltern (\*) dargestellt.

## Bedienung

Abhängig von der Projektierung stehen Ihnen zur Bedienung des EA-Feldes folgende Möglichkeiten zur Verfügung:

- Standardbedienung: Geben Sie einen Wert in das EA-Feld ein.
- Ereignis: Durch die Bedienung des EA-Feldes, z. B. durch Aktivieren, wird ein Ereignis ausgelöst. An das Ereignis kann die Abarbeitung einer Funktionsliste projektiert sein.

## Verhalten bei Bedienung

### Grenzwertprüfung bei numerischen Werten

Für Variablen können Grenzwerte projektiert sein. Wenn Sie einen Wert eingeben, der außerhalb dieser Grenzen liegt, wird dieser nicht übernommen, z. B. 80 beim Grenzwert 78. In diesem Fall wird am Bediengerät eine Systemmeldung ausgegeben, wenn ein Meldefenster projektiert ist. Der ursprüngliche Wert wird wieder angezeigt.

### Nachkommastellen bei numerischen Werten

Der Projektteur kann für ein numerisches Eingabefeld die Anzahl der Nachkommastellen festlegen. Wenn Sie einen Wert in ein solches EA-Feld eingeben, wird die Anzahl der Nachkommastellen überprüft.

- Zu viel eingegebene Nachkommastellen werden ignoriert.
- Zu wenig eingegebene Nachkommastellen werden mit "0" aufgefüllt

### Verdeckte Eingabe

Bei der verdeckten Eingabe wird für jedes Zeichen ein "\*" angezeigt. Das Datenformat des Eingabewerts ist nicht erkennbar.

### Verhalten beim Wechsel zwischen Eingabefeldern

Wenn durch Bedienung innerhalb eines Bildes von einem Eingabefeld zu einem anderen gewechselt wird und dadurch die Bildschirmtastatur angezeigt wird, so wird für das vorherige Feld das Ereignis "Feld verlassen" nicht sofort ausgelöst, sondern erst nach Schließen der Bildschirmtastatur.

## Touch- und Tasten-Bedienung


### Touch-Bedienung

Um das EA-Feld zu bedienen, gehen Sie folgendermaßen vor:

1. Berühren Sie das EA-Feld am Touch-Screen des Bediengeräts. Die Bildschirmtastatur wird automatisch eingeblendet.
2. Geben Sie über die Bildschirmtastatur den gewünschten Wert ein.
3. Bestätigen Sie die Eingabe am Bediengerät mit <Enter> oder verwerfen Sie die Eingabe mit <Esc>.









Nachdem Sie die Eingabe bestätigt oder verworfen haben, wird die Bildschirmtastatur automatisch wieder ausgeblendet.

## Tasten-Bedienung

Aktivieren Sie das EA-Feld z.B. mit einem oder mehreren  entsprechend der projektierten Tab-Reihenfolge. Die Anwahl wird durch einen Farbumschlag des Feldinhalts signalisiert.



## Vorgehen

Um das EA-Feld zu bedienen, gehen Sie folgendermaßen vor:

1. Positionieren Sie den Cursor mit  und einer Cursor-Taste.
2. Die Markierung des Feldinhalts wird aufgehoben. Geben Sie den gewünschten Wert ein.
3. Drücken Sie . Das Objekt wechselt in den speziellen Editiermodus. Jetzt ist immer nur ein Zeichen des Feldes markiert.
  - Sie können mit den Cursortasten  /  durch eine Zeichentabelle scrollen.
  - Mit den Cursortasten  /  können Sie zur nächsten oder vorherigen Stelle der Eingabe wechseln.
4. Bestätigen Sie Ihre Eingabe mit  oder verwerfen Sie die Eingabe mit .

---

### Hinweis

Zur Eingabe der hexadezimalen  en "A" bis "F" für numerische Werte schalten Sie die Eingabetasten mit der Taste  in die Buchstabenbelegung um.

---

## Grafikanzeige

### Verwendung

Die Grafikanzeige dient zur Anzeige von Grafik.



## Darstellung

Die Darstellung der Grafik ist abhängig von der Projektierung. Die Grafikanzeige passt sich z. B. automatisch an die Größe der Grafik an.

---

### Hinweis

Wenn Sie in WinCC Bildern Bitmaps mit der Einstellung "Transparente Farbe" verwenden, benötigt die Darstellung auf den Panel-Bediengeräten eine hohe Performance. Um eine bessere Performance zu erreichen, deaktivieren Sie die Einstellung "Transparente Farbe" in den Eigenschaften des jeweiligen Grafikobjekts. Diese Einschränkung gilt insbesondere bei der Verwendung von Bitmaps als Hintergrundbilder.

---

## Bedienung

Die Grafikanzeige dient zur reinen Anzeige und kann nicht bedient werden.

## Grafisches EA-Feld

### Überblick

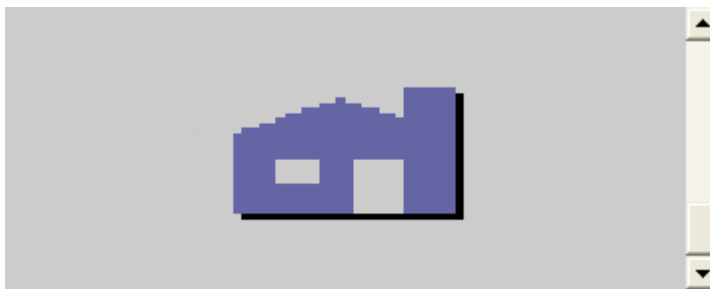
### Verwendung

Ein grafisches EA-Feld kann in Runtime folgende Funktion haben:

- Ausgabe von Einträgen einer Grafikliste
- Kombinierte Eingabe und Ausgabe

Beispiel zur Verwendung als Ausgabefeld:

Um in Runtime den Zustand eines Ventils anzuzeigen, wird im grafischen EA-Feld entweder die Darstellung eines geschlossenen oder eines geöffneten Ventils ausgegeben.



## Bedienung

Abhängig von der Projektierung stehen Ihnen zur Bedienung des grafischen EA-Feldes folgende Möglichkeiten zur Verfügung:

- Standardbedienung: Wählen Sie einen Eintrag aus der Grafikliste aus.
- Ereignis: Durch die Bedienung des grafischen EA-Feldes, z.B. durch Aktivieren, wird ein Ereignis ausgelöst. An das Ereignis kann die Abarbeitung einer Funktionsliste projektiert sein.

## Verhalten bei Bedienung

Wenn im grafischen EA-Feld eine Kakteengrafik angezeigt wird, wurde im Projekt für einen bestimmten Wert keine Grafik zur Ausgabe definiert.

Die Aktivierung wird Ihnen am Bediengerät dadurch angezeigt, dass der Inhalt des grafischen EA-Feldes seine Farbe ändert.

Der Rahmen im 3D-Stil wird nur in einem Ausgabefeld grafisch dargestellt.

## Touch- und Tasten-Bedienung

### Touch-Bedienung






Berühren Sie das grafische EA-Feld am Touch-Screen des Bediengeräts. Der Auswahlmodus ist aktiviert.

Wählen Sie die gewünschte Grafik mit der Bildlaufleiste aus.

Übernehmen Sie die gewünschte Grafik durch Berühren der Grafik oder werfen Sie die Auswahl durch Berühren eines anderen Bildobjekts.

### Tasten-Bedienung

So bedienen Sie ein grafisches EA-Feld am Tastengerät:

| Schritt                         | Vorgehen   |
|---------------------------------|--|
| 1 Grafisches EA-Feld anwählen   | z. B.  |
| 2 Auswahlmodus aktivieren       |         |
| 3 Eintrag auswählen             |         |
| 4 Markierung übernehmen<br>oder |         |
| Auswahl abbrechen               |         |

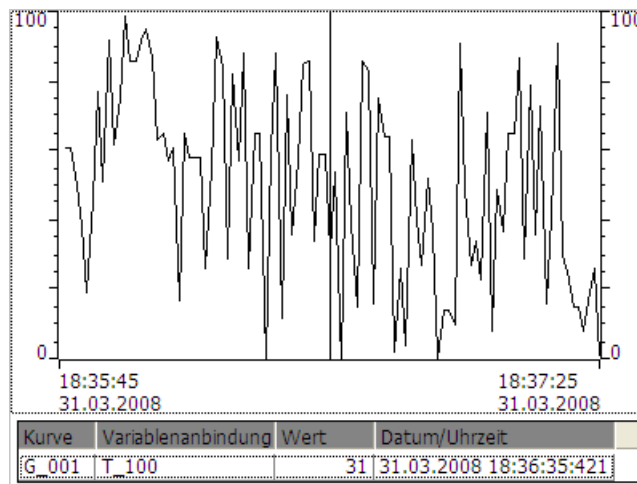


## Kurvenanzeige

### Überblick

### Verwendung

Die Kurvenanzeige ist ein dynamisches Anzeigeobjekt. In der Kurvenanzeige können sowohl laufende, aktuelle Prozessdaten als auch - wenn vom Bediengerät unterstützt - Prozessdaten aus einem Archiv kontinuierlich dargestellt werden.



### Darstellung

Die Darstellung der Kurvenanzeige ist abhängig von der Projektierung. In einer Kurvenanzeige können auch mehrere Kurven gleichzeitig dargestellt werden, damit der Benutzer z.B. verschiedene Prozessverläufe vergleichen kann. Wenn der dargestellte Prozesswert die projektierten Grenzwerte überschreitet oder unterschreitet, kann die Grenzwertverletzung durch Farbwechsel der Kurve angezeigt werden.

Des Weiteren kann ein Lineal das Ablesen von Prozesswerten aus der Kurvenanzeige erleichtern. Das Lineal zeigt den zu einem X-Wert gehörenden Kurvenwert an.










### Bedienung

Abhängig von der Projektierung können Sie:

- Den dargestellten Zeitabschnitt erweitern.
- Den dargestellten Zeitabschnitt verkleinern.
- Um eine Anzeigebreite zurück blättern.
- Um eine Anzeigebreite vorwärts blättern.
- Die Kurvenaufzeichnung anhalten und wieder fortsetzen.

### Bedienelemente

Die Schaltflächen haben folgende Funktionen:

| Bedienelement  | Funktion  |
|--|---|
|   | Blättert zum Beginn der Kurvenaufzeichnung zurück. Dort werden die Startwerte angezeigt, mit denen die Kurvenaufzeichnung begonnen hat. |
|   | Vergrößert den dargestellten Zeitabschnitt  |
|   | Verkleinert den dargestellten Zeitabschnitt   |
|   | Bewegt das Lineal rückwärts (nach links).   |
|   | Bewegt das Lineal vorwärts (nach rechts).   |
|   | Blättert um eine Anzeigenbreite rückwärts (nach links).   |
|   | Blättert um eine Anzeigenbreite vorwärts (nach rechts).   |
|   | Blendet das Lineal ein oder aus. Das Lineal zeigt den zu einem X-Wert gehörenden Y-Wert an.   |
|  | Hält die Kurvenaufzeichnung an oder setzt die Kurvenaufzeichnung fort   |

### Touch- und Tasten-Bedienung



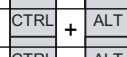




#### Vorgehen

Berühren Sie das gewünschte Bedienelement der Kurvenanzeige am Touch-Screen des Bediengeräts.

#### Vorgehen

Aktivieren Sie die Kurvenanzeige z.B. mit  über die projizierte Tab-Reihenfolge.

Die folgende Tabelle zeigt die möglichen Tastenkombinationen:

| Tasten  | Funktion  |
|---|---|
|  | Blättert zum Beginn der Kurvenaufzeichnung zurück. Dort werden die Startwerte angezeigt, mit denen die Kurvenaufzeichnung begonnen hat. |
|  | Vergrößert den dargestellten Zeitabschnitt.   |
|  | Verkleinert den dargestellten Zeitabschnitt.  |
|  | Bewegt das Lineal rückwärts (nach links).   |
|  | Bewegt das Lineal vorwärts (nach rechts).   |
|  | Blättert um eine Anzeigenbreite rückwärts (nach links).   |
|  | Blättert um eine Anzeigenbreite vorwärts (nach rechts).   |

## Schaltfläche

### Überblick

### Verwendung

Eine Schaltfläche ist eine virtuelle Taste am Bildschirm des Bediengeräts, die mit einer oder mehreren Funktionen belegt sein kann.



### Darstellung

Die Darstellung der Schaltfläche ist abhängig vom Schaltflächentyp.

- Schaltfläche mit Text: Der auf der Schaltfläche dargestellte Text gibt Auskunft über den Zustand der Schaltfläche.
- Schaltfläche mit Grafik: Die auf der Schaltfläche dargestellte Grafik gibt Auskunft über den Zustand der Schaltfläche.
- Unsichtbar: Die Schaltfläche ist in Runtime nicht sichtbar.

### Bedienung

Abhängig von der Projektierung stehen Ihnen zur Bedienung der Schaltfläche folgende Möglichkeiten zur Verfügung:

- Standardbedienung: Klicken Sie die Schaltfläche.
- Ereignis: Durch die Bedienung der Schaltfläche, z.B. durch Klicken, wird ein Ereignis ausgelöst. An das Ereignis kann die Abarbeitung einer Funktionsliste projiziert sein.

### Verhalten bei Bedienung

Nach Bedienung kann eine optische Rückmeldung erfolgen. Beachten Sie jedoch, dass die optische Rückmeldung nur die erfolgte Bedienung anzeigt und nicht, ob die projizierten Funktionen tatsächlich ausgeführt wurden.




### Touch- und Tasten-Bedienung

### Vorgehen

Berühren Sie die Schaltfläche am Touch-Screen des Bediengeräts.

## Vorgehen

So bedienen Sie eine Schaltfläche am Tastengerät:

1. Wählen Sie die Schaltfläche mit einer Cursor-Taste an, z. B. 
2. Drücken Sie anschließend entweder die Taste  oder die Taste 

## Schalter

### Überblick

### Verwendung

Der Schalter ist ein Bedienobjekt und Anzeigeobjekt mit den beiden Zuständen "Gedrückt" und "Losgelassen". Schalter können den Zustand eines Anlagenteils signalisieren, das vom Bediengerät aus nicht zu sehen ist, z.B. ein Motor. Gleichzeitig können Sie damit den Zustand des betreffenden Anlagenteils am Bediengerät ändern.



### Darstellung

Die Darstellung des Schalters ist abhängig vom Schaltertyp.

- Schalter: Der Schalter verfügt über einen Schieber, dessen Position Auskunft über den Zustand des Schalters gibt.
- Schalter mit Text: Der auf dem Schalter dargestellte Text gibt Auskunft über den Zustand des Schalters.
- Schalter mit Grafik: Die auf dem Schalter dargestellte Grafik gibt Auskunft über den Zustand des Schalters.

### Bedienung

Abhängig von der Projektierung stehen Ihnen zur Bedienung des Schalters folgende Möglichkeiten zur Verfügung:

- Standardbedienung: Klicken Sie den Schalter.
- Ereignis: Durch die Bedienung des Schalters, z.B. durch Klicken, wird ein Ereignis ausgelöst. An das Ereignis kann die Abarbeitung einer Funktionsliste projektiert sein.

### Verhalten bei Bedienung

Ein Schalter hat zwei stabile Zustände: Wenn Sie den Schalter betätigen, wechselt er in den jeweils anderen Zustand. Diesen Zustand behält der Schalter bis zur nächsten Bedienung bei.

## Touch- und Tasten-Bedienung




### Touch-Bedienung

Die Touch-Bedienung des Schalters unterscheidet sich abhängig vom Typ des Schalters:

- Wenn für den Schalter ein Schieber angezeigt wird:  
Ziehen Sie den Schieber am Touch-Screen des Bediengeräts in die neue Position oder Doppelklicken Sie auf den Schieberbereich.
- Wenn für den Schalter nur ein Text oder eine Grafik angezeigt wird:  
Berühren Sie den Schalter am Touch-Screen des Bediengeräts.

### Tasten-Bedienung

So bedienen Sie einen Schalter am Tastengerät:

- Wählen Sie den Schalter mit einer Cursor-Taste an, z. B. 
- Drücken Sie anschließend entweder die Taste  oder die Taste 

## Symbolisches EA-Feld

### Überblick

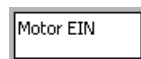
### Verwendung

Ein symbolisches EA -Feld kann in Runtime folgende Funktion haben:

- Ausgabe von Einträgen einer Textliste
- Kombinierte Eingabe und Ausgabe

Beispiel zur Verwendung als kombiniertes Eingabefeld und Ausgabefeld:

Um in Runtime einen Motor zu steuern, wählen Sie aus der Textliste einen der Texte "Motor EIN" oder "Motor AUS". Der Motor wird entsprechend der Auswahl gestartet oder angehalten. Das symbolische EA-Feld zeigt den jeweiligen Zustand des Motors an.



### Bedienung

Abhängig von der Projektierung stehen Ihnen zur Bedienung des symbolischen EA-Feldes folgende Möglichkeiten zur Verfügung:

- Standardbedienung: Wählen Sie einen Eintrag aus der Textliste aus.
- Ereignis: Durch die Bedienung des symbolischen EA-Feldes, z.B. durch Aktivieren, wird ein Ereignis ausgelöst. An das Ereignis kann die Abarbeitung einer Funktionsliste projiziert sein.

### Verhalten bei Bedienung

Wenn das symbolische EA-Feld eine leere Textzeile in der Auswahlliste enthält, ist im Projekt kein Eintrag definiert worden. Die Aktivierung wird Ihnen am Bediengerät dadurch angezeigt, dass der Inhalt des symbolischen EA-Feldes seine Farbe ändert.

### Touch- und Tasten-Bedienung

#### Touch-Bedienung






Berühren Sie das symbolische EA-Feld am Touch-Screen des Bediengeräts. Die vordefinierten Einträge werden in der Auswahlliste angezeigt.

Wenn die Auswahlliste eine Bildlaufleiste besitzt: Berühren Sie die Bildlaufleiste am Touch-Screen des Bediengeräts. Bewegen Sie die Bildlaufleiste unter stetiger Berührung des Touch-Screens in die gewünschte Richtung.

Wählen Sie den gewünschten Eintrag und übernehmen Sie den zugehörigen Wert der Variablen, indem Sie den Eintrag am Touch-Screen berühren. Die Auswahlliste wird geschlossen und der Eintrag angezeigt. Das symbolische EA-Feld besitzt weiterhin den Fokus.

#### Tasten-Bedienung

So bedienen Sie ein symbolisches EA-Feld am Tastengerät:

| Schritt                         | Vorgehen  |
|---------------------------------|---|
| 1 Symbolisches EA-Feld anwählen | z. B.  Das symbolische EA-Feld ist markiert.  |
| 2 Auswahlliste öffnen           |  Die Auswahlliste wird aufgeklappt.  |
| 3 Eintrag auswählen             |  Bewegt den Cursor zeilenweise.  |
| 4 Markierung übernehmen<br>Oder |  Der markierte Eintrag wird gültig. Das Auswahlfeld wird geschlossen.                |
| Markierung abbrechen            |  Der ursprüngliche Wert wird wieder rekonstruiert. Das Auswahlfeld wird geschlossen. |

#### 10.11.4.3 Sicherheit im Projekt

### Überblick

#### Aufbau des Sicherheitssystems

Der Projektteur kann die Bedienung eines Projekts durch ein Sicherheitssystem schützen.

Das Sicherheitssystem basiert auf Berechtigungen, Benutzergruppen und Benutzern.

Wenn Sie ein Bedienobjekt mit Zugriffsschutz im Projekt bedienen, müssen Sie sich zunächst am Bediengerät anmelden. Dazu wird ein Anmeldedialog angezeigt, in dem Sie den Benutzernamen und das Kennwort eingeben. Nach dem Anmelden dürfen Sie die Bedienobjekte bedienen, für die Sie die notwendigen Berechtigungen besitzen.

Der Projekteur kann den Anmeldedialog auch über ein eigenes Bedienobjekt zur Verfügung stellen.

Ebenso kann der Projekteur ein Bedienobjekt zum Abmelden zur Verfügung stellen. Nach dem Abmelden dürfen Sie Objekte mit Zugriffsschutz nicht mehr bedienen, sondern müssen sich erneut anmelden.

## Benutzergruppen und Berechtigungen

Der Projekteur legt die Benutzergruppen projektspezifisch an. Die Gruppen "Administratoren" und "Benutzer" sind standardmäßig in jedem Projekt enthalten. Den Benutzergruppen sind Berechtigungen zugeordnet. Im Projekt ist für jedes Objekt und jede Funktion differenziert festgelegt, welche Berechtigung für die Bedienung erforderlich ist.

## Benutzer und Kennwörter

Jeder Benutzer ist jeweils genau einer Benutzergruppe zugeordnet.

Folgende Personen dürfen Benutzer anlegen und Kennwörter für diese vergeben:

- Der Projekteur bei der Projektierung
- Der Administrator am Bediengerät
- Ein Benutzer mit der Berechtigung zur Benutzerverwaltung am Bediengerät

Unabhängig von der Benutzergruppe darf jeder Benutzer sein eigenes Kennwort ändern.

## Abmeldezeiten

Für jeden Benutzer ist im System eine Abmeldezeit projektierbar. Wenn die Zeit zwischen zwei beliebigen Benutzeraktionen, z. B. Werteingabe oder Bildwechsel, länger wird als die Abmeldezeit, wird der Benutzer automatisch abgemeldet. Wenn er Objekte mit Zugriffsschutz weiter bedienen will, muss er sich erneut anmelden.

## Sichern und Wiederherstellen

Die Benutzerdaten werden verschlüsselt und netzausfallsicher am Bediengerät gespeichert.

Die am Bediengerät eingerichteten Benutzer, Kennwörter, Gruppenzuordnungen und Abmeldezeiten können Sie sichern und wiederherstellen. Sie vermeiden so die erneute Eingabe der Daten an einem anderen Bediengerät.

---

**Hinweis**

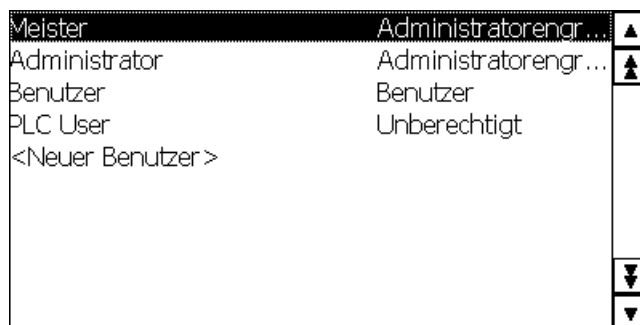
Die aktuell gültigen Benutzerdaten werden in folgenden Fällen überschrieben:

- Abhängig von den Einstellungen bei einem erneuten Laden des Projekts
  - Beim Wiederherstellen eines gesicherten Projekts
  - Beim Importieren der Benutzerverwaltung über ein Bedienobjekt. Die erneut geladenen oder wiederhergestellten Benutzerdaten und Kennwörter sind sofort gültig.
- 

## Einfache Benutzeranzeige

### Verwendung

Auf Bediengeräten mit kleiner Display-Größe wird die einfache Benutzeranzeige zur Anzeige von Benutzern am Bediengerät verwendet.



---

**Hinweis**

Das Objekt "Einfache Benutzeranzeige" kann nicht mit einem Skript dynamisiert werden.

---

### Darstellung

Die Darstellung ist abhängig von den Berechtigungen.

- Beim Administrator oder einem Benutzer mit der Berechtigung zur Benutzerverwaltung werden in der Benutzeranzeige alle auf dem Bediengerät vorhandenen Benutzer angezeigt.
- Bei einem Benutzer ohne die Berechtigung zur Benutzerverwaltung wird nur der eigene Benutzereintrag angezeigt.



## Bedienung

Abhängig von der Projektierung können Sie:

- Benutzer verwalten, z.B. neu anlegen, löschen.
- Vorhandene Benutzerdaten ändern.
- Benutzerdaten exportieren oder importieren.

---

### Hinweis

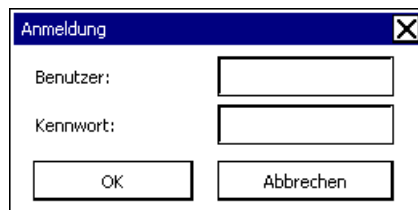
An einem Bediengerät ist die Anzahl auf 100 Benutzer und einem PLC User begrenzt. Diese Grenze gilt nicht für PCs. An einem PC wird die maximale Anzahl der Benutzer durch den physikalischen Speicher begrenzt.

---

## Benutzer anmelden

### Anmeldedialog

Zum Anmelden am Sicherheitssystem des Bediengeräts verwenden Sie den Anmeldedialog. Im Anmeldedialog geben Sie den Benutzernamen und das Kennwort ein.



Der Anmeldedialog wird in folgenden Fällen geöffnet:

- Sie bedienen ein Bedienobjekt mit Zugriffsschutz.
- Sie bedienen ein Bedienobjekt, das zum Aufblenden des Anmeldedialogs projiziert wurde.
- Sie aktivieren den Eintrag "<ENTER>" in der einfachen Benutzeranzeige.
- Sie aktivieren einen leeren Eintrag in der erweiterten Benutzeranzeige.
- Beim Start des Projekts wird der Anmeldedialog abhängig von der Projektierung automatisch angezeigt.

### Voraussetzung

Der Anmeldedialog ist geöffnet.




## Vorgehen bei Touch-Bedienung

Gehen Sie folgendermaßen vor:

1. Geben Sie Benutzer und Kennwort ein.  
Berühren Sie dazu das jeweilige Eingabefeld. Die alphanumerische Bildschirmtastatur wird eingeblendet.
2. Bestätigen Sie die Anmeldung mit der Schaltfläche "OK".

## Vorgehen bei Tasten-Bedienung

Gehen Sie folgendermaßen vor:

1. Markieren Sie mit der Taste  innerhalb des Anmeldedialogs das Eingabefeld "Benutzer".
2. Geben Sie den Benutzernamen mit den Systemtasten ein.  
Zur Eingabe von Buchstaben schalten Sie den Ziffernblock mit der Taste  in die Buchstabenbelegung um.
3. Markieren Sie mit der Taste  das Eingabefeld "Kennwort".
4. Geben Sie das Kennwort mit den Systemtasten ein.
5. Bestätigen Sie die Eingaben mit der Schaltfläche "OK".

---

### Hinweis

Beim Eingeben des Benutzernamens wird nicht zwischen Groß- und Kleinschreibung unterschieden.

Beim Eingeben des Kennworts müssen Sie die Groß- und Kleinschreibung beachten.

---

## Ergebnis

Nach dem erfolgreichen Anmelden am Sicherheitssystem können Sie zugriffsgeschützte Funktionen am Bediengerät ausführen, für die Sie die Berechtigung besitzen.

Wenn Sie das Kennwort falsch eingegeben haben und ein Meldfenster projiziert wurde, wird eine Fehlermeldung angezeigt.

## Benutzer abmelden

### Voraussetzung

Sie haben sich am Sicherheitssystem des Bediengeräts angemeldet.

### Vorgehen

Zum Abmelden haben Sie folgende Möglichkeiten:

- Sie bedienen ein Bedienobjekt, das zum Abmelden projiziert wurde.
- Wenn Sie das Projekt nicht bedienen und die Abmeldezeit überschritten ist, werden Sie automatisch abgemeldet.

Wenn Sie ein falsches Kennwort eingeben, werden Sie ebenfalls automatisch abgemeldet.

## Ergebnis

Sie sind nicht mehr am Projekt angemeldet. Um ein Bedienobjekt mit Zugriffsschutz zu bedienen, müssen Sie sich erneut anmelden.

## Benutzer anlegen

### Voraussetzung

- Die Benutzeranzeige ist geöffnet.
- Sie besitzen die Berechtigung zur Benutzerverwaltung oder Sie sind Administrator.
- Eine Benutzergruppe ist angelegt.

---

#### Hinweis

In Runtime müssen Sie einen Benutzer einer Benutzergruppe zuordnen. Die Benutzergruppe wird im Engineering System angelegt. Die Bezeichnung der Benutzergruppe ist sprachabhängig.

---

#### Hinweis

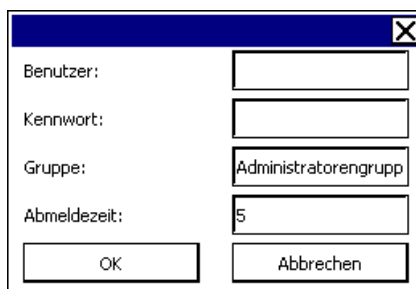
Beim Kennwort dürfen Sie folgende Zeichen nicht verwenden:

- Leerzeichen
  - Sonderzeichen \* ? . % / \ ' "
- 



## Benutzer anlegen mit einfacher Benutzeranzeige

Gehen Sie wie folgendermaßen vor:

1. Klicken Sie in der Benutzeranzeige den Eintrag "<Neuer Benutzer>". Ein Dialog wird geöffnet.



2. Geben Sie den gewünschten Benutzernamen und ein Kennwort ein. Berühren Sie dazu das jeweilige Eingabefeld. Die alphanumerische Bildschirmtastatur wird eingeblendet.
3. Klicken Sie auf das Eingabefeld der Gruppe. Ein Dialog wird geöffnet.

4. Ordnen Sie den Benutzer einer Gruppe zu. Mit den Schaltflächen  und  scrollen Sie in der Auswahlliste.
5. Berühren Sie den gewünschten Eintrag in der Auswahlliste.  
Der ausgewählte Eintrag wird dann als Eingabe übernommen.
6. Berühren Sie das Eingabefeld "Abmeldezeit". Die Bildschirmtastatur wird eingeblendet.
7. Geben Sie einen Wert zwischen 0 und 60 für die Abmeldezeit in Minuten an. Der Wert 0 steht für "kein automatisches Abmelden".
8. Bestätigen Sie die Eingaben mit der Schaltfläche "OK".

## Ergebnis

Der neue Benutzer ist angelegt.

## Benutzer ändern

### Voraussetzung

Die Benutzeranzeige ist geöffnet.

Von Ihrer Berechtigung hängt ab, welche Daten Sie ändern dürfen:

- Sie sind Administrator oder ein Benutzer mit der Berechtigung zur Benutzerverwaltung. In diesen Fällen dürfen Sie in der Benutzeranzeige die Daten für alle auf dem Bediengerät vorhandenen Benutzer ändern:
  - Benutzername
  - Gruppenzuordnung
  - Kennwort
  - Abmeldezeit
- Sie sind ein Benutzer ohne die Berechtigung zur Benutzerverwaltung. In diesem Fall dürfen Sie nur folgende eigene Benutzerdaten ändern:
  - Kennwort
  - Abmeldezeit, wenn in der Projektierung vorgesehen

---

#### Hinweis

Für den Benutzer "Admin" dürfen Sie nur die Abmeldezeit und das Kennwort ändern.

Für den Benutzer "PLC\_User" dürfen Sie nur die Abmeldezeit ändern. Dieser Benutzer wird für das Anmelden über die Steuerung verwendet.

---

#### Hinweis

Änderungen in der Benutzeranzeige in Runtime sind sofort wirksam. Änderungen in Runtime werden im Engineering System nicht aktualisiert.

Beim Laden der Benutzerverwaltung auf das Bediengerät werden alle Änderungen in der Benutzeranzeige überschrieben.

---

## Benutzerdaten ändern mit einfacher Benutzeranzeige

Gehen Sie wie folgendermaßen vor:

1. Berühren Sie in der Benutzeranzeige den Benutzer, dessen Benutzerdaten Sie ändern wollen.
2. Gehen Sie beim Ändern der Daten genauso vor, wie beim Anlegen eines Benutzers.

## Benutzerdaten ändern mit erweiterter Benutzeranzeige

Gehen Sie folgendermaßen vor:

1. Berühren Sie in der Benutzeranzeige die Benutzerdaten, die Sie ändern wollen.
2. Gehen Sie beim Ändern der Daten genauso vor, wie beim Anlegen eines Benutzers.

## Ergebnis

Die Benutzerdaten für den Benutzer sind geändert.

## Benutzer löschen

### Voraussetzung

- Ein Bild ist geöffnet, das die Benutzeranzeige enthält.
- Zum Löschen eines Benutzers müssen Sie Administrator sein oder die Berechtigung zur Benutzerverwaltung besitzen.

---

### Hinweis

Änderungen in der Benutzeranzeige in Runtime sind sofort wirksam. Änderungen in Runtime werden im Engineering System nicht aktualisiert.

Beim Laden der Benutzerverwaltung auf das Bediengerät werden alle Änderungen in der Benutzeranzeige überschrieben.

---



## Vorgehen bei Touch-Bedienung

Gehen Sie folgendermaßen vor:

1. Berühren Sie in der Benutzeranzeige den Benutzer, den Sie löschen wollen.
2. Löschen Sie den Benutzernamen.

## Vorgehen bei Tasten-Bedienung

Gehen Sie folgendermaßen vor:

1. Markieren Sie die Benutzeranzeige mit der Taste  oder mit den Cursortasten.
2. Markieren Sie in der Benutzeranzeige mit den Cursortasten den Benutzer.
3. Um den Benutzer zu löschen, drücken Sie die Taste .

## Ergebnis

Der Benutzer ist gelöscht und darf sich nicht mehr am Projekt anmelden.

### 10.11.4.4 Meldungen bedienen

## Überblick

## Meldungen

Meldungen zeigen am Bediengerät Ereignisse und Zustände an, die in der Anlage bzw. im Prozess oder am Bediengerät auftreten. Ein Zustand wird bei dessen Eintritt gemeldet.

Für eine Meldung können folgende Meldeereignisse auftreten:

- Kommen
- Gehen
- Quittieren

Der Projekteur legt fest, welche Meldungen vom Benutzer quittiert werden müssen.

Eine Meldung kann folgende Informationen enthalten:

- Datum
- Uhrzeit
- Meldetext
- Ereignistext
- Störort
- Zustand
- Meldeklasse
- Meldenummer
- Meldegruppe
- Diagnostizierbar

## Meldeklassen

Meldungen sind verschiedenen Meldeklassen zugeordnet. Die Auswahl ist abhängig vom Bediengerät.

- "Warnings"  
Meldungen dieser Klasse zeigen normalerweise Zustände in der Anlage an, z.B. "Motor eingeschaltet". Meldungen dieser Meldeklasse müssen nicht quittiert werden.
- "Errors"  
Meldungen dieser Klasse müssen immer quittiert werden. Störmeldungen zeigen normalerweise kritische Störungen in der Anlage an, z.B. "Motortemperatur zu hoch".

- "System"  
Systemmeldungen zeigen Zustände oder Ereignisse des Bediengeräts selbst an. Systemmeldungen informieren z.B. über Fehlbedienungen oder Störungen in der Kommunikation.
- "Diagnosis Events"  
SIMATIC-Diagnosemeldungen zeigen Zustände und Ereignisse der Steuerungen SIMATIC S7 an.

---

**Hinweis**

**Geräteabhängigkeit**

Diagnosemeldungen stehen für Basic Panels nicht zur Verfügung.

---

- STEP 7-Meldeklassen  
Die in STEP 7 projektierten Meldeklassen stehen auch dem Bediengerät zur Verfügung.

---

**Hinweis**

**Geräteabhängigkeit**

STEP 7-Meldeklassen stehen für Basic Panels nicht zur Verfügung.

---

- Benutzerdefinierte Meldeklassen  
Die Eigenschaften dieser Meldeklasse werden bei der Projektierung festgelegt .

## Meldepuffer

Die Meldeereignisse werden in einem internen Puffer gespeichert. Die Größe dieses Meldepuffers hängt vom Typ des Bediengeräts ab.

## Meldeprotokoll

Wenn das Protokollieren von Meldungen im Projekt aktiviert wurde, werden Meldeereignisse direkt auf dem angeschlossenen Drucker ausgegeben.

Für jede Meldung kann einzeln projektiert sein, ob sie protokolliert werden soll. Der Druck einer solchen Meldung wird veranlasst, wenn die Meldeereignisse "Kommen" und "Gehen" eintreten.

Falls Meldungen der Meldeklasse "System" gedruckt werden sollen, muss dies über den zugehörigen Meldepuffer veranlasst werden. Dann wird der Meldepufferinhalt vollständig ausgedruckt. Zu diesem Zweck muss im Projekt ein Bedienobjekt, um den Druck auszulösen, projektiert worden sein.

---

**Hinweis**

**Geräteabhängigkeit**

Meldeprotokolle stehen für Basic Panels nicht zur Verfügung.

---

## Meldearchiv

Wenn ein Meldearchiv projiziert ist, werden die Meldeereignisse in diesem Meldearchiv gespeichert. Die Kapazität des Archivs ist durch das Speichermedium und die Systemgrenzen begrenzt.

---

### Hinweis

#### Geräteabhängigkeit

Meldearchive stehen für Basic Panels nicht zur Verfügung.

---

## Meldeanzeige

In der Meldeanzeige werden ausgewählte Meldungen oder Meldeereignisse aus dem Meldepuffer oder Meldearchiv angezeigt. Über die Projektierung wird festgelegt, ob die auftretenden Meldungen quittiert werden müssen. Über die Projektierung kann die Anzeige so gefiltert werden, dass nur Meldungen angezeigt werden, die in Ihrem Meldetext eine bestimmte Zeichenkette enthalten.

## Meldefenster

Das Meldefenster, wenn projiziert, zeigt alle anstehenden oder zu quittierenden Meldungen der jeweiligen Meldeklasse an. Das Meldefenster öffnet sich, sobald eine neue Meldung eintrifft.

Die Sortierung der angezeigten Meldungen ist projektierbar. Entweder wird die aktuelle oder die älteste Meldung zuerst angezeigt. Im Meldefenster kann weiterhin der genaue Störort mit Datum/Uhrzeit des Meldeereignisses angezeigt werden. Über die Projektierung kann die Anzeige so gefiltert werden, dass nur Meldungen angezeigt werden, die in Ihrem Meldetext eine bestimmte Zeichenkette enthalten.

## Meldeindikator

Der Meldeindikator ist ein grafisches Symbol, das am Bildschirm angezeigt wird, wenn eine Meldung der festgelegten Meldeklasse kommt.

Der Meldeindikator kann zwei Zustände haben:

- Blinkend: Mindestens eine unquitierte Meldung steht an.
- Statisch: Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen. Die angezeigte Zahl bedeutet die Anzahl noch anstehender Meldungen.



## Einfache Meldeanzeige, Meldefenster

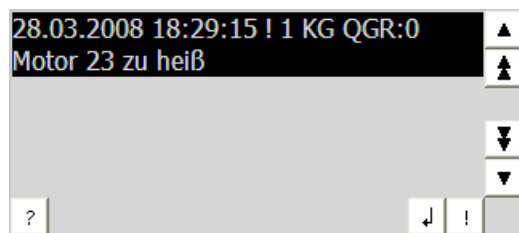
### Verwendung

In der einfachen Meldeanzeige werden ausgewählte Meldungen oder Meldeereignisse aus dem Meldepuffer oder Meldearchiv angezeigt. Aussehen und Bedienung des einfachen Meldefensters entsprechen dem der einfachen Meldeanzeige.

#### Hinweis

Das Objekt "Einfache Meldeanzeige" kann nicht mit einem Skript dynamisiert werden.

Im Engineering System können Sie im Eigenschaftsfenster in der Gruppe "Animationen" z. B. die Sichtbarkeit eines Objekts dynamisieren. Das Objekt "Einfache Meldeanzeige" unterstützt in Runtime keine Animation. Wenn Sie eine Animation projiziert haben und z. B. eine Konsistenzprüfung des Projekts durchführen, dann wird im Ausgabe-Fenster eine Fehlermeldung ausgegeben.



### Darstellung

Abhängig von der Projektierung werden in der Meldeanzeige verschiedene Spalten mit Informationen zu einer Meldung oder einem Meldeereignis angezeigt.

Zur Unterscheidung der verschiedenen Meldeklassen enthält die erste Spalte in der Meldeanzeige ein Symbol:

| Symbol                       | Meldeklasse                     |
|------------------------------|---------------------------------|
| !                            | "Errors"                        |
| (leer)                       | "Warnings"                      |
| (abhängig von Projektierung) | Benutzerdefinierte Meldeklassen |
| \$                           | "System"                        |

### Bedienung

Abhängig von der Projektierung können Sie:

- Meldungen quittieren
- Meldungen bearbeiten

## Bedienelemente

Die Schaltflächen haben folgende Funktionen:

| Schaltfläche | Funktion   |
|--------------|--|
|              | Meldung quittieren   |
|              | Meldung bearbeiten   |
|              | Hilfetext für eine Meldung anzeigen  |
|              | Anzeige des kompletten Meldetextes zur ausgewählten Meldung in einem separaten Fenster, dem Meldetextfenster.<br>Im Meldetextfenster können Sie Meldetexte anzeigen, die mehr Platz erfordern, als in der Meldeanzeige vorhanden ist. Schließen Sie das Meldetextfenster mit |
|              | Blättert um eine Meldung nach oben   |
|              | Blättert in der Meldeanzeige seitenweise nach oben   |
|              | Blättert in der Meldeanzeige seitenweise nach unten  |
|              | Blättert um eine Meldung nach unten  |

## Darstellung der Bedienelemente

Die Darstellung der Schaltflächen zur Bedienung der einfachen Meldeanzeige ist abhängig von der projektierten Größe. Prüfen Sie daher auf dem Bediengerät, ob alle benötigten Schaltflächen zur Verfügung stehen.

## Anstehende Meldungen erkennen

### Einleitung

Das Anstehen quittierpflichtiger Meldungen erkennen Sie an folgenden Kennzeichen:

- Bei einem Bediengerät mit Tasten: Die LED in der Taste leuchtet.
- Abhängig von der Projektierung: Ein Meldeindikator wird auf dem Bildschirm angezeigt.

Ob eine Meldung quittierpflichtig ist, hängt von der Projektierung ab. Die Quittierpflicht wird durch die Meldeklasse bestimmt, der eine Meldung angehört.

### LED in der Taste "ACK"

An einem Bediengerät mit Tasten befindet sich eine LED in der Taste . Die LED leuchtet, solange quittierpflichtige Meldungen anstehen, die Sie noch nicht quittiert haben.

Wenn Sie alle quittierpflichtigen Meldungen quittieren, erlischt die LED.

## Meldeindikator

Der Meldeindikator ist ein grafisches Symbol, das abhängig von der Projektierung auf anstehende oder zu quittierende Meldungen hinweist.



Bild 10-2 Meldeindikator mit drei anstehenden Meldungen

## Darstellung

Der Meldeindikator kann zwei Zustände haben:

- **Blinkend:**  
Solange noch Meldungen unquittiert sind, blinkt der Meldeindikator. Die angezeigte Zahl bedeutet die Anzahl noch anstehender Meldungen. Der Projektteur kann Funktionen projektieren, die beim Bedienen des Meldeindikators ausgeführt werden
- **Statisch:** Die Meldungen sind quittiert, aber mindestens eine davon ist noch nicht gegangen.




## Einblenden von Dialogen

Der eingeblendete Meldeindikator wird z. B. vom Anmeldedialog, der Hilfe und von Meldetextfenstern überdeckt. Wenn diese geschlossen werden, ist der Meldeindikator wieder sichtbar.

## Hilfetext für Meldung anzeigen



### Vorgehen bei Touch-Bedienung

Um den Hilfetext anzuzeigen, gehen Sie folgendermaßen vor:

1. Berühren Sie in der Meldeanzeige oder im Meldefenster die gewünschte Meldung. Die Meldung ist markiert.
2. Berühren Sie die Schaltfläche  in der einfachen Meldeanzeige bzw.  in der erweiterten Meldeanzeige. Wenn ein Hilfetext für die Meldung projiziert wurde, wird er angezeigt.
3. Schließen Sie das Fenster zur Anzeige des Hilfetexts mit der Schaltfläche .

### Vorgehen bei Tasten-Bedienung

Um den Hilfetext anzuzeigen, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Meldeanzeige die gewünschte Meldung.
2. Drücken Sie die Taste .
3. Schließen Sie den Hilfetext mit der Taste .



## Meldung quittieren

### Voraussetzung

Die quittierpflichtige Meldung wird im Meldefenster oder in der Meldeanzeige angezeigt.

### Vorgehen bei Touch-Bedienung







Um eine Meldung zu quittieren, gehen Sie folgendermaßen vor:

1. Berühren Sie in der Meldeanzeige oder im Meldefenster die gewünschte Meldung. Die Meldung ist markiert.
2. Berühren Sie die Schaltfläche  in der einfachen Meldeanzeige bzw.  in der erweiterten Meldeanzeige.

### Vorgehen bei Tasten-Bedienung

Innerhalb einer Meldeanzeige oder eines Meldefensters besteht eine Tabulatorreihenfolge, mit der Sie Bedienelemente und die zuletzt ausgewählte Meldung über die Tastatur anwählen können.

Um eine Meldung zu quittieren, gehen Sie folgendermaßen vor:

1. Markieren Sie mit der Taste  die gewünschte Meldeanzeige oder das gewünschte Meldefenster.
2. Wählen Sie die gewünschte Meldung aus. Verwenden Sie dazu die Tasten , ,  oder .
3. Drücken Sie die Taste .

### Alternative Bedienung

Abhängig von der Projektierung können Sie eine Meldung auch mit einer Funktionstaste quittieren.

### Ergebnis

Die Meldung ist quittiert. Wenn die Meldung zu einer Meldegruppe gehört, sind alle Meldungen der zugehörigen Meldegruppe quittiert.

## Meldung bearbeiten

### Einleitung

Der Projektteur kann für jede Meldung zusätzliche Funktionen projektieren. Diese Funktionen werden ausgeführt, wenn die Meldung bearbeitet wird.

---

#### Hinweis

Wenn Sie eine unquitierte Meldung bearbeiten, wird diese automatisch quittiert.



---

### Voraussetzung

Die zu bearbeitende Meldung wird im Meldefenster bzw. in der Meldeanzeige angezeigt.










### Vorgehen bei Touch-Bedienung

Um eine Meldung zu bearbeiten, gehen Sie folgendermaßen vor:

1. Berühren Sie in der Meldeanzeige oder im Meldefenster die gewünschte Meldung. Die Meldung ist markiert.
2. Berühren Sie die Schaltfläche  in der einfachen Meldeanzeige bzw.  in der erweiterten Meldeanzeige.

### Vorgehen bei Tasten-Bedienung

Um eine Meldung zu bearbeiten, gehen Sie folgendermaßen vor:

1. Markieren Sie mit der Taste  die gewünschte Meldeanzeige oder das gewünschte Meldefenster.
2. Wählen Sie die gewünschte Meldung an. Verwenden Sie dazu die Tasten , ,  oder .
3. Drücken Sie so oft die Taste , bis die Schaltfläche  in der einfachen Meldeanzeige, bzw.  in der erweiterten Meldeanzeige markiert ist.
4. Führen Sie die Bedienung mit der Taste  aus.

### Ergebnis

Die zusätzlichen Funktionen für die Meldung werden ausgeführt. Nähere Hinweise hierzu finden Sie gegebenenfalls in Ihrer Anlagendokumentation.

### 10.11.4.5 Rezepturen bedienen

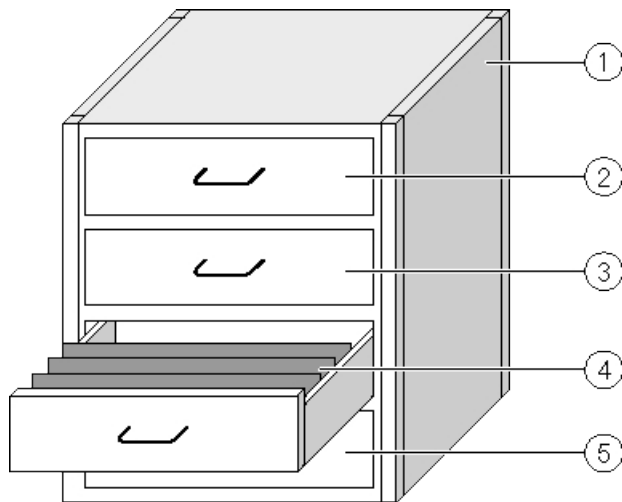
#### Aufbau einer Rezeptur

#### Rezepturen

Sie können die Rezeptursammlung zur Produktion einer Produktfamilie mit einem Aktenschrank vergleichen. Eine Rezeptur, nach der ein Produkt hergestellt wird, entspricht einer Schublade des Aktenschranks.

Beispiel:

Bei einer Anlage zur Getränkeherstellung werden Rezepturen für unterschiedliche Geschmacksrichtungen benötigt. Es gibt z. B. jeweils eine Rezeptur für Getränke der Geschmacksrichtungen Orange, Traube, Apfel und Kirsch.



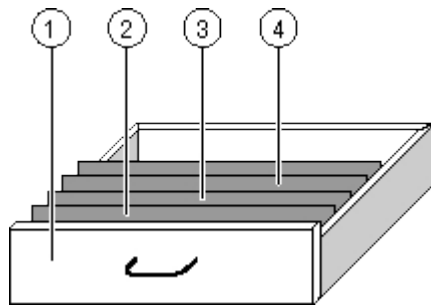
|                |                  |  |
|----------------|------------------|--|
| ① Aktenschrank | Rezeptursammlung | Rezepturen einer Fruchtsaftanlage      |
| ② Schublade    | Rezeptur         | Getränke der Geschmacksrichtung Orange |
| ③ Schublade    | Rezeptur         | Getränke der Geschmacksrichtung Traube |
| ④ Schublade    | Rezeptur         | Getränke der Geschmacksrichtung Apfel  |
| ⑤ Schublade    | Rezeptur         | Getränke der Geschmacksrichtung Kirsch |

#### Rezepturdatensätze

Die Schubladen des Aktenschranks sind mit Hängeordnern gefüllt. Die Hängeordner repräsentieren Datensätze, die für die Herstellung verschiedener Produktvarianten notwendig sind.

Beispiel:

Produktvarianten der Geschmacksrichtung Apfel sind z. B. Getränk, Saft oder Nektar.



- |   |             |                   |  |
|---|-------------|-------------------|--|
| ① | Schublade   | Rezeptur          | Getränkevarianten der Geschmacksrichtung Apfel |
| ② | Hängeordner | Rezepturdatensatz | Apfelgetränk                                   |
| ③ | Hängeordner | Rezepturdatensatz | Apfelnektar                                    |
| ④ | Hängeordner | Rezepturdatensatz | Apfelsaft                                      |

## Elemente

Im Bild des Aktenschrankes enthält jeder Hängeordner die gleiche Anzahl Blätter. Jedes Blatt im Hängeordner entspricht einem Element des Rezepturdatensatzes. Alle Datensätze einer Rezeptur enthalten die gleichen Elemente. Die Datensätze unterscheiden sich jedoch im Wert der einzelnen Elemente.

Beispiel:

Alle Getränke enthalten die gleichen Bestandteile: Wasser, Konzentrat, Zucker und Aroma. Die Datensätze für Getränk, Saft oder Nektar unterscheiden sich jedoch in der Menge des Zuckers, der bei der Herstellung verwendet wird.

## Rezepturen im Projekt

### Übersicht

Wenn in einem Projekt Rezepturen eingesetzt werden, wirken folgende Komponenten zusammen:

- **Rezepturspeicher des Bediengeräts**  
Im Rezepturspeicher des Bediengeräts werden Rezepturen in Form von Datensätzen gespeichert.
- **Rezepturanzeige / Rezepturbild**  
Am Bediengerät werden Rezepturen in der Rezepturanzeige oder in einem Rezepturbild angezeigt und bearbeitet.
  - In der Rezepturanzeige werden die Rezepturdatsätze aus dem internen Speicher des Bediengeräts angezeigt und bearbeitet.
  - Im Rezepturbild werden die Werte der Rezepturvariablen angezeigt und bearbeitet.

---

#### **Hinweis**

Dieselbe Rezepturvariable kann in verschiedenen Rezepturen projiziert sein. Wenn Sie den Wert einer Rezepturvariable ändern, ändert sich durch die Synchronisierung der Wert der Rezepturvariable in allen Rezepturen.

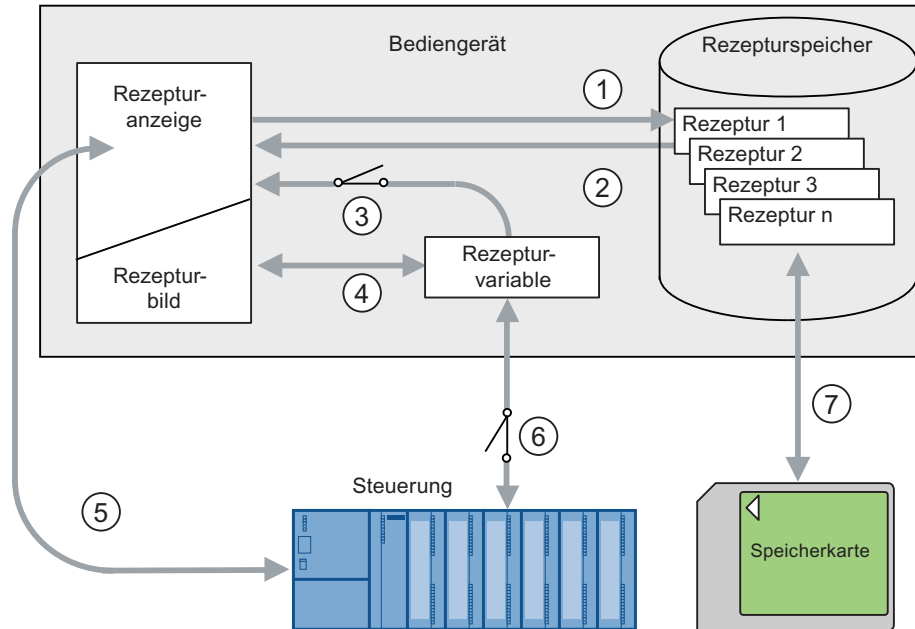
---

- **Rezepturvariablen**  
Die Rezepturvariablen enthalten Rezepturdaten. Wenn Sie Rezepturen in einem Rezepturbild bearbeiten, werden die Rezepturwerte in Rezepturvariablen abgelegt. Wann die Werte der Rezepturvariablen mit der Steuerung ausgetauscht werden, hängt von der Projektierung ab.



## Datenfluss

Das folgende Bild zeigt den Datenfluss in einem Projekt mit Rezepturen:



- ① Rezepturdatensatz bearbeiten, speichern oder löschen.
- ② Rezepturdatensatz anzeigen.
- ③ Rezepturvariablen synchronisieren oder nicht synchronisieren.
- ④ Rezepturvariablen im Rezepturbild anzeigen und bearbeiten.
- ⑤ Datensätze aus der Rezepturanzeige in die Steuerung schreiben oder Datensätze aus der Steuerung lesen und in der Rezepturanzeige anzeigen.
- ⑥ Rezepturvariablen sind zur Steuerung online oder offline.
- ⑦ Rezepturdatensatz auf Speicherkarte exportieren oder importieren.

## Einfache Rezepturanzeige

### Darstellung

Die einfache Rezepturanzeige besteht aus drei Anzeigebereichen:

- Rezepturliste
- Datensatzliste
- Elementliste

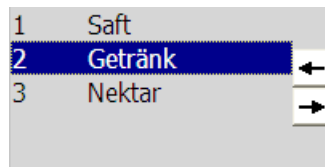


Bild 10-3 Einfache Rezepturanzeige - Beispiel mit Datensatzliste

Jeder Anzeigebereich wird von der einfachen Rezepturanzeige separat am Bediengerät dargestellt. Jeden dieser Anzeigebereiche bedienen Sie mit einem kontextspezifischen Menü. Standardmäßig startet die einfache Rezepturanzeige mit der Rezepturliste.

### Bedienung

Sie können die einfache Rezepturanzeige abhängig von der Projektierung wie folgt bedienen:

- Rezepturdatensatz neu anlegen, ändern, kopieren oder löschen
- Rezepturdatensatz aus Steuerung lesen oder zur Steuerung übertragen

### Bedienelement der einfachen Rezepturanzeige

Zur Bedienung der einfachen Rezepturanzeige wechseln Sie zwischen den Anzeigebereichen und den Kontextmenüs.



Die folgende Tabelle zeigt die Bedienung des Anzeigebereichs.

|                         | Taste | Funktion   |
|-------------------------|-------|--|
| Berühren eines Eintrags |       | Der nächstniedrige Anzeigebereich, d. h. die Datensatzliste oder die Elementliste wird geöffnet. |
|                         |       | Der vorherige Anzeigebereich wird geöffnet.  |
|                         |       | Das Kontextmenü des Anzeigebereichs wird geöffnet.   |

Die folgende Tabelle zeigt die Bedienung der Kontextmenüs:





|                          | Taste                              | Funktion  |
|--------------------------|------------------------------------|---|
|                          |                                    | Das Menü wird geschlossen.<br>Der Anzeigebereich wird geöffnet. |
| Berühren des Menübefehls | Eingabe der Nummer des Menübefehls | Der Menübefehl wird ausgeführt.                                 |

## Kontextspezifische Menüs der einfachen Rezepturanzeige







Zu jedem Anzeigebereich können Sie mit der Schaltfläche  oder mit der Taste  ein Kontextmenü aufrufen. In dem Kontextmenü werden die Befehle angezeigt, die für den aktuellen Anzeigebereich verfügbar sind. Jedem Befehl ist eine Nummer zugeordnet. Wenn Sie die Nummer des Befehls eingeben, wird der Befehl ausgeführt. Alternativ können Sie bei einigen Befehlen die Systemtasten nutzen.

Der Umfang ist abhängig vom Bediengerät.

- Rezepturliste



| Nr. | Menübefehl         | Tasten  | Funktion   |
|-----|--------------------|---|--|
| 0   | Neu                |  +  | Für die markierte Rezeptur wird ein neuer Rezepturdatensatz erstellt.<br>Wenn ein Startwert projektiert wurde, wird dieser im Eingabefeld angezeigt. |
| 1   | Hilfetext anzeigen |    | Der für die einfache Rezepturanzeige projektierte Hilfetext wird angezeigt.  |
| 2   | Öffnen             |    | Die Datensatzliste der markierten Rezeptur wird geöffnet.  |

- Datensatzliste



|  | Menübefehl         | Tasten  | Funktion  |
|--|--------------------|---|---|
|  | Neu                |  +  | Ein neuer Rezepturdatensatz wird erstellt.<br>Wenn ein Startwert projektiert wurde, wird dieser im Eingabefeld angezeigt. |
|  | Löschen            |    | Der markierte Datensatz wird gelöscht.  |
|  | Speichern unter    |   | Der markierte Datensatz unter einem anderen Namen gespeichert. Für die Eingabe des Namens wird ein Dialog geöffnet.       |
|  | Umbenennen         |   | Der markierte Datensatz wird umbenannt. Für die Eingabe des Namens wird ein Dialog geöffnet.                              |
|  | Öffnen             |    | Die Elementliste des markierten Datensatzes wird geöffnet.  |
|  | Zurück             |    | Die Rezepturliste wird geöffnet.  |
|  | Zur Steuerung      |   | Die angezeigten Werte des markierten Datensatzes werden vom Bediengerät in die Steuerung übertragen.                      |
|  | Von Steuerung      |   | Die Rezepturwerte aus der Steuerung werden am Bediengerät in der Rezepturanzeige angezeigt.                               |
|  | Hilfetext anzeigen |    | Der für die einfache Rezepturanzeige projektierte Hilfetext wird angezeigt.   |

- Elementliste





|  | Menübefehl    | Tasten | Funktion   |
|--|---------------|--------|--|
|  | Speichern     |        | Der markierte Datensatz wird gespeichert.  |
|  | Zur Steuerung |        | Die angezeigten Werte des markierten Datensatzes werden vom Bediengerät in die Steuerung übertragen. |
|  | Von Steuerung |        | Die Rezepturwerte aus der Steuerung werden am Bediengerät in der Rezepturanzeige angezeigt.          |

|  | Menübefehl         | Tasten  | Funktion   |
|--|--------------------|---|--|
|  | Speichern unter    |   | Der markierte Datensatz wird unter einem neuen Namen gespeichert. Für die Eingabe des Namens wird ein Dialog geöffnet. |
|  | Hilfetext anzeigen |  | Der für die einfache Rezepturanzeige projektierte Hilfetext wird angezeigt.  |
|  | Umbenennen         |   | Der markierte Datensatz wird umbenannt. Für die Eingabe des Namens wird ein Dialog geöffnet.                           |
|  | Zurück             |  | Die Datensatzliste wird geöffnet.  |

### Maus-Bedienung oder Touch-Bedienung der einfachen Rezepturanzeige

1. Wählen Sie in der Rezepturanzeige die gewünschte Rezeptur.
2. Bedienen Sie die Schaltfläche .  
Das Kontextmenü wird geöffnet.
3. Wählen Sie den gewünschten Menübefehl.  
Der gewünschte Menübefehl wird ausgeführt.
4. Alternativ öffnen Sie in der Rezepturanzeige die gewünschte Rezeptur.  
Die Datensatzliste wird angezeigt.
5. Öffnen Sie den gewünschten Datensatz. Alternativ öffnen Sie mit der Schaltfläche  das Kontextmenü und wählen einen Menübefehl.  
Der gewünschte Menübefehl wird ausgeführt.

### Tastatur-Bedienung der einfachen Rezepturanzeige

1. Drücken Sie so oft die Taste , bis die einfache Rezepturanzeige markiert ist.
2. Wählen Sie mit den Cursortasten die gewünschte Rezeptur.
3. Drücken Sie die Taste .  
Das Kontextmenü wird geöffnet.
4. Drücken Sie so oft die Cursortaste , bis der gewünschte Menübefehl ausgewählt ist.
5. Bestätigen Sie den Menübefehl mit der Taste .
6. Alternativ drücken Sie die Nummer des gewünschten Menübefehls.  
Der gewünschte Menübefehl wird ausgeführt.

### Rezepturdatensatz erstellen

#### Einleitung

Einen neuen Rezepturdatensatz legen Sie in der Rezepturliste oder in der Datensatzliste an. Anschließend geben Sie die Werte für den neuen Datensatz in der Elementliste ein und speichern den Datensatz.

## Voraussetzung

Ein Bild mit einer einfachen Rezepturanzeige wird angezeigt.

## Vorgehen

Um einen Rezepturdatensatz anzulegen, gehen Sie folgendermaßen vor:

1. Wenn die Rezepturliste mehrere Rezepturen enthält: Markieren Sie die Rezeptur, für die Sie einen neuen Rezepturdatensatz erstellen wollen.
2. Öffnen Sie das Menü der Rezepturliste.
3. Wählen Sie den Menübefehl "0 Neu".  
Ein neuer Datensatz wird angelegt.  
Die Elementliste des neuen Datensatzes wird geöffnet.
4. Geben Sie Werte für die Elemente des Datensatzes ein.  
Abhängig von der Projektierung können die Variablen des Datensatzes mit Standardwerten belegt sein.
5. Öffnen Sie das Menü der Elementliste und wählen Sie den Menübefehl "0 Speichern".
6. Geben Sie einen Namen für den neuen Datensatz an.
7. Bestätigen Sie die Eingaben.  
Wenn Sie die neue Datensatznummer in eine vorhandene Datensatznummer ändern, wird der bestehende Datensatz überschrieben.

## Ergebnis

Der neue Rezepturdatensatz ist in der ausgewählten Rezeptur gespeichert.

## Rezepturdatensatz bearbeiten

### Einleitung

Die Werte der Rezepturdatensätze bearbeiten und speichern Sie in einer Rezepturanzeige.

### Abgleich mit der Steuerung

Wenn Sie die aktuellen Rezepturwerte aus der Steuerung in der einfachen Rezepturanzeige anzeigen wollen, müssen Sie zunächst in der Elementliste die aktuellen Werte mit dem Menübefehl "2 Von Steuerung" aus der Steuerung lesen.

Die in der Rezepturanzeige geänderten Werte werden erst dann in der Steuerung wirksam, wenn Sie den geänderten Datensatz mit dem Menübefehl "1 Zur Steuerung" zur Steuerung übertragen.

## Voraussetzung

Ein Bild mit einer Rezepturanzeige wird angezeigt.

## Vorgehen

Um einen Rezepturdatensatz zu bearbeiten, gehen Sie folgendermaßen vor:

1. Wenn die Rezepturliste mehrere Rezepturen enthält: Markieren Sie die Rezeptur, die den gewünschten Rezepturdatensatz enthält.
2. Öffnen Sie die Datensatzliste.
3. Wählen Sie den Rezepturdatensatz, den Sie ändern wollen.
4. Öffnen Sie die Elementliste.
5. Ändern Sie die Werte der Elemente wie gewünscht.
6. Speichern Sie Ihre Änderungen mit dem Menübefehl "0 Speichern".

## Ergebnis

Der geänderte Rezepturdatensatz wird in der ausgewählten Rezeptur gespeichert.

## Rezepturdatensatz löschen

### Einleitung

Sie können alle nicht benötigten Datensätze löschen.

### Voraussetzung

Ein Bild mit einer einfachen Rezepturanzeige wird angezeigt.

## Vorgehen bei Touch-Bedienung

Um einen neuen Rezepturdatensatz zu löschen, gehen Sie folgendermaßen vor:

1. Wenn die Rezepturliste mehrere Rezepturen enthält: Markieren Sie die Rezeptur, die den gewünschten Rezepturdatensatz enthält.
2. Öffnen Sie die Datensatzliste.
3. Markieren Sie den Datensatz, den Sie löschen wollen.
4. Öffnen Sie das Menü.
5. Wählen Sie den Menübefehl "1 Löschen".

## Vorgehen bei Tasten-Bedienung

Um einen neuen Rezepturdatensatz zu löschen, gehen Sie folgendermaßen vor:

1. Wenn die Rezepturliste mehrere Rezepturen enthält: Markieren Sie die Rezeptur, die den gewünschten Rezepturdatensatz enthält.
2. Öffnen Sie die Datensatzliste.

3. Markieren Sie den Datensatz, den Sie löschen wollen.
4. Drücken Sie die Taste .

## Ergebnis

Der Datensatz ist gelöscht.

## Rezepturdatsatz aus der Steuerung lesen

### Einleitung

Die Werte von Rezepturelementen werden über Variablen mit der Steuerung ausgetauscht.

Im laufenden Projekt können Sie direkt in der Anlage die Werte ändern, die auch in den Rezepturen im Bediengerät hinterlegt sind. Z. B. ist dies der Fall, wenn ein Ventil direkt an der Anlage weiter geöffnet wurde als in der Rezeptur hinterlegt ist. Die Werte der Variablen am Bediengerät stimmen dann möglicherweise nicht mehr mit den Werten in der Steuerung überein.

Um die Rezepturwerte abzugleichen, lesen Sie die Werte aus der Steuerung und zeigen sie in der Rezepturanzeige an.

### Voraussetzung

Ein Bild mit einer einfachen Rezepturanzeige wird angezeigt.

### Vorgehen

Um einen Rezepturdatsatz zur Steuerung zu übertragen, gehen Sie folgendermaßen vor:

1. Wenn die Rezepturliste mehrere Rezepturen enthält: Markieren Sie die Rezeptur, die den gewünschten Rezepturdatsatz enthält.
2. Öffnen Sie die Elementliste des Rezepturdatsatzes, in den Sie die Werte aus der Steuerung übernehmen wollen.
3. Öffnen Sie das Menü.
4. Wählen Sie den Menübefehl "2 Von Steuerung".  
Die Werte werden aus der Steuerung gelesen.
5. Wenn Sie die angezeigten Werte im Bediengerät speichern wollen, wählen Sie den Menübefehl "0 Speichern".

## Ergebnis

Die Werte wurden aus der Steuerung gelesen, sind am Bediengerät sichtbar und im ausgewählten Rezepturdatensatz gespeichert.

---

### Hinweis

#### Basic Panels

Bei Basic Panels kann der Menübefehl "Von Steuerung" auch für die Datensatzliste projiziert worden sein. In diesem Fall können Sie den Menübefehl "Von Steuerung" auch in der Datensatzliste wählen.

---

## Rezepturdatensatz zur Steuerung übertragen

### Einleitung

Damit ein geänderter Rezepturdatensatz im Prozess wirksam wird, müssen Sie die Werte in die Steuerung übertragen.

Es werden immer die in der Rezepturanzeige angezeigten Werte zur Steuerung übertragen.

### Voraussetzung

Ein Bild mit einer einfachen Rezepturanzeige wird angezeigt.

### Vorgehensweise

Um einen Rezepturdatensatz zur Steuerung zu übertragen, gehen Sie folgendermaßen vor:

1. Wenn die Rezepturliste mehrere Rezepturen enthält: Markieren Sie die Rezeptur, die den gewünschten Rezepturdatensatz enthält.
2. Öffnen Sie die Elementliste des Rezepturdatensatzes, dessen Werte Sie zur Steuerung übertragen wollen.
3. Öffnen Sie das Menü.
4. Wählen Sie den Menübefehl "1 Zur Steuerung".

## Ergebnis

Die Werte des Rezepturdatensatzes wurden zur Steuerung übertragen und werden im Prozess wirksam.

---

### Hinweis

#### Basic Panels

Bei Basic Panels kann der Menübefehl "Zur Steuerung" auch für die Datensatzliste projiziert worden sein. In diesem Fall können Sie den Menübefehl "Zur Steuerung" auch in der Datensatzliste wählen.

---



## 10.12 Leistungsmerkmale

### 10.12.1 Engineering System

#### Engineering System

Die folgenden Tabellen unterstützen Sie bei der Abschätzung, ob Ihr Projekt noch innerhalb der Leistungsmerkmale für das Engineering System liegt.

Zusätzlich zu den angegebenen Grenzen ist noch die Begrenzung durch den zur Verfügung stehenden Hauptspeicher zu beachten. WinCC nutzt betriebssystembedingt maximal 2 Gbyte des vorhandenen Hauptspeichers. Ein Hauptspeicher größer als 2 Gbyte ist dennoch sinnvoll, wenn Sie auf dem PC viele Applikationen mit hohem Speicherbedarf gleichzeitig nutzen.

#### Systemgrenzen für ein Projekt

|  | WinCC   |
|--|---------|
| Anzahl Bediengeräte im Projekt                                     | 35      |
| Anzahl HMI-Variablen <sup>1)</sup>                                 | 80.000  |
| Anzahl Archivvariablen   | 8.000   |
| Anzahl Bausteine (Bildbausteine, Anwenderdatentypen) <sup>3)</sup> | 10.000  |
| Anzahl Bilder  | 3.000   |
| Anzahl Bildobjekte pro Bild  | 3.000   |
| Anzahl Bildobjekte   | 320.000 |
| Anzahl Meldungen <sup>2) 3)</sup>                                  | 20.000  |
| Anzahl Texte <sup>3)</sup>   | 300.000 |
| Anzahl Textlisten und Grafiklisten <sup>3)</sup>                   | 10.000  |
| Anzahl Einträge je Textliste                                       | 3.000   |
| Anzahl Sprachen  | 32      |
| Anzahl Globale Bibliotheken <sup>3)</sup>                          | 20      |
| Anzahl Objekte in der Projektbibliothek <sup>3)</sup>              | 300.000 |

1) Inklusive Archivvariablen.

2) Mit durchschnittlich 5 Texten und einem dynamischen Parameter

3) Inklusive der im Bereich "PLC programmieren" projektierten Objekte

#### Systemgrenzen für ein HMI-Gerät

|                                    | WinCC  |
|------------------------------------|--------|
| Anzahl HMI-Variablen <sup>1)</sup> | 80.000 |
| Anzahl Archivvariablen             | 8.000  |

|                                       | WinCC   |
|---------------------------------------|---------|
| Anzahl Archive                        | 500     |
| Anzahl Bilder                         | 1000    |
| Anzahl Bildobjekte pro Bild           | 3.000   |
| Anzahl Bildobjekte                    | 320.000 |
| Anzahl Funktionslisten                | 30.000  |
| Anzahl Animationen und lokale Skripte | 50.000  |
| Anzahl benutzerdefinierte Funktionen  | 1.000   |
| Anzahl Aufgaben                       | 500     |
| Anzahl Meldungen <sup>2)</sup>        | 20.000  |
| Anzahl Rezepturen                     | 1.000   |
| Anzahl Rezepturelemente               | 10.000  |
| Anzahl Texte                          | 100.000 |
| Anzahl Textlisten und Grafiklisten    | 1.000   |
| Anzahl Einträge je Textliste          | 3.000   |
| Anzahl Benutzer                       | 200     |
| Anzahl Protokolle                     | 300     |

- 1) Inklusive Archivvariablen.
- 2) Mit durchschnittlich 5 Texten und einem dynamischen Parameter

### Systemgrenzen bei der Migration

Sie können Projekte migrieren, die in einem oder mehreren Punkten über den angegebenen Systemgrenzen liegen.

Wenn durch die Migration ein Projekt entsteht, dessen Grenzen die angegebenen Systemgrenzen überschreitet, werden Sie durch eine Meldung darauf aufmerksam gemacht. Um den sicheren Betrieb in WinCC zu gewährleisten, müssen Sie nach der Migration das Projekt so anpassen, dass die oben angegebenen Systemgrenzen eingehalten werden.

### 10.12.2 Basic Panel

#### Basic Panel

Die folgende Tabelle unterstützt Sie bei der Abschätzung, ob Ihr Projekt noch innerhalb der Leistungsmerkmale für das Bediengerät liegt.

Die angegebenen Maximalwerte sind nicht additiv. Die Funktionsfähigkeit von Projektierungen, in denen alle Systemgrenzen ausgenutzt werden, kann auf den Geräten nicht gewährleistet werden.

Zusätzlich zu den angegebenen Grenzen ist noch die Begrenzung durch den zur Verfügung stehenden Projektierungs-Speicher zu beachten.

## Variablen

|                             | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic           | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|-----------------------------|----------------|----------------|---------------------------|-----------------|------------------|-----------------|
| Anzahl Variablen im Projekt | 250            | 500            | 250 (mono)<br>500 (color) | 500             | 500              | 500             |
| Anzahl PowerTags            | --             | --             | --                        | --              | --               | --              |
| Anzahl Elemente je Array    | 100            | 100            | 100                       | 100             | 100              | 100             |
| Anzahl Lokale Variablen     | --             | --             | --                        | --              | --               | --              |

## Meldungen

|                                   | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|-----------------------------------|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Meldeklassen               | 32             | 32             | 32              | 32              | 32               | 32              |
| Anzahl Bitmeldungen               | 200            | 200            | 200             | 200             | 200              | 200             |
| Anzahl Analogmeldungen            | 15             | 15             | 15              | 15              | 15               | 15              |
| Länge einer Meldung in Zeichen    | 80             | 80             | 80              | 80              | 80               | 80              |
| Anzahl Prozesswerte je Meldung    | 8              | 8              | 8               | 8               | 8                | 8               |
| Größe des Meldepuffers            | 256            | 256            | 256             | 256             | 256              | 256             |
| Anzahl anstehende Meldeereignisse | 64             | 64             | 64              | 64              | 64               | 64              |

## Bilder

|  | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|--|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Bilder                                  | 50             | 50             | 50              | 50              | 50               | 50              |
| Anzahl Felder pro Bild                         | 30             | 30             | 30              | 30              | 30               | 30              |
| Anzahl Variablen pro Bild                      | 30             | 30             | 30              | 30              | 30               | 30              |
| Anzahl Komplexe Objekte pro Bild <sup>1)</sup> | 5              | 5              | 5               | 5               | 5                | 5               |
| Anzahl Array-Elemente pro Bild <sup>2)</sup>   | 100            | 100            | 100             | 100             | 100              | 100             |

- 1) Komplexe Objekte sind: Balken, Schieberegler, Symbolbibliothek, Uhr und alle Objekte aus dem Bereich Controls.
- 2) In Rezepturen enthaltene Array-Elemente zählen dabei mit.

## Rezepturen

|  | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|--|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Rezepturen                                      | 5              | 5              | 5               | 5               | 5                | 5               |
| Anzahl Elemente pro Rezeptur <sup>1)</sup>             | 20             | 20             | 20              | 20              | 20               | 20              |
| Nutzdatenlänge in Byte pro Datensatz                   | --             | --             | --              | --              | --               | --              |
| Anzahl Datensätze pro Rezeptur                         | 20             | 20             | 20              | 20              | 20               | 20              |
| Reservierter Speicher für Datensätze im internen Flash | 40 kB          | 40 kB          | 40 kB           | 40 kB           | 40 kB            | 40 kB           |

- 1) Bei Verwendung von Arrays, zählt jedes Arrayelement als ein Rezepturelement

## Archive

|  | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|--|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Archive   | --             | --             | --              | --              | --               | --              |
| Anzahl Einträge je Archiv (incl. aller Archive-segmente) <sup>1)</sup> | --             | --             | --              | --              | --               | --              |
| Anzahl Archivsegmente  | --             | --             | --              | --              | --               | --              |
| zyklischer Trigger für die Archivierung von Variablen                  | --             | --             | --              | --              | --               | --              |
| Anzahl archivierbarer Variablen je Archiv                              | --             | --             | --              | --              | --               | --              |

- 1) Bei der Archivierungsmethode "segmentiertes Umlaufarchiv" gilt die Anzahl der Einträge für die Gesamtheit aller Folgearchive. Das Produkt aus der Anzahl der Folgearchive und der Anzahl der Datensätze pro Folgearchiv darf die Systemgrenze nicht überschreiten

## Kurven

|               | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|---------------|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Kurven | 25             | 25             | 25              | 25              | 25               | 25              |

## Textlisten und Grafiklisten

|                     | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|---------------------|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Grafiklisten | 100            | 100            | 100             | 100             | 100              | 100             |
| Anzahl Textlisten   | 150            | 150            | 150             | 150             | 150              | 150             |

|  | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|--|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl der Einträge pro Text- oder Grafikliste | 30             | 30             | 30              | 30              | 30               | 30              |
| Anzahl Grafikobjekte                           | 500            | 500            | 500             | 500             | 500              | 500             |
| Anzahl Textelemente                            | 500            | 500            | 500             | 500             | 500              | 500             |

## Skripte

|                | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|----------------|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Skripte | --             | --             | --              | --              | --               | --              |

## Kommunikation

|  | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|--|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Verbindungen                                  | 4              | 4              | 4               | 4               | 4                | 4               |
| Anzahl Verbindungen basierend auf "SIMATIC HMI HTTP" | --             | --             | --              | --              | --               | --              |

Bei der Kommunikation mit S7-1200 Steuerungen ist zu beachten, dass pro Steuerung maximal 200 Variablen projiziert werden sollten. Wenn mehrere HMI-Geräte auf eine Steuerung zugreifen, gilt diese Grenze für alle HMI-Geräte gemeinsam.

Auf eine S7-1200 können maximal 4 HMI-Geräte gleichzeitig zugreifen. Von einem Panel können maximal 8 Steuerungen angesprochen werden.

## Hilfesystem

|                                    | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|------------------------------------|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Länge eines Hilfetextes in Zeichen | 320            | 320            | 320             | 320             | 320              | 320             |

## Sprachen

|                        | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|------------------------|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Runtimesprachen | 5              | 5              | 5               | 5               | 5                | 5               |

## Aufgabenplaner

|  | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|--|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Zeitgetriggerte Aufgaben <sup>1)</sup> | --             | --             | --              | --              | --               | --              |

- 1) Ereignisgetriggerte Aufgaben sind für die Systemgrenzen nicht relevant

## Benutzerverwaltung

|                        | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|------------------------|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Anzahl Benutzergruppen | 50             | 50             | 50              | 50              | 50               | 50              |
| Anzahl Berechtigungen  | 32             | 32             | 32              | 32              | 32               | 32              |
| Anzahl Benutzer        | 50             | 50             | 50              | 50              | 50               | 50              |

## Projekt

|                                | KP300<br>Basic | KP400<br>Basic | KTP400<br>Basic | KTP600<br>Basic | KTP1000<br>Basic | TP1500<br>Basic |
|--------------------------------|----------------|----------------|-----------------|-----------------|------------------|-----------------|
| Größe der Projektdatei "*.srt" | 512 kB         | 512 kB         | 512 kB          | 512 kB          | 1024 kB          | 1024 kB         |

## 10.12.3 Allgemeine technische Daten

### 10.12.3.1 Empfohlene Drucker

#### Empfohlene Drucker

Die aktuelle Liste der empfohlenen Drucker für die Bediengeräte finden Sie im Internet:

Link zur aktuellen Druckerliste (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=11376409&caller=view>)

#### Hinweis

Alle Bediengeräte außer PC und Panel-PC unterstützen maximal einen Drucker über USB-Anschluss, auch wenn physikalisch mehrere Anschlüsse vorhanden sind.

## Siehe auch

Druckerliste (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=11376409&caller=view>)

### 10.12.3.2 Speicherplatzbedarf von Rezepturen für Basic Panels

#### Einleitung

Die nachfolgende Berechnung des Speicherbedarfs von Rezepturen ist für Basic Panels sowie die Geräte OP 77A und TP 177A gültig.

#### Einschränkungen

Der im Bediengerät vorhandene Speicherplatz für Rezepturen beträgt 39 Kilobyte. Dieser Speicherplatz darf nicht überschritten werden. Der Gesamtspeicherplatz für Rezepturen wird folgendermaßen berechnet: Summe aller Rezepturen + Rezeptur mit größtem Speicherbedarf.

Jede einzelne Rezeptur darf eine maximale Speichergröße von 19 Kilobyte nicht überschreiten.

#### Speicherbedarf berechnen

Der erforderliche Speicherbedarf pro Rezeptur in Kilobyte errechnet sich aus den drei Summanden D1 + D2 + D3.

Es gilt:

- D1 = Anzahl Datensätze x M

Für M (Größe eines Datensatzes) gilt:

$M = 1 \times \text{Anzahl der Elemente eines Bytes} + 2 \times \text{Anzahl der Elemente von 2 Bytes} + 4 \times \text{Anzahl der Elemente von 4 Bytes} + 8 \times \text{Anzahl der Elemente von 8 Bytes} + K$

Für K (Größe der String-Elemente) gilt:

$K = \text{Anzahl der String-Elemente} \times (\text{Größe des Strings} + 1) \times 2$

- D2 - Datensatzgröße

$D2 = 4 + \text{Anzahl der Sprachen} \times 8 + \text{Anzahl der Sprachen} \times (4 + 4 \times \text{Anzahl der Datensätze} + (\text{Länge des Datensatznamens} + 1) \times 2 \times \text{Anzahl der Datensätze}) + 8 + 8 \times \text{Anzahl der Datensätze}$

Oder umformuliert:

$D2 = 12 + 8 \times \text{Anzahl der Datensätze} + \text{Anzahl der Sprachen} \times (12 + \text{Anzahl der Datensätze} \times (4 + (\text{Länge des Datensatznamens} + 1) \times 2))$

- D3 Gemeinsam genutzter Speicher  
D3 = 14 + Anzahl der Elemente

---

**Hinweis**

Die Berechnung für Arrays und Einzelelemente kann wie oben beschrieben durchgeführt werden.

---

## 10.13 Umstieg auf WinCC V12

### 10.13.1 Übersicht zum Umstieg auf WinCC V12

#### Übersicht zum Kapitel "Umstieg auf WinCC V12"

SIMATIC WinCC V12 bietet eine Reihe funktioneller Änderungen. Einige Funktionen unterscheiden sich von den Funktionen, die Sie aus gewohnten Umgebungen wie WinCC V7 oder WinCC flexible her kennen.

Dieses Dokument gibt einen Überblick über spezielle Funktionen und Vorgehensweisen in SIMATIC WinCC V12.

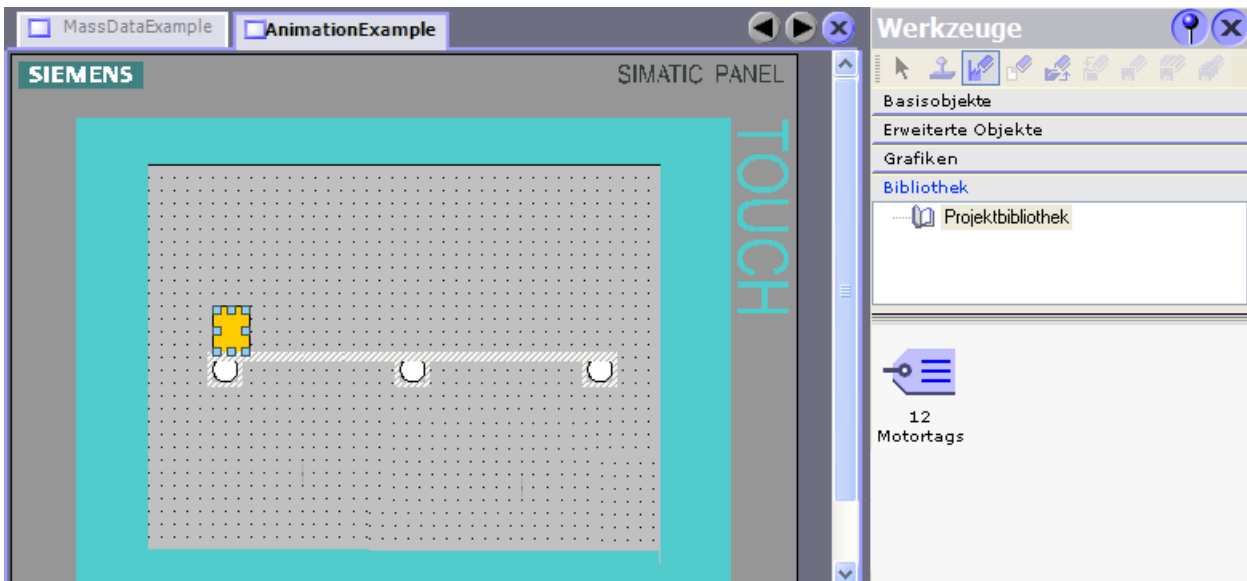
Es sind Funktionen und Vorgehensweisen, die sich grundlegend von den Versionen WinCC V7 und WinCC flexible unterscheiden oder einen anderen Namen tragen.



## 10.13.2 Bibliotheken

### Bibliotheken in WinCC flexible

Bibliotheken sind Sammlungen von bereits konfigurierten Bildobjekten. Sie erweitern die Menge der verfügbaren Bildobjekte und steigern die Effektivität beim Projektieren, da Bibliotheksobjekte ohne erneute Konfigurierung immer wieder verwendet werden können.



In WinCC flexible können Sie zwei Bibliothekstypen anlegen:

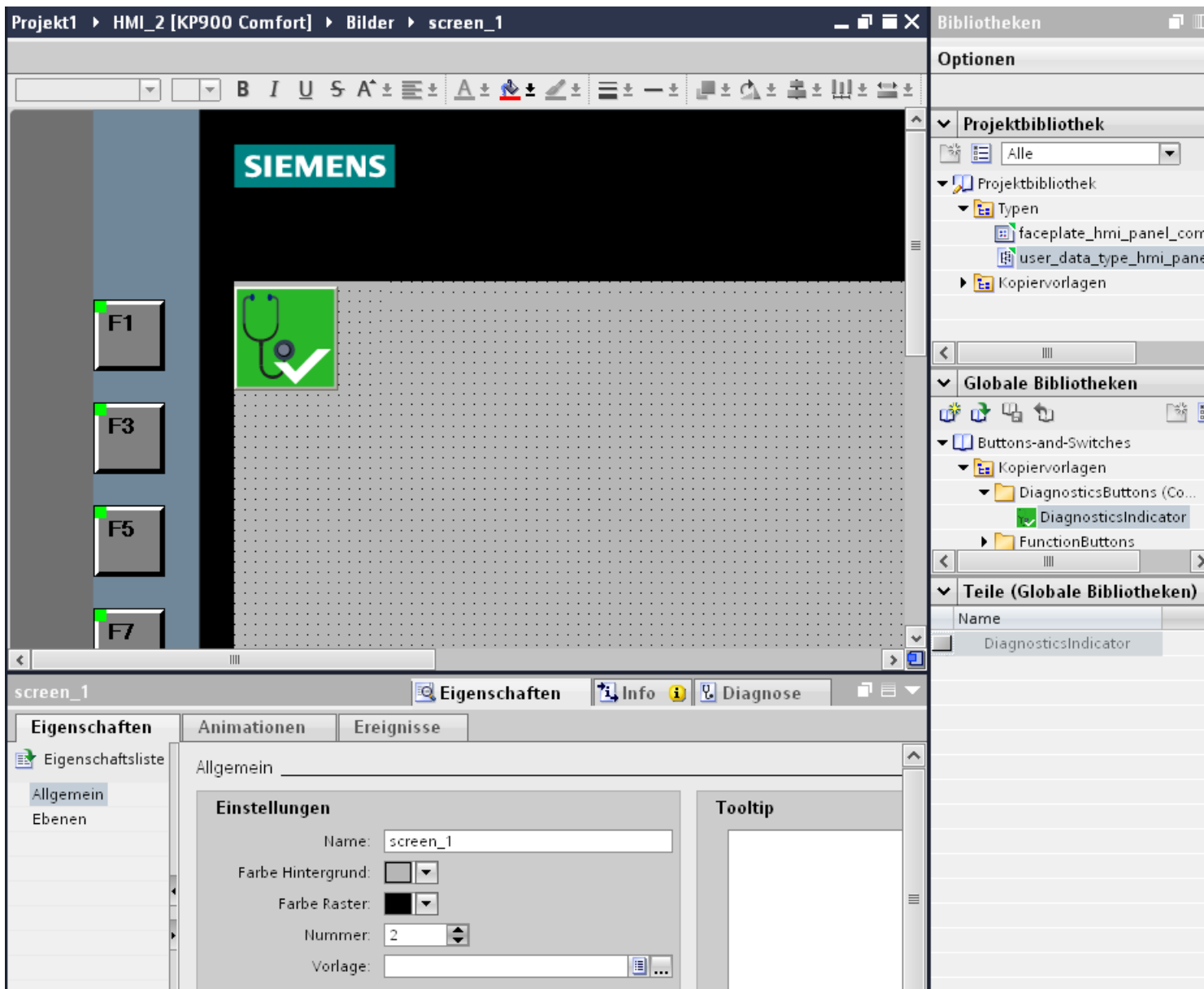
- Projektbibliothek
- Globale Bibliothek

Eine Bibliothek kann alle WinCC flexible Objekte, z. B. Bilder, Variablen, Grafikobjekte oder Meldungen enthalten.

### Wie projettierte ich Bibliotheken in WinCC V12?

In WinCC V12 projektieren Sie ebenfalls die "Projektbibliothek" und die "Globale Bibliothek".

Im Vergleich zu WinCC flexible können Sie in Bibliotheken keine Systemfunktionen ablegen.



Sowohl die "Projektbibliothek" als auch die "Globale Bibliothek" enthalten die beiden Ordner "Kopiervorlagen" und "Typen". Sie können die Bibliotheksobjekte entweder als Kopiervorlage oder als Typ erzeugen bzw. verwenden.

- **Kopiervorlagen**  
Um voneinander unabhängige Kopien des Bibliotheksobjekts zu erzeugen, verwenden Sie Kopiervorlagen.
- **Typen**  
Von Objekten des Ordners "Typen" erstellen Sie Instanzen und verwenden die Instanzen in Ihrem Projekt. Die Instanzen sind an ihren jeweiligen Typ gebunden. Änderungen einer Instanz ändern auch alle anderen Instanzen. Typen werden in der Task Card "Bibliotheken" mit einem grünen Dreieck gekennzeichnet.
- **Verwalten der Bibliotheksobjekte**  
Sie kopieren und verschieben Bibliotheksobjekte nur innerhalb der gleichen Bibliothek.

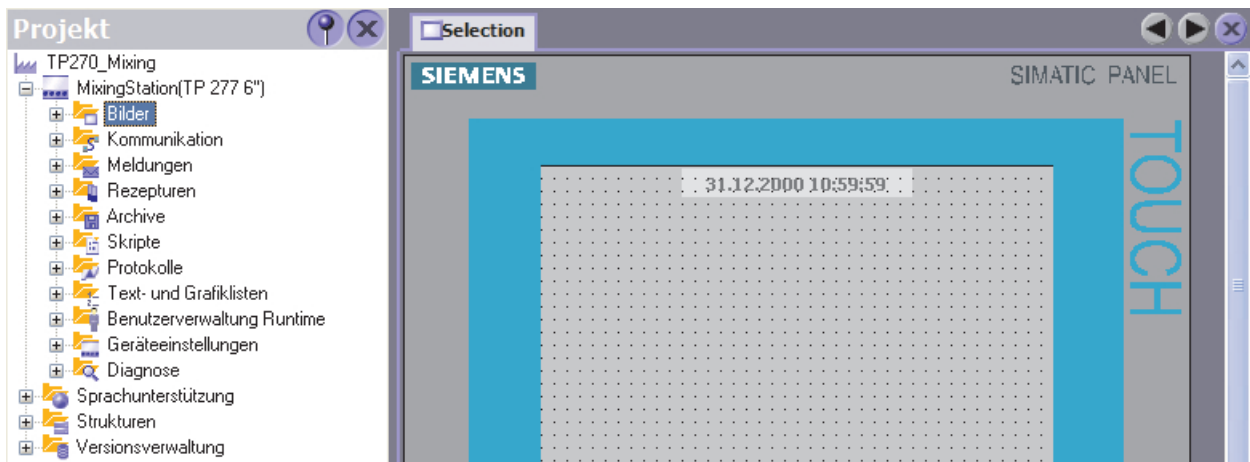
Nähere Hinweise hierzu finden Sie unter:

Bibliotheken in WinCC (Seite 3842)

### 10.13.3 Bilder und Vorlagen

#### Bilder und Vorlagen in WinCC flexible

Mit WinCC flexible erstellen Sie Bilder zum Bedienen und Beobachten von Maschinen und Anlagen. Um Bilder zu erstellen, stehen Ihnen vordefinierte Objekte zur Verfügung, mit denen Sie Ihre Anlage nachbilden, Prozessabläufe anzeigen und Prozesswerte vorgeben.



Für jedes Bediengerät existiert im Projekt eine Vorlage, in dem Sie Funktionstasten und Objekte für Ihr Projekt zentral projektieren können.

Jedes Bild, das auf der Vorlage basiert, enthält die in der Vorlage projizierten Funktionstasten und Objekte. Wenn Sie in der Vorlage ein Objekt oder die Belegung einer Funktionstaste ändern, ändert sich das Objekt in allen Bildern, die auf der Vorlage basieren.

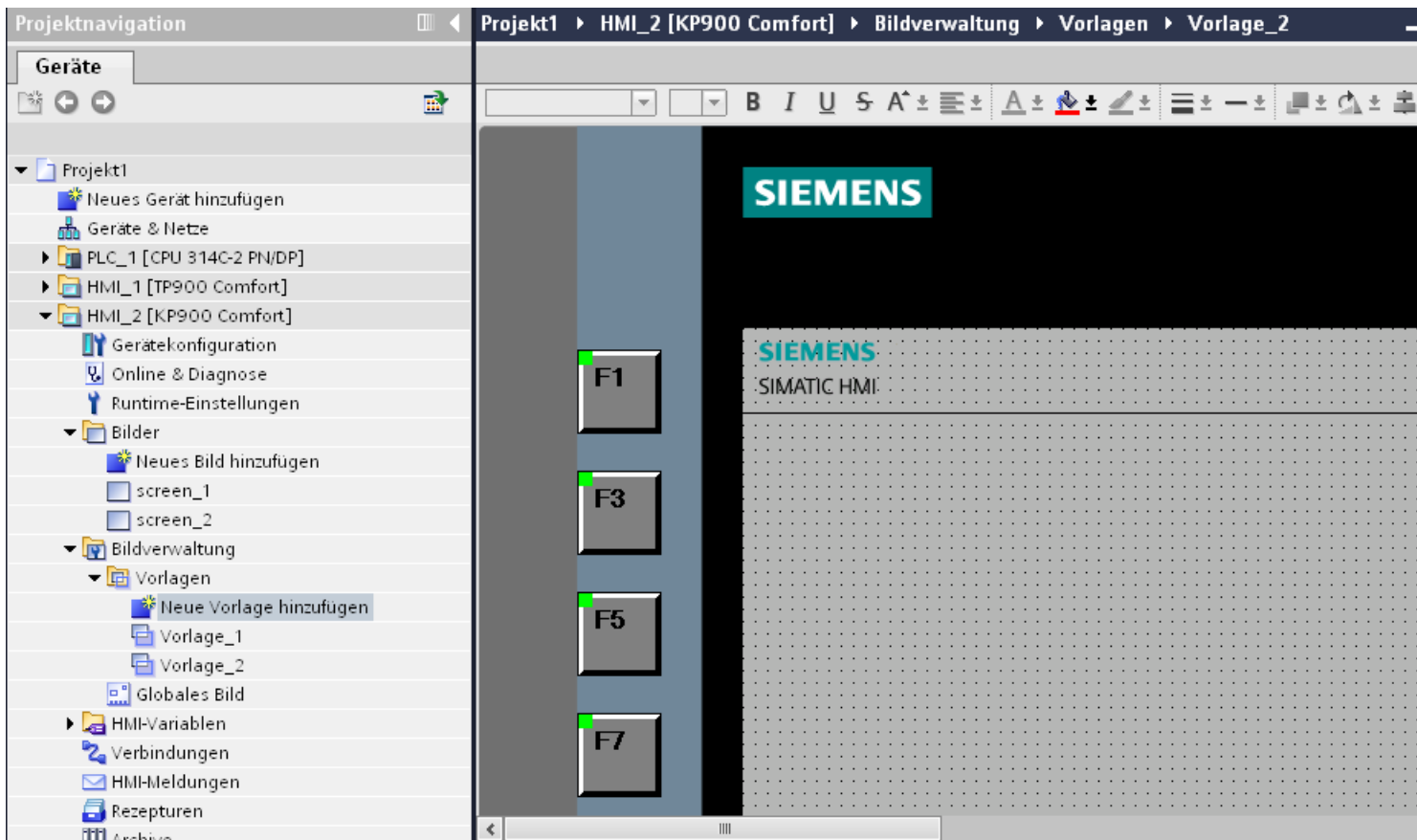
#### Wie projektieren ich Bilder und Vorlagen in WinCC V12?

In WinCC V12 projektieren Sie neben "Bildern" auch "Vorlagen" und ein "Globales Bild".

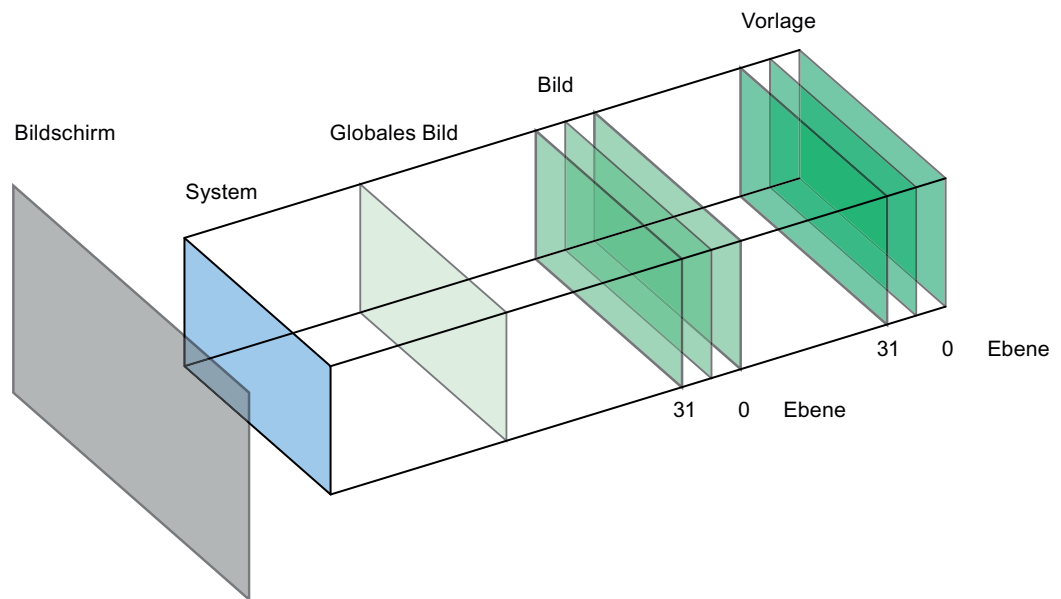
Innerhalb einer Vorlage legen Sie Funktionen und Objekte fest, die dann für alle Bilder auf der Grundlage dieser Vorlage gelten. Sie können in WinCC mehrere Vorlagen anlegen.

Für alle Bilder eines Bediengeräts legen Sie im "Globalen Bild" Elemente fest, die unabhängig von der verwendeten Vorlage sind. Als globale Objekte stehen die Objekte "Meldefenster", "Meldeindikator" zur Verfügung. Bei Bediengeräten mit Funktionstasten belegen Sie die Funktionstasten innerhalb des Editors "Globales Bild".

Für Comfort Panels haben Sie auch die Möglichkeit ein "System-Diagnosefenster" im Globalen Bild zu projektieren.



Bei der Anzeige der Bilder in Runtime gilt mit Ausnahme der Controls folgende Reihenfolge:



Nähere Hinweise hierzu finden Sie unter:  
Grundlagen zu Bildern (Seite 3025)



## Technologiefunktionen einsetzen

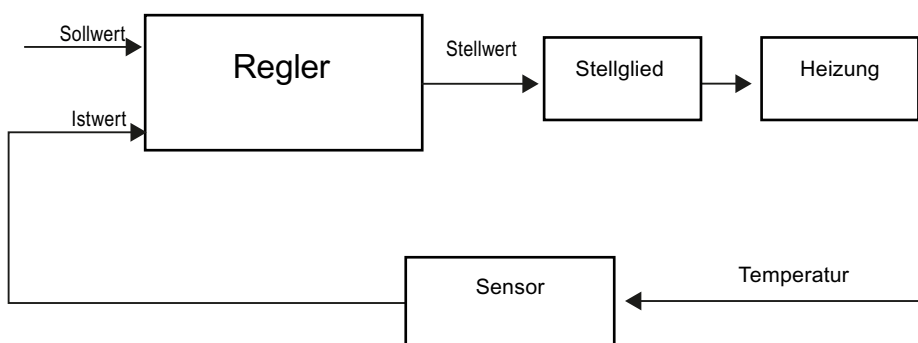
### 11.1 PID-Regelung

#### 11.1.1 Grundlagen zum Regeln

##### 11.1.1.1 Regelkreis und Stellglieder

#### Regelkreis

Ein einfaches Beispiel für einen Regelkreis ist die Regelung der Raumtemperatur durch eine Heizung. Die Raumtemperatur wird mit einem Sensor gemessen und einem Regler zugeführt. Dieser vergleicht die aktuelle Raumtemperatur mit einem Sollwert und berechnet einen Ausgangswert (Stellwert) für die Ansteuerung der Heizung.



Ein korrekt eingestellter PID-Regler erreicht den Sollwert möglichst schnell und hält ihn dann konstant. Nach einer Änderung des Ausgangswerts ändert sich der Istwert häufig erst zeitlich verzögert. Dieses Verhalten soll durch den Regler ausgeglichen werden.

#### Stellglieder

Das Stellglied ist Bestandteil des Regelkreises und wird vom Regler beeinflusst. Dadurch wird ein Masse- oder Energiefluss verändert.

Die folgende Tabelle gibt einen Überblick über den Einsatz der Stellglieder

| Einsatz für                          | Stellglied                                      |
|--------------------------------------|---|
| flüssigen und gasförmigen Massefluss | Ventil, Klappe, Schieber                        |
| festen Massefluss, z. B. Schüttgüter | Abzugschieber, Förderband, Vibratorrinne        |
| elektrischen Energiefluss            | Schaltkontakt, Schaltschütz, Relais, Thyristor  |
|                                      | Stellwiderstand, Stelltransformator, Transistor |

Stellglieder werden folgendermaßen unterschieden:

- Proportionale Stellglieder mit stetigem Stellsignal  
Proportional zum Ausgangswert werden Öffnungsgrade, Drehwinkel oder Positionen eingenommen. Der Ausgangswert wirkt innerhalb des Stellbereiches in analoger Weise auf den Prozess ein.  
Zu den Stellgliedern dieser Gruppe gehören federbelastete pneumatische Antriebe, aber auch motorische Antriebe mit Stellungsrückmeldung, bei denen ein Stellungsregelkreis gebildet wird.  
Der Ausgangswert wird von einem kontinuierlichen Regler erzeugt, z. B. PID\_Compact.
- Proportionale Stellglieder mit pulswertenmoduliertem Signal  
Bei diesen Stellgliedern wird im Takt der Abtastzeit ein Impuls ausgegeben, dessen Länge proportional zum Ausgangswert ist. Das Stellglied - z. B. ein Heizwiderstand oder ein Kühlaggregat - wird taktsynchron je nach Ausgangswert unterschiedlich lange eingeschaltet.  
Das Stellsignal kann entweder unipolar die Zustände "Ein" oder "Aus" annehmen oder bipolar z. B. die Werte "Auf/Zu", "Vorwärts/Rückwärts", "Beschleunigen/Bremsen" usw. repräsentieren.  
Der Ausgangswert wird von einem Zweipunktregler erzeugt, z. B. PID\_Compact mit Pulsweitenmodulation.
- Integral wirkende Stellglieder mit Dreipunktstellsignal  
Häufig werden Stellglieder durch Motoren betätigt, bei denen die Einschaltdauer proportional zum Verstellweg des Drosselorgans ist. Dazu zählen z. B. Ventile, Klappen und Schieber. Trotz unterschiedlichster Bauformen ist diesen Stellgliedern gemeinsam, dass sie der Wirkung eines I-Anteils am Streckeneingang entsprechen.  
Der Ausgangswert wird von einem Schrittreger erzeugt, z. B. PID\_3Step.

### 11.1.1.2 Regelstrecken

Die Eigenschaften einer Regelstrecke sind durch verfahrenstechnische und maschinentechnische Gegebenheiten festgelegt und kaum zu beeinflussen. Ein gutes Regelergebnis kann nur erreicht werden durch Auswahl eines für diese Regelstrecke geeigneten Reglertyps und dessen Anpassung an das Zeitverhalten der Regelstrecke. Genaue Kenntnisse des Typs und der Kenndaten der Regelstrecke sind deshalb für die Konfiguration von P-, I- und D-Anteil des Reglers unerlässlich.

### Regelstreckentypen

Regelstrecken werden klassifiziert anhand ihres Zeitverhaltens auf die sprunghafte Veränderung des Ausgangswerts.

Folgende Regelstrecken werden unterschieden:

- Regelstrecken mit Ausgleich
  - P-Regelstrecken
  - PT1-Regelstrecken
  - PT2-Regelstrecken
- Regelstrecken ohne Ausgleich
- Regelstrecken mit und ohne Totzeit



## Regelstrecken mit Ausgleich

### P-Regelstrecken

Bei Proportionalstrecken folgt der Istwert nahezu unverzögert dem Ausgangswert. Das Verhältnis von Istwert zu Ausgangswert wird angegeben durch den Proportionalbeiwert Gain der Regelstrecke.

Beispiele:

- Ventilschieber in einem Rohrsystem
- Spannungsteiler
- Untersetzung in Hydrauliksystemen

### PT1-Regelstrecken

Bei einer PT1-Regelstrecke ändert sich der Istwert zunächst proportional zur Änderung des Ausgangswerts. Mit der Zeit ändert sich der Istwert langsamer - also verzögert - bis ein Endwert erreicht ist.

Beispiele:

- Feder-Dämpfer-System
- Ladung von RC-Gliedern
- Wasserbehälter, der mit Dampf aufgeheizt wird.

Häufig gelten gleiche Zeitkonstanten für Heiz- und Kühlvorgänge oder Lade- und Entladekurven. Unterscheiden sich die die Zeitkonstanten, wird die Regelung deutlich schwieriger.

### PT2-Regelstrecken

Bei einer PT2-Regelstrecke ändert sich der Istwert bei einem Ausgangswertsprung zunächst nicht, steigt dann mit zunehmender Steigung an und strebt danach mit abnehmender Steigung dem Sollwert zu. Die Regelstrecke zeigt ein proportionales Übertragungsverhalten mit Verzögerung 2. Ordnung.

Beispiele:

- Druckregelung
- Durchflussregelung
- Temperaturregelung

## Regelstrecken ohne Ausgleich

Regelstrecken ohne Ausgleich zeigen integrales Verhalten. Der Istwert strebt einen unendlich großen Wert an.

Beispiel:

- Zufluss einer Flüssigkeit in einen Behälter

## Regelstrecken mit Totzeit

Eine Totzeit ist immer die Lauf- oder Transportzeit, bis eine Änderung am Systemeingang am Systemausgang gemessen werden kann.

Bei Regelstrecken mit Totzeit ändert sich der Istwert um die Totzeit verspätet.

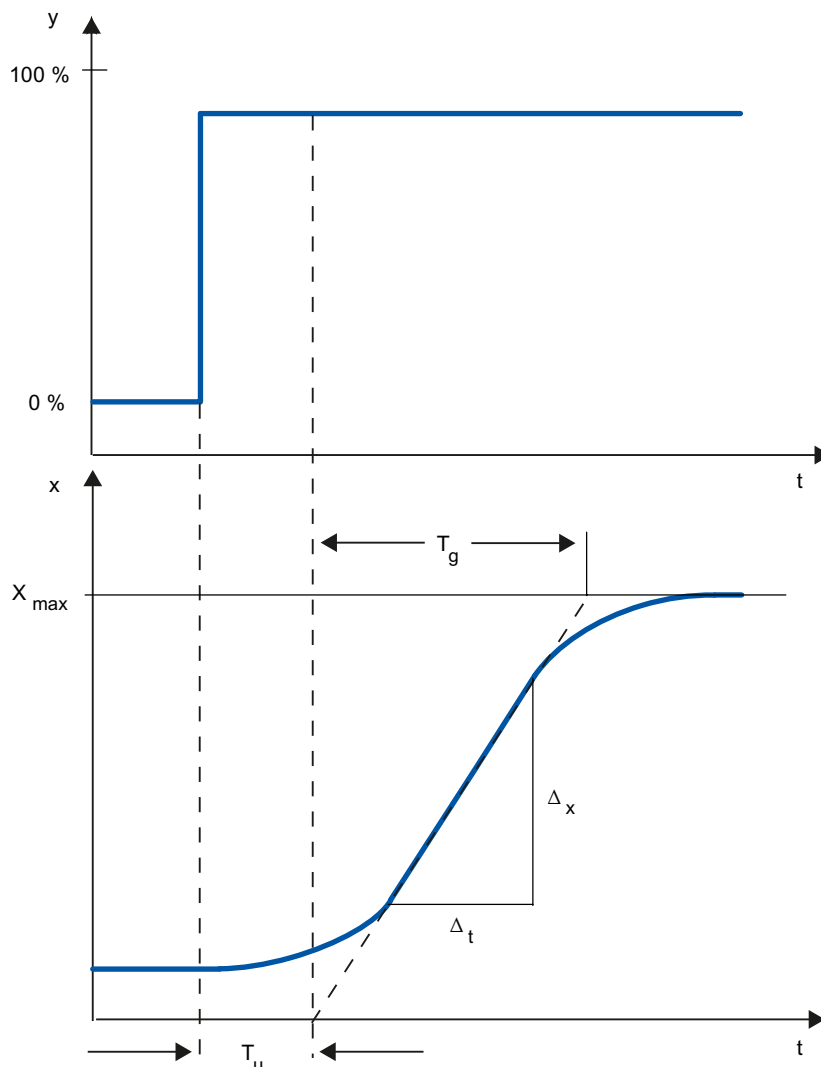
Beispiel:

Förderband

### 11.1.1.3 Kennwerte der Regelstrecke

#### Ermitteln des Zeitverhaltens aus der Sprungantwort

Das Zeitverhalten der Regelstrecke kann durch den zeitlichen Verlauf des Istwerts  $x$  nach sprunghafter Änderung des Ausgangswerts  $y$  ermittelt werden. Die meisten Regelstrecken sind Regelstrecken mit Ausgleich.



Das Zeitverhalten lässt sich näherungsweise durch die Größen Verzugszeit  $T_u$ , Ausgleichszeit  $T_g$  und Maximalwert  $X_{\max}$  festlegen. Die Größen werden durch Anlegen von Tangenten an den Maximalwert und an den Wendepunkt der Sprungantwort ermittelt. Die Aufnahme der

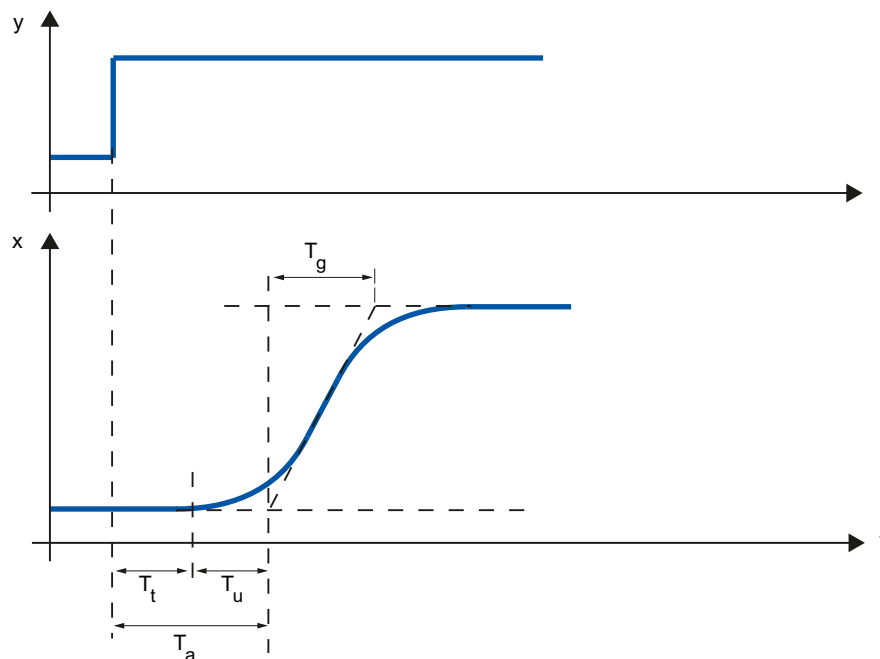
Übergangsfunktion bis zum Maximalwert ist in vielen Fällen nicht möglich, weil der Istwert bestimmte Werte nicht überschreiten darf. Dann wird die Anstiegsgeschwindigkeit  $v_{\max}$  zur Identifikation der Regelstrecke verwendet ( $v_{\max} = \Delta_x / \Delta_t$ ).

Die Regelbarkeit der Regelstrecke lässt sich abschätzen aus dem Verhältnis  $T_u / T_g$  bzw.  $T_u \times v_{\max} / X_{\max}$ . Es gilt:

| Streckentyp | $T_u / T_g$ | Regelbarkeit der Regelstrecke |
|-------------|-------------|-------------------------------|
| I           | < 0,1       | gut regelbar                  |
| II          | 0,1 bis 0,3 | noch regelbar                 |
| III         | > 0,3       | schwer regelbar               |

### Einfluss der Totzeit auf die Regelbarkeit einer Regelstrecke

Eine Regelstrecke mit Totzeit und Ausgleich zeigt folgendes Verhalten auf einen Sprung des Ausgangswerts.



- $T_t$  Totzeit
- $T_u$  Verzugszeit
- $T_g$  Ausgleichszeit
- $y$  Ausgangswert
- $x$  Istwert

Die Regelbarkeit einer Regelstrecke mit Totzeit und Ausgleich wird durch das Verhältnis von  $T_t$  zu  $T_g$  bestimmt.  $T_t$  muss klein sein gegenüber  $T_g$ . Es muss gelten:

$$T_t / T_g \leq 1$$

### Reaktionsgeschwindigkeit von Regelstrecken

Nach folgenden Werten lassen sich Regelstrecken beurteilen:

$T_u < 0,5 \text{ min}$ ,  $T_g < 5 \text{ min}$  = schnelle Regelstrecke

$T_u > 0,5 \text{ min}$ ,  $T_g > 5 \text{ min}$  = langsame Regelstrecke

### Kennwerte einiger Regelstrecken

| Physikalische Größe    | Regelstrecke                           | Verzugszeit $T_u$ | Ausgleichszeit $T_g$ | Anstiegsgeschwindigkeit $v_{\max}$ |
|------------------------|--|-------------------|----------------------|------------------------------------|
| Temperatur             | kleiner, elektrisch beheizter Ofen     | 0,5 bis 1 min     | 5 bis 15 min         | bis 60 K/min                       |
|                        | großer, elektrisch beheizter Glühofen  | 1 bis 5 min       | 10 bis 20 min        | bis 20 K/min                       |
|                        | großer, gasbeheizter Glühofen          | 0,2 bis 5 min     | 3 bis 60 min         | 1 bis 30 K/min                     |
|                        | Destillationskolonne                   | 1 bis 7 min       | 40 bis 60 min        | 0,1 bis 0,5 °C/s                   |
|                        | Autoklaven (2,5 m³)                    | 0,5 bis 0,7 min   | 10 bis 20 min        | keine Angabe                       |
|                        | Hochdruck-Autoklaven                   | 12 bis 15 min     | 200 bis 300 min      | keine Angabe                       |
|                        | Dampfüberhitzer                        | 30 s bis 2,5 min  | 1 bis 4 min          | 2 °C/s                             |
|                        | Spritzgießmaschinen                    | 0,5 bis 3 min     | 3 bis 30 min         | 5 bis 20 K/min                     |
|                        | Extruder                               | 1 bis 6 min       | 5 bis 60 min         |                                    |
|                        | Verpackungsmaschinen                   | 0,5 bis 4 min     | 3 bis 40 min         | 2 bis 35 K/min                     |
| Raumbeheizung          | 1 bis 5 min                            | 10 bis 60 min     | 1 °C/min             |                                    |
| Durchfluss             | Rohrleitung mit Gas                    | 0 bis 5 s         | 0,2 bis 10 s         | nicht relevant                     |
|                        | Rohrleitung mit Flüssigkeit            | keine             | keine                |                                    |
| Druck                  | Gasrohrleitung                         | keine             | 0,1 s                | nicht relevant                     |
|                        | Trommelkessel mit Gas- oder Ölfeuerung | keine             | 150 s                | nicht relevant                     |
|                        | Trommelkessel mit Schlägermühlen       | 1 bis 2 min       | 2 bis 5 min          | nicht relevant                     |
| Behälterstand (Niveau) | Trommelkessel                          | 0,6 bis 1 min     | keine Angabe         | 0,1 bis 0,3 cm/s                   |
| Drehzahl               | Kleiner elektrischer Antrieb           | keine             | 0,2 bis 10 s         | nicht relevant                     |
|                        | Großer elektrischer Antrieb            | keine             | 5 bis 40 s           | nicht relevant                     |
|                        | Dampfturbine                           | keine             | keine Angabe         | 50 min <sup>-1</sup>               |
| Elektrische Spannung   | Kleine Generatoren                     | keine             | 1 bis 5 s            | nicht relevant                     |
|                        | Große Generatoren                      | keine             | 5 bis 10 s           | nicht relevant                     |

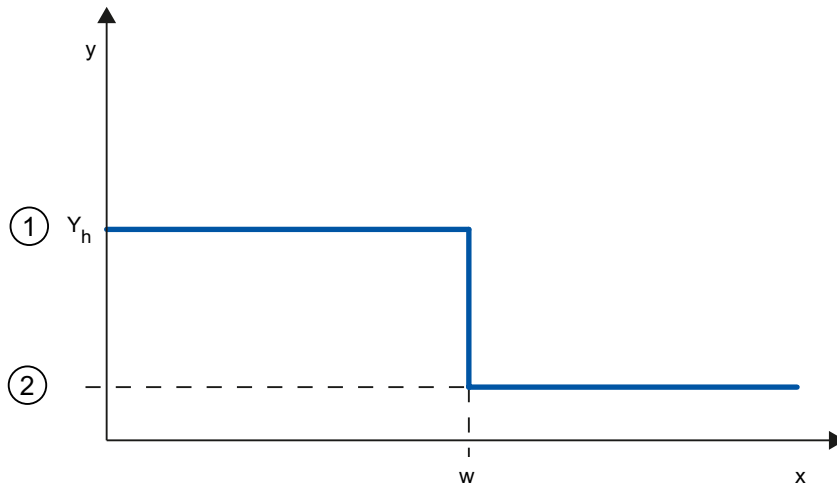
#### 11.1.1.4 Impulsregler

#### Zweipunktregler ohne Rückführung

Zweipunktregler haben als Schaltfunktion den Zustand "EIN" und "AUS". Dies entspricht 100 % bzw. 0 % Leistung. Durch dieses Verhalten tritt eine Dauerschwingung des Istwerts  $x$  um den Sollwert  $w$  auf.

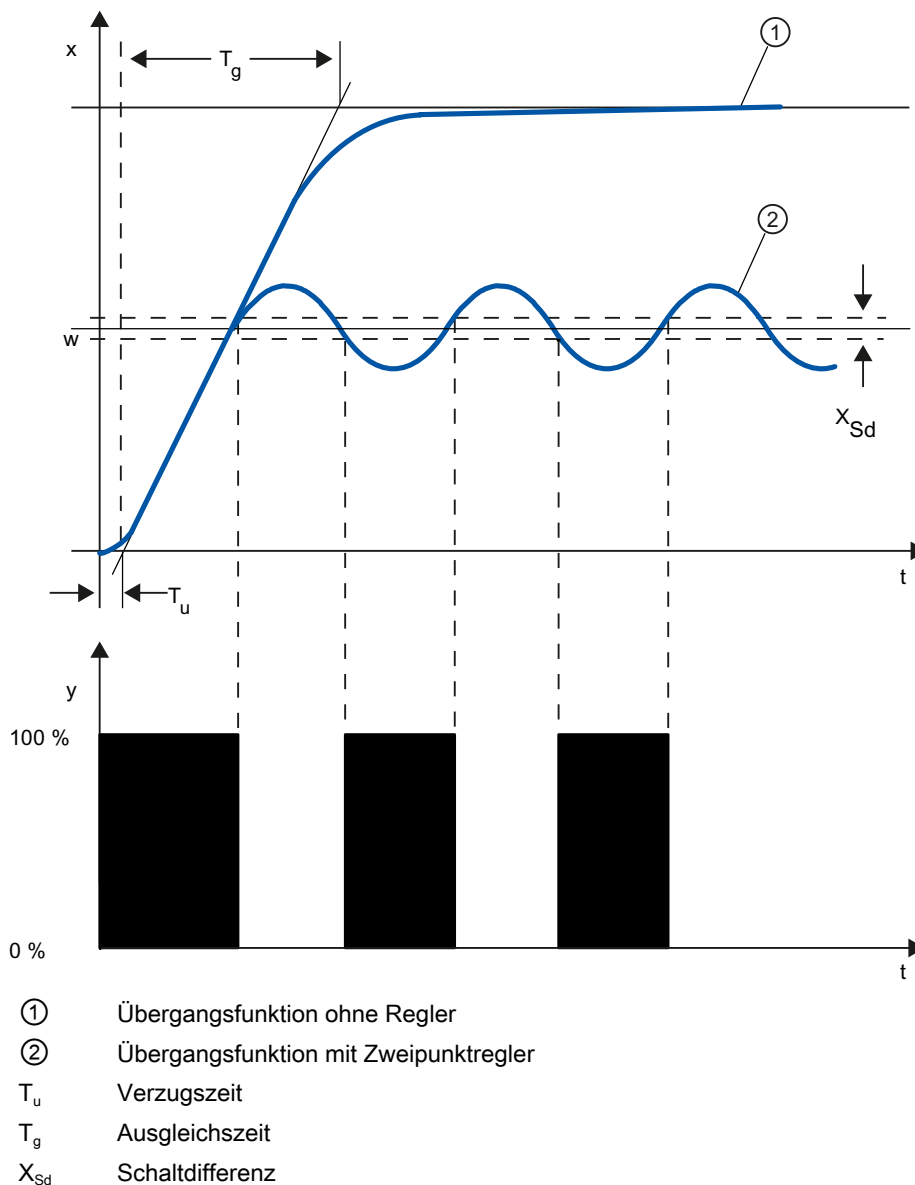
Amplitude und Periodendauer der Schwingung wachsen mit dem Verhältnis von Verzugszeit  $T_u$  zur Ausgleichzeit  $T_g$  der Regelstrecke. Diese Regler werden hauptsächlich für einfache Temperaturregelungen (z. B. für elektrisch direktbeheizte Öfen) oder als Grenzwertmelder eingesetzt.

Die folgende Grafik zeigt die Kennlinie eines Zweipunktreglers



- ① EIN
- ② AUS
- $Y_h$  Stellbereich
- $w$  Sollwert

Die folgende Grafik zeigt die Regelfunktion eines Zweipunktreglers



### Zweipunktregler mit Rückführung

Das Verhalten von Zweipunktreglern bei Regelstrecken mit größeren Verzugszeiten, z. B. Öfen, bei denen der Nutzraum von der Heizung getrennt ist, lässt sich durch elektronische Rückführungen verbessern.

Durch die Rückführung wird die Schaltfrequenz des Reglers erhöht, wodurch sich die Amplitude des Istwerts verkleinert. Außerdem können die Regelergebnisse im dynamischen Betrieb wesentlich verbessert werden. Die Grenze für die Schaltfrequenz ist durch die Ausgangsstufe gesetzt. Sie sollte bei mechanischen Stellgliedern wie Relais und Schützen 1 bis 5 Schaltungen pro Minute nicht überschreiten. Bei binären Spannungs- und Stromausgängen mit nachgeschalteten Thyristor- oder Triacstellern können hohe Schaltfrequenzen gewählt werden, die weit über der Grenzfrequenz der Regelstrecke liegen.

Da die Schaltimpulse am Ausgang der Regelstrecke nicht mehr feststellbar sind, erhält man vergleichbare Ergebnisse wie mit kontinuierlichen Reglern.

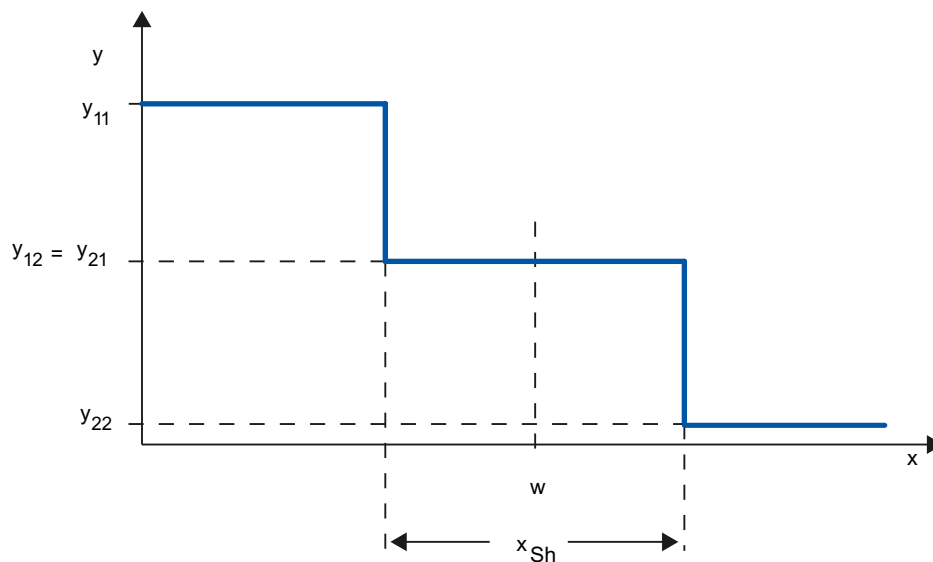
Der Ausgangswert wird durch Pulsweitenmodulation des Ausgangswerts eines kontinuierlichen Reglers erzeugt

Zweipunktregler mit Rückführung werden zur Temperaturregelung in Öfen, an Verarbeitungsmaschinen der Kunststoff-, Textil-, Papier-, Gummi- und Lebensmittelindustrie sowie für Heiz- und Kühlgeräte eingesetzt.

## Dreipunktregler

Dreipunktregler werden für Heizen/Kühlen eingesetzt. Diese Regler haben als Ausgang zwei Schaltpunkte. Durch elektronische Rückführstrukturen werden die Regelergebnisse optimiert. Einsatzgebiete für derartige Regler sind Wärme-, Kälte-, Klimakammern und Werkzeugbeheizungen für kunststoffverarbeitende Maschinen.

Die folgende Grafik zeigt die Kennlinie eines Dreipunktreglers

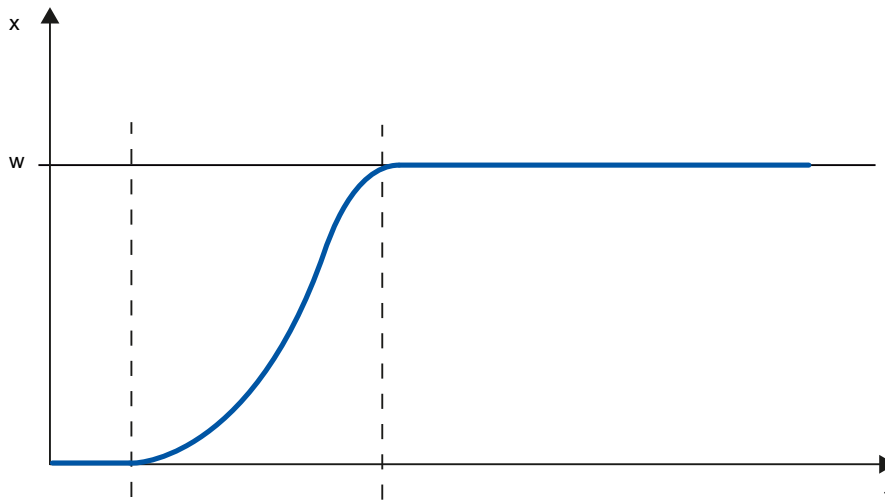


|          |   |
|----------|---|
| y        | Ausgangswert, z. B.<br>y11 = 100 % Heizung<br>y12 = 0 % Heizung<br>y21 = 0 % Kühlung<br>y22 = 100 % Kühlung |
| x        | Physikalische Größe des Istwerts, z. B. Temperatur in °C  |
| w        | Sollwert  |
| $x_{Sh}$ | Abstand zwischen Schaltpunkt 1 und Schaltpunkt 2  |

### 11.1.1.5 Führungs- und Störverhalten

#### Führungsverhalten

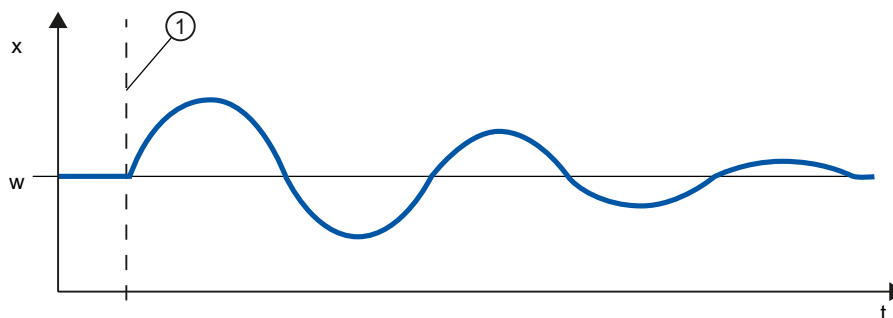
Der Istwert soll einer Änderung des Sollwerts möglichst schnell folgen. Das Führungsverhalten ist umso besser, je kürzer die Zeit, in der der neue Sollwert erreicht wird und je geringer die Schwankung des Istwerts ist.



|   |          |
|---|----------|
| x | Istwert  |
| w | Sollwert |

#### Störverhalten

Der Sollwert wird durch Störgrößen beeinflusst. Der Regler muss die dadurch auftretende Regeldifferenz möglichst schnell beseitigen. Das Störverhalten ist umso besser, je kürzer die Zeit, in der der Sollwert wieder erreicht wird und je geringer die Schwankung des Istwerts.



|   |                           |
|---|---------------------------|
| x | Istwert                   |
| w | Sollwert                  |
| ① | Einwirken einer Störgröße |



Eine Störgröße wird durch Regler mit I-Anteil ausgeglichen. Die Qualität der Regelung wird durch eine konstant wirkende Störgröße nicht verringert, da die Regeldifferenz relativ konstant ist. Eine dynamische Störgröße beeinflusst die Qualität der Regelung stärker, da die Regeldifferenz schwankt. Erst über den sich langsam aufbauenden I-Anteil wird die Regeldifferenz wieder abgebaut.

Eine messbare Störgröße kann in den Regelkreis einfließen. Die Reaktion des Reglers kann dadurch erheblich beschleunigt werden.

#### 11.1.1.6 Regelverhalten bei unterschiedlichen Rückführstrukturen

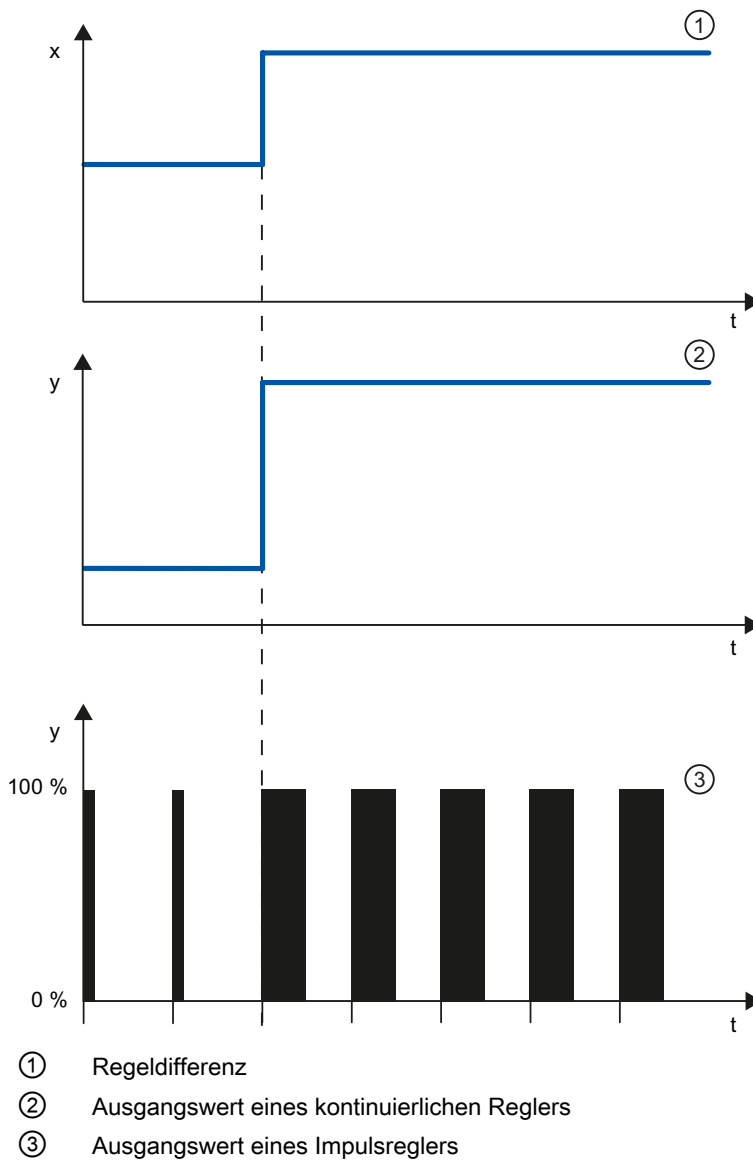
##### Regler-Verhaltensweisen

Je genauer der Regler an das Zeitverhalten der Regelstrecke angepasst ist, desto genauer stellt er den Sollwert ein und reagiert optimal auf Störgrößen.

Die Rückführschaltung kann Proportional-Verhalten (P), Proportional-Differential-Verhalten (PD), Proportional-Integral-Verhalten (PI) oder Proportional-Integral-Differential-Verhalten (PID) haben.

Wenn eine Sprungfunktion auf die Regeldifferenz gegeben wird, entstehen je nach Regler unterschiedliche Sprungantworten.

### Sprungantwort eines P-Reglers



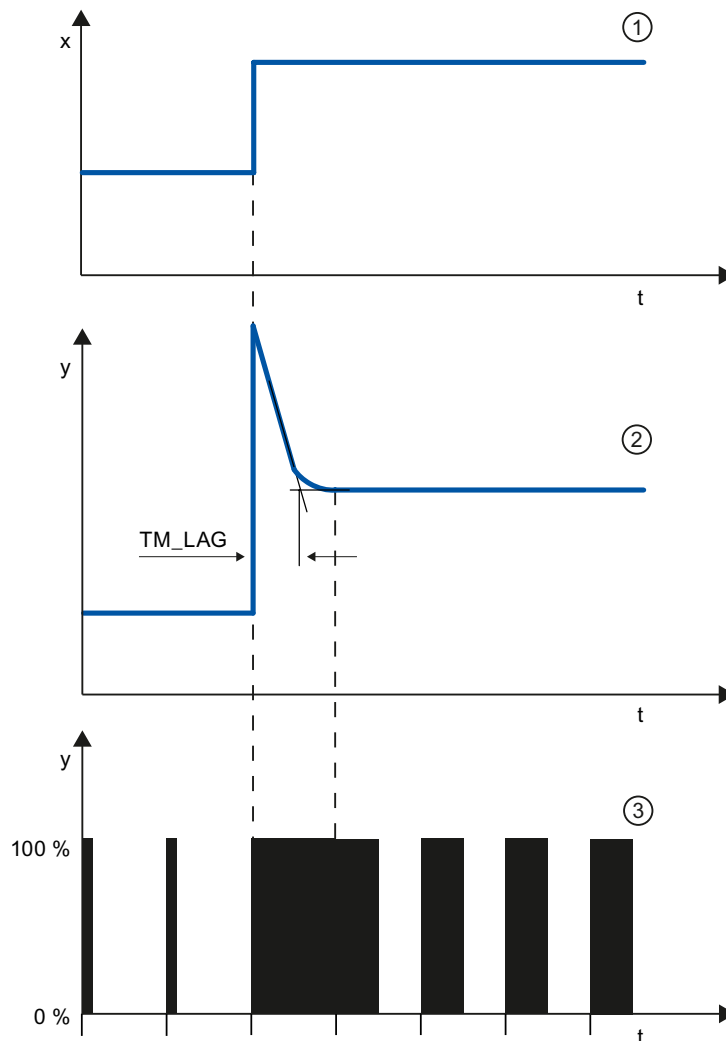
#### Formel für P-Regler

Ausgangswert und Regeldifferenz sind direkt proportional, d. h.:

Ausgangswert = Proportionalbeiwert × Regeldifferenz

$$y = \text{GAIN} \times x$$

### Sprungantwort eines PD-Reglers



- ① Regeldifferenz
- ② Ausgangswert eines kontinuierlichen Reglers
- ③ Ausgangswert eines Impulsreglers
- TM\_LAG Verzögerung des D-Anteils

#### Formel für PD-Regler

Für die Sprungantwort des PD-Reglers im Zeitbereich gilt:

$$y = \text{GAIN} \cdot X_W \cdot \left( 1 + \frac{\text{TD}}{\text{TM\_LAG}} \cdot e^{-\frac{t}{\text{TM\_LAG}}} \right)$$

t = Zeitdauer seit dem Sprung der Regeldifferenz

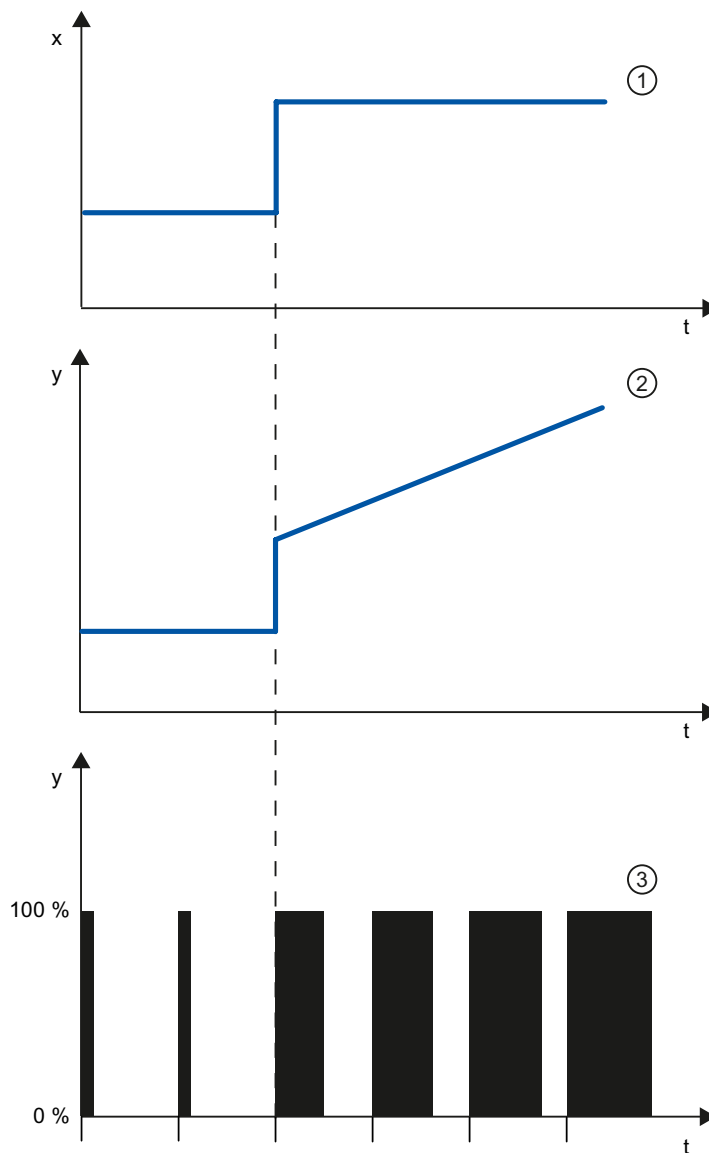
Der D-Anteil erzeugt einen Ausgangswert in Abhängigkeit von der Geschwindigkeit, mit der sich der Istwert ändert. Ein reiner D-Anteil ist nicht zur Regelung geeignet, da nur eine

Änderung des Istwerts eine Änderung des Ausgangswerts bewirkt. Wenn der Istwert konstant bleibt, ändert sich auch der Ausgangswert nicht mehr.

In Verbindung mit einem P-Anteil verbessert der D-Anteil das Störverhalten. Störungen werden nicht vollkommen ausgeregelt. Vorteilhaft ist das gute dynamische Verhalten. Beim Anfahren und bei der Sollwertänderung wird ein gut gedämpfter, schwingungsfreier Übergang erreicht.

Ein Regler mit D-Anteil ist nicht angebracht, wenn eine Regelstrecke pulsierende Messgrößen hat, z. B. bei Druck- oder Durchflussregelungen.

### Sprungantwort eines PI-Reglers



- ① Regeldifferenz
- ② Ausgangswert eines kontinuierlichen Reglers
- ③ Ausgangswert eines Impulsreglers

Ein I-Anteil im Regler summiert die Regeldifferenz über die Zeit. Dadurch wird der Regler so lange nachstellen, bis die Regeldifferenz aufgehoben ist. Bei einem reinen P-Regler tritt eine dauerhafte Regeldifferenz auf. Diese kann durch einen I-Anteil im Regler behoben werden.

In der Praxis ist, je nach Anforderung an das Regelverhalten, eine Kombination aus P-Anteil, I-Anteil und D-Anteil ideal. Das Zeitverhalten der einzelnen Anteile lässt sich durch die Reglerparameter Proportionalbeiwert GAIN, Nachstellzeit TI (I-Anteil) und Vorhaltezeit TD (D-Anteil) beschreiben.

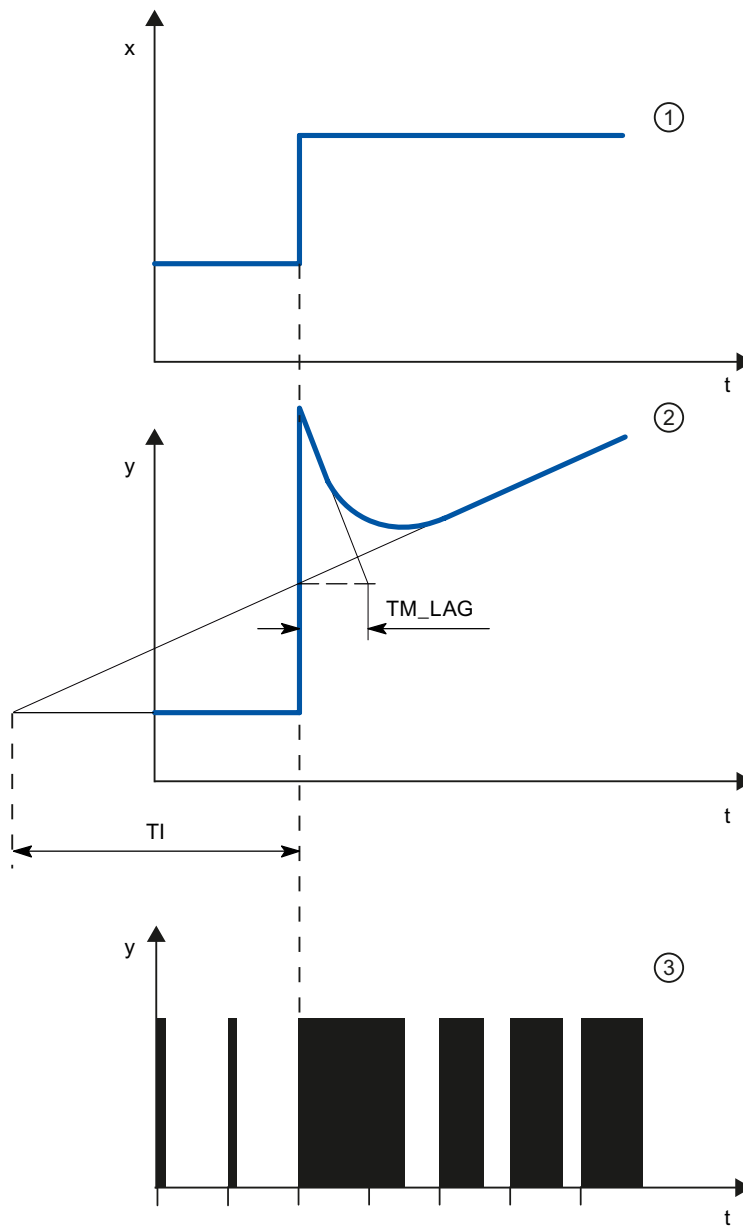
#### Formel für PI-Regler

Für die Sprungantwort des PI-Reglers im Zeitbereich gilt:

$$y = \text{GAIN} \cdot X_W \cdot \left( 1 + \frac{1}{\text{TI} \cdot t} \right)$$

t = Zeitdauer seit dem Sprung der Regeldifferenz

Sprungantwort eines PID-Reglers



- ① Regeldifferenz
- ② Ausgangswert eines kontinuierlichen Reglers
- ③ Ausgangswert eines Impulsreglers
- TM\_LAG Verzögerung des D-Anteils
- T<sub>i</sub> Integrationszeit

Formel für PID-Regler

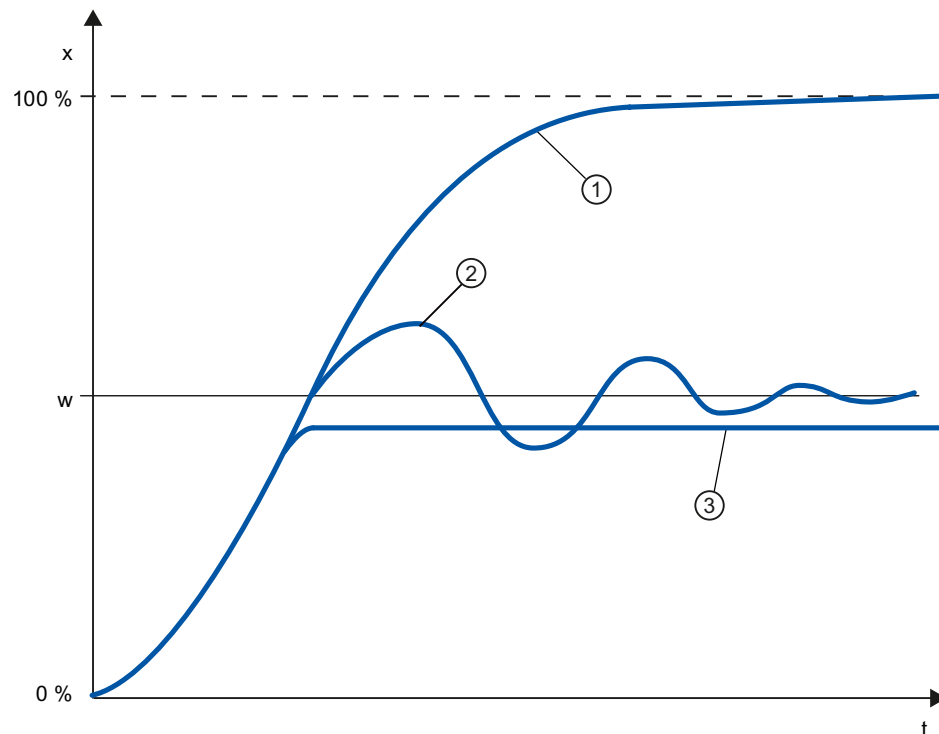
Für die Sprungantwort des PID-Reglers im Zeitbereich gilt:

$$y = \text{GAIN} \cdot X_w \cdot \left( 1 + \frac{1}{\text{TI} \cdot t} + \frac{\text{TD}}{\text{TM\_LAG}} \cdot e^{-\frac{t}{\text{TM\_LAG}}} \right)$$

t = Zeitdauer seit dem Sprung der Regeldifferenz

### Verhalten einer Regelstrecke bei verschiedenen Reglerstrukturen

Die meisten in der Verfahrenstechnik vorkommenden Regelungen lassen sich mit einem Regler mit PI-Verhalten beherrschen. Bei langsamen Regelstrecken mit großer Verzugszeit, z. B. Temperaturregelungen, lässt sich das Regelergebnis durch einen Regler mit PID-Verhalten verbessern.



- |   |             |
|---|-------------|
| ① | Kein Regler |
| ② | PID-Regler  |
| ③ | PD-Regler   |
| w | Sollwert    |
| x | Istwert     |




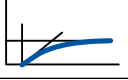
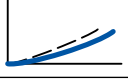
Regler mit PI- und PID-Verhalten haben den Vorteil, dass nach dem Einschwingen der Istwert keine Abweichung zum Sollwert zeigt. Der Istwert schwingt beim Anfahren über den Sollwert.

### 11.1.1.7 Wahl der Reglerstruktur bei gegebener Regelstrecke

#### Auswahl der geeigneten Reglerstrukturen

Sie erhalten nur dann ein optimales Regelergebnis, wenn Sie eine Reglerstruktur wählen, die zur Regelstrecke passt und sich innerhalb bestimmter Grenzen an die Regelstrecke anpassen lässt.

Die folgende Tabelle gibt einen Überblick, welche Reglerstruktur für welche Regelstrecke geeignet ist.

| Regelstrecke  |                 | Reglerstruktur |                  |                  |               |
|---|-----------------|----------------|------------------|------------------|---------------|
|   |                 | P              | PD               | PI               | PID           |
|    | nur mit Totzeit | ungeeignet     | ungeeignet       | geeignet         | ungeeignet    |
|    | PT1 mit Totzeit | ungeeignet     | ungeeignet       | gut geeignet     | gut geeignet  |
|   | PT2 mit Totzeit | ungeeignet     | bedingt geeignet | gut geeignet     | gut geeignet  |
|  | höhere Ordnung  | ungeeignet     | ungeeignet       | bedingt geeignet | gut geeignet  |
|  | ohne Ausgleich  | gut geeignet   | gut geeignet     | gut geeignet     | gute geeignet |

Die folgende Tabelle gibt einen Überblick, welche Reglerstruktur für welche physikalische Größe geeignet ist.

| Physikalische Größe | Reglerstruktur   |              |   |                   |
|---------------------|--|--------------|---|-------------------|
|                     | P  | PD           | PI  | PID               |
|                     | bleibende Regeldifferenz                                     |              | keine bleibende Regeldifferenz  |                   |
| Temperatur          | für geringe Ansprüche und bei P-Strecken mit $T_d/T_g < 0,1$ | gut geeignet | für gehobene Ansprüche die am besten geeigneten Reglerstrukturen (von besonders angepassten Spezialreglern abgesehen) |                   |
| Druck               | geeignet, wenn keine nennenswerte Verzugszeit                | ungeeignet   | für gehobene Ansprüche die am besten geeigneten Reglerstrukturen (von besonders angepassten Spezialreglern abgesehen) |                   |
| Durchfluss          | ungeeignet, weil erforderlicher GAIN-Bereich meist zu groß   | ungeeignet   | brauchbar, aber I-Regler allein oft besser  | kaum erforderlich |



### 11.1.1.8 Einstellung der PID-Parameter

#### Faustformel für die Parametereinstellung

| Reglerstruktur | Einstellung   |
|----------------|---|
| P              | $GAIN \approx v_{max} \times T_u [^\circ C]$  |
| PI             | $GAIN \approx 1,2 \times v_{max} \times T_u [^\circ C]$   |
| PD             | $GAIN \approx 0,83 \times v_{max} \times T_u [^\circ C]$<br>$TD \approx 0,25 \times v_{max} \times T_u [min]$<br>$TM\_LAG \approx 0,5 \times TD [min]$                    |
| PID            | $GAIN \approx 0,83 \times v_{max} \times T_u [^\circ C]$<br>$TI \approx 2 \times T_u [min]$<br>$TD \approx 0,4 \times T_u [min]$<br>$TM\_LAG \approx 0,5 \times TD [min]$ |
| PD/PID         | $GAIN \approx 0,4 \times v_{max} \times T_u [^\circ C]$<br>$TI \approx 2 \times T_u [min]$<br>$TD \approx 0,4 \times T_u [min]$<br>$TM\_LAG \approx 0,5 \times TD [min]$  |

Statt  $v_{max} = \Delta_x / \Delta_t$  kann  $X_{max} / T_g$  eingesetzt werden.

Bei Reglern mit PID-Struktur ist die Einstellung von Nachstellzeit und Vorhaltezeit meist miteinander gekoppelt.

Das Verhältnis  $TI / TD$  liegt zwischen 4 und 5 und ist für die meisten Regelstrecken optimal.

Das Nichteinhalten der Vorhaltezeit  $TD$  ist bei PD-Reglern unkritisch.

Bei PI- oder PID-Reglern treten Regelschwingungen auf, wenn die Nachstellzeit  $TI$  um mehr als die Hälfte zu klein gewählt wurde.

Eine zu große Nachstellzeit verlangsamt das Ausregeln von Störungen. Man kann nicht erwarten, dass die Regelkreise nach den ersten Parametereinstellungen "optimal" arbeiten. Erfahrungsgemäß ist ein Nachjustieren immer dann erforderlich, wenn eine "schwer regelbare" Strecke mit  $T_u / T_g > 0,3$  vorliegt.

## 11.1.2 Projektieren eines Software-Reglers

### 11.1.2.1 Übersicht der Software-Regler

Für die Projektierung eines Software-Reglers benötigen Sie eine Anweisung mit dem Regelalgorithmus und ein Technologieobjekt. Das Technologieobjekt für einen Software-Regler entspricht dem Instanz-DB der Anweisung. Im Technologieobjekt wird die Konfiguration des Reglers gespeichert. Im Unterschied zu den Instanz-DBs anderer Anweisungen, werden Technologieobjekte nicht bei den Programmressourcen abgelegt, sondern unter `CPU > Technologieobjekte`.

**Technologieobjekte und Anweisungen**

| CPU                     | Bibliothek   | Anweisung        | Technologieobjekt                                      | Beschreibung   |
|-------------------------|--|------------------|--|--|
| S7-1200                 | Compact PID  | PID_Compact V1.X | PID_Compact V1.X                                       | Universeller PID-Regler mit integrierter Optimierung |
| S7-1200                 |  | PID_3Step V1.X   | PID_3Step V1.X   | PID-Regler mit integrierter Optimierung für Ventile  |
| S7-1500<br>S7-1200 V4.x |  | PID_Compact V2.X | PID_Compact V2.X                                       | Universeller PID-Regler mit integrierter Optimierung |
| S7-1500<br>S7-1200 V4.x |  | PID_3Step V2.X   | PID_3Step V2.X   | PID-Regler mit integrierter Optimierung für Ventile  |
| S7-1500/300/400         | PID Basisfunktionen                                  | CONT_C           | CONT_C   | Kontinuierlicher Regler                              |
| S7-1500/300/400         |  | CONT_S           | CONT_S   | Schrittregler für integrierende Stellglieder         |
| S7-1500/300/400         |  | PULSEGEN         | -  | Pulsgenerator für proportional wirkende Stellglieder |
| S7-1500/300/400         |  | TCONT_CP         | TCONT_CP   | Kontinuierlicher Temperaturregler mit Pulsgenerator  |
| S7-1500/300/400         |  | TCONT_S          | TCONT_S  | Temperaturregler für integrierende Stellglieder      |
| S7-300/400              | PID Self-Tuner                                       | TUN_EC           | TUN_EC   | Optimierung eines kontinuierlichen Reglers           |
| S7-300/400              |  | TUN_ES           | TUN_ES   | Optimierung eines Schrittreglers                     |
| S7-300/400              | Standard PID Control (Optionspaket PID Professional) | PID_CP           | PID_CP   | Kontinuierlicher Regler mit Pulsegenerator           |
| S7-300/400              |  | PID_ES           | PID_ES   | Schrittregler für integrierende Stellglieder         |
| S7-300/400              |  | LP_SCHED         | -  | Regleraufrufe verteilen                              |
| S7-300/400              | Modular PID Control (Optionspaket PID Professional)  | A_DEAD_B         | -  | Störsignale aus Regeldifferenz filtern               |
| S7-300/400              |  | CRP_IN           | -  | Analoges Eingangssignal skalieren                    |
| S7-300/400              |  | CRP_OUT          | -  | Analoges Ausgangssignal skalieren                    |
| S7-300/400              |  | DEAD_T           | -  | Eingangssignal verzögert ausgeben                    |
| S7-300/400              |  | DEADBAND         | -  | Kleine Schwankungen des Istwerts unterdrücken        |
| S7-300/400              |  | DIF              | -  | Eingangssignal über die Zeit differenzieren          |
| S7-300/400              |  | ERR_MON          | -  | Regeldifferenz überwachen                            |
| S7-300/400              |  | INTEG            | -  | Eingangssignal über die Zeit integrieren             |
| S7-300/400              |  | LAG1ST           | -  | Verzögerungsglied 1. Ordnung                         |
| S7-300/400              |  | LAG2ND           | -  | Verzögerungsglied 2. Ordnung                         |
| S7-300/400              |  | LIMALARM         | -  | Grenzwerte melden                                    |
| S7-300/400              |  | LIMITER          | -  | Stellwert begrenzen                                  |
| S7-300/400              |  | LMNGEN_C         | -  | Stellwert für kontinuierlichen Regler ermitteln      |
| S7-300/400              |  | LMNGEN_S         | -  | Stellwert für Schrittregler ermitteln                |
| S7-300/400              |  | NONLIN           | -  | Gebersignal linearisieren                            |
| S7-300/400              |  | NORM             | -  | Istwert physikalisch normieren                       |
| S7-300/400              | OVERRIDE   | -                | Stellwerte von 2 PID-Reglern auf 1 Stellglied schalten |  |

| CPU        | Bibliothek | Anweisung  | Technologieobjekt | Beschreibung  |
|------------|------------|------------|-------------------|---|
| S7-300/400 |            | PARA_CTL   | -                 | Parametersätze umschalten                               |
| S7-300/400 |            | PID        | -                 | PID Algorithmus   |
| S7-300/400 |            | PUSLEGEN_M | -                 | Pulse für proportional wirkende Stellglieder generieren |
| S7-300/400 |            | RMP_SOAK   | -                 | Sollwerte nach Zeitplan vorgeben                        |
| S7-300/400 |            | ROC_LIM    | -                 | Änderungsgeschwindigkeit begrenzen                      |
| S7-300/400 |            | SCALE_M    | -                 | Istwert skalieren                                       |
| S7-300/400 |            | SP_GEN     | -                 | Sollwert manuell vorgeben                               |
| S7-300/400 |            | SPLT_RAN   | -                 | Stellwertbereiche aufteilen                             |
| S7-300/400 |            | SWITCH     | -                 | Analogwerte schalten                                    |
| S7-300/400 |            | LP_SCHED_M | -                 | Regleraufrufe verteilen                                 |

### 11.1.2.2 Schritte für das Projektieren eines Software-Reglers

Alle SW-Regler projektieren Sie nach demselben Schema:

| Schritt | Beschreibung   |
|---------|--|
| 1       | Technologieobjekt hinzufügen (Seite 4053)                  |
| 2       | Technologieobjekt konfigurieren (Seite 4054)               |
| 3       | Anweisung im Anwenderprogramm aufrufen (Seite 4055)        |
| 4       | Technologieobjekt in Gerät laden (Seite 4056)              |
| 5       | Software-Regler in Betrieb nehmen (Seite 4058)             |
| 6       | Optimierte PID-Parameter im Projekt speichern (Seite 4058) |
| 7       | Werte vergleichen (Seite 4059)                             |
| 8       | Instanzen eines Technologieobjektes anzeigen (Seite 4061)  |

### 11.1.2.3 Technologieobjekte hinzufügen

#### Technologieobjekt im Projektnavigator hinzufügen

Beim Hinzufügen eines Technologieobjekts wird ein Instanz-DB der Anweisung zu diesem Technologieobjekt erzeugt. In diesem Instanz-DB wird die Konfiguration des Technologieobjekts hinterlegt.

#### Voraussetzung

Ein Projekt mit einer CPU ist angelegt.

## Vorgehen

Um ein Technologieobjekt hinzuzufügen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Doppelklicken Sie "Neues Objekt hinzufügen".  
Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Klicken Sie auf die Schaltfläche "PID-Regler".  
Alle verfügbaren PID-Regler für diese CPU werden angezeigt.
5. Wählen Sie die Anweisung für das Technologieobjekt, z. B. PID\_Compact.
6. Geben Sie im Eingabefeld "Name" einen individuellen Namen für das Technologieobjekt ein.
7. Wählen Sie die Option "manuell", falls Sie die vorgeschlagene Datenbausteinnummer des Instanz-DB ändern möchten.
8. Klicken Sie auf "Weitere Informationen", wenn Sie eigene Informationen zum Technologieobjekt hinterlegen möchten.
9. Bestätigen Sie mit "OK".

## Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt. Das Technologieobjekt wird verwendet, wenn die Anweisung zu diesem Technologieobjekt in einem Weckalarm-OB aufgerufen wird.

---

### Hinweis

Sie können im unteren Bereich des Dialogfelds das Optionskästchen "Neu hinzufügen und öffnen" aktivieren. Dadurch wird die Konfiguration des Technologieobjekts nach dem Hinzufügen geöffnet.

---

### 11.1.2.4 Technologieobjekte konfigurieren

Die Eigenschaften eines Technologieobjekts auf einer S7-1200 CPU können Sie auf zwei Arten konfigurieren.

- Im Inspektorfenster des Programmiereditors
- Im Konfigurationseditor

Die Eigenschaften eines Technologieobjekts auf einer S7-300/400 CPU können Sie nur im Konfigurationseditor konfigurieren.

### Inspektorfenster des Programmiereditors

Im Inspektorfenster des Programmiereditors können Sie nur die Parameter konfigurieren, die für den Betrieb notwendig sind.

Auch im Online-Modus werden die Offline-Werte der Parameter angezeigt. Sie können die Online-Werte nur in Fenster Inbetriebnahme ändern.

Um das Inspektorfenster eines Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Doppelklicken Sie den Baustein (Weckalarm-OB), in welchem Sie die Anweisung des SW-Reglers aufrufen.  
Der Baustein wird im Arbeitsbereich geöffnet.
3. Klicken Sie auf die Anweisung des SW-Reglers.
4. Wählen Sie im Inspektorfenster nacheinander die Register "Eigenschaften" und "Konfiguration".

## Konfigurationsfenster

Für jedes Technologieobjekt gibt es ein spezifisches Konfigurationsfenster, in dem Sie alle Eigenschaften konfigurieren.

Um das Konfigurationsfenster eines Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie das Objekt "Konfiguration".

## Symbole

Symbole in der Bereichsnavigation des Konfigurations- und Inspektorfensters zeigen weitere Details zur Vollständigkeit der Konfiguration:

|   |  |
|---|--|
| ✓ | <b>Die Konfiguration enthält Voreinstellungswerte und ist vollständig.</b><br>Die Konfiguration enthält ausschließlich Voreinstellungswerte. Mit diesen Voreinstellungswerten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.  |
| ✔ | <b>Die Konfiguration enthält vom Anwender definierte Werte und ist vollständig</b><br>Alle Eingabefelder der Konfiguration enthalten gültige Werte, und mindestens ein Voreinstellungswert wurde geändert.   |
| ✘ | <b>Die Konfiguration ist unvollständig oder fehlerhaft</b><br>Mindestens ein Eingabefeld oder eine Klappliste enthält keinen oder einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an. |

Die Eigenschaften eines Technologieobjekts sind detailliert im Kapitel für das Technologieobjekt beschrieben.

### 11.1.2.5 Anweisung im Anwenderprogramm aufrufen

Die Anweisung des Software-Reglers muss in einem Weckalarm-OB aufgerufen werden. Die Abtastzeit des Software-Reglers wird durch den zeitlichen Abstand der Aufrufe im Weckalarm-OB bestimmt.

## Voraussetzung

Der Weckalarm-OB ist angelegt und die Zykluszeit des Weckalarm-OBs ist korrekt konfiguriert.

## Vorgehen

Um die Anweisung im Anwenderprogramm aufzurufen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Programmbausteine".
3. Doppelklicken Sie den Weckalarm-OB.  
Der Baustein wird im Arbeitsbereich geöffnet.
4. Öffnen Sie im Fenster "Anweisungen" die Gruppe "Technologie" und den Ordner "PID Control".  
Der Ordner enthält alle Anweisungen für Software-Regler, die auf der CPU projiziert werden können.
5. Wählen Sie eine Anweisung und ziehen Sie diese per Drag & Drop in Ihren Weckalarm-OB.  
Der Dialog "Aufrufoptionen" wird geöffnet.
6. Wählen Sie aus der Liste "Name" ein Technologieobjekt oder geben Sie den Namen für ein neues Technologieobjekt ein.

## Ergebnis

Wenn das Technologieobjekt noch nicht besteht, wird es hinzugefügt. Die Anweisung wird in den Weckalarm-OB eingefügt. Das Technologieobjekt ist diesem Aufruf der Anweisung zugeordnet.

### 11.1.2.6 Technologieobjekte in Gerät laden

Eine neue oder geänderte Konfiguration des Technologieobjekts muss für den Onlinebetrieb in die CPU geladen werden. Beim Laden permanenter Daten gelten folgende Besonderheiten:

- **Software (nur Änderungen)**
  - S7-1200, S7-1500:  
Remanente Daten bleiben erhalten.
  - S7-300/400:  
Remanente Daten werden sofort aktualisiert. Die CPU geht nicht in Stop.
- **PLC-Programm in Gerät laden und zurücksetzen**
  - S7-1200, S7-1500:  
Remanente Daten werden beim nächsten Übergang von Stop in RUN aktualisiert. Das PLC-Programm kann nur vollständig geladen werden.
  - S7-300/400:  
Remanente Daten werden beim nächsten Übergang von Stop in RUN aktualisiert.

---

## Remanente Daten auf eine CPU S7-1200 oder S7-1500 laden

---

### Hinweis

Das Laden und zurücksetzen des PLC-Programms bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie das PLC-Programm laden und zurücksetzen!

---

Um remanente Daten zu laden, gehen Sie folgendermaßen vor:

1. Selektieren Sie den Eintrag der CPU in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "PLC-Programm in Gerät laden und zurücksetzen".
  - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet. Stellen Sie in diesem Fall alle notwendigen Parameter für die Verbindung ein und klicken Sie auf "Laden".
  - Wurde die Online-Verbindung definiert, werden die Projektdaten übersetzt, wenn dies erforderlich ist und der Dialog "Vorschau laden" wird geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
3. Kontrollieren Sie die Meldungen.  
Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
4. Klicken Sie auf "Laden".  
Das vollständige PLC-Programm wird geladen und der Dialog "Ergebnisse laden" geöffnet. Dieser Dialog zeigt Ihnen den Status und die Aktionen nach dem Ladevorgang an.
5. Wenn die Baugruppen nach dem Laden direkt wieder gestartet werden sollen, aktivieren Sie das Optionskästchen "Alle starten".
6. Schließen Sie den Dialog "Ergebnisse laden" mit "Fertig stellen".

## Ergebnis

Das komplette PLC-Programm wird in das Gerät geladen. Bausteine, die nur online im Gerät existieren, werden gelöscht. Durch das Laden aller betroffenen Bausteine und das Löschen nicht benötigter Bausteine im Gerät werden Inkonsistenzen zwischen den Bausteinen im Anwenderprogramm vermieden.

Ob der Ladevorgang erfolgreich war, können Sie an den Meldungen im Inspektorfenster unter "Info > Allgemein" erkennen.

### 11.1.2.7 Software-Regler in Betrieb nehmen

#### Vorgehen

Um den Arbeitsbereich "Inbetriebnahme" eines Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner "Technologieobjekte".
2. Öffnen Sie in der Projektnavigation das Technologieobjekt.
3. Doppelklicken Sie das Objekt "Inbetriebnahme".

Die Funktionen zur Inbetriebnahme sind für jeden Regler spezifisch und dort beschrieben.

### 11.1.2.8 Optimierte PID-Parameter im Projekt speichern


Der Software-Regler wird in der CPU optimiert. Dadurch stimmen die Werte im Instanz-DB auf der CPU nicht mehr mit denen im Projekt überein.

Um die PID-Parameter im Projekt mit den optimierten PID-Parametern in der CPU zu aktualisieren, gehen Sie folgendermaßen vor:

#### Voraussetzung

- Eine Onlineverbindung zur CPU ist aufgebaut und die CPU befindet sich im Betriebszustand "RUN".
- Die Funktionen des Inbetriebnahmefensters sind mit der Schaltfläche "Start" freigegeben.

#### Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Öffnen Sie ein Technologieobjekt.
4. Doppelklicken Sie auf "Inbetriebnahme".
5. Klicken Sie auf das Symbol  "PID-Parameter laden".
6. Speichern Sie das Projekt.

#### Ergebnis

Die aktuell wirkenden PID-Parameter werden in den Projektdaten gesichert. Beim erneuten Laden der Projektdaten in die CPU werden die optimierten Parameter verwendet.



### 11.1.2.9 Werte vergleichen







#### Vergleichsanzeige und Randbedingungen

Die Funktion "Werte vergleichen" bietet folgende Möglichkeiten:

- Vergleich der konfigurierten Startwerte des Projekts mit den Startwerten in der CPU und den Aktualwerten
- Direktes Bearbeiten der Aktualwerte und der Startwerte des Projekts
- Sofortiges Erkennen und Anzeigen von Eingabefehlern mit Korrekturvorschlägen
- Sicherung der Aktualwerte im Projekt
- Übertragen der Startwerte des Projekts in die CPU als Aktualwerte

#### Symbole und Bedienelemente

Folgende Symbole und Bedienelemente stehen zur Verfügung:

| Symbol  | Funktion  |
|---|---|
|   | Startwert in CPU ist gleich dem konfigurierten Startwert im Projekt                                   |
|  | Startwert in CPU ist ungleich dem konfigurierten Startwert im Projekt                                 |
|  | Der Vergleich Startwert in CPU und konfiguriertem Startwert im Projekt kann nicht durchgeführt werden |
|  | Aktualwerte werden im Projekt gesichert   |
|  | Aktualisierte Startwerte im Projekt werden in die CPU übertragen (Einstellwerte initialisieren)       |
|  | Dialog "Werte vergleichen" wird geöffnet  |

#### Randbedingungen

Die Funktion "Werte vergleichen" ist uneingeschränkt verfügbar für S7-1200 und S7-1500.

Für S7-300 und S7-400 gilt folgende Einschränkung:

Im Beobachtungsmodus kann eine S7-300/S7-400 die Startwerte nicht in die CPU übertragen. Diese Werte können mit "Werte vergleichen" nicht online angezeigt werden.

Die Aktualwerte des Technologieobjekts werden angezeigt und sind direkt veränderbar.



#### Werte vergleichen

Im Folgenden wird das Vorgehen am Beispiel der "PID-Parameter" gezeigt.

## Voraussetzungen

- Ein Projekt mit einem Software-Regler ist konfiguriert.
- Das Projekt ist in die CPU geladen.
- Im Projektnavigator ist der Konfigurationsdialog geöffnet.

## Vorgehen

1. Öffnen Sie in der Projektnavigation den gewünschten Software-Regler.
2. Doppelklicken Sie auf das Objekt "Konfiguration".
3. Navigieren Sie im Konfigurationsfenster zum Dialog "PID-Parameter".
4. Klicken Sie auf das Symbol , um den Beobachtungsmodus zu aktivieren.  
Die Symbole und Bedienelemente (Seite 4059) der Funktion "Werte vergleichen" werden hinter den Parametern angezeigt.
5. Klicken Sie in das Eingabefeld des gewünschten Parameters und ändern Sie die Parameterwerte manuell durch direkte Eingabe.
  - Wenn der Hintergrund des Eingabefelds grau ist, sind die Werte nur lesbar, nicht änderbar.
  - Um im Dialog "PID-Parameter" die Werte zu ändern, aktivieren Sie vorher die manuelle Eingabe durch Anklicken des Optionskästchens "Manuelle Eingabe aktivieren".
6. Klicken Sie auf das Symbol , um den Dialog der Startwerte zu öffnen.  
Dieser Dialog zeigt zwei Werte des Parameters an:
  - Startwert in CPU: Im oberen Teil wird der Startwert in der CPU angezeigt.
  - Startwert im Projekt: Im unteren Teil wird der konfigurierte Startwert im Projekt angezeigt.
7. Tragen Sie den gewünschten Wert in das Eingabefeld für das Projekt ein.

## Fehlererkennung

Die Eingabe von fehlerhaften Werten wird erkannt. In diesem Fall werden Korrekturen angeboten.

Wenn Sie einen syntaktisch falschen Wert eingeben, wird unter dem Parameter ein Rollout mit der entsprechenden Fehlermeldung geöffnet. Der falsche Wert wird nicht übernommen.

Wenn Sie einen technologisch falschen Wert eingeben, wird ein Dialog geöffnet, in dem der Fehler gemeldet und ein Korrekturvorschlag angezeigt wird:


- Durch Klicken auf "Nein" können Sie diesen Korrekturvorschlag annehmen und Ihre Eingabe korrigieren.
- Durch Klicken auf "OK" übernehmen Sie den falschen Wert.

### ACHTUNG


#### Fehlfunktion des Regler


Technologisch falsche Werte können zu Fehlfunktionen des Reglers führen.

## Aktualwerte sichern

Durch Klicken auf das Symbol  übertragen Sie die Aktualwerte des Reglers in die Startwerte Ihres konfigurierten Projekts.

## Projektwerte in die CPU übertragen

Durch Klicken auf das Symbol  übertragen Sie die konfigurierten Werte Ihres Projekts in die CPU.

|   |
|---|
|  <b>VORSICHT</b>   |
| <b>Personen- und Sachschäden verhindern!</b>  |
| Das Laden und Zurücksetzen des Anwenderprogramms bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen! |
| Vergewissern Sie sich, dass keine gefährlichen Zustände eintreten können, bevor Sie das Anwenderprogramm laden und zurücksetzen!  |

### 11.1.2.10 Instanz-DB eines Technologieobjekts anzeigen

Zu jedem Technologieobjekt wird ein Instanz-DB angelegt, in dem die Parameter und statischen Variablen gespeichert sind.

## Vorgehen

Um den Instanz-DB eines Technologieobjekts anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner "Technologieobjekte".
3. Markieren Sie ein Technologieobjekt.
4. Wählen Sie im Kontextmenü den Befehl "Öffnen im Editor".

## 11.1.3 PID\_Compact einsetzen

### 11.1.3.1 Technologieobjekt PID\_Compact

Das Technologieobjekt "PID\_Compact" stellt einen kontinuierlichen PID-Regler mit integrierter Optimierung zur Verfügung. Alternativ können Sie einen Impulsregler konfigurieren. Es sind Hand- und Automatikbetrieb möglich.

Der PID-Regler erfasst innerhalb eines Regelkreises fortlaufend den gemessenen Istwert und vergleicht diesen mit dem gewünschten Sollwert. Aus der sich ergebenden Regeldifferenz errechnet die Anweisung PID\_Compact einen Ausgangswert, durch den der Istwert möglichst schnell und stabil an den Sollwert angeglichen wird. Der Ausgangswert setzt sich beim PID-Regler aus drei Anteilen zusammen:

## 11.1 PID-Regelung

- **P-Anteil**  
Der P-Anteil des Ausgangswerts steigt proportional zur Regeldifferenz.
- **I-Anteil**  
Der I-Anteil des Ausgangswerts steigt solange bis die Regeldifferenz ausgeglichen ist.
- **D-Anteil**  
Der D-Anteil steigt mit wachsender Änderungsgeschwindigkeit der Regeldifferenz. Der Istwert wird möglichst schnell an den Sollwert angeglichen. Nimmt die Änderungsgeschwindigkeit der Regeldifferenz wieder ab, so verringert sich der D-Anteil wieder.

Die Anweisung PID\_Compact berechnet die P-, I-, und D-Parameter für Ihre Regelstrecke selbstständig während der "Erstoptimierung". Die Parameter können über eine "Nachoptimierung" weiter optimiert werden. Sie brauchen die Parameter nicht manuell ermitteln.

### Weitere Informationen

- Übersicht der Software-Regler (Seite 4051)
- Technologieobjekte hinzufügen (Seite 4053)
- Technologieobjekte konfigurieren (Seite 4054)
- PID\_Compact V2 konfigurieren (Seite 4062)
- PID\_Compact V1 konfigurieren (Seite 4078)

### 11.1.3.2 PID\_Compact V2

#### PID\_Compact V2 konfigurieren

#### Grundeinstellungen

#### Einleitung

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID\_Compact:

- Physikalische Größe
- Regelsinn
- Anlaufverhalten nach Reset
- Sollwert (nur im Inspektorfenster)
- Istwert (nur im Inspektorfenster)
- Ausgangswert (nur im Inspektorfenster)

## Sollwert, Istwert und Ausgangswert

Sollwert, Istwert und Ausgangswert können Sie nur im Inspektorfenster des Programmiereditors konfigurieren. Für jeden Wert wählen Sie die Quelle:

- Instanz-DB  
Es wird der Wert verwendet der im Instanz-DB gespeichert ist.  
Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden.  
An der Anweisung darf kein Wert stehen.  
Änderung über HMI möglich.
- Anweisung  
Es wird der Wert verwendet, der an der Anweisung verschaltet ist.  
Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben.  
Änderung über HMI nicht möglich.

## Regelungsart

### Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Sollwert, Istwert und Störgröße. Sollwert, Istwert und Störgröße werden in dieser Einheit angezeigt.

## Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn.

PID\_Compact arbeitet nicht mit negativer Proportionalverstärkung. Um durch einen höheren Ausgangswert den Istwert zu verringern, aktivieren Sie das Optionskästchen "Invertieren des Regelsinns".

Beispiele

- Durch Öffnen eines Ablaufventils sinkt der Füllstand eines Behälters.
- Durch eine höhere Kühlleistung sinkt die Temperatur.

## Anlaufverhalten

1. Um nach Neustart der CPU in die Betriebsart "Inaktiv" zu wechseln, deaktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".  
Um nach Neustart der CPU in die Betriebsart zu wechseln, die an Mode gespeichert ist, aktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".
2. Wählen Sie in der Klappliste "Mode setzen auf" die Betriebsart aus, die nach einem vollständigen Laden in Gerät aktiviert werden soll.  
Nach einem vollständigen Laden in Gerät startet PID\_Compact in der gewählten Betriebsart. Bei jedem weiteren Neustart, startet PID\_Compact in der Betriebsart, die zuletzt an Mode gespeichert war.

Beispiel

Sie haben das Optionskästchen "Nach CPU Neustart Mode aktivieren" aktiviert und in der Liste "Mode setzen auf" den Eintrag "Erstoptimierung" gewählt. Nach einem vollständigen

Laden in Gerät startet PID\_Compact in der Betriebsart "Erstoptimierung". Wenn die Erstoptimierung noch aktiv ist, startet PID\_Compact nach dem Neustart der CPU wieder in der Betriebsart "Erstoptimierung". Wenn die Erstoptimierung erfolgreich beendet wurde und der Automatikbetrieb aktiv ist, startet PID\_Compact nach dem Neustart der CPU im "Automatikbetrieb".

## Sollwert

### Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80 °C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist.  
Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert Zeit gesteuert zu ändern.

## Istwert

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID\_Compact den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

### Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input\_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

## Ausgangswert

PID\_Compact stellt drei Ausgangswerte zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- Output\_PER  
Das Stellglied wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10V, 4...20mA.
- Output  
Der Ausgangswert muss über das Anwenderprogramm aufbereitet werden, z.B. weil das Stellglied nichtlineares Verhalten zeigt.
- Output\_PWM  
Das Stellglied wird über einen Digitalausgang gesteuert. Über eine Pulsweitenmodulation werden variable Ein- und Ausschaltzeiten gebildet.

## Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output\_PER (analog)".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output".
2. Wählen Sie "Instanz-DB".  
Der berechnete Ausgangswert wird im Instanz-DB gespeichert.
3. Verwenden Sie für die Aufbereitung des Ausgangswerts den Ausgangsparameter Output.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Digital- oder Analogausgang der CPU zum Stellglied.

Um den digitalen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output\_PWM".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Digitalausgangs ein.

## Istwerteinstellungen

### Istwert skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Input\_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input\_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input\_PER anhand eines unteren und oberen Wertepaars skaliert.

## Vorgehen

Um den Istwert zu skalieren, gehen Sie folgendermaßen vor:

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID\_Compact.
2. Verschalten Sie in den Grundeinstellungen Input\_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung".

Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

## Istwertgrenzen

Als Grenzwerte müssen Sie für Ihre Regelstrecke eine sinnvolle absolute Ober- und Untergrenze des Istwerts festlegen. Sobald diese Grenzen über- oder unterschritten werden, tritt ein Fehler auf (ErrorBits = 0001h). Die Optimierung wird abgebrochen, wenn die Istwertgrenzen überschritten werden. Wie PID\_Compact im Fehlerfall im Automatikbetrieb reagiert, konfigurieren Sie bei den Ausgangswerteinstellungen.

## Erweiterte Einstellungen

### Istwertüberwachung

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über oder unterschritten, so wird an der Anweisung PID\_Compact eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning\_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning\_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

## Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID\_Compact verhält sich folgendermaßen:



| Istwert             | InputWarnin<br>g_H | InputWarnin<br>g_L | ErrorBi<br>ts | Betriebsart   |
|---------------------|--------------------|--------------------|---------------|---|
| > 98 °C             | TRUE               | FALSE              | 0001h         | Inaktiv oder<br>Ersatzausgangswert mit<br>Fehlerüberwachung |
| ≤ 98 °C und > 90 °C | TRUE               | FALSE              | 0000h         | Automatikbetrieb  |
| ≤ 90 °C und ≥ 10 °C | FALSE              | FALSE              | 0000h         | Automatikbetrieb  |
| < 10°C und ≥ 0 °C   | FALSE              | TRUE               | 0000h         | Automatikbetrieb  |
| < 0 °C              | FALSE              | TRUE               | 0001h         | Inaktiv oder<br>Ersatzausgangswert mit<br>Fehlerüberwachung |

Wie PID\_Compact beim Über- bzw. Unterschreiten der Ober- oder Untergrenze Istwert reagiert, konfigurieren Sie bei den Ausgangswerteinstellungen.

### Siehe auch

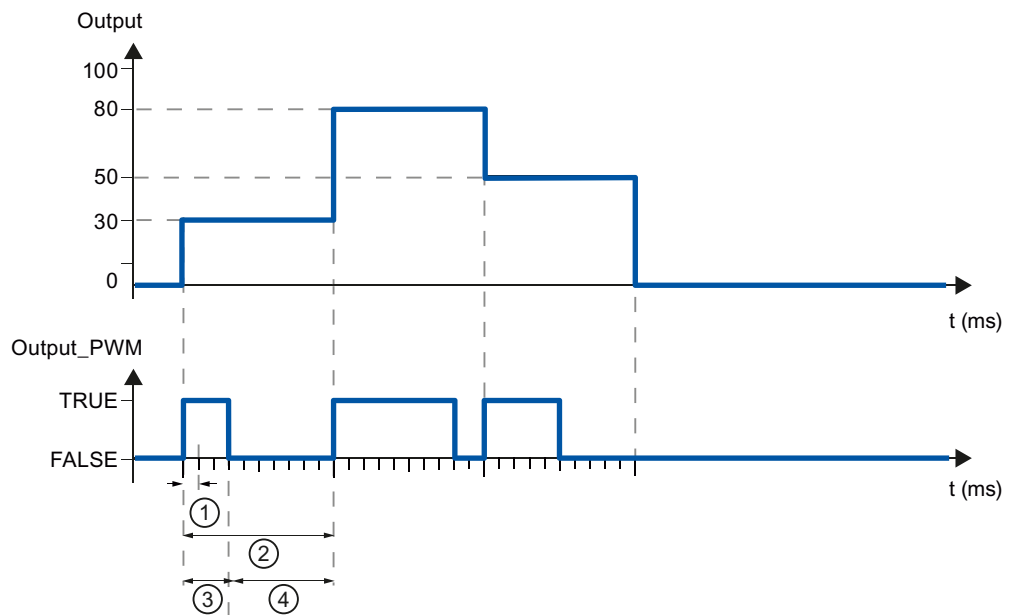
Parameter State und Mode V2 (Seite 2691)

### PWM-Begrenzungen

Über eine Pulsweitenmodulation wird der Wert am Ausgangsparameter Output in eine Impulsfolge transformiert, die am Ausgangsparameter Output\_PWM ausgegeben wird. Output wird in der Abtastzeit PID-Algorithmus berechnet, Output\_PWM wird in der Abtastzeit PID\_Compact ausgegeben.

Die Abtastzeit PID-Algorithmus wird während der Erst- oder Nachoptimierung ermittelt. Wenn Sie die PID-Parameter manuell einstellen, müssen Sie dort auch die Abtastzeit PID-Algorithmus konfigurieren. Die Abtastzeit PID\_Compact entspricht der Zykluszeit des aufrufenden OBs.

Die Impulsdauer ist proportional zum Wert an Output und beträgt stets ein ganzzahliges Vielfaches der Abtastzeit PID\_Compact.



- ① Abtastzeit PID\_Compact
- ② Abtastzeit PID-Algorithmus
- ③ Impulsdauer
- ④ Pausendauer

Die "Minimale Einschaltzeit" und die "Minimale Ausschaltzeit" werden auf ein ganzzahliges Vielfaches der Abtastzeit PID\_Compact gerundet.

Ein Impuls oder eine Pause wird nie kürzer als die minimale Ein- oder Ausschaltzeiten. Die Ungenauigkeiten, die sich dadurch ergeben, werden aufsummiert und im nächsten Zyklus ausgeglichen.

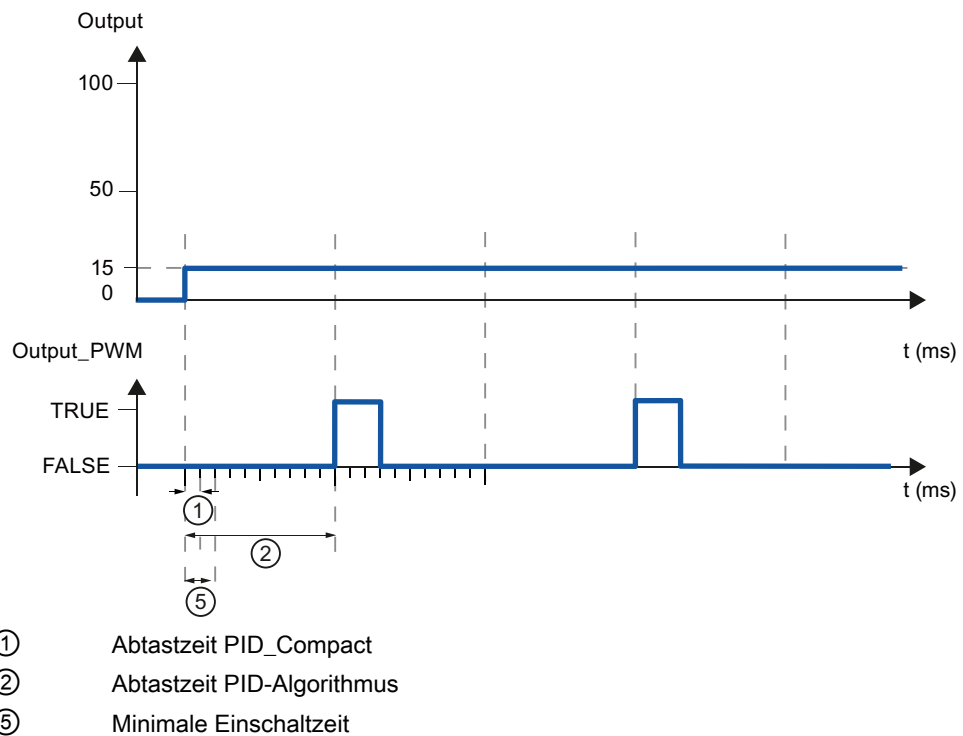
**Beispiel**

Abtastzeit PID\_Compact = 100 ms

Abtastzeit PID-Algorithmus = 1000 ms

Minimale Einschaltzeit = 200 ms

Output beträgt konstant 15 %. Der kleinste Impuls, den PID\_Compact ausgeben kann, entspricht 20%. Im ersten Zyklus wird kein Impuls ausgegeben. Im zweiten Zyklus wird der nicht ausgegebene Impuls des ersten Zyklus zum Impuls des zweiten Zyklus addiert.



Um die Schalthäufigkeit zu verringern und das Stellglied zu schonen, verlängern Sie die minimalen Ein- und Ausschaltzeiten.

Wenn Sie "Output" oder "Output\_PER" verwenden, müssen Sie die minimale Einschaltzeit und die minimale Ausschaltzeit auf den Wert 0.0 konfigurieren.

#### Hinweis

Die minimalen Ein- und Ausschaltzeiten wirken nur auf den Ausgangsparameter Output\_PWM und werden nicht für evtl. in der CPU integrierte Impulsgeneratoren verwendet.

## Ausgangswert

### Ausgangswertgrenzen

Konfigurieren Sie im Konfigurationsfenster "Ausgangswertgrenzen" die absoluten Grenzen Ihres Ausgangswerts in Prozent. Absolute Ausgangswertgrenzen werden weder im Handbetrieb noch im Automatikbetrieb über- oder unterschritten. Wird im Handbetrieb ein Ausgangswert außerhalb der Grenzen vorgegeben, so wird der effektive Wert in der CPU auf die konfigurierten Grenzen begrenzt.

Die Ausgangswertgrenzen müssen zum Regelsinn passen.

Die gültigen Werte für die Ausgangswertgrenzen hängen vom verwendeten Output ab.

|            |                    |
|------------|--------------------|
| Output     | -100.0 bis 100.0 % |
| Output_PER | -100.0 bis 100.0 % |
| Output_PWM | 0.0 bis 100.0 %    |

## Verhalten im Fehlerfall

|  |
|--|
| <b>ACHTUNG</b>   |
| <b>Ihre Anlage kann beschädigt werden.</b>   |
| Wenn Sie im Fehlerfall "Aktueller Wert für die Fehlerdauer" oder "Ersatzausgangswert für die Fehlerdauer" ausgeben, bleibt PID_Compact im Automatikbetrieb. Dadurch können die Istwertgrenzen überschritten und Ihre Anlage beschädigt werden. |
| Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.   |

PID\_Compact ist so voreingestellt, dass im Fehlerfall die Regelung in den meisten Fällen aktiv bleibt. Wenn im Regelbetrieb häufig Fehler auftreten, wird durch diese Voreinstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter Errorbits und beheben Sie die Fehlerursache.

Im Fehlerfall gibt PID\_Compact einen konfigurierbaren Ausgangswert aus:

- Null (Inaktiv)  
PID\_Compact gibt bei allen Fehlern 0.0 als Ausgangswert aus und schaltet in die Betriebsart "Inaktiv". Der Regler wird erst durch eine fallende Flanke an Reset oder eine steigende Flanke an ModeActivate reaktiviert.

- Aktuellen Wert für die Fehlerdauer  
Wenn im **Automatikbetrieb** folgende Fehler aufgetreten sind, kehrt PID\_Compact in den Automatikbetrieb zurück, sobald die Fehler nicht mehr anstehen.  
Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID\_Compact im Automatikbetrieb:

- 0001h: Der Parameter "Input" ist außerhalb der Istwertgrenzen.
- 0800h: Abtastzeitfehler
- 40000h: Ungültiger Wert am Parameter Disturbance.

Wenn im **Automatikbetrieb** einer oder mehrere der folgenden Fehler auftreten, wechselt PID\_Compact in die Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" und gibt den letzten gültigen Ausgangswert aus:

- 0002h: Ungültiger Wert am Parameter Input\_PER.
- 0200h: Ungültiger Wert am Parameter Input.
- 0400h: Berechnung des Ausgangswerts fehlgeschlagen.
- 1000h: Ungültiger Wert am Parameter Setpoint.

Wenn im **Handbetrieb** ein Fehler auftritt, verwendet PID\_Compact weiterhin den Handwert als Ausgangswert. Wenn der Handwert ungültig ist, wird der Ersatzausgangswert verwendet. Wenn Handwert und Ersatzausgangswert ungültig sind, wird die Untergrenze Ausgangswert verwendet.

Wenn während einer **Erst- oder Nachoptimierung** der folgende Fehler auftritt, bleibt PID\_Compact in der aktiven Betriebsart:

- 0020h: Die Erstopoptimierung ist während der Nachoptimierung nicht erlaubt.

Bei allen anderen Fehlern bricht PID\_Compact die Optimierung ab und wechselt in die Betriebsart, aus der die Optimierung gestartet wurde.

Sobald keine Fehler nicht mehr anstehen, kehrt PID\_Compact in den Automatikbetrieb zurück.

- Ersatzausgangswert für die Fehlerdauer  
PID\_Compact gibt den Ersatzausgangswert aus.  
Wenn der folgende Fehler auftritt, bleibt PID\_Compact in der Betriebsart "Ersatzausgangswert mit Fehlerüberwachung" und gibt die Untergrenze Ausgangswert aus:

- 20000h: Ungültiger Wert an der Variablen SubstituteOutput.

Bei allen anderen Fehler verhält sich PID\_Compact wie bei "Aktuellen Wert für die Fehlerdauer" beschrieben.

## Siehe auch

Parameter State und Mode V2 (Seite 2691)

### PID-Parameter

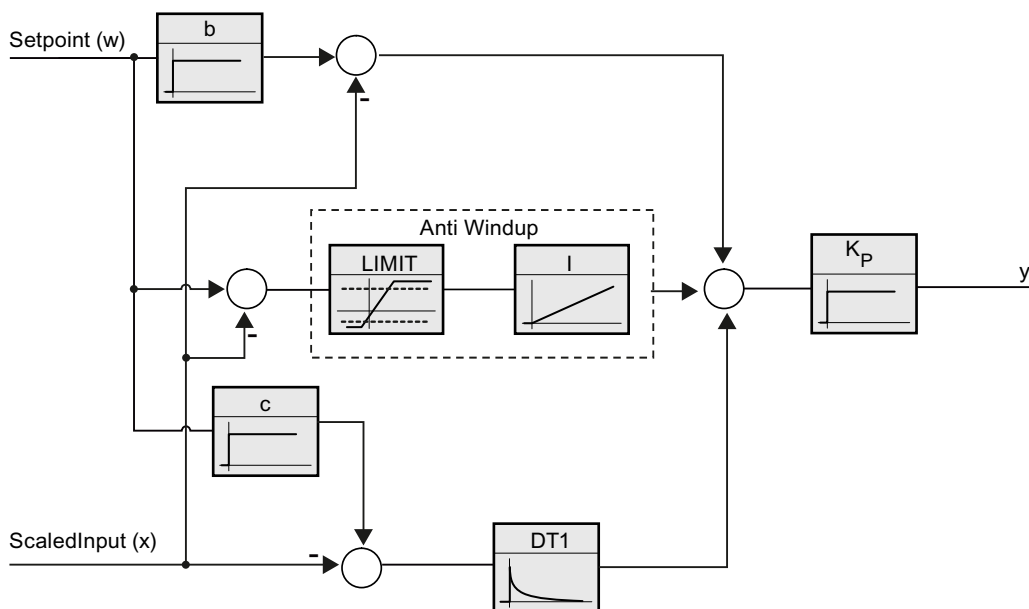
Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt. Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst. Sie brauchen die PID-Parameter nicht manuell eingeben.

Der PID-Algorithmus arbeitet nach folgender Formel:

$$y = K_p \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

| Symbol         | Beschreibung   |
|----------------|--|
| y              | Ausgangswert des PID-Algorithmus   |
| K <sub>p</sub> | Proportionalverstärkung  |
| s              | Laplace-Operator   |
| b              | Gewichtung des P-Anteils   |
| w              | Sollwert   |
| x              | Istwert  |
| T <sub>i</sub> | Integrationszeit   |
| a              | Koeffizient Differenzierverzug (Differenzierverzug T <sub>1</sub> = a × T <sub>D</sub> ) |
| T <sub>D</sub> | Differenzierzeit   |
| c              | Gewichtung des D-Anteils   |

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:



Alle PID-Parameter sind remanent. Wenn Sie die PID-Parameter manuell eingeben, müssen Sie PID\_Compact vollständig laden.

Technologieobjekte in Gerät laden (Seite 4056)

### Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID\_Compact arbeitet nicht mit einer negativen Proportionalverstärkung. Den Regelsinn invertieren Sie unter Grundeinstellungen > Regelungsart.

### Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0.

### Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

### Koeffizient Differenziervorgang

Die Wirkung des D-Anteils wird durch den Koeffizient Differenziervorgang verzögert.

Differenziervorgang = Differenzierzeit × Koeffizient Differenziervorgang

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

### Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

### Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

### Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID\_Compact werden bei jedem Aufruf durchgeführt.

Wenn Sie Output\_PWM verwenden, wird die Genauigkeit des Ausgangssignals bestimmt durch das Verhältnis von Abtastzeit PID-Algorithmus zu Zykluszeit des OB. Die Abtastzeit PID-Algorithmus entspricht der Periodendauer der Pulsweitenmodulation. Die Zykluszeit sollte mindestens das 10-fache der Abtastzeit PID-Algorithmus betragen.

### Regel für Optimierung

Wählen Sie in der Klappliste "Reglerstruktur" aus, ob PI oder PID-Parameter berechnet werden.

- **PID**  
Berechnet während der Erst- und Nachoptimierung PID-Parameter
- **PI**  
Berechnet während der Erst- und Nachoptimierung PI-Parameter
- **Benutzerdefiniert**  
Wenn Sie über ein Anwenderprogramm unterschiedliche Reglerstrukturen für Erst- und Nachoptimierung eingestellt haben, wird "Benutzerdefiniert" in der Klappliste angezeigt.

### PID\_Compact V2 in Betrieb nehmen

#### Erstoptimierung

Die Erstoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Dies ist am ehesten in den Betriebsarten "Inaktiv" oder "Handbetrieb" gegeben. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

#### Voraussetzung

- Die Anweisung "PID\_Compact" wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- PID\_Compact befindet sich in der Betriebsart "Handbetrieb", "Inaktiv" oder "Automatikbetrieb".



- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwertüberwachung").
- Die Differenz zwischen Sollwert und Istwert ist größer als 30 % der Differenz zwischen Obergrenze Istwert und Untergrenze Istwert.
- Der Abstand zwischen Sollwert und Istwert ist > 50% des Sollwerts.

## Vorgehen

Um die "Erstoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID\_Compact > Inbetriebnahme".
2. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Erstoptimierung".
3. Klicken Sie auf das Symbol "Start".
  - Eine Online-Verbindung wird aufgebaut.
  - Die Aufzeichnung der Werte wird gestartet.
  - Die Erstoptimierung wird gestartet.
  - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

---

### Hinweis

Klicken Sie auf das Symbol "Stop", wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

---

## Ergebnis

Wurde die Erstoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID\_Compact wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn eine Erstoptimierung nicht möglich ist, verhält sich PID\_Compact wie unter Verhalten im Fehlerfall konfiguriert.

## Siehe auch

Parameter State und Mode V2 (Seite 2691)

## Nachoptimierung

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-

Parameter aus der Erstopptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

PID\_Compact versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

### Voraussetzung

- Die Anweisung PID\_Compact wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen.
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
- Es werden keine Störungen erwartet.
- PID\_Compact befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

### Ablauf abhängig von Startsituation

Die Nachoptimierung können Sie aus den Betriebsarten "Inaktiv", "Automatikbetrieb" oder "Handbetrieb" starten. Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- Automatikbetrieb  
Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.  
PID\_Compact regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- Inaktiv oder Handbetrieb  
Wenn die Voraussetzungen für eine Erstopptimierung erfüllt sind, wird eine Erstopptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. Ist die Erstopptimierung nicht möglich, verhält sich PID\_Compact wie unter Verhalten im Fehlerfall konfiguriert.  
Wenn sich der Istwert für eine Erstopptimierung bereits zu nah am Sollwert befindet, wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen.

## Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Nachoptimierung".
2. Klicken Sie auf das Symbol "Start".
  - Eine Online-Verbindung wird aufgebaut.
  - Die Aufzeichnung der Werte wird gestartet.
  - Der Ablauf der Nachoptimierung wird gestartet.
  - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

---

### Hinweis

Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

---

## Ergebnis

Wenn während der Nachoptimierung keine Fehler auftraten, wurden die PID-Parameter optimiert. PID\_Compact wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der "Nachoptimierung" Fehler auftraten, verhält sich PID\_Compact wie unter Verhalten im Fehlerfall konfiguriert.

## Siehe auch

Parameter State und Mode V2 (Seite 2691)

## Betriebsart "Handbetrieb"


Im Folgenden wird beschrieben, wie Sie die Betriebsart "Handbetrieb" im Inbetriebnahmefenster des Technologieobjekts "PID\_Compact" nutzen können. Der Handbetrieb ist auch möglich, wenn ein Fehler ansteht.

## Voraussetzung

- Die Anweisung "PID\_Compact" wird in einem Weckalarm-OB aufgerufen.
- Es wurde eine Onlineverbindung zur CPU aufgebaut und die CPU befindet sich im Betriebszustand "RUN".

## Vorgehen

Verwenden Sie im Inbetriebnahmefenster "Handbetrieb", wenn Sie die Regelstrecke durch Vorgabe eines Handwerts testen möchten. Um einen Handwert vorzugeben, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Symbol "Start".
2. Aktivieren Sie im Bereich "Online-Zustand des Reglers" das Kontrollkästchen "Handbetrieb".  
PID\_Compact arbeitet im Handbetrieb. Der zuletzt aktuelle Ausgangswert bleibt aktiv.
3. Tragen Sie im Feld "Output" den Handwert in der Einheit % ein.
4. Klicken Sie auf das Symbol .

## Ergebnis

Der Handwert wird in die CPU geschrieben und ist sofort wirksam.

Deaktivieren Sie das Kontrollkästchen "Handbetrieb", wenn der Ausgangswert wieder durch den PID-Regler vorgegeben werden soll. Der Wechsel in den Automatikbetrieb erfolgt stoßfrei.

## Siehe auch

Parameter State und Mode V2 (Seite 2691)

### 11.1.3.3 PID\_Compact V1

## PID\_Compact V1 konfigurieren

## Grundeinstellungen

## Einleitung

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID\_Compact:

- Physikalische Größe
- Regelsinn
- Anlaufverhalten nach Reset
- Sollwert (nur im Inspektorfenster)
- Istwert (nur im Inspektorfenster)
- Ausgangswert (nur im Inspektorfenster)

## Sollwert, Istwert und Ausgangswert

Sollwert, Istwert und Ausgangswert können Sie nur im Inspektorfenster des Programmiereditors konfigurieren. Für jeden Wert wählen Sie die Quelle:

- Instanz-DB  
Es wird der Wert verwendet der im Instanz-DB gespeichert ist.  
Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden.  
An der Anweisung darf kein Wert stehen.  
Änderung über HMI möglich.
- Anweisung  
Es wird der Wert verwendet, der an der Anweisung verschaltet ist.  
Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben.  
Änderung über HMI nicht möglich.

## Regelungsart

### Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Soll- und Istwert. Soll- und Istwert werden in dieser Einheit angezeigt.

## Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn.

PID\_Compact arbeitet nicht mit negativer Proportionalverstärkung. Um durch einen höheren Ausgangswert den Istwert zu verringern, aktivieren Sie das Optionskästchen "Invertieren des Regelsinns".

Beispiele

- Durch Öffnen eines Ablaufventils sinkt der Füllstand eines Behälters.
- Durch eine höhere Kühlleistung sinkt die Temperatur.

## Anlaufverhalten nach Reset

Um nach Neustart der CPU sofort in die zuletzt aktive Betriebsart zu wechseln, aktivieren Sie das Optionskästchen "Nach CPU Neustart letzte Betriebsart aktivieren".

Wenn das Optionskästchen deaktiviert ist, bleibt PID\_Compact in der Betriebsart "Inaktiv".

## Sollwert

### Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80 °C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist.  
Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert Zeit gesteuert zu ändern.

## Istwert

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID\_Compact den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

### Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input\_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

## Ausgangswert

PID\_Compact stellt drei Ausgangswerte zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- Output\_PER  
Das Stellglied wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10V, 4...20mA.
- Output  
Der Ausgangswert muss über das Anwenderprogramm aufbereitet werden, z.B. weil das Stellglied nichtlineares Verhalten zeigt.
- Output\_PWM  
Das Stellglied wird über einen Digitalausgang gesteuert. Über eine Pulsweitenmodulation werden variable Ein- und Ausschaltzeiten gebildet.

## Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output\_PER (analog)".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output".
2. Wählen Sie "Instanz-DB".  
Der berechnete Ausgangswert wird im Instanz-DB gespeichert.
3. Verwenden Sie für die Aufbereitung des Ausgangswerts den Ausgangsparameter Output.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Digital- oder Analogausgang der CPU zum Stellglied.

Um den digitalen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output\_PWM".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Digitalausgangs ein.

## Istwerteinstellung

Konfigurieren Sie im Konfigurationsfenster "Istwerteinstellungen" die Normierung Ihres Istwerts und legen Sie die absoluten Istwertgrenzen fest.

## Istwert skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Input\_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input\_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input\_PER anhand eines unteren und oberen Wertepaars skaliert.

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID\_Compact.
2. Verschalten Sie in den Grundeinstellungen Input\_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung".

Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

### Istwert überwachen

Legen Sie die absolute Ober- und Untergrenze des Istwerts fest. Sobald im Betrieb diese Grenzen über oder unterschritten werden, so schaltet die Regelung ab und der Ausgangswert wird auf 0% gesetzt. Als Grenzwerte müssen Sie für Ihre Regelstrecke sinnvolle Werte eingeben. Während der Optimierung sind sinnvolle Grenzwerte wichtig, um optimale PID-Parameter zu erhalten.

Die Vorbelegung der "Obergrenze Istwert" beträgt 120 %. Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird keine Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID\_Compact wechselt in die Betriebsart "Inaktiv".



#### **WARNUNG**

Wenn Sie als Istwertgrenzen sehr hohe Werte (z. B.  $-3,4 \cdot 10^{38} \dots +3,4 \cdot 10^{38}$ ) einstellen, wird die Überwachung des Istwerts deaktiviert. Dann kann es durch einen Fehler zu Schäden an Ihrer Anlage kommen.

### Siehe auch

Istwertüberwachung (Seite 4083)

PWM-Begrenzungen (Seite 4083)

Ausgangswertgrenzen (Seite 4085)

PID-Parameter (Seite 4086)



## Erweiterte Einstellungen

### Istwertüberwachung

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über oder unterschritten, so wird an der Anweisung PID\_Compact eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning\_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning\_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

### Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID\_Compact verhält sich folgendermaßen:

| Istwert             | InputWarning_H | InputWarning_L | Betriebsart      |
|---------------------|----------------|----------------|------------------|
| > 98 °C             | TRUE           | FALSE          | Inaktiv          |
| ≤ 98 °C und > 90 °C | TRUE           | FALSE          | Automatikbetrieb |
| ≤ 90 °C und ≥ 10 °C | FALSE          | FALSE          | Automatikbetrieb |
| < 10 °C und ≥ 0 °C  | FALSE          | TRUE           | Automatikbetrieb |
| < 0 °C              | FALSE          | TRUE           | Inaktiv          |

### Siehe auch

Istwerteinstellung (Seite 4081)

PWM-Begrenzungen (Seite 4083)

Ausgangswertgrenzen (Seite 4085)

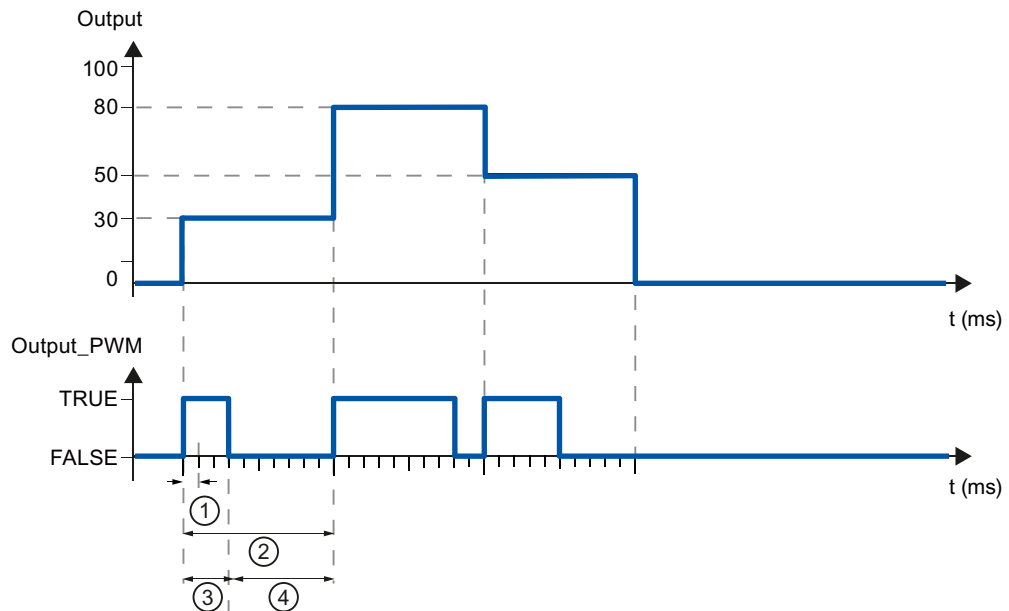
PID-Parameter (Seite 4086)

### PWM-Begrenzungen

Über eine Pulsweitenmodulation wird der Wert am Ausgangsparameter Output in eine Impulsfolge transformiert, die am Ausgangsparameter Output\_PWM ausgegeben wird. Output wird in der Abtastzeit PID-Algorithmus berechnet, Output\_PWM wird in der Abtastzeit PID\_Compact ausgegeben.

Die Abtastzeit PID-Algorithmus wird während der Erst- oder Nachoptimierung ermittelt. Wenn Sie die PID-Parameter manuell einstellen, müssen Sie dort auch die Abtastzeit PID-Algorithmus konfigurieren. Die Abtastzeit PID\_Compact entspricht der Zykluszeit des aufrufenden OBs.

Die Impulsdauer ist proportional zum Wert an Output und beträgt stets ein ganzzahliges Vielfaches der Abtastzeit PID\_Compact.



- ① Abtastzeit PID\_Compact
- ② Abtastzeit PID-Algorithmus
- ③ Impulsdauer
- ④ Pausendauer

Die "Minimale Einschaltzeit" und die "Minimale Ausschaltzeit" werden auf ein ganzzahliges Vielfaches der Abtastzeit PID\_Compact gerundet.

Ein Impuls oder eine Pause wird nie kürzer als die minimale Ein- oder Ausschaltzeiten. Die Ungenauigkeiten, die sich dadurch ergeben, werden aufsummiert und im nächsten Zyklus ausgeglichen.

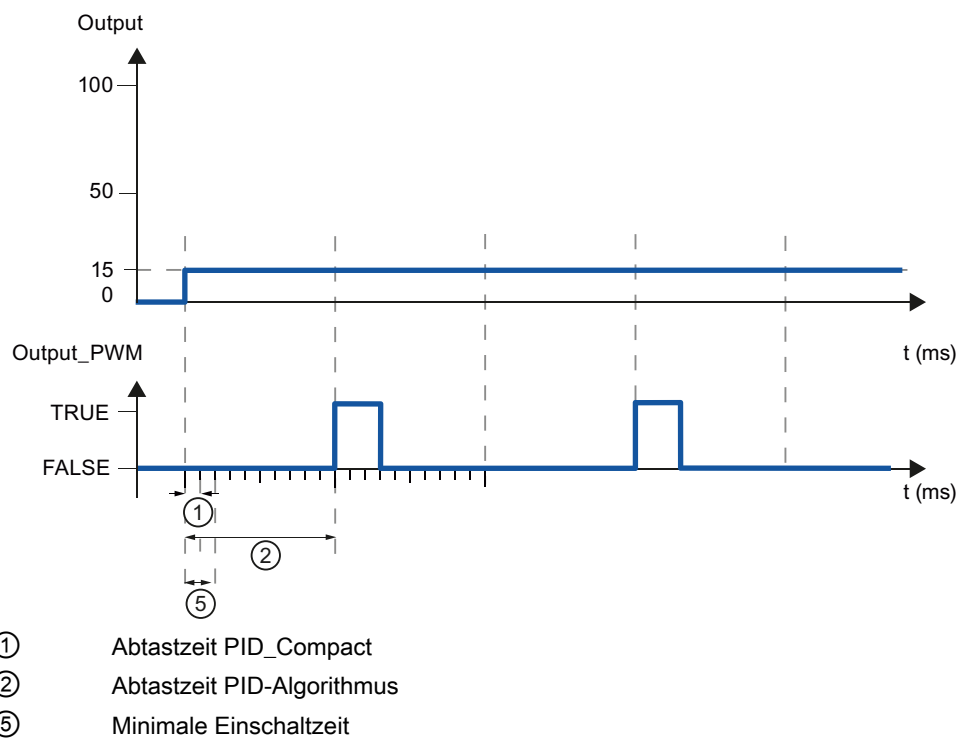
**Beispiel**

Abtastzeit PID\_Compact = 100 ms

Abtastzeit PID-Algorithmus = 1000 ms

Minimale Einschaltzeit = 200 ms

Output beträgt konstant 15 %. Der kleinste Impuls, den PID\_Compact ausgeben kann, entspricht 20%. Im ersten Zyklus wird kein Impuls ausgegeben. Im zweiten Zyklus wird der nicht ausgegebene Impuls des ersten Zyklus zum Impuls des zweiten Zyklus addiert.



Um die Schalthäufigkeit zu verringern und das Stellglied zu schonen, verlängern Sie die minimalen Ein- und Ausschaltzeiten.

Wenn Sie "Output" oder "Output\_PER" verwenden, müssen Sie die minimale Einschaltzeit und die minimale Ausschaltzeit auf den Wert 0.0 konfigurieren.

#### Hinweis

Die minimalen Ein- und Ausschaltzeiten wirken nur auf den Ausgangsparameter Output\_PWM und werden nicht für evtl. in der CPU integrierte Impulsgeneratoren verwendet.

#### Siehe auch

IstwertEinstellung (Seite 4081)

Istwertüberwachung (Seite 4083)

Ausgangswertgrenzen (Seite 4085)

PID-Parameter (Seite 4086)

#### Ausgangswertgrenzen

Konfigurieren Sie im Konfigurationsfenster "Ausgangswertgrenzen" die absoluten Grenzen Ihres Ausgangswerts in Prozent. Absolute Ausgangswertgrenzen werden weder im Handbetrieb noch im Automatikbetrieb über- oder unterschritten. Wird im Handbetrieb ein Ausgangswert außerhalb der Grenzen vorgegeben, so wird der effektive Wert in der CPU auf die konfigurierten Grenzen begrenzt.

Die gültigen Werte für die Ausgangswertgrenzen hängen vom verwendeten Output ab.

|            |                  |
|------------|------------------|
| Output     | -100.0 bis 100.0 |
| Output_PER | -100.0 bis 100.0 |
| Output_PWM | 0.0 bis 100.0    |

Im Fehlerfall setzt PID\_Compact den Ausgangswert auf 0.0. Daher muss 0.0 immer innerhalb der Ausgangswertgrenzen liegen. Um eine Untergrenze Ausgangswert größer 0.0 zu erhalten, müssen Sie im Anwenderprogramm zu Output und Output\_PER einen Offset addieren.

**Siehe auch**

- Istwerteinstellung (Seite 4081)
- Istwertüberwachung (Seite 4083)
- PWM-Begrenzungen (Seite 4083)
- PID-Parameter (Seite 4086)

**PID-Parameter**

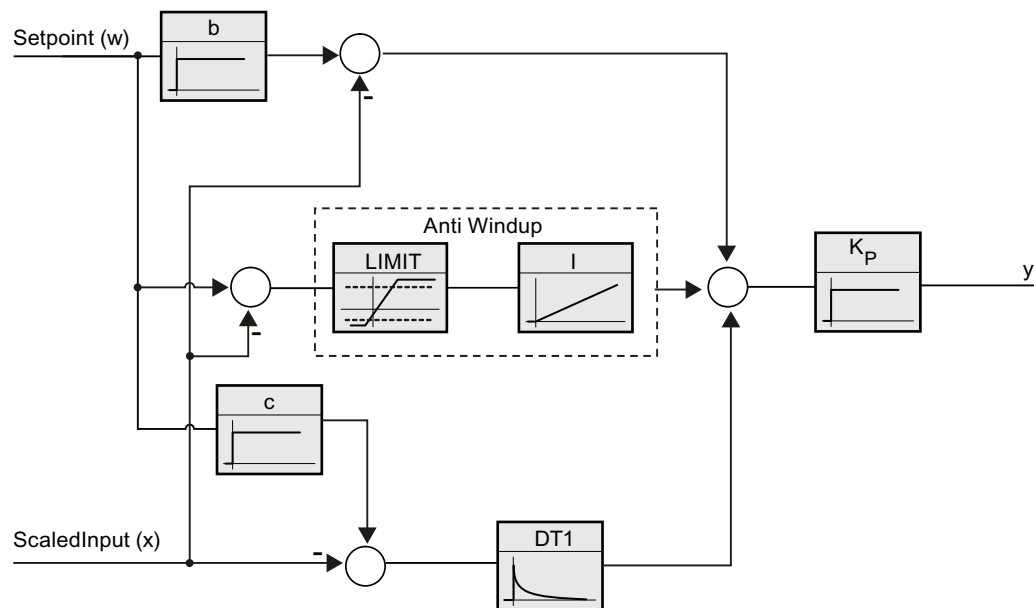
Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt. Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst. Sie brauchen die PID-Parameter nicht manuell eingeben.

Der PID-Algorithmus arbeitet nach folgender Formel:

$$y = K_p \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

| Symbol         | Beschreibung   |
|----------------|--|
| y              | Ausgangswert des PID-Algorithmus   |
| K <sub>p</sub> | Proportionalverstärkung  |
| s              | Laplace-Operator   |
| b              | Gewichtung des P-Anteils   |
| w              | Sollwert   |
| x              | Istwert  |
| T <sub>i</sub> | Integrationszeit   |
| a              | Koeffizient Differenzierverzug (Differenzierverzug T1 = a × T <sub>D</sub> ) |
| T <sub>D</sub> | Differenzierzeit   |
| c              | Gewichtung des D-Anteils   |

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:



Alle PID-Parameter sind remanent. Wenn Sie die PID-Parameter manuell eingeben, müssen Sie PID\_Compact vollständig laden.

Auto-Hotspot

### Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID\_Compact arbeitet nicht mit einer negativen Proportionalverstärkung. Den Regelsinn invertieren Sie unter Grundeinstellungen > Regelungsart.

### Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0.

### Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

### Koeffizient Differenziervorgang

Die Wirkung des D-Anteils wird durch den Koeffizient Differenziervorgang verzögert.

Differenziervorgang = Differenzierzeit × Koeffizient Differenziervorgang

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

### Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

### Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

### Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Zykluszeit gerundet. Alle anderen Funktionen von PID\_Compact werden bei jedem Aufruf durchgeführt.

Wenn Sie Output\_PWM verwenden, wird die Genauigkeit des Ausgangssignals bestimmt durch das Verhältnis von Abtastzeit PID-Algorithmus zu Zykluszeit des OB. Die Abtastzeit PID-Algorithmus entspricht der Periodendauer der Pulsweitenmodulation. Die Zykluszeit sollte mindestens das 10-fache der Abtastzeit PID-Algorithmus betragen.

### Regel für Optimierung

Wählen Sie in der Klappliste "Reglerstruktur" aus, ob PI oder PID-Parameter berechnet werden.

- **PID**  
Berechnet während der Erst- und Nachoptimierung PID-Parameter
- **PI**  
Berechnet während der Erst- und Nachoptimierung PI-Parameter
- **Benutzerdefiniert**  
Wenn Sie über ein Anwenderprogramm unterschiedliche Reglerstrukturen für Erst- und Nachoptimierung eingestellt haben, wird "Benutzerdefiniert" in der Klappliste angezeigt.

### Siehe auch

Technologieobjekte in Gerät laden (Seite 4056)

## PID\_Compact V1 in Betrieb nehmen

### Inbetriebnahme

Das Inbetriebnahmefenster hilft Ihnen bei der Inbetriebnahme des PID-Reglers. In der Kurvenanzeige können Sie die Werte für Sollwert, Istwert und Ausgangswert entlang der Zeitachse beobachten. Folgende Funktionen werden im Inbetriebnahmefenster unterstützt:

- Ersoptimierung des Reglers
- Nachoptimierung des Reglers  
Verwenden Sie die Nachoptimierung, wenn Sie eine Feinjustage der PID-Parameter wünschen.
- Beobachtung der laufenden Regelung im Kurvenfenster
- Test der Regelstrecke durch Vorgabe eines manuellen Ausgangswerts

Für alle Funktionen muss eine Onlineverbindung zur CPU aufgebaut sein.

### Grundsätzliche Bedienung

- Wählen Sie in der Klappliste "Aktualisierungszeit" die gewünschte Aktualisierungszeit aus. Alle Werte im Inbetriebnahmefenster werden in der ausgewählten Aktualisierungszeit aktualisiert.
- Klicken Sie auf das Symbol "Start" in der Gruppe Messung, wenn Sie die Inbetriebnahmefunktionen nutzen möchten.  
Die Aufzeichnung der Werte wird gestartet. In der Kurvenanzeige werden die aktuellen Werte für Sollwert, Istwert und Ausgangswert eingetragen. Die Bedienung des Inbetriebnahmefensters wird freigegeben.
- Klicken Sie auf das Symbol "Stop", wenn Sie die Inbetriebnahmefunktionen beenden möchten.  
Die in der Kurvenanzeige aufgezeichneten Werte können weiterhin analysiert werden.

Mit Schließen des Inbetriebnahmefensters wird die Aufzeichnung in der Kurvenanzeige beendet und die aufgezeichneten Werte werden gelöscht.

### Siehe auch

Ersoptimierung (Seite 4089)

Nachoptimierung (Seite 4091)

Betriebsart "Handbetrieb" (Seite 4093)

### Ersoptimierung

Die Ersoptimierung ermittelt die Prozessantwort auf einen Sprung des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts

signifikant größer ist als das Rauschen. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

### Voraussetzung

- Die Anweisung "PID\_Compact" wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- PID\_Compact befindet sich in der Betriebsart "Inaktiv" oder "Handbetrieb".
- Der Sollwert darf während der Optimierung nicht geändert werden. Sonst wird PID\_Compact deaktiviert.
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwertüberwachung").
- Die Differenz zwischen Sollwert und Istwert ist größer als 30 % der Differenz Obergrenze Istwert und Untergrenze Istwert.
- Der Abstand zwischen Sollwert und Istwert ist > 50% des Sollwerts.

### Vorgehen

Um die "Erstoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID\_Compact > Inbetriebnahme".
2. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Erstoptimierung".
3. Klicken Sie auf das Symbol "Start".
  - Eine Online-Verbindung wird aufgebaut.
  - Die Aufzeichnung der Werte wird gestartet.
  - Die Erstoptimierung wird gestartet.
  - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

---

#### Hinweis

Klicken Sie auf das Symbol "Stop", wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

---

### Ergebnis

Wurde die Erstoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID\_Compact wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn eine Erstoptimierung nicht möglich ist, wechselt PID\_Compact in die Betriebsart "Inaktiv".



## Siehe auch

Parameter State und sRet.i\_Mode V1 (Seite 2709)

Inbetriebnahme (Seite 4089)

Nachoptimierung (Seite 4091)

Betriebsart "Handbetrieb" (Seite 4093)

## Nachoptimierung

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstopptimierung.

PID\_Compact versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwerts. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwerts beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

## Voraussetzung

- Die Anweisung PID\_Compact wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwertüberwachung").
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
- Es werden keine Störungen erwartet.
- Der Sollwert darf während der Optimierung nicht geändert werden.
- PID\_Compact befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

### Ablauf abhängig von Startsituation

Die Nachoptimierung können Sie aus den Betriebsarten "Inaktiv", "Automatikbetrieb" oder "Handbetrieb" starten. Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- **Automatikbetrieb**  
Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.  
PID\_Compact regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- **Inaktiv oder Handbetrieb**  
Wenn die Voraussetzungen für eine Erstopptimierung erfüllt sind, wird eine Erstopptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung. Ist die Erstopptimierung nicht möglich, wechselt PID\_Compact in die Betriebsart "Inaktiv".  
Wenn sich der Istwert für eine Erstopptimierung bereits zu nah am Sollwert befindet, wird versucht den Sollwert mit minimalem oder maximalem Ausgangswert zu erreichen. Das kann ein erhöhtes Überschwingen verursachen.

### Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Nachoptimierung".
2. Klicken Sie auf das Symbol "Start".
  - Eine Online-Verbindung wird aufgebaut.
  - Die Aufzeichnung der Werte wird gestartet.
  - Der Ablauf der Nachoptimierung wird gestartet.
  - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

---

#### Hinweis

Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

---

### Ergebnis

Wurde die Nachoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID\_Compact wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der "Nachoptimierung" Fehler auftraten, wechselt PID\_Compact in die Betriebsart "Inaktiv".

## Siehe auch

Parameter State und sRet.i\_Mode V1 (Seite 2709)

Inbetriebnahme (Seite 4089)

Erstoptimierung (Seite 4089)

Betriebsart "Handbetrieb" (Seite 4093)

## Betriebsart "Handbetrieb"


Im Folgenden wird beschrieben, wie Sie die Betriebsart "Handbetrieb" im Inbetriebnahmefenster des Technologieobjekts "PID Compact" nutzen können.

## Voraussetzung

- Die Anweisung "PID\_Compact" wird in einem Weckalarm-OB aufgerufen.
- Es wurde eine Onlineverbindung zur CPU aufgebaut und die CPU befindet sich im Betriebszustand "RUN".
- Die Funktionen des Inbetriebnahmefensters sind über das Symbol "Messung Ein" freigegeben.

## Vorgehen

Verwenden Sie im Inbetriebnahmefenster "Handbetrieb", wenn Sie die Regelstrecke durch Vorgabe eines Handwerts testen möchten. Um einen Handwert vorzugeben, gehen Sie folgendermaßen vor:

1. Aktivieren Sie im Bereich "Online-Zustand des Relgers" das Kontrollkästchen "Handbetrieb".  
PID\_Compact arbeitet im Handbetrieb. Der zuletzt aktuelle Ausgangswert bleibt aktiv.
2. Tragen Sie im Feld "Output" den Handwert in der Einheit % ein.
3. Klicken Sie auf das Steuersymbol .

## Ergebnis

Der Handwert wird in die CPU geschrieben und ist sofort wirksam.

---

### Hinweis

PID\_Compact überwacht den Istwert weiterhin. Wenn die Istwertgrenzen überschritten werden, wird PID\_Compact deaktiviert.

---

Deaktivieren Sie das Kontrollkästchen "Handbetrieb", wenn der Ausgangswert wieder durch den PID-Regler vorgegeben werden soll. Der Wechsel in den Automatikbetrieb erfolgt stoßfrei.

## Siehe auch

- Parameter State und sRet.i\_Mode V1 (Seite 2709)
- Inbetriebnahme (Seite 4089)
- Erstoptimierung (Seite 4089)
- Nachoptimierung (Seite 4091)

## 11.1.4 PID\_3Step einsetzen

### 11.1.4.1 Technologieobjekt PID\_3Step

Das Technologieobjekt PID\_3Step stellt einen PID-Regler mit Optimierung für Ventile oder Stellglieder mit integrierendem Verhalten zur Verfügung.

Sie können folgende Regler konfigurieren:

- Dreipunktschrittregler mit Stellungsrückmeldung
- Dreipunktschrittregler ohne Stellungsrückmeldung
- Ventilregler mit analogem Ausgangswert

PID\_3Step erfasst innerhalb eines Regelkreises fortlaufend den gemessenen Istwert und vergleicht diesen mit dem Sollwert. Aus der sich ergebenden Regeldifferenz errechnet PID\_3Step einen Ausgangswert, durch den der Istwert den Sollwert möglichst schnell und stabil erreicht. Der Ausgangswert setzt sich beim PID-Regler aus drei Anteilen zusammen:

- **P-Anteil**  
Der P-Anteil des Ausgangswerts steigt proportional zur Regeldifferenz.
- **I-Anteil**  
Der I-Anteil des Ausgangswerts steigt solange bis die Regeldifferenz ausgeglichen ist.
- **D-Anteil**  
Der D-Anteil steigt mit wachsender Änderungsgeschwindigkeit der Regeldifferenz. Der Istwert wird möglichst schnell an den Sollwert angeglichen. Nimmt die Änderungsgeschwindigkeit der Regeldifferenz wieder ab, so verringert sich der D-Anteil wieder.

Die Anweisung PID\_3Step berechnet die P-, I-, und D-Parameter für Ihre Regelstrecke selbstständig während der Erstoptimierung. Die Parameter können über eine Nachoptimierung weiter optimiert werden. Sie brauchen die Parameter nicht manuell ermitteln.

## Weitere Informationen

- Übersicht der Software-Regler (Seite 4051)
- Technologieobjekte hinzufügen (Seite 4053)
- Technologieobjekte konfigurieren (Seite 4054)
- PID\_3Step V2 konfigurieren (Seite 4095)
- PID\_3Step V1 konfigurieren (Seite 4111)

## 11.1.4.2 PID\_3Step V2

### PID\_3Step V2 konfigurieren

#### Grundeinstellungen

##### Einleitung

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID\_3Step:

- Physikalische Größe
- Regelsinn
- Anlaufverhalten nach Reset
- Sollwert (nur im Inspektorfenster)
- Istwert (nur im Inspektorfenster)
- Ausgangswert (nur im Inspektorfenster)
- Stellungsrückmeldung (nur im Inspektorfenster)

##### Sollwert, Istwert, Ausgangswert und Stellungsrückmeldung

Sollwert, Istwert, Ausgangswert und Stellungsrückmeldung können Sie nur im Inspektorfenster des Programmiereditors konfigurieren. Für jeden Wert wählen Sie die Quelle:

- Instanz-DB  
Es wird der Wert verwendet der im Instanz-DB gespeichert ist.  
Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden.  
An der Anweisung darf kein Wert stehen.  
Änderung über HMI möglich.
- Anweisung  
Es wird der Wert verwendet, der an der Anweisung verschaltet ist.  
Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben.  
Änderung über HMI nicht möglich.

##### Regelungsart

##### Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Sollwert, Istwert und Störgröße. Sollwert, Istwert und Störgröße werden in dieser Einheit angezeigt.

##### Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn.

PID\_3Step arbeitet nicht mit negativer Proportionalverstärkung. Um durch einen höheren Ausgangswert den Istwert zu verringern, aktivieren Sie das Optionskästchen "Invertieren des Regelsinns".

Beispiele

- Durch Öffnen eines Ablaufventils sinkt der Füllstand eines Behälters.
- Durch eine höhere Kühlleistung sinkt die Temperatur.

## Anlaufverhalten

1. Um nach Neustart der CPU in die Betriebsart "Inaktiv" zu wechseln, deaktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".  
Um nach Neustart der CPU in die Betriebsart zu wechseln, die an Mode gespeichert ist, aktivieren Sie das Optionskästchen "Nach CPU Neustart Mode aktivieren".
2. Wählen Sie in der Klappliste "Mode setzen auf" die Betriebsart aus, die nach einem vollständigen Laden in Gerät aktiviert werden soll.  
Nach einem vollständigen Laden in Gerät startet PID\_3Step in der gewählten Betriebsart. Bei jedem weiteren Neustart, startet PID\_3Step in der Betriebsart, die zuletzt an Mode gespeichert war.

Beispiel

Sie haben das Optionskästchen "Nach CPU Neustart Mode aktivieren" aktiviert und in der Liste "Mode setzen auf" den Eintrag "Erstoptimierung" gewählt. Nach einem vollständigen Laden in Gerät startet PID\_3Step in der Betriebsart "Erstoptimierung". Wenn die Erstoptimierung noch aktiv ist, startet PID\_3Step nach dem Neustart der CPU wieder in der Betriebsart "Erstoptimierung". Wenn die Erstoptimierung erfolgreich beendet wurde und der Automatikbetrieb aktiv ist, startet PID\_3Step nach dem Neustart der CPU im "Automatikbetrieb".

## Sollwert

### Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80 °C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist. Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert Zeit gesteuert zu ändern.

## Istwert

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID\_3Step den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

## Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input\_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

## Stellungsrückmeldung

Die Konfiguration der Stellungsrückmeldung ist abhängig vom eingesetzten Stellglied.

- Stellglied ohne Stellungsrückmeldung
- Stellglied mit digitalen Anschlagssignalen
- Stellglied mit analoger Stellungsrückmeldung
- Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen

## Stellglied ohne Stellungsrückmeldung

Um PID\_3Step für ein Stellglied ohne Stellungsrückmeldung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Kein Feedback".

## Stellglied mit digitalen Anschlagssignalen

Um PID\_3Step für ein Stellglied mit Anschlagssignalen zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Kein Feedback".
2. Aktivieren Sie das Optionskästchen "Anschlagssignale Stellglied".
3. Wählen Sie für Actuator\_H und Actuator\_L als Quelle "Anweisung".
4. Geben Sie für Actuator\_H und Actuator\_L die Adressen der Digitaleingänge ein.

### Stellglied mit analoger Stellungsrückmeldung

Um PID\_3Step für ein Stellglied mit analoger Stellungsrückmeldung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Feedback" oder "Feedback\_PER".
  - Bei Feedback\_PER verwenden Sie den Wert des Analogeingangs direkt. Die Skalierung von Feedback\_PER konfigurieren Sie in den Stellgliedeinstellungen.
  - Bei Feedback bereiten Sie den Wert des Analogeingangs über Ihr Anwenderprogramm auf.
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs oder die Variable Ihres Anwenderprogramms ein.

### Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen

Um PID\_3Step für ein Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Feedback" oder "Feedback\_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs oder die Variable Ihres Anwenderprogramms ein.
4. Aktivieren Sie das Optionskästchen "Anschlagssignale Stellglied".
5. Wählen Sie für Actuator\_H und Actuator\_L als Quelle "Anweisung".
6. Geben Sie für Actuator\_H und Actuator\_L die Adressen der Digitaleingänge ein.

### Ausgangswert

PID\_3Step stellt einen analogen Ausgangswert (Output\_PER) und digitale Ausgangswerte (Output\_UP, Output\_DN) zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- Output\_PER  
Das Stellglied wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10V, 4...20mA.
- Output\_UP, Output\_DN  
Das Stellglied wird über zwei Digitalausgänge gesteuert.

### Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output (analog)".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den digitalen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:



1. Wählen Sie in der Klappliste "Output" den Eintrag "Output (digital)".
2. Wählen Sie für Output\_UP und Output\_DN "Anweisung".
3. Geben Sie die Adressen der Digitalausgänge ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den zum Stellglied passenden Eintrag.
2. Wählen Sie "Anweisung".
3. Geben Sie den Namen der Variablen an, die Sie für die Aufbereitung des Ausgangswerts verwenden.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Analog- oder Digitalausgang der CPU zum Stellglied.

## Istwerteinstellungen

### Istwert skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Input\_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input\_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input\_PER anhand eines unteren und oberen Wertepaars skaliert.

## Vorgehen

Um den Istwert zu skalieren, gehen Sie folgendermaßen vor:

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID\_3Step.
2. Verschalten Sie in den Grundeinstellungen Input\_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung".

Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

## Istwertgrenzen

Als Grenzwerte müssen Sie für Ihre Regelstrecke eine sinnvolle absolute Ober- und Untergrenze des Istwerts festlegen. Sobald diese Grenzen über- oder unterschritten werden, tritt ein Fehler auf (ErrorBits = 0001h). Die Optimierung wird abgebrochen, wenn die

Istwertgrenzen überschritten werden. Wie PID\_3Step im Fehlerfall im Automatikbetrieb reagiert, legen Sie bei den Stellgliedeinstellungen fest.

## Stellgliedeinstellungen

### Stellglied

#### Stellglied spezifische Zeiten

Um das Stellglied vor Beschädigung zu schützen, konfigurieren Sie die Motorstellzeit, die minimale Einschaltzeit und die minimale Ausschaltzeit. Die Daten finden Sie im Datenblatt des Stellglieds.

Die Motorstellzeit ist die Zeit in Sekunden, die der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen. Die Motorstellzeit kann während der Inbetriebnahme ermittelt werden.

Die Motorstellzeit ist remanent. Wenn Sie die Motorstellzeit manuell ändern, müssen Sie PID\_3Step vollständig laden.

Technologieobjekte in Gerät laden (Seite 4056)

Wenn Sie Output\_UP und Output\_DN verwenden, verringern Sie die Schalthäufigkeit mit der minimalen Ein- und Ausschaltzeit.

Im Automatikbetrieb werden die berechneten Ein- oder Ausschaltzeiten summiert und erst wirksam, wenn die Summe größer oder gleich der minimalen Ein- oder Ausschaltzeit ist.

Im Handbetrieb wird das Stellglied durch Manual\_UP = TRUE oder Manual\_DN = TRUE mindestens um die minimale Ein- oder Ausschaltzeit angesteuert.

#### Verhalten im Fehlerfall

PID\_3Step ist so voreingestellt, dass im Fehlerfall die Regelung in den meisten Fällen aktiv bleibt. Wenn im Regelbetrieb häufig Fehler auftreten, wird durch diese Voreinstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter Errorbits und beheben Sie die Fehlerursache.

#### **ACHTUNG**

##### **Ihre Anlage kann beschädigt werden.**

Wenn Sie im Fehlerfall "Aktueller Wert für die Fehlerdauer" oder "Ersatzausgangswert für die Fehlerdauer" ausgeben, bleibt PID\_3Step auch beim Überschreiten der Istwertgrenzen im Automatikbetrieb. Dadurch kann Ihre Anlage beschädigt werden.

Konfigurieren Sie für Ihre Regelstrecke ein Verhalten im Fehlerfall, das Ihre Anlage vor Beschädigung schützt.

Im Fehlerfall gibt PID\_3Step einen konfigurierbaren Ausgangswert aus:

- Aktueller Wert  
PID\_3Step ist ausgeschaltet und verändert die Position des Stellglieds nicht mehr.
- Aktueller Wert für die Fehlerdauer  
Die Reglerfunktionen von PID\_3Step sind ausgeschaltet und die Position des Stellglieds wird nicht mehr verändert.  
Wenn im Automatikbetrieb folgende Fehler aufgetreten sind, kehrt PID\_3Step in den Automatikbetrieb zurück, sobald die Fehler nicht mehr anstehen.
  - 0002h: Ungültiger Wert am Parameter Input\_PER.
  - 0200h: Ungültiger Wert am Parameter Input.
  - 0400h: Berechnung des Ausgangswerts fehlgeschlagen.
  - 1000h: Ungültiger Wert am Parameter Setpoint.
  - 2000h: Ungültiger Wert am Parameter Feedback\_PER.
  - 4000h: Ungültiger Wert am Parameter Feedback.
  - 8000h: Fehler bei der digitalen Stellungsrückmeldung.
  - 20000h: Ungültiger Wert an der Variablen SavePosition.Wenn einer oder mehrere der folgenden Fehler auftreten, bleibt PID\_3Step im Automatikbetrieb:
  - 0001h: Der Parameter Input ist außerhalb der Istwertgrenzen.
  - 0800h: Abtastzeitfehler
  - 40000h: Ungültiger Wert am Parameter Disturbance.Wenn im Handbetrieb ein Fehler auftritt, bleibt PID\_3Step im Handbetrieb.  
Wenn während der Optimierung oder Stellzeitmessung ein Fehler auftritt, wechselt PID\_3Step in die Betriebsart, aus der die Optimierung oder Stellzeitmessung gestartet wurde. Nur bei folgendem Fehler wird die Optimierung nicht abgebrochen:
  - 0020h: Die Erstopoptimierung ist während der Nachoptimierung nicht erlaubt.
- Ersatzausgangswert  
PID\_3Step fährt das Stellglied auf den Ersatzausgangswert und schaltet sich aus.
- Ersatzausgangswert für die Fehlerdauer  
PID\_3Step fährt das Stellglied auf den Ersatzausgangswert. Nach dem Erreichen des Ersatzausgangswerts verhält sich PID\_3Step wie bei "Aktueller Wert für die Fehlerdauer" beschrieben.

Den Ersatzausgangswert tragen Sie in "%" ein.

Bei Stellgliedern ohne analoge Stellungsrückmeldung können nur die Ersatzausgangswerte 0 % und 100 % exakt angefahren werden. Ein Ersatzausgangswert ungleich 0 % oder 100 % wird über eine intern simulierte Stellungsrückmeldung angefahren. Mit diesem Verfahren kann der Ersatzausgangswert aber nie exakt angefahren werden.

Bei Stellgliedern mit analoger Stellungsrückmeldung können alle Ersatzausgangswerte exakt angefahren werden.

## Stellungsrückmeldung skalieren

### Stellungsrückmeldung skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Feedback\_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in % umrechnen. Im Anzeigefeld "Feedback" wird die aktuelle Konfiguration angezeigt.

Feedback\_PER wird anhand eines unteren und oberen Wertepaars skaliert.

1. Geben Sie in den Eingabefeldern "Unterer Anschlag" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Oberer Anschlag" und "Oben" das obere Wertepaar ein.

"Unterer Anschlag" muss kleiner sein als "Oberer Anschlag"; "Unten" muss kleiner sein als "Oben".

Die gültigen Werte für "Oberer Anschlag" und "Unterer Anschlag" hängen ab von:

- Kein Feedback, Feedback, Feedback\_PER
- Output (analog), Output (digital)

| Output           | Feedback      | Unterer Anschlag          | Oberer Anschlag             |
|------------------|---------------|---------------------------|-----------------------------|
| Output (digital) | Kein Feedback | nicht einstellbar (0.0 %) | nicht einstellbar (100.0 %) |
| Output (digital) | Feedback      | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |
| Output (digital) | Feedback_PER  | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |
| Output (analog)  | Kein Feedback | nicht einstellbar (0.0 %) | nicht einstellbar (100.0 %) |
| Output (analog)  | Feedback      | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |
| Output (analog)  | Feedback_PER  | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |

## Ausgangswertgrenzen

### Ausgangswert begrenzen

Nur während der Stellzeitmessung können die Ausgangswertgrenzen über- oder unterschritten werden. In allen anderen Betriebsarten wird der Ausgangswert auf diese Werte begrenzt.

Geben Sie in den Eingabefeldern "Obergrenze Ausgangswert" und "Untergrenze Ausgangswert" die absoluten Ausgangswertgrenzen ein. Die Ausgangswertgrenzen müssen innerhalb "Unterer Anschlag" und "Oberer Anschlag" liegen.

Wenn kein Feedback vorhanden ist und Output (digital) eingestellt ist, können Sie den Ausgangswert nicht begrenzen. Output\_UP und Output\_DN werden dann bei Actuator\_H = TRUE oder Actuator\_L = TRUE zurück gesetzt. Wenn keine Anschlagsignale vorhanden sind, werden Output\_UP und Output\_DN nach einer Verfahrzeit von 150% der Motorstellzeit zurück gesetzt.

## Erweiterte Einstellungen

### Istwertüberwachung

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über oder unterschritten, so wird an der Anweisung PID\_3Step eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning\_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning\_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

### Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID\_3Step verhält sich folgendermaßen:

| Istwert             | InputWarning_H | InputWarning_L | ErrorBits | Betriebsart      |
|---------------------|----------------|----------------|-----------|------------------|
| > 98 °C             | TRUE           | FALSE          | 0001h     | Wie konfiguriert |
| ≤ 98 °C und > 90 °C | TRUE           | FALSE          | 0000h     | Automatikbetrieb |
| ≤ 90 °C und ≥ 10 °C | FALSE          | FALSE          | 0000h     | Automatikbetrieb |
| < 10 °C und ≥ 0 °C  | FALSE          | TRUE           | 0000h     | Automatikbetrieb |
| < 0 °C              | FALSE          | TRUE           | 0001h     | Wie konfiguriert |

Wie PID\_3Step beim Über- oder Unterschreiten der Obergrenze bzw. Untergrenze Istwert reagiert, konfigurieren Sie bei den Stellglicheinstellungen.

### PID-Parameter

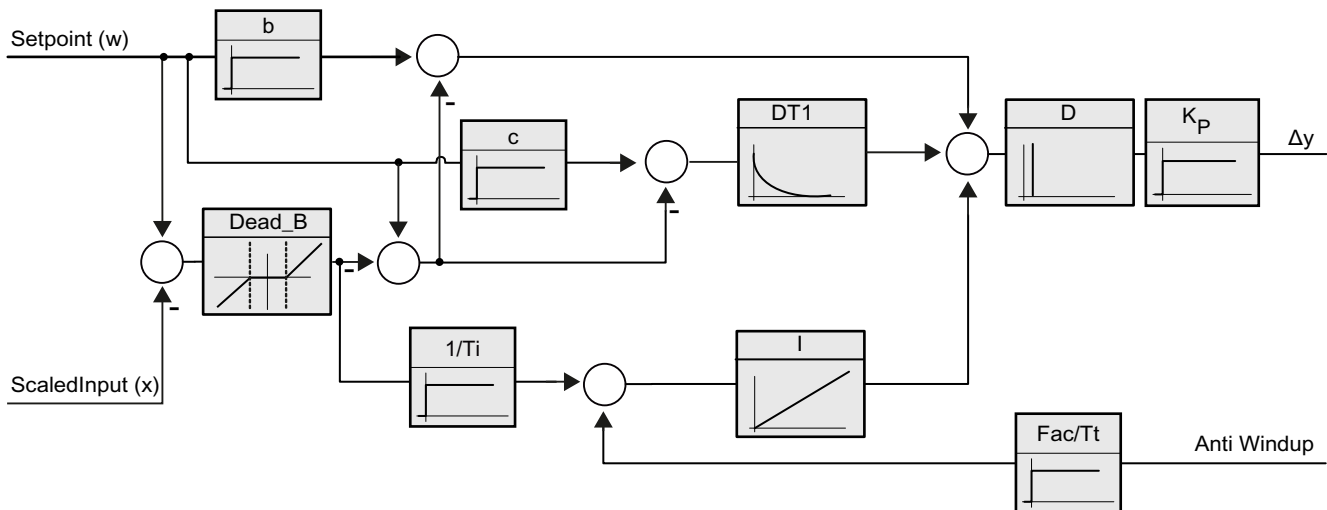
Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt. Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst. Sie brauchen die PID-Parameter nicht manuell eingeben.

Der PID-Algorithmus arbeitet nach folgender Formel:

$$\Delta y = K_p \cdot s \cdot \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

| Symbol     | Beschreibung  |
|------------|---|
| $\Delta y$ | Ausgangswert des PID-Algorithmus  |
| $K_p$      | Proportionalverstärkung   |
| s          | Laplace-Operator  |
| b          | Gewichtung des P-Anteils  |
| w          | Sollwert  |
| x          | Istwert   |
| $T_i$      | Integrationszeit  |
| a          | Koeffizient für den Differenzierverzug (Differenzierverzug $T_1 = a \times T_D$ ) |
| $T_D$      | Differenzierzeit  |
| c          | Gewichtung des D-Anteils  |

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:



Alle PID-Parameter sind remanent. Wenn Sie die PID-Parameter manuell eingeben, müssen Sie PID\_3Step vollständig laden.

Technologieobjekte in Gerät laden (Seite 4056)

### Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID\_3Step arbeitet nicht mit einer negativen Proportionalverstärkung. Den Regelsinn invertieren Sie unter Grundeinstellungen > Regelungsart.

### Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0.

### Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

### Koeffizient Differenziervorgang

Die Wirkung des D-Anteils wird durch den Koeffizient Differenziervorgang verzögert.

Differenziervorgang = Differenzierzeit × Koeffizient Differenziervorgang

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

### Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

### Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

### Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Abtastzeit PID\_3Step gerundet. Alle anderen Funktionen von PID\_3Step werden bei jedem Aufruf durchgeführt.

### Totzonenbreite

Die Totzone unterdrückt die Rauschanteile im eingeschwungenen Reglerzustand. Die Totzonenbreite gibt die Größe der Totzone an. Bei einer Totzonenbreite von 0.0 ist die Totzone abgeschaltet.

## PID\_3Step V2 in Betrieb nehmen

### Erstoptimierung

Die Erstoptimierung ermittelt die Prozessantwort auf einen Impuls des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Dies ist am ehesten in den Betriebsarten "Inaktiv" oder "Handbetrieb" gegeben. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Der Sollwert wird während der Erstoptimierung eingefroren.

### Voraussetzung

- Die Anweisung PID\_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- Die Motorstellzeit ist konfiguriert oder gemessen.
- PID\_3Step befindet sich in der Betriebsart "Inaktiv", "Handbetrieb" oder "Automatikbetrieb".
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").

### Vorgehen

Um die Erstoptimierung durchzuführen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID\_3Step > Inbetriebnahme".
2. Wählen Sie im Arbeitsbereich "Optimierung" in der Klappliste "Optimierungsart" den Eintrag "Erstoptimierung".
3. Klicken Sie auf das Symbol "Start".
  - Eine Online-Verbindung wird aufgebaut.
  - Die Aufzeichnung der Werte wird gestartet.
  - Die Erstoptimierung wird gestartet.
  - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

---

#### Hinweis

Klicken Sie auf das Symbol "Stop", wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

---



## Ergebnis

Wurde die Erstopptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID\_3Step wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn eine Erstopptimierung nicht möglich ist, verhält sich PID\_3Step wie unter Verhalten im Fehlerfall konfiguriert.

## Nachoptimierung

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungsverhalten als die PID-Parameter aus der Erstopptimierung. Die besten PID-Parameter erhalten Sie, wenn Sie Erst- und Nachoptimierung durchführen.

PID\_3Step versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwertes. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwertes beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Der Sollwert wird während der Nachoptimierung eingefroren.

## Voraussetzung

- Die Anweisung PID\_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- Die Motorstellzeit ist konfiguriert oder gemessen.
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
- Es werden keine Störungen erwartet.
- PID\_3Step befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

## Ablauf abhängig von Startsituation

Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- **Automatikbetrieb**  
Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.  
PID\_3Step regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- **Inaktiv oder Handbetrieb**  
Zuerst wird immer eine Ersoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.

## Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Nachoptimierung".
2. Klicken Sie auf das Symbol "Start".
  - Eine Online-Verbindung wird aufgebaut.
  - Die Aufzeichnung der Werte wird gestartet.
  - Der Ablauf der Nachoptimierung wird gestartet.
  - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

---

### Hinweis

Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

---

## Ergebnis

Wenn während der Nachoptimierung keine Fehler auftraten, wurden die PID-Parameter optimiert. PID\_3Step wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der Nachoptimierung Fehler auftraten, verhält sich PID\_3Step wie unter Verhalten im Fehlerfall konfiguriert.

## Mit manuellen PID-Parametern in Betrieb nehmen

### Voraussetzung

- Die Anweisung PID\_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Reset = FALSE
- Die Motorstellzeit ist konfiguriert oder gemessen.
- PID\_3Step befindet sich in der Betriebsart "Inaktiv".
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").

### Vorgehen

Um PID\_3Step mit manuellen PID-Parametern in Betrieb zu nehmen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID\_3Step > Konfiguration".
2. Klicken Sie im Konfigurationsfenster auf "Erweiterte Einstellungen > PID-Parameter".
3. Aktivieren Sie das Optionskästchen "Manuelle Eingabe aktivieren".
4. Geben Sie die PID-Parameter ein.
5. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID\_3Step > Inbetriebnahme".
6. Stellen Sie eine Online-Verbindung zur CPU her.
7. Laden Sie die PID-Parameter auf die CPU.
8. Klicken Sie auf das Symbol "Start PID\_3Step".

### Ergebnis

PID\_3Step wechselt in den Automatikbetrieb und regelt mit den aktuellen PID-Parametern.

### Siehe auch

PID-Parameter (Seite 4103)

### Motorstellzeit messen

### Einleitung

PID\_3Step benötigt die Motorstellzeit so genau wie möglich, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren.

Wenn Sie Stellglieder mit Stellungsrückmeldung oder mit Anschlagssignalen verwenden, können Sie während der Inbetriebnahme die Motorstellzeit messen. Die Ausgangswertgrenzen werden während der Motorstellzeitmessung nicht berücksichtigt. Das Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

Wenn weder eine Stellungsrückmeldung noch Anschlagssignale verfügbar sind, kann die Motorstellzeit nicht gemessen werden.

### Stellglieder mit analoger Stellungsrückmeldung

Um die Motorstellzeit mit Stellungsrückmeldung zu messen, gehen Sie folgendermaßen vor:

#### Voraussetzung

- In den Grundeinstellungen ist Feedback oder Feedback\_PER ausgewählt und das Signal ist verschaltet.
  - Eine Online-Verbindung zur CPU ist aufgebaut.
1. Aktivieren Sie das Optionsfeld "Stellungsrückmeldung verwenden".
  2. Geben Sie im Eingabefeld "Zielstellung" ein, wohin das Stellglied bewegt werden soll. Die aktuelle Stellungsrückmeldung (Startstellung) wird angezeigt. Die Differenz zwischen "Zielstellung" und "Stellungsrückmeldung" muss mindestens 50 % des zulässigen Ausgangswertbereichs sein.
  3. Klicken Sie auf das Symbol "Start".

### Ergebnis

Das Stellglied wird von der Startstellung in die Zielstellung gefahren. Die Zeitmessung startet sofort und endet, wenn das Stellglied die Zielstellung erreicht hat. Die Motorstellzeit wird berechnet nach der Formel:

$$\text{Motorstellzeit} = (\text{Obergrenze Ausgangswert} - \text{Untergrenze Ausgangswert}) \times \text{Messzeit} / \text{BETRAG}(\text{Zielstellung} - \text{Startstellung}).$$

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Instanz-Datenbaustein auf der CPU gespeichert und im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist und ActivateRecoverMode = TRUE ist, wechselt PID\_3Step in die Betriebsart, aus der die Stellzeitmessung gestartet wurde. Wenn die Stellzeitmessung beendet ist und ActivateRecoverMode = FALSE ist, wechselt PID\_3Step in die Betriebsart "Inaktiv".

---

#### Hinweis

Um die gemessene Motorstellzeit in das Projekt zu übernehmen, klicken Sie auf das Symbol  "Gemessene Stellzeit laden".

---

### Stellglieder mit Anschlagssignalen

Um die Stellzeit von Stellgliedern mit Anschlagssignalen zu messen, gehen Sie folgendermaßen vor:

#### Voraussetzung

- In den Grundeinstellungen ist das Optionskästchen "Anschlagsignale" aktiviert und Actuator\_H und Actuator\_L sind verschaltet.
- Eine Online-Verbindung zur CPU ist aufgebaut.

Um die Motorstellzeit mit Anschlagsignalen zu messen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionsfeld "Anschlagsignale des Stellglieds verwenden".
2. Wählen Sie die Richtung, in die das Stellglied bewegt werden soll.
  - Öffnen - Schließen - Öffnen  
Das Stellglied wird zuerst bis zum oberen Anschlag gefahren, dann zum unteren Anschlag und erneut zum oberen Anschlag.
  - Schließen - Öffnen - Schließen  
Das Stellglied wird zuerst bis zum unteren Anschlag gefahren, dann zum oberen Anschlag und erneut zum unteren Anschlag.
3. Klicken Sie auf das Symbol "Start".

## Ergebnis

Das Stellglied wird in der gewählten Richtung bewegt. Die Zeitmessung startet, wenn das Stellglied den ersten Anschlag erreicht hat und endet, wenn das Stellglied diesen Anschlag zum zweiten Mal erreicht. Die gemessene Zeit geteilt durch zwei ergibt die Motorstellzeit.

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Instanz-Datenbaustein auf der CPU gespeichert und im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist und ActivateRecoverMode = TRUE ist, wechselt PID\_3Step in die Betriebsart, aus der die Stellzeitmessung gestartet wurde. Wenn die Stellzeitmessung beendet ist und ActivateRecoverMode = FALSE ist, wechselt PID\_3Step in die Betriebsart "Inaktiv".

## Stellzeitmessung abbrechen

Wenn Sie die Stellzeitmessung mit der Schaltfläche Stop abbrechen, wechselt PID\_3Step in die Betriebsart "Inaktiv".

### 11.1.4.3 PID\_3Step V1

## PID\_3Step V1 konfigurieren

### Grundeinstellungen

#### Einleitung

Konfigurieren Sie im Inspektorfenster, bzw. im Konfigurationsfenster in den "Grundeinstellungen" folgende Eigenschaften des Technologieobjekts PID\_3Step:

- Physikalische Größe
- Regelsinn

- Anlaufverhalten nach Reset
- Sollwert (nur im Inspektorfenster)
- Istwert (nur im Inspektorfenster)
- Ausgangswert (nur im Inspektorfenster)
- Stellungsrückmeldung (nur im Inspektorfenster)

### Sollwert, Istwert, Ausgangswert und Stellungsrückmeldung

Sollwert, Istwert, Ausgangswert und Stellungsrückmeldung können Sie nur im Inspektorfenster des Programmiereditors konfigurieren. Für jeden Wert wählen Sie die Quelle:

- Instanz-DB  
Es wird der Wert verwendet der im Instanz-DB gespeichert ist.  
Wert muss im Instanz-DB vom Anwenderprogramm aktualisiert werden.  
An der Anweisung darf kein Wert stehen.  
Änderung über HMI möglich.
- Anweisung  
Es wird der Wert verwendet, der an der Anweisung verschaltet ist.  
Bei jedem Aufruf der Anweisung wird der Wert in den Instanz-DB geschrieben.  
Änderung über HMI nicht möglich.

### Regelungsart

#### Physikalische Größe

Wählen Sie in der Gruppe "Regelungsart" die physikalische Größe und Einheit für Soll- und Istwert. Soll- und Istwert werden in dieser Einheit angezeigt.

#### Regelsinn

Meist soll mit einer Erhöhung des Ausgangswerts eine Erhöhung des Istwerts erreicht werden. In diesem Fall spricht man von einem normalen Regelsinn.

PID\_3Step arbeitet nicht mit negativer Proportionalverstärkung. Um durch einen höheren Ausgangswert den Istwert zu verringern, aktivieren Sie das Optionskästchen "Invertieren des Regelsinns".

Beispiele

- Durch Öffnen eines Ablaufventils sinkt der Füllstand eines Behälters.
- Durch eine höhere Kühlleistung sinkt die Temperatur.

#### Anlaufverhalten nach Reset

Um nach Neustart der CPU sofort in die zuletzt aktive Betriebsart zu wechseln, aktivieren Sie das Optionskästchen "Nach CPU Neustart letzte Betriebsart aktivieren".

Wenn das Optionskästchen deaktiviert ist, bleibt der PID\_3Step in der Betriebsart "Inaktiv".

## Sollwert

### Vorgehen

Um einen festen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Instanz-DB".
2. Geben Sie einen Sollwert ein, z. B. 80 °C.
3. Löschen Sie ggf. einen Eintrag an der Anweisung.

Um einen variablen Sollwert vorzugeben, gehen Sie folgendermaßen vor:

1. Wählen Sie "Anweisung".
2. Geben Sie den Namen der REAL-Variablen ein, in der der Sollwert gespeichert ist.  
Die REAL-Variable können Sie programmgesteuert mit unterschiedlichen Werten belegen, z. B. um den Sollwert Zeit gesteuert zu ändern.

## Istwert

Wenn Sie den Wert des Analogeingangs direkt verwenden, skaliert PID\_3Step den Wert des Analogeingangs in die physikalische Größe.

Wenn Sie den Wert des Analogeingangs erst aufbereiten wollen, müssen Sie ein eigenes Programm für die Aufbereitung schreiben. Zum Beispiel ist der Istwert nicht direkt proportional zum Wert am Analogeingang. Der aufbereitete Istwert muss im Gleitpunktformat vorliegen.

### Vorgehen

Um den Wert des Analogeingangs direkt zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input\_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs ein.

Um den aufbereiteten Istwert im Gleitpunktformat zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Input" den Eintrag "Input".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie den Namen der Variablen ein, in der der aufbereitete Istwert gespeichert ist.

## Stellungsrückmeldung

Die Konfiguration der Stellungsrückmeldung ist abhängig vom eingesetzten Stellglied.

- Stellglied ohne Stellungsrückmeldung
- Stellglied mit digitalen Anschlagssignalen
- Stellglied mit analoger Stellungsrückmeldung
- Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen

### Stellglied ohne Stellungsrückmeldung

Um PID\_3Step für ein Stellglied ohne Stellungsrückmeldung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Kein Feedback".

### Stellglied mit digitalen Anschlagssignalen

Um PID\_3Step für ein Stellglied mit Anschlagssignalen zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Kein Feedback".
2. Aktivieren Sie das Optionskästchen "Anschlagssignale Stellglied".
3. Wählen Sie für Actuator\_H und Actuator\_L als Quelle "Anweisung".
4. Geben Sie für Actuator\_H und Actuator\_L die Adressen der Digitaleingänge ein.

### Stellglied mit analoger Stellungsrückmeldung

Um PID\_3Step für ein Stellglied mit analoger Stellungsrückmeldung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Feedback" oder "Feedback\_PER".
  - Bei Feedback\_PER verwenden Sie den Wert des Analogeingangs direkt. Die Skalierung von Feedback\_PER konfigurieren Sie in den Stellgliedeinstellungen.
  - Bei Feedback bereiten Sie den Wert des Analogeingangs über Ihr Anwenderprogramm auf.
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs oder die Variable Ihres Anwenderprogramms ein.

### Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen

Um PID\_3Step für ein Stellglied mit analoger Stellungsrückmeldung und Anschlagssignalen zu konfigurieren, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Feedback" den Eintrag "Feedback" oder "Feedback\_PER".
2. Wählen Sie als Quelle "Anweisung".
3. Geben Sie die Adresse des Analogeingangs oder die Variable Ihres Anwenderprogramms ein.
4. Aktivieren Sie das Optionskästchen "Anschlagssignale Stellglied".
5. Wählen Sie für Actuator\_H und Actuator\_L als Quelle "Anweisung".
6. Geben Sie für Actuator\_H und Actuator\_L die Adressen der Digitaleingänge ein.



## Ausgangswert

PID\_3Step stellt einen analogen Ausgangswert (Output\_PER) und digitale Ausgangswerte (Output\_UP, Output\_DN) zur Verfügung. Welchen Ausgangswert Sie verwenden, hängt von Ihrem Stellglied ab.

- Output\_PER  
Das Stellglied wird über einen analogen Ausgang angesprochen und mit einem kontinuierlichen Signal gesteuert, z. B. 0...10V, 4...20mA.
- Output\_UP, Output\_DN  
Das Stellglied wird über zwei Digitalausgänge gesteuert.

## Vorgehen

Um den analogen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output (analog)".
2. Wählen Sie "Anweisung".
3. Geben Sie die Adresse des Analogausgangs ein.

Um den digitalen Ausgangswert zu verwenden, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den Eintrag "Output (digital)".
2. Wählen Sie für Output\_UP und Output\_DN "Anweisung".
3. Geben Sie die Adressen der Digitalausgänge ein.

Um den Ausgangswert über das Anwenderprogramm aufzubereiten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Output" den zum Stellglied passenden Eintrag.
2. Wählen Sie "Anweisung".
3. Geben Sie den Namen der Variablen an, die Sie für die Aufbereitung des Ausgangswerts verwenden.
4. Übertragen Sie den aufbereiteten Ausgangswert über einen Analog- oder Digitalausgang der CPU zum Stellglied.

## IstwertEinstellung

Konfigurieren Sie im Konfigurationsfenster "IstwertEinstellungen" die Normierung Ihres Istwerts und legen Sie die absoluten Istwertgrenzen fest.

## Istwert skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Input\_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in die physikalische Größe des Istwerts umrechnen. Im Anzeigefeld Input\_PER wird die aktuelle Konfiguration angezeigt.

Wenn der Istwert direkt proportional zum Wert des Analogeingangs ist, wird Input\_PER anhand eines unteren und oberen Wertepaars skaliert.

1. Geben Sie in den Eingabefeldern "Skalierter unterer Istwert" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Skalierter oberer Istwert" und "Oben" das obere Wertepaar ein.

In der HW-Konfiguration sind Voreinstellungen für die Wertepaare hinterlegt. Um die Wertepaare aus der HW-Konfiguration zu verwenden, gehen Sie folgendermaßen vor:

1. Markieren Sie im Programmiereditor die Anweisung PID\_3Step.
2. Verschalten Sie in den Grundeinstellungen Input\_PER mit einem Analogeingang.
3. Klicken Sie in den Istwerteinstellungen auf die Schaltfläche "Automatische Einstellung". Die bestehenden Werte werden mit den Werten aus der HW-Konfiguration überschrieben.

### Istwert überwachen

Legen Sie die absolute Ober- und Untergrenze des Istwerts fest. Als Grenzwerte müssen Sie für Ihre Regelstrecke sinnvolle Werte eingeben. Während der Optimierung sind sinnvolle Grenzwerte wichtig, um optimale PID-Parameter zu erhalten. Die Vorbelegung der "Obergrenze Istwert" beträgt 120 %. Am Peripherie-Eingang kann der Istwert maximal 18 % über dem Normbereich (Übersteuerungsbereich) liegen. Wegen einer Überschreitung der "Obergrenze Istwert" wird mit dieser Einstellung kein Fehler mehr gemeldet. Nur Drahtbruch und Kurzschluss werden erkannt und PID\_3Step verhält sich wie unter Verhalten im Fehlerfall konfiguriert.

#### **ACHTUNG**

##### **Ihre Anlage kann beschädigt werden.**

Wenn Sie als Istwertgrenzen sehr hohe Werte (z. B.  $-3,4 \cdot 10^{38} \dots +3,4 \cdot 10^{38}$ ) einstellen, wird die Überwachung des Istwerts deaktiviert. Dann kann es durch einen Fehler zu Schäden an Ihrer Anlage kommen. Konfigurieren Sie für Ihre Regelstrecke sinnvolle Istwertgrenzen.

### Stellgliedeinstellung

#### Stellglied spezifische Zeiten

Um das Stellglied vor Beschädigung zu schützen, konfigurieren Sie die Motorstellzeit, die minimale Einschaltzeit und die minimale Ausschaltzeit. Die Daten finden Sie im Datenblatt des Stellglieds.

Die Motorstellzeit ist die Zeit in Sekunden, die der Motor benötigt, um das Stellglied vom geschlossenen in den geöffneten Zustand zu bewegen. Das Stellglied wird maximal 110% der Motorstellzeit in eine Richtung bewegt. Die Motorstellzeit können Sie während der Inbetriebnahme messen.

Wenn Sie Output\_UP und Output\_DN verwenden, verringern Sie die Schalzhäufigkeit mit der minimalen Ein- und Ausschaltzeit.

Im Automatikbetrieb werden die berechneten Ein- oder Ausschaltzeiten summiert und erst wirksam, wenn die Summe größer oder gleich der minimalen Ein- oder Ausschaltzeit ist.

Im Handbetrieb wird durch eine steigende Flanke an Manual\_UP oder Manual\_DN das Stellglied mindestens um die minimale Ein- oder Ausschaltzeit angesteuert.

## Verhalten im Fehlerfall

PID\_3Step ist so voreingestellt, dass im Fehlerfall die Regelung in den meisten Fällen aktiv bleibt. Wenn im Regelbetrieb häufig Fehler auftreten, wird durch diese Voreinstellung das Regelverhalten verschlechtert. Überprüfen Sie dann den Parameter Errorbits und beheben Sie die Fehlerursache.

Im Fehlerfall gibt PID\_3Step einen konfigurierbaren Ausgangswert aus:

- Aktueller Wert  
PID\_3Step ist ausgeschaltet und verändert die Position des Stellglieds nicht mehr.
- Aktueller Wert für die Fehlerdauer  
Die Reglerfunktionen von PID\_3Step sind ausgeschaltet und die Position des Stellglieds wird nicht mehr verändert.  
Wenn im Automatikbetrieb folgende Fehler aufgetreten sind, kehrt PID\_3Step in den Automatikbetrieb zurück, sobald die Fehler nicht mehr anstehen.
  - 0002h: Ungültiger Wert am Parameter Input\_PER.
  - 0200h: Ungültiger Wert am Parameter Input.
  - 0800h: Abtastzeitfehler
  - 1000h: Ungültiger Wert am Parameter Setpoint.
  - 2000h: Ungültiger Wert am Parameter Feedback\_PER.
  - 4000h: Ungültiger Wert am Parameter Feedback.
  - 8000h: Fehler bei der digitalen Stellungsrückmeldung.

Wenn im Handbetrieb einer dieser Fehler auftritt, bleibt PID\_3Step im Handbetrieb. Wenn während der Optimierung oder Stellzeitmessung ein Fehler auftritt, wird PID\_3Step ausgeschaltet.

- Ersatzausgangswert  
PID\_3Step fährt das Stellglied auf den Ersatzausgangswert und schaltet sich aus.
- Ersatzausgangswert für die Fehlerdauer  
PID\_3Step fährt das Stellglied auf den Ersatzausgangswert. Nach dem Erreichen des Ersatzausgangswerts verhält sich PID\_3Step wie bei "Aktueller Wert für die Fehlerdauer" beschrieben.

Den Ersatzausgangswert tragen Sie in "%" ein.

Bei Stellgliedern ohne analoge Stellungsrückmeldung können nur die Ersatzausgangswerte 0 % und 100 % exakt angefahren werden. Damit der obere oder untere Anschlag erreicht wird, wird das Stellglied mit 110% der Motorstellzeit in eine Richtung gefahren. Die Anschlagssignale werden vorrangig berücksichtigt. Ein Ersatzausgangswert ungleich 0 % oder 100 % wird über eine intern simulierte Stellungsrückmeldung angefahren. Mit diesem Verfahren kann der Ersatzausgangswert aber nie exakt angefahren werden.

Bei Stellgliedern mit analoger Stellungsrückmeldung können alle Ersatzausgangswerte exakt angefahren werden.

### Stellungsrückmeldung skalieren

Wenn Sie in der Grundeinstellung die Verwendung von Feedback\_PER konfiguriert haben, dann müssen Sie den Wert des Analogeingangs in % umrechnen. Im Anzeigefeld "Feedback" wird die aktuelle Konfiguration angezeigt.

Feedback\_PER wird anhand eines unteren und oberen Wertepaars skaliert.

1. Geben Sie in den Eingabefeldern "Unterer Anschlag" und "Unten" das untere Wertepaar ein.
2. Geben Sie in den Eingabefeldern "Oberer Anschlag" und "Oben" das obere Wertepaar ein.

"Unterer Anschlag" muss kleiner sein als "Oberer Anschlag"; "Unten" muss kleiner sein als "Oben".

Die gültigen Werte für "Oberer Anschlag" und "Unterer Anschlag" hängen ab von:

- Kein Feedback, Feedback, Feedback\_PER
- Output (analog), Output (digital)

| Output           | Feedback      | Unterer Anschlag          | Oberer Anschlag             |
|------------------|---------------|---------------------------|-----------------------------|
| Output (digital) | Kein Feedback | nicht einstellbar (0.0 %) | nicht einstellbar (100.0 %) |
| Output (digital) | Feedback      | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |
| Output (digital) | Feedback_PER  | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |
| Output (analog)  | Kein Feedback | nicht einstellbar (0.0 %) | nicht einstellbar (100.0 %) |
| Output (analog)  | Feedback      | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |
| Output (analog)  | Feedback_PER  | -100.0 % oder 0.0 %       | 0.0 % oder +100.0 %         |

### Ausgangswert begrenzen

Nur während der Stellzeitmessung können die Ausgangswertgrenzen über- oder unterschritten werden. In allen anderen Betriebsarten wird der Ausgangswert auf diese Werte begrenzt.

Geben Sie in den Eingabefeldern "Obergrenze Ausgangswert" und "Untergrenze Ausgangswert" die absoluten Ausgangswertgrenzen ein. Die Ausgangswertgrenzen müssen innerhalb "Unterer Anschlag" und "Oberer Anschlag" liegen.

Wenn kein Feedback vorhanden ist und Output (digital) eingestellt ist, können Sie den Ausgangswert nicht begrenzen. Die Digitalausgänge entweder bei Actuator\_H = TRUE oder Actuator\_L = TRUE oder nach einer Verfahzeit von 110% der Motorstellzeit zurück gesetzt.

## Erweiterte Einstellungen

### Istwertüberwachung

Konfigurieren Sie im Konfigurationsfenster "Istwertüberwachung" eine untere und eine obere Warngrenze des Istwerts. Wird im Betrieb eine der Warngrenzen über oder unterschritten, so wird an der Anweisung PID\_3Step eine Warnung angezeigt:

- Am Ausgangsparameter InputWarning\_H, wenn die obere Warngrenze überschritten wurde
- Am Ausgangsparameter InputWarning\_L, wenn die untere Warngrenze unterschritten wurde

Die Warngrenzen müssen innerhalb der Ober- und Untergrenze Istwert liegen.

Wenn Sie keine Werte eingeben, werden die Ober- und Untergrenze Istwert verwendet.

### Beispiel

Obergrenze Istwert = 98 °C ; Obere Warngrenze = 90 °C

Untere Warngrenze = 10 °C ; Untergrenze Istwert = 0 °C

PID\_3Step verhält sich folgendermaßen:

| Istwert             | InputWarning_H | InputWarning_L | Betriebsart      |
|---------------------|----------------|----------------|------------------|
| > 98 °C             | TRUE           | FALSE          | Inaktiv          |
| ≤ 98 °C und > 90 °C | TRUE           | FALSE          | Automatikbetrieb |
| ≤ 90 °C und ≥ 10 °C | FALSE          | FALSE          | Automatikbetrieb |
| < 10 °C und ≥ 0 °C  | FALSE          | TRUE           | Automatikbetrieb |
| < 0 °C              | FALSE          | TRUE           | Inaktiv          |

### PID-Parameter

Im Konfigurationsfenster "PID-Parameter" werden die PID-Parameter angezeigt. Während der Optimierung werden die PID-Parameter an Ihre Regelstrecke angepasst. Sie brauchen die PID-Parameter nicht manuell eingeben.

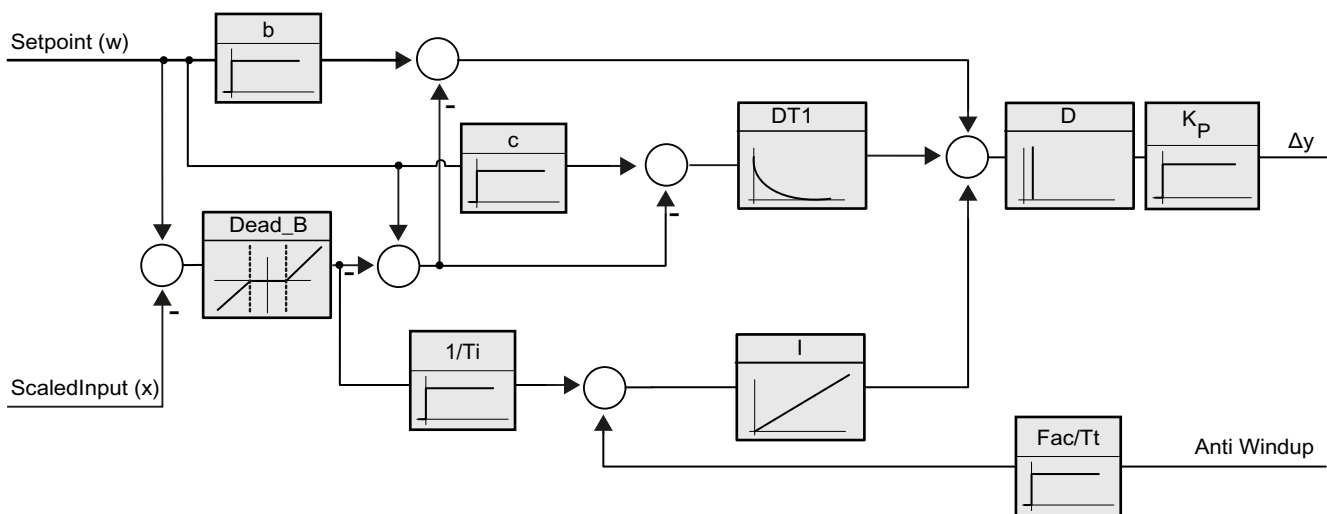
Der PID-Algorithmus arbeitet nach folgender Formel:

$$\Delta y = K_p \cdot s \cdot \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

| Symbol     | Beschreibung                     |
|------------|----------------------------------|
| $\Delta y$ | Ausgangswert des PID-Algorithmus |
| $K_p$      | Proportionalverstärkung          |

|       |   |
|-------|---|
| s     | Laplace-Operator  |
| b     | Gewichtung des P-Anteils  |
| w     | Sollwert  |
| x     | Istwert   |
| $T_I$ | Integrationszeit  |
| a     | Koeffizient für den Differenzierverzug (Differenzierverzug $T_1 = a \times T_D$ ) |
| $T_D$ | Differenzierzeit  |
| c     | Gewichtung des D-Anteils  |

Die folgende Grafik zeigt, wie die Parameter in den PID-Algorithmus einfließen:



Alle PID-Parameter sind remanent. Wenn Sie die PID-Parameter manuell eingeben, müssen Sie PID\_3Step vollständig laden.

Auto-Hotspot

### Proportionalverstärkung

Der Wert gibt die Proportionalverstärkung des Reglers an. PID\_3Step arbeitet nicht mit einer negativen Proportionalverstärkung. Den Regelsinn invertieren Sie unter Grundeinstellungen > Regelungsart.

### Integrationszeit

Die Integrationszeit bestimmt das Zeitverhalten des I-Anteils. Abschalten des I-Anteils erfolgt mit Integrationszeit = 0.0.

### Differenzierzeit

Die Differenzierzeit bestimmt das Zeitverhalten des D-Anteils. Abschalten des D-Anteils erfolgt mit Differenzierzeit = 0.0.

### Koeffizient Differenzierversug

Die Wirkung des D-Anteils wird durch den Koeffizient Differenzierversug verzögert.

Differenzierversug = Differenzierzeit × Koeffizient Differenzierversug

- 0.0: D-Anteil wirkt nur für einen Zyklus und ist damit fast nicht wirksam.
- 0.5: Dieser Wert hat sich in der Praxis für Regelstrecken mit **einer** dominierenden Zeitkonstanten bewährt.
- > 1.0: Je größer der Koeffizient, desto stärker wird die Wirkung des D-Anteils verzögert.

### Gewichtung des P-Anteils

Sie können bei Sollwertänderungen den P-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: P-Anteil bei Sollwertänderung voll wirksam
- 0.0: P-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der P-Anteil immer voll wirksam.

### Gewichtung des D-Anteils

Sie können bei Sollwertänderungen den D-Anteil abschwächen.

Sinnvoll sind Werte von 0.0 bis 1.0.

- 1.0: D-Anteil bei Sollwertänderung voll wirksam
- 0.0: D-Anteil bei Sollwertänderung nicht wirksam

Bei Änderung des Istwerts ist der D-Anteil immer voll wirksam.

### Abtastzeit PID-Algorithmus

Da die Regelstrecke eine gewisse Zeit benötigt, um auf eine Änderung des Ausgangswerts zu reagieren, ist es sinnvoll, den Ausgangswert nicht in jedem Zyklus zu berechnen. Die Abtastzeit PID-Algorithmus ist die Zeit zwischen zwei Ausgangswertberechnungen. Sie wird während der Optimierung ermittelt und auf ein Vielfaches der Abtastzeit PID\_3Step gerundet. Alle anderen Funktionen von PID\_3Step werden bei jedem Aufruf durchgeführt.

### Totzonbreite

Die Totzone unterdrückt die Rauschanteile im eingeschwungenen Reglerzustand. Die Totzonbreite gibt die Größe der Totzone an. Bei einer Totzonbreite von 0.0 ist die Totzone abgeschaltet.

### Siehe auch

Technologieobjekte in Gerät laden (Seite 4056)

## PID\_3Step V1 in Betrieb nehmen

### Inbetriebnahme

Im Arbeitsbereich "Optimierung" können Sie Sollwert, Istwert und Ausgangswert in Abhängigkeit von der Zeit beobachten. Folgende Funktionen zur Inbetriebnahme werden im Kurvenschreiber unterstützt:

- Ersoptimierung des Reglers
- Nachoptimierung des Reglers
- Beobachtung der laufenden Regelung im Kurvenfenster

Für alle Funktionen muss eine Onlineverbindung zur CPU aufgebaut sein.

### Grundsätzliche Bedienung

- Wählen Sie in der Klappliste "Aktualisierungszeit" die gewünschte Aktualisierungszeit aus. Alle Werte im Arbeitsbereich Optimierung werden in der ausgewählten Aktualisierungszeit aktualisiert.
- Klicken Sie auf das Symbol "Start" in der Gruppe Messung, wenn Sie die Inbetriebnahmefunktionen nutzen möchten. Die Aufzeichnung der Werte wird gestartet. In der Kurvenanzeige werden die aktuellen Werte für Sollwert, Istwert und Ausgangswert eingetragen. Die Bedienung des Inbetriebnahmefensters wird freigegeben.
- Klicken Sie auf das Symbol "Stop", wenn Sie die Inbetriebnahmefunktionen beenden möchten. Die in der Kurvenanzeige aufgezeichneten Werte können weiterhin analysiert werden.
- Mit Schließen des Inbetriebnahmefensters wird die Aufzeichnung in der Kurvenanzeige beendet und die aufgezeichneten Werte werden gelöscht.

### Ersoptimierung

Die Ersoptimierung ermittelt die Prozessantwort auf einen Impuls des Ausgangswerts und sucht den Wendepunkt. Aus der maximalen Steigung und der Totzeit der Regelstrecke werden die optimalen PID-Parameter berechnet.

Je stabiler der Istwert ist, desto leichter und genauer können die PID-Parameter ermittelt werden. Ein Rauschen des Istwerts ist solange akzeptabel, wie der Anstieg des Istwerts signifikant größer ist als das Rauschen. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Der Sollwert wird während der Ersoptimierung eingefroren.

### Voraussetzung

- Die Anweisung PID\_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE



- PID\_3Step befindet sich in der Betriebsart "Inaktiv" oder "Handbetrieb".
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").

## Vorgehen

Um die Erstopptimierung durchzuführen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID\_3Step > Inbetriebnahme".
2. Wählen Sie im Arbeitsbereich "Optimierung" in der Klappliste "Optimierungsart" den Eintrag "Erstopptimierung".
3. Klicken Sie auf das Symbol "Start".
  - Eine Online-Verbindung wird aufgebaut.
  - Die Aufzeichnung der Werte wird gestartet.
  - Die Erstopptimierung wird gestartet.
  - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

---

### Hinweis

Klicken Sie auf das Symbol "Stop" wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

---

## Ergebnis

Wurde die Erstopptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID\_3Step wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn eine Erstopptimierung nicht möglich ist, wechselt PID\_3Step in die Betriebsart "Inaktiv".

## Nachoptimierung

Die Nachoptimierung generiert eine konstante, begrenzte Schwingung des Istwertes. Aus Amplitude und Frequenz dieser Schwingung werden die PID-Parameter für den Arbeitspunkt optimiert. Aus den Ergebnissen werden alle PID-Parameter neu berechnet. Die PID-Parameter aus der Nachoptimierung zeigen meist ein besseres Führungs- und Störverhalten als die PID-Parameter aus der Erstopptimierung.

PID\_3Step versucht automatisch eine Schwingung zu erzeugen, die größer ist als das Rauschen des Istwertes. Die Nachoptimierung wird nur geringfügig von der Stabilität des Istwertes beeinflusst. Die PID-Parameter werden gesichert bevor sie neu berechnet werden.

Der Sollwert wird während der Nachoptimierung eingefroren.

### Voraussetzung

- Die Anweisung PID\_3Step wird in einem Weckalarm-OB aufgerufen.
- ManualEnable = FALSE
- Die Motorstellzeit ist konfiguriert oder gemessen.
- Der Sollwert und der Istwert befinden sich innerhalb der konfigurierten Grenzen (siehe Konfiguration "Istwerteinstellungen").
- Der Regelkreis ist am Arbeitspunkt eingeschwungen. Der Arbeitspunkt ist erreicht, wenn der Istwert dem Sollwert entspricht.
- Es werden keine Störungen erwartet.
- PID\_3Step befindet sich in der Betriebsart Inaktiv, Automatikbetrieb oder Handbetrieb.

### Ablauf abhängig von Startsituation

Die Nachoptimierung verläuft folgendermaßen beim Start aus:

- Automatikbetrieb  
Wenn Sie die vorhandenen PID-Parameter durch die Optimierung verbessern wollen, starten Sie die Nachoptimierung aus dem Automatikbetrieb.  
PID\_3Step regelt solange mit den vorhandenen PID-Parametern, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.
- Inaktiv oder Handbetrieb  
Zuerst wird immer eine Erstopoptimierung gestartet. Mit den ermittelten PID-Parametern wird solange geregelt, bis der Regelkreis eingeschwungen ist und die Voraussetzungen für eine Nachoptimierung erfüllt sind. Erst dann startet die Nachoptimierung.

### Vorgehen

Um die "Nachoptimierung" durchzuführen gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste "Optimierungsart" den Eintrag "Nachoptimierung".
2. Klicken Sie auf das Symbol "Start".
  - Eine Online-Verbindung wird aufgebaut.
  - Die Aufzeichnung der Werte wird gestartet.
  - Der Ablauf der Nachoptimierung wird gestartet.
  - Im Feld "Status" werden Ihnen die aktuellen Arbeitsschritte und evtl. auftretende Fehler angezeigt. Der Fortschrittsbalken zeigt den Fortschritt des aktuellen Arbeitsschritts an.

---

#### Hinweis

Klicken Sie in der Gruppe "Optimierungsart" auf das Symbol "Stop", wenn der Fortschrittsbalken 100% erreicht hat und von einer Blockade der Optimierung ausgegangen werden muss. Prüfen Sie die Konfiguration des Technologieobjekts und starten Sie ggf. die Optimierung erneut.

---

## Ergebnis

Wurde die Nachoptimierung ohne Fehlermeldung durchlaufen, so wurden die PID-Parameter optimiert. PID\_3Step wechselt in den Automatikbetrieb und verwendet die optimierten Parameter. Die optimierten PID-Parameter bleiben bei Netz-AUS und Neustart der CPU erhalten.

Wenn während der Nachoptimierung Fehler auftraten, wechselt PID\_3Step in die Betriebsart "Inaktiv".

## Mit manuellen PID-Parametern in Betrieb nehmen

### Vorgehen

Um PID\_3Step mit manuellen PID-Parametern in Betrieb zu nehmen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID\_3Step > Konfiguration".
2. Klicken Sie im Konfigurationsfenster auf "Erweiterte Einstellungen > PID-Parameter".
3. Aktivieren Sie das Optionskästchen "Manuelle Eingabe aktivieren".
4. Geben Sie die PID-Parameter ein.
5. Doppelklicken Sie in der Projektnavigation auf den Eintrag "PID\_3Step > Inbetriebnahme".
6. Stellen Sie eine Online-Verbindung zur CPU her.
7. Laden Sie die PID-Parameter auf die CPU.
8. Klicken Sie auf das Symbol "Regler aktivieren".

## Ergebnis

PID\_3Step wechselt in den Automatikbetrieb und regelt mit den aktuellen PID-Parametern.

## Motorstellzeit messen

### Einleitung

PID\_3Step benötigt die Motorstellzeit so genau wie möglich, um ein gutes Regelergebnis zu erreichen. Die Angaben in der Dokumentation des Stellglieds sind gemittelte Werte für diesen Typ Stellglieder. Für das konkret verwendete Stellglied kann der Wert variieren.


Wenn Sie Stellglieder mit Stellungsrückmeldung oder mit Anschlagsignalen verwenden, können Sie während der Inbetriebnahme die Motorstellzeit messen. Die Ausgangswertgrenzen werden während der Motorstellzeitmessung nicht berücksichtigt. Das Stellglied kann bis zum oberen oder unteren Anschlag verfahren werden.

Wenn weder eine Stellungsrückmeldung noch Anschlagsignale verfügbar sind, kann die Motorstellzeit nicht gemessen werden.

## Stellglieder mit analoger Stellungsrückmeldung

Um die Motorstellzeit mit Stellungsrückmeldung zu messen, gehen Sie folgendermaßen vor:

### Voraussetzung

- In den Grundeinstellungen ist Feedback oder Feedback\_PER ausgewählt und das Signal ist verschaltet.
  - Eine Online-Verbindung zur CPU ist aufgebaut.
1. Aktivieren Sie das Optionsfeld "Stellungsrückmeldung verwenden".
  2. Geben Sie im Eingabefeld "Zielstellung" ein, wohin das Stellglied bewegt werden soll. Die aktuelle Stellungsrückmeldung (Startstellung) wird angezeigt. Die Differenz zwischen "Zielstellung" und "Stellungsrückmeldung" muss mindestens 50 % des zulässigen Ausgangswertbereichs sein.
  3. Klicken Sie auf das Symbol  "Stellzeitmessung starten".

## Ergebnis


Das Stellglied wird von der Startstellung in die Zielstellung gefahren. Die Zeitmessung startet sofort und endet, wenn das Stellglied die Zielstellung erreicht hat. Die Motorstellzeit wird berechnet nach der Formel:

$$\text{Motorstellzeit} = (\text{Obergrenze Ausgangswert} - \text{Untergrenze Ausgangswert}) \times \text{Messzeit} / \text{BETRAG}(\text{Zielstellung} - \text{Startstellung}).$$

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Instanz-Datenbaustein auf der CPU gespeichert und im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wechselt PID\_3Step in die Betriebsart "Inaktiv".

---

### Hinweis

Um die gemessene Motorstellzeit in das Projekt zu übernehmen, klicken Sie auf das Symbol  "Gemessene Stellzeit laden".

---


## Stellglieder mit Anschlagssignalen

Um die Stellzeit von Stellgliedern mit Anschlagssignalen zu messen, gehen Sie folgendermaßen vor:

### Voraussetzung

- In den Grundeinstellungen ist das Optionskästchen "Anschlagssignale" aktiviert und Actuator\_H und Actuator\_L sind verschaltet.
- Eine Online-Verbindung zur CPU ist aufgebaut.

Um die Motorstellzeit mit Anschlagssignalen zu messen, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionsfeld "Anschlagsignale des Stellglieds verwenden".
2. Wählen Sie die Richtung, in die das Stellglied bewegt werden soll.
  - Öffnen - Schließen - Öffnen  
Das Stellglied wird zuerst bis zum oberen Anschlag gefahren, dann zum unteren Anschlag und erneut zum oberen Anschlag.
  - Schließen - Öffnen - Schließen  
Das Stellglied wird zuerst bis zum unteren Anschlag gefahren, dann zum oberen Anschlag und erneut zum unteren Anschlag.
3. Klicken Sie auf das Symbol  "Stellzeitmessung starten".

## Ergebnis

Das Stellglied wird in der gewählten Richtung bewegt. Die Zeitmessung startet, wenn das Stellglied den ersten Anschlag erreicht hat und endet, wenn das Stellglied diesen Anschlag zum zweiten Mal erreicht. Die gemessene Zeit geteilt durch zwei ergibt die Motorstellzeit.

Fortschritt und Status der Stellzeitmessung werden angezeigt. Die gemessene Stellzeit wird im Instanz-Datenbaustein auf der CPU gespeichert und im Feld "gemessene Stellzeit" angezeigt. Wenn die Stellzeitmessung beendet ist, wechselt PID\_3Step in die Betriebsart "Inaktiv".

## Stellzeitmessung abbrechen

Wenn Sie die Stellzeitmessung abbrechen, wechselt PID\_3Step sofort in die Betriebsart "Inaktiv". Das Stellglied wird nicht mehr bewegt. Sie können PID\_3Step im Kurvenschreiber wieder aktivieren.

## 11.2 S7-1200 Motion Control einsetzen

### 11.2.1 Einleitung

#### 11.2.1.1 Motionfunktionalität der CPU S7-1200

Das TIA-Portal unterstützt Sie zusammen mit der "Motion Control" Funktionalität der CPU S7-1200 bei der Steuerung von Schrittmotoren und Servomotoren mit Impulsschnittstelle:

- Im TIA-Portal konfigurieren Sie das Technologieobjekt "Achse" und "Auftragstabelle". Mit Hilfe dieser Technologieobjekte steuert die CPU S7-1200 die Impuls- und Richtungsausgänge zur Ansteuerung der Antriebe.
- Im Anwenderprogramm steuern Sie mit Hilfe von Motion Control-Anweisungen die Achse und initiieren hiermit Bewegungsaufträge Ihres Antriebs.

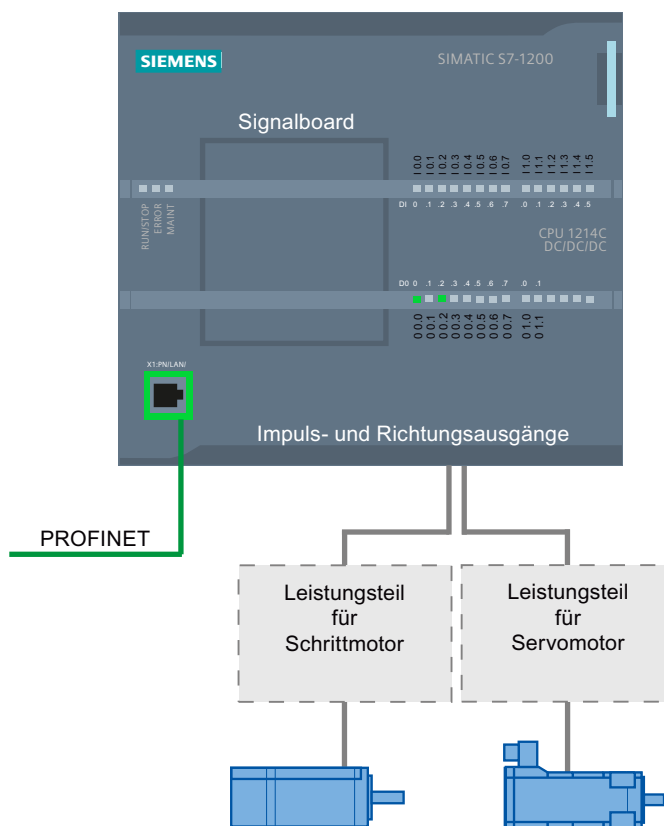
Eine Multimedia-Einführung finden Sie im Internet (<http://www.automation.siemens.com/mcms/topics/en/simatic/simatic-technology/integrated-functions/simatic-s7-1200/Pages/Default.aspx>).

**Siehe auch**

- Hardwarekomponenten für Motion Control (Seite 4128)
- Einbindung des Technologieobjekts Achse (Seite 4141)
- Verwendung des Technologieobjekts Auftragstabelle (Seite 4170)
- Werkzeuge des Technologieobjekts Auftragstabelle (Seite 4171)

**11.2.1.2 Hardwarekomponenten für Motion Control**

Die nachfolgende Darstellung zeigt den prinzipiellen Hardwareaufbau für eine Motion Control-Anwendung mit der CPU S7-1200.



**CPU S7-1200**

Die CPU S7-1200 vereint die Funktionalität einer speicherprogrammierbaren Steuerung mit der Motion Control-Funktionalität für den Betrieb von Schrittmotoren und Servomotoren mit Impulsschnittstelle. Die Motion Control-Funktionalität übernimmt die Steuerung und Überwachung der Antriebe.

Die DC/DC/DC Varianten der CPU S7-1200 verfügen über On-board Ausgänge zur direkten Ansteuerung der Antriebe. Die Relais-Varianten der CPU benötigen zur Ansteuerung eines Antriebs eines der nachfolgend beschriebenen Signalboards.

## Signalboard

Mit den Signalboards erweitern Sie die CPU um weitere Ein- und Ausgänge. Die digitalen Ausgänge können bei Bedarf als Impuls- und Richtungsausgänge zur Ansteuerung von Antrieben verwendet werden.

Bei CPUs mit Relais-Ausgängen kann das Impulssignal nicht auf den on-board Ausgängen ausgegeben werden, da die Relais die erforderlichen Schaltfrequenzen nicht unterstützen. Um auf diesen CPUs den PTO (Pulse Train Output) nutzen zu können, muss ein Signalboard mit digitalen Ausgängen verwendet werden.

Beim Einsatz einer DC/DC/DC Variante der CPU S7-1200 zusammen mit einem Signalboard bleibt die maximale Anzahl der ansteuerbaren Antriebe auf "2 / 4" (MLFB - Bestellnummer xxxxxxx-1xx30-xxxx / xxxxxxx-1xx31-xxxx) begrenzt.

## PROFINET

Über die PROFINET-Schnittstelle stellen Sie die Onlineverbindung zwischen der CPU S7-1200 und dem Programmiergerät her. Neben den Onlinefunktionen der CPU stehen Ihnen für Motion Control zusätzliche Inbetriebnahme- und Diagnosefunktionen zur Verfügung.

## Maximale Anzahl ansteuerbarer Antriebe

Die maximale Anzahl der ansteuerbaren Antriebe für die unterschiedlichen CPU-Ausprägungen entnehmen Sie der nachfolgenden Tabelle:

| CPU  |           | Signalboard |                       |                        |                    |                       |                   |
|--|-----------|-------------|-----------------------|------------------------|--------------------|-----------------------|-------------------|
|  |           | Ohne        | DI2/DO2 x DC24V 20kHz | DI2/DO2 x DC24V 200kHz | DO4 x DC24V 200kHz | DI2/DO2 x DC5V 200kHz | DO4 x DC5V 200kHz |
| CPU 1211C, CPU 1212C, CPU 1214C<br>(MLFB - Bestellnummer xxxxxxx-1xx30-xxxx) | DC/DC/DC  | 2           | 2                     | 2                      | 2                  | 2                     | 2                 |
|  | AC/DC/RLY | -           | 1                     | 1                      | 2                  | 1                     | 2                 |
|  | DC/DC/RLY | -           | 1                     | 1                      | 2                  | 1                     | 2                 |
| CPU 1211C<br>(MLFB - Bestellnummer xxxxxxx-1xx31-xxxx)                       | DC/DC/DC  | 2           | 3                     | 3                      | 4                  | 3                     | 4                 |
|  | AC/DC/RLY | -           | 1                     | 1                      | 2                  | 1                     | 2                 |
|  | DC/DC/RLY | -           | 1                     | 1                      | 2                  | 1                     | 2                 |
| CPU 1212C<br>(MLFB - Bestellnummer xxxxxxx-1xx31-xxxx)                       | DC/DC/DC  | 3           | 4                     | 4                      | 4                  | 4                     | 4                 |
|  | AC/DC/RLY | -           | 1                     | 1                      | 2                  | 1                     | 2                 |
|  | DC/DC/RLY | -           | 1                     | 1                      | 2                  | 1                     | 2                 |
| CPU 1214C<br>(MLFB - Bestellnummer xxxxxxx-1xx31-xxxx)                       | DC/DC/DC  | 4           | 4                     | 4                      | 4                  | 4                     | 4                 |

| CPU       |           | Signalboard |   |   |   |   |   |
|-----------|-----------|-------------|---|---|---|---|---|
|           | AC/DC/RLY | -           | 1 | 1 | 2 | 1 | 2 |
|           | DC/DC/RLY | -           | 1 | 1 | 2 | 1 | 2 |
| CPU 1215C | DC/DC/DC  | 4           | 4 | 4 | 4 | 4 | 4 |
|           | AC/DC/RLY | -           | 1 | 1 | 2 | 1 | 2 |
|           | DC/DC/RLY | -           | 1 | 1 | 2 | 1 | 2 |

### Grenzfrequenzen der Impulsausgänge

Für die Impulsausgänge gelten folgende Grenzfrequenzen:

| Impulsausgang                                      | Grenzfrequenzen für Technologieobjekt "Achse" V1.0  | Grenzfrequenzen ab Technologieobjekt "Achse" V2.0   |
|--|---|---|
| On-board (MLFB - Bestellnummer xxxxxxx-1xx30-xxxx) | $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$  | $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$  |
| On-board (MLFB - Bestellnummer xxxxxxx-1xx31xxxx)  | $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$ (PTO 1+2)<br>$2 \text{ Hz} \leq f \leq 20 \text{ kHz}$ (PTO 3+4) | $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$ (PTO 1+2)<br>$2 \text{ Hz} \leq f \leq 20 \text{ kHz}$ (PTO 3+4) |
| Signalboard DI2/DO2 x DC24V 20kHz                  | $2 \text{ Hz} \leq f \leq 20 \text{ kHz}$   | $2 \text{ Hz} \leq f \leq 20 \text{ kHz}$   |
| Signalboard DI2/DO2 x DC24V 200kHz                 | $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$  | $2 \text{ Hz} \leq f \leq 200 \text{ kHz}$  |
| Signalboard DO4 x DC24V 200kHz                     | $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$  | $2 \text{ Hz} \leq f \leq 200 \text{ kHz}$  |
| Signalboard DI2/DO2 x DC5V 200kHz                  | $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$  | $2 \text{ Hz} \leq f \leq 200 \text{ kHz}$  |
| Signalboard DO4 x DC5V 200kHz                      | $2 \text{ Hz} \leq f \leq 100 \text{ kHz}$  | $2 \text{ Hz} \leq f \leq 200 \text{ kHz}$  |

### Bestellinformationen

Die nachfolgend aufgelisteten Bestellinformationen gelten für die aktuell installierte Lieferstufe (ohne evt. installierte Hardware Support Packages) des TIA-Portals.

| Bezeichnung         | MLFB - Bestellnummer |
|---------------------|----------------------|
| CPU 1211C DC/DC/DC  | 6ES7211-1Ax3x-0XB0   |
| CPU 1211C AC/DC/RLY | 6ES7211-1Bx3x-0XB0   |
| CPU 1211C DC/DC/RLY | 6ES7211-1Hx3x-0XB0   |
| CPU 1212C DC/DC/DC  | 6ES7212-1Ax3x-0XB0   |
| CPU 1212C AC/DC/RLY | 6ES7212-1Bx3x-0XB0   |
| CPU 1212C DC/DC/RLY | 6ES7212-1Hx3x-0XB0   |
| CPU 1214C DC/DC/DC  | 6ES7214-1Ax3x-0XB0   |
| CPU 1214C AC/DC/RLY | 6ES7214-1Bx3x-0XB0   |
| CPU 1214C DC/DC/RLY | 6ES7214-1Hx3x-0XB0   |
| CPU 1215C DC/DC/DC  | 6ES7215-1AG31-0XB0   |
| CPU 1215C AC/DC/RLY | 6ES7215-1BG31-0XB0   |



| Bezeichnung                        | MLFB - Bestellnummer |
|------------------------------------|----------------------|
| CPU 1215C DC/DC/RLY                | 6ES7215-1HG31-0XB0   |
| Signalboard DI2/DO2 x DC24V 20kHz  | 6ES7223-0BD30-0XB0   |
| Signalboard DI2/DO2 x DC24V 200kHz | 6ES7223-3BD30-0XB0   |
| Signalboard DO4 x DC24V 200kHz     | 6ES7222-1BD30-0XB0   |
| Signalboard DI2/DO2 x DC5V 200kHz  | 6ES7223-3AD30-0XB0   |
| Signalboard DO4 x DC5V 200kHz      | 6ES7222-1AD30-0XB0   |

Neuere Hardware-Komponenten können Sie über ein Hardware Support Package (HSP) installieren. Die Hardware-Komponente steht danach im Hardware-Katalog zur Verfügung.

### Siehe auch

Motionfunktionalität der CPU S7-1200 (Seite 4127)

Für Motion Control relevante Ausgänge der CPU (Seite 4131)

## 11.2.2 Grundlagen für das Arbeiten mit S7-1200 Motion Control

### 11.2.2.1 Für Motion Control relevante Ausgänge der CPU

#### Impuls- und Richtungsausgang

Zur Ansteuerung eines Schrittmotorantriebs oder eines Servomotorantriebs mit Impulsschnittstelle stellt die CPU einen Impulsausgang und einen Richtungsausgang zur Verfügung. Über den Impulsausgang erhält der Antrieb die notwendigen Impulse zur Bewegung des Motors. Der Richtungsausgang steuert die Fahrtrichtung des Antriebs.

Impuls- und Richtungsausgang sind zueinander fest zugeordnet. Als Impuls- und Richtungsausgänge können On-board CPU-Ausgänge oder Ausgänge eines Signalboards verwendet werden. Die Auswahl zwischen On-board CPU-Ausgängen und Ausgängen des Signalboards treffen Sie in der Gerätekonfiguration unter Impulsgeneratoren (PTO/PWM) im Register "Eigenschaften".

Die mögliche Adresszuordnung der Impuls- und Richtungsausgänge entnehmen Sie der nachfolgenden Tabelle:

| CPU S7-1200   | Ohne Signalboard |        |               |        | Signalboards<br>DI2/DO2 *) |        |               |        | Signalboards<br>DO4 **) |        |                  |        |
|---|------------------|--------|---------------|--------|----------------------------|--------|---------------|--------|-------------------------|--------|------------------|--------|
|   | Ausgänge PTO1    |        | Ausgänge PTO2 |        | Ausgänge PTO1              |        | Ausgänge PTO2 |        | Ausgänge<br>PTO1        |        | Ausgänge<br>PTO2 |        |
|   | Imp.             | Richt. | Imp.          | Richt. | Imp.                       | Richt. | Imp.          | Richt. | Imp.                    | Richt. | Imp.             | Richt. |
| CPU 1211C,<br>CPU 1212C,<br>CPU 1214C<br>(DC/DC/DC) | Ax.0             | Ax.1   | Ax.2          | Ax.3   | Ax.0                       | Ax.1   | Ax.2          | Ax.3   | Ax.0                    | Ax.1   | Ax.2             | Ax.3   |
|   |                  |        |               |        | Ay.0                       | Ay.1   |               |        | Ay.0                    | Ay.1   | Ay.2             | Ay.3   |

| CPU S7-1200  | Ohne Signalboard |        |               |        | Signalboards<br>DI2/DO2 *) |        |               |        | Signalboards<br>DO4 **) |        |                  |        |
|--|------------------|--------|---------------|--------|----------------------------|--------|---------------|--------|-------------------------|--------|------------------|--------|
|  | Ausgänge PTO1    |        | Ausgänge PTO2 |        | Ausgänge PTO1              |        | Ausgänge PTO2 |        | Ausgänge<br>PTO1        |        | Ausgänge<br>PTO2 |        |
|  | Imp.             | Richt. | Imp.          | Richt. | Imp.                       | Richt. | Imp.          | Richt. | Imp.                    | Richt. | Imp.             | Richt. |
| CPU 1211C,<br>CPU 1212C,<br>CPU 1214C<br>(AC/DC/RLY) | -                | -      | -             | -      | Ay.0                       | Ay.1   | -             | -      | Ay.0                    | Ay.1   | Ay.2             | Ay.3   |
| CPU 1211C,<br>CPU 1212C,<br>CPU 1214C<br>(DC/DC/RLY) | -                | -      | -             | -      | Ay.0                       | Ay.1   | -             | -      | Ay.0                    | Ay.1   | Ay.2             | Ay.3   |

x = Anfangsbyteadresse der On-board CPU-Ausgänge (Standardwert = 0)

y = Anfangsbyteadresse der Signalboard-Ausgänge (Standardwert = 4)

\* Wird eine CPU-Variante DC/DC/DC zusammen mit einem Signalboard DI2/DO2 eingesetzt, so können die Signale des PTO1 über die On-board CPU-Ausgänge oder über das Signalboard ausgegeben werden.

\*\* Wird eine CPU-Variante DC/DC/DC zusammen mit einem Signalboard DO4 eingesetzt, so können sowohl die Signale für PTO1 als auch für PTO2 über die On-board CPU-Ausgänge oder über das Signalboard ausgegeben werden.

## Antriebssignale

Für Motion Control können Sie optional eine Antriebsschnittstelle für "Antriebsfreigabe" und "Antrieb bereit" parametrieren. Bei Verwendung der Antriebsschnittstelle kann der digitale Ausgang für die Antriebsfreigabe und der digitale Eingang für "Antrieb bereit" frei gewählt werden.

### Hinweis

Wurde der PTO (Pulse Train Output) aktiviert und einer Achse zugeordnet, so übernimmt die Firmware die Kontrolle über den zugehörigen Impuls- und Richtungsausgang.

Mit der Übernahme der Kontrolle wird auch die Verbindung zwischen dem Prozessabbild und dem Peripherie-Ausgang getrennt. Der Anwender hat zwar die Möglichkeit das Prozessabbild von Impuls- und Richtungsausgang über Anwenderprogramm oder Beobachtungstabelle zu beschreiben, dieses wird jedoch nicht auf den Peripherie-Ausgang übertragen werden. Dementsprechend ist es auch nicht möglich den Peripherie-Ausgang über Anwenderprogramm oder Beobachtungstabelle zu beobachten. Die gelesenen Informationen spiegeln den Wert des Prozessabbilds wieder, welche nicht mit dem realen Zustand des Peripherie-Ausgangs übereinstimmen.

Bei allen anderen, nicht fest durch die CPU-Firmware verwendeten CPU-Ausgängen, kann der Status des Peripherie-Ausgangs wie gewohnt über das Prozessabbild gesteuert bzw. überwacht werden.

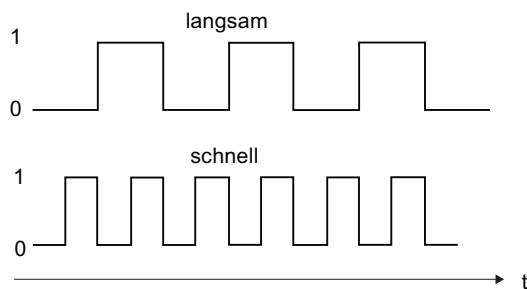
## Siehe auch

- Prinzip der Impulsschnittstelle (Seite 4133)
- Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang (Seite 4134)
- Hardware- und Software-Endschalter (Seite 4135)
- Ruckbegrenzung (Seite 4136)
- Referenzieren (Seite 4137)
- Hardwarekomponenten für Motion Control (Seite 4128)
- Einbindung des Technologieobjekts Achse (Seite 4141)
- Werkzeuge des Technologieobjekts Achse (Seite 4143)

### 11.2.2.2 Prinzip der Impulsschnittstelle

In Abhängigkeit zu den Einstellungen des Schrittmotors, hat jeder Impuls das Verfahren des Schrittmotors um einen definierten Winkel zur Folge. Ist der Schrittmotor auf beispielsweise 1000 Impulse pro Umdrehung eingestellt, verfährt der Schrittmotor pro Impuls um  $0,36^\circ$ .

Die Geschwindigkeit des Schrittmotors wird über die Anzahl der Impulse pro Zeiteinheit bestimmt.



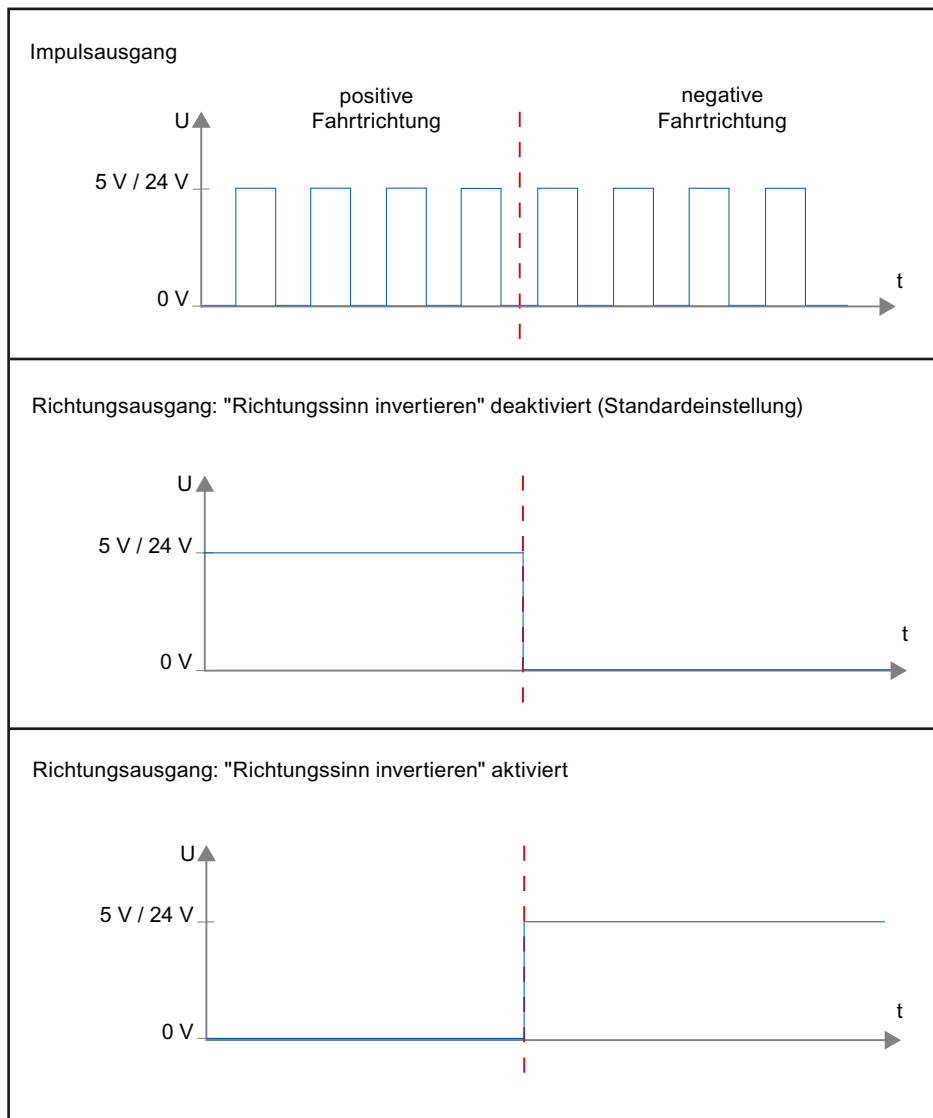
(Die hier getroffenen Aussagen gelten auch für Servomotoren mit Impulsschnittstelle)

## Siehe auch

- Für Motion Control relevante Ausgänge der CPU (Seite 4131)
- Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang (Seite 4134)
- Hardware- und Software-Endschalter (Seite 4135)
- Ruckbegrenzung (Seite 4136)
- Referenzieren (Seite 4137)
- Einbindung des Technologieobjekts Achse (Seite 4141)
- Werkzeuge des Technologieobjekts Achse (Seite 4143)

### 11.2.2.3 Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang

Der Richtungsausgang der CPU gibt die Fahrtrichtung des Antriebs vor. Den Richtungssinn konfigurieren Sie in der Konfiguration der Achse unter "Mechanik". Die Zusammenhänge zwischen Konfiguration, Richtungsausgang und Fahrtrichtung entnehmen Sie der nachfolgenden Darstellung:



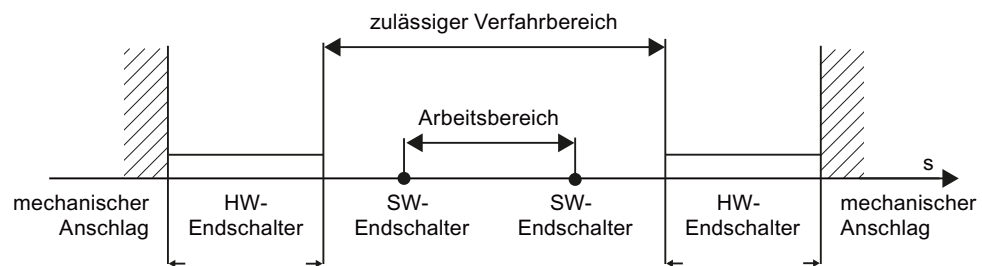
Ist in der Konfiguration "Richtungssinn invertieren" deaktiviert, so wird bei positiver Fahrtrichtung ein 5 V / 24 V-Pegel am Richtungsausgang ausgegeben (die angegebene Spannung ist abhängig von der verwendeten Hardware). Wurde in der Konfiguration "Richtungssinn invertieren" aktiviert, so wird bei positiver Fahrtrichtung ein 0 V-Pegel am Richtungsausgang ausgegeben.

## Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 4131)  
Prinzip der Impulsschnittstelle (Seite 4133)  
Hardware- und Software-Endschalter (Seite 4135)  
Ruckbegrenzung (Seite 4136)  
Referenzieren (Seite 4137)  
Einbindung des Technologieobjekts Achse (Seite 4141)  
Werkzeuge des Technologieobjekts Achse (Seite 4143)

### 11.2.2.4 Hardware- und Software-Endschalter

Begrenzen Sie mit Hardware- und Software-Endschaltern den "zulässigen Verfahrbereich" und den "Arbeitsbereich" ihres Technologieobjekts Achse. Die Zusammenhänge können Sie der nachfolgenden Darstellung entnehmen:



Bei den Hardware-Endschaltern handelt es sich um Endlagenschalter, welche den maximal "zulässigen Verfahrbereich" der Achse begrenzen. Hardware-Endschalter sind physikalische Schaltelemente, die an alarmfähigen Eingängen der CPU angeschlossen werden müssen.

Mit Software-Endschaltern wird der "Arbeitsbereich" der Achse begrenzt. Sie sollten bezogen auf den Verfahrbereich innerhalb der Hardware-Endschalter liegen. Da die Positionen der Software-Endschalter flexibel eingestellt werden können, kann der Arbeitsbereich der Achse je nach aktuellem Verfahrprofil individuell angepasst werden. Im Gegensatz zu Hardware-Endschaltern werden Software-Endschalter nur über die Software realisiert und benötigen keine eigenen Schaltelemente.

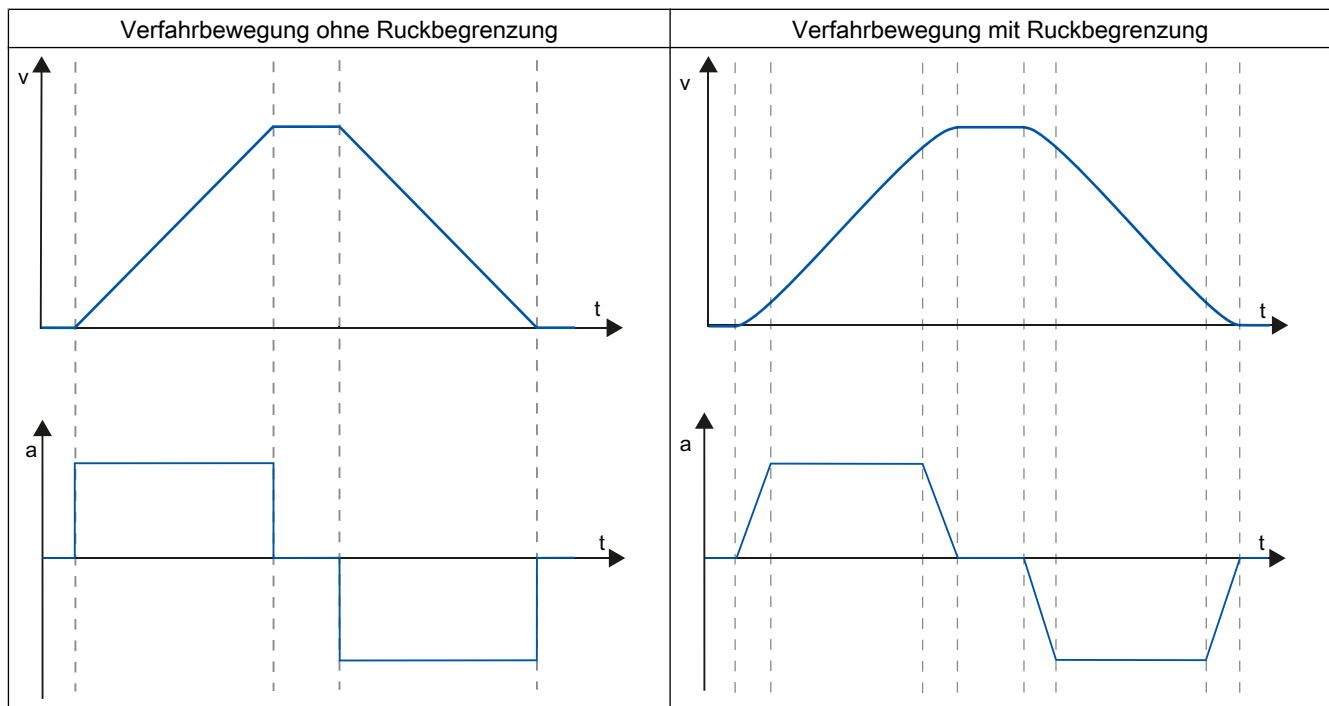
Hardware- und Software-Endschalter müssen vor der Verwendung in der Konfiguration, bzw. im Anwenderprogramm aktiviert werden. Software-Endschalter sind erst nach dem Referenzieren der Achse wirksam.

**Siehe auch**

- Für Motion Control relevante Ausgänge der CPU (Seite 4131)
- Prinzip der Impulsschnittstelle (Seite 4133)
- Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang (Seite 4134)
- Ruckbegrenzung (Seite 4136)
- Referenzieren (Seite 4137)
- Einbindung des Technologieobjekts Achse (Seite 4141)
- Werkzeuge des Technologieobjekts Achse (Seite 4143)
- Positionsüberwachung (Seite 4151)

**11.2.2.5 Ruckbegrenzung**

Mit der Ruckbegrenzung reduzieren Sie die Belastungen ihrer Mechanik während einer Beschleunigungs- und Verzögerungsrampe. Der Wert für die Beschleunigung und Verzögerung wird bei aktiver Ruckbegrenzung nicht abrupt geändert, sondern sanft auf- und abgebaut. Das nachfolgende Bild zeigt den Verlauf von Geschwindigkeit und Beschleunigung ohne und mit Ruckbegrenzung:



Mit der Ruckbegrenzung ergibt sich ein "verrundetes" Geschwindigkeitsprofil der Achsbewegung. Hiermit ist z. B. ein sanftes Anfahren und Abbremsen eines Förderbandes gewährleistet.

## Siehe auch

- Verhalten der Achse beim Verwenden der Ruckbegrenzung (Seite 4160)
- Für Motion Control relevante Ausgänge der CPU (Seite 4131)
- Prinzip der Impulsschnittstelle (Seite 4133)
- Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang (Seite 4134)
- Hardware- und Software-Endschalter (Seite 4135)
- Referenzieren (Seite 4137)
- Einbindung des Technologieobjekts Achse (Seite 4141)
- Werkzeuge des Technologieobjekts Achse (Seite 4143)

### 11.2.2.6 Referenzieren

Unter Referenzieren versteht man das Abgleichen der Achskoordinate des Technologieobjekts auf die reale, physikalische Position des Antriebs. Da sich bei positionsgesteuerten Achsen die Eingaben und Anzeigen zur Position auf genau diese Achskoordinate beziehen, ist eine Übereinstimmung zur realen Situation von äußerster Wichtigkeit. Nur so kann gewährleistet werden, dass die absolute Zielposition der Achse auch am Antrieb exakt erreicht wird.

Bei der CPU S7-1200 erfolgt das Referenzieren der Achse mit der Motion Control-Anweisung "MC\_Home". Es wird zwischen den folgenden Referenziermodi unterschieden:

#### Referenziermodi

- **Aktives Referenzieren**  
Beim aktiven Referenzieren führt die Motion Control-Anweisung "MC\_Home" die notwendige Referenzpunktfahrt durch. Beim Erkennen des Referenzpunktschalters wird die Achse entsprechend der Konfiguration referenziert. Laufende Verfahrbewegungen werden abgebrochen.
- **Passives Referenzieren**  
Beim passiven Referenzieren führt die Motion Control-Anweisung "MC\_Home" keine Referenzierbewegung durch. Die dafür notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen realisiert werden. Beim Erkennen des Referenzpunktschalters wird die Achse entsprechend der Konfiguration referenziert. Laufende Verfahrbewegungen werden beim Start des passiven Referenzierens nicht abgebrochen.

- **Direktes Referenzieren Absolut**  
Die Achsposition wird ohne Berücksichtigung des Referenzpunktschalters gesetzt. Laufende Verfahrbewegungen werden nicht abgebrochen. Der Wert des Eingangsparameters "Position" der Motion Control-Anweisung "MC\_Home" wird sofort als Referenzpunkt der Achse gesetzt.
- **Direktes Referenzieren Relativ**  
Die Achsposition wird ohne Berücksichtigung des Referenzpunktschalters gesetzt. Laufende Verfahrbewegungen werden nicht abgebrochen. Für die Achsposition nach dem Referenzieren gilt:  
Neue Achsposition = Aktuelle Achsposition + Wert des Parameters "Position" der Anweisung "MC\_Home".

### Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 4131)

Prinzip der Impulsschnittstelle (Seite 4133)

Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang (Seite 4134)

Hardware- und Software-Endschalter (Seite 4135)

Ruckbegrenzung (Seite 4136)

Einbindung des Technologieobjekts Achse (Seite 4141)

Werkzeuge des Technologieobjekts Achse (Seite 4143)

Referenzieren (Technologieobjekt "Achse" ab V2.0) (Seite 4163)

### 11.2.3 Leitfaden zum Einsatz von Motion Control

Der hier beschriebene Leitfaden zeigt die grundsätzliche Vorgehensweise, um Motion Control mit der CPU S7-1200 einzusetzen.

#### Voraussetzung

Um das Technologieobjekt "Achse" einzusetzen muss ein Projekt mit einer CPU S7-1200 angelegt sein.

#### Vorgehen

Gehen Sie in der nachfolgend empfohlenen Reihenfolge vor, um Motion Control mit der CPU S7-1200 einzusetzen. Folgen Sie hierzu den aufgelisteten Links:

1. Technologieobjekt Achse hinzufügen (Seite 4145)
2. Arbeiten mit dem Konfigurationsdialog (Seite 4146)
3. Laden in CPU (Seite 4190)



4. Funktionstest der Achse im Inbetriebnahmefenster (Seite 4191)
5. Programmieren (Seite 4194)
6. Achssteuerung diagnostizieren (Seite 4214)

## 11.2.4 Versionsübersicht

Die Zusammenhänge der relevanten Versionen für S7-1200 Motion Control können Sie der nachfolgenden Aufstellung entnehmen:

### Version der Technologie

Die aktuell ausgewählte Version der Technologie können Sie in der Task Card Anweisungen > Technologie > Motion Control > S7-1200 Motion Control und im Dialog "Neues Objekt hinzufügen" überprüfen. Die Version der Technologie wählen Sie in der Task Card Anweisungen > Technologie > Motion Control > S7-1200 Motion Control. Wird im Dialog "Neues Objekt hinzufügen" ein Technologieobjekt mit einer alternativen Version hinzugefügt, so wird die Version der Technologie ebenfalls umgeschaltet.

---

#### Hinweis

Die Auswahl einer alternativen Version der Technologie betrifft auch die Version der Motion Control-Anweisungen (Task Card). Die Technologieobjekte und Motion Control-Anweisungen werden erst beim Übersetzen, bzw. beim "Laden in Gerät" entsprechend der gewählten Version der Technologie konvertiert.

---

### Version des Technologieobjekts

Die Version eines Technologieobjekts kann im Inspektorfenster im Register "Eigenschaften > Allgemein > Information" im Feld "Version" überprüft werden.

Wählen Sie zum Ändern der Version in der Task Card der Anweisungen > Technologie die gewünschte Version und wählen Sie den Menübefehl Bearbeiten > Übersetzen. Wird im Dialog "Neues Objekt hinzufügen" ein Technologieobjekt mit einer alternativen Version hinzugefügt, so wird die Version der Technologieobjekte ebenfalls umgeschaltet.

Beachten Sie evtl. Fehleranzeigen beim Übersetzen und beseitigen Sie die Ursachen der angezeigten Fehler. Wiederholen Sie die Übersetzung, bis die Übersetzung fehlerfrei abgeschlossen wird.

Überprüfen Sie anschließend die Konfiguration der Technologieobjekte.

### Version der Motion Control-Anweisung

Gehen Sie zum Überprüfen der Version einer Motion Control-Anweisung wie folgt vor:

1. Öffnen Sie im Navigator die Ordner Programmbausteine > Systembausteine > Programmressourcen und markieren sie die gewünschte Motion Control-Anweisung.
2. Wählen Sie den Menübefehl Bearbeiten > Eigenschaften.
3. Im Register Information finden Sie im Feld Version die Version der Motion Control-Anweisung.

Entspricht die eingesetzte Version der Motion Control-Anweisung nicht der nachfolgenden Kompatibilitätsliste, so werden die entsprechenden Motion Control-Anweisungen im Programmeditor markiert.

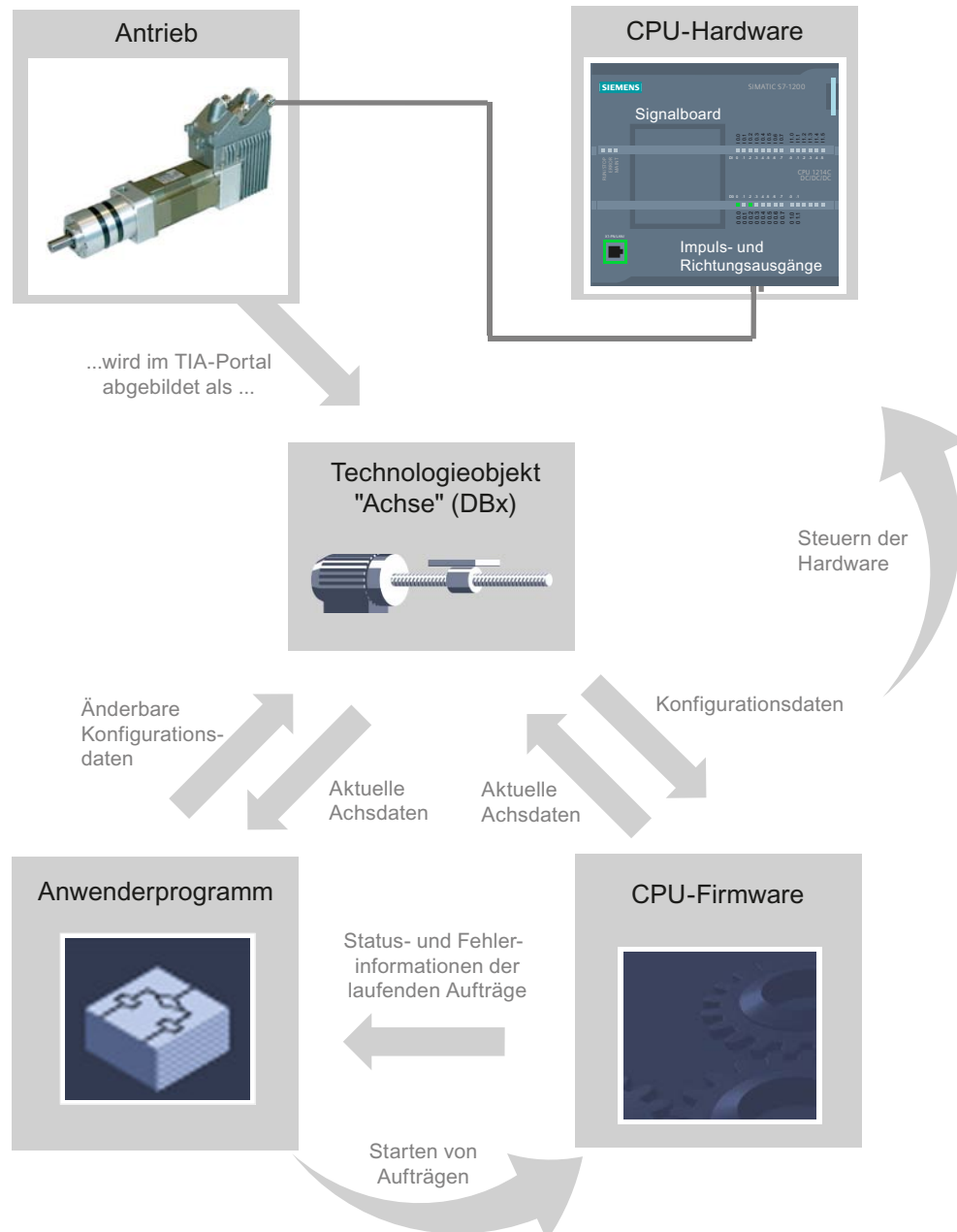
### Kompatibilitätsliste

| Technologie |   | CPU                    | Technologieobjekt                   | Motion Control-Anweisung   |
|-------------|---|------------------------|-------------------------------------|--|
| V1.0        |   | V1.0, V2.0, V2.1, V2.2 | Achse V1.0                          | MC_Power V1.0<br>MC_Reset V1.0<br>MC_Home V1.0<br>MC_Halt V1.0<br>MC_MoveAbsolute V1.0<br>MC_MoveRelative V1.0<br>MC_MoveVelocity V1.0<br>MC_MoveJog V1.0  |
| V2.0        | Neuerungen: <ul style="list-style-type: none"> <li>• Ruckbegrenzung</li> <li>• Auftragstabelle</li> <li>• MC_ChangeDynamic</li> </ul> | V2.1, V2.2             | Achse V2.0,<br>Auftragstabelle V2.0 | MC_Power V2.0<br>MC_Reset V2.0<br>MC_Home V2.0<br>MC_Halt V2.0<br>MC_MoveAbsolute V2.0<br>MC_MoveRelative V2.0<br>MC_MoveVelocity V2.0<br>MC_MoveJog V2.0<br>MC_CommandTable V2.0<br>MC_ChangeDynamic V2.0 |
| V3.0        | Neuerung:<br>Laden im Betriebszustand<br>RUN  | V2.2                   | Achse V3.0,<br>Auftragstabelle V3.0 | MC_Power V3.0<br>MC_Reset V3.0<br>MC_Home V3.0<br>MC_Halt V3.0<br>MC_MoveAbsolute V3.0<br>MC_MoveRelative V3.0<br>MC_MoveVelocity V3.0<br>MC_MoveJog V3.0<br>MC_CommandTable V3.0<br>MC_ChangeDynamic V3.0 |

## 11.2.5 Technologieobjekt Achse

### 11.2.5.1 Einbindung des Technologieobjekts Achse

In der nachfolgenden Darstellung werden die Zusammenhänge der Hardware- und Softwarekomponenten dargestellt, die bei der Verwendung des Technologieobjekts "Achse" zum Einsatz kommen:



## CPU-Hardware

Über die CPU-Hardware wird der physikalische Antrieb gesteuert und überwacht.

## Antrieb

Der Antrieb stellt die Einheit aus Leistungsteil und Motor dar. Es können Schrittmotoren oder Servomotoren mit Impulsschnittstelle verwendet werden.

## Technologieobjekt "Achse"

Der physikalische Antrieb inklusive Mechanik wird im TIA-Portal als Technologieobjekt "Achse" abgebildet. Konfigurieren Sie hierzu das Technologieobjekt "Achse" mit den nachfolgenden Parametern:

- Auswahl des zu verwendenden PTOs (Pulse Train Output) und Konfiguration der Antriebsschnittstelle
- Parameter zur Mechanik und zur Getriebeübersetzung des Antriebs (bzw. der Maschine oder Anlage)
- Parameter zur Positionsüberwachung, zu Dynamikparametern und zum Referenzieren

Die Konfiguration des Technologieobjekts "Achse" wird im Technologieobjekt (Datenbaustein) gespeichert. Dieser Datenbaustein bildet gleichzeitig die Schnittstelle zwischen Anwenderprogramm und CPU-Firmware. Zur Laufzeit des Anwenderprogramms werden im Datenbaustein des Technologieobjekts die aktuellen Achsdaten abgelegt.

## Anwenderprogramm

Über das Anwenderprogramm starten Sie über Motion Control-Anweisungen Aufträge in der CPU-Firmware. Folgende Aufträge zur Steuerung der Achse sind möglich:

- Achse absolut positionieren
- Achse relativ positionieren
- Achse mit Drehzahlvorgabe fahren
- Achsaufträge als Bewegungssequenz ausführen (ab Technologie V2.0)
- Achse im Tipp-Betrieb fahren
- Achse anhalten
- Achse referenzieren; Referenzpunkt setzen
- Fehler quittieren

Über die Eingangsparameter der Motion Control-Anweisungen und die Konfiguration der Achse bestimmen Sie die Parameter des Auftrags. Über die Ausgangs-Parameter der Anweisung erhalten Sie zeitaktuelle Informationen zum Status und zu evtl. Fehlern des Auftrags.

Bevor Sie einen Auftrag für die Achse starten, müssen Sie die Achse mit der Motion Control-Anweisung "MC\_Power" freigegeben.

Über die Variablen des Technologieobjekts können Sie im Anwenderprogramm Konfigurationsdaten und aktuelle Achsdaten auslesen. Einzelne, änderbare Variablen des

Technologieobjekts (z. B. die aktuelle Beschleunigung) können Sie vom Anwenderprogramm aus ändern.

## CPU Firmware

Die im Anwenderprogramm angestoßenen Motion Control-Aufträge werden in der CPU-Firmware abgearbeitet. Beim Einsatz der Achssteuertafel erfolgt der Anstoß der Motion Control-Aufträge über die Bedienung der Achssteuertafel. Entsprechend der Konfiguration der Achse erfüllt die CPU-Firmware folgende Aufgaben:

- Berechnen des genauen Bewegungsprofils für Bewegungsaufträge und Notstoppsituationen
- Steuerung der Antriebsfreigabe, sowie des Impuls- und Richtungssignals
- Überwachung des Antriebs, sowie der Hardware- und Software-Endschalter
- Zeitaktuelles Rückmelden von Status- und Fehlerinformationen der Aufträge an die Motion Control-Anweisungen im Anwenderprogramm
- Schreiben von aktuellen Achsdaten in den Datenbaustein des Technologieobjekts

## Siehe auch

Für Motion Control relevante Ausgänge der CPU (Seite 4131)

Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang (Seite 4134)

Werkzeuge des Technologieobjekts Achse (Seite 4143)

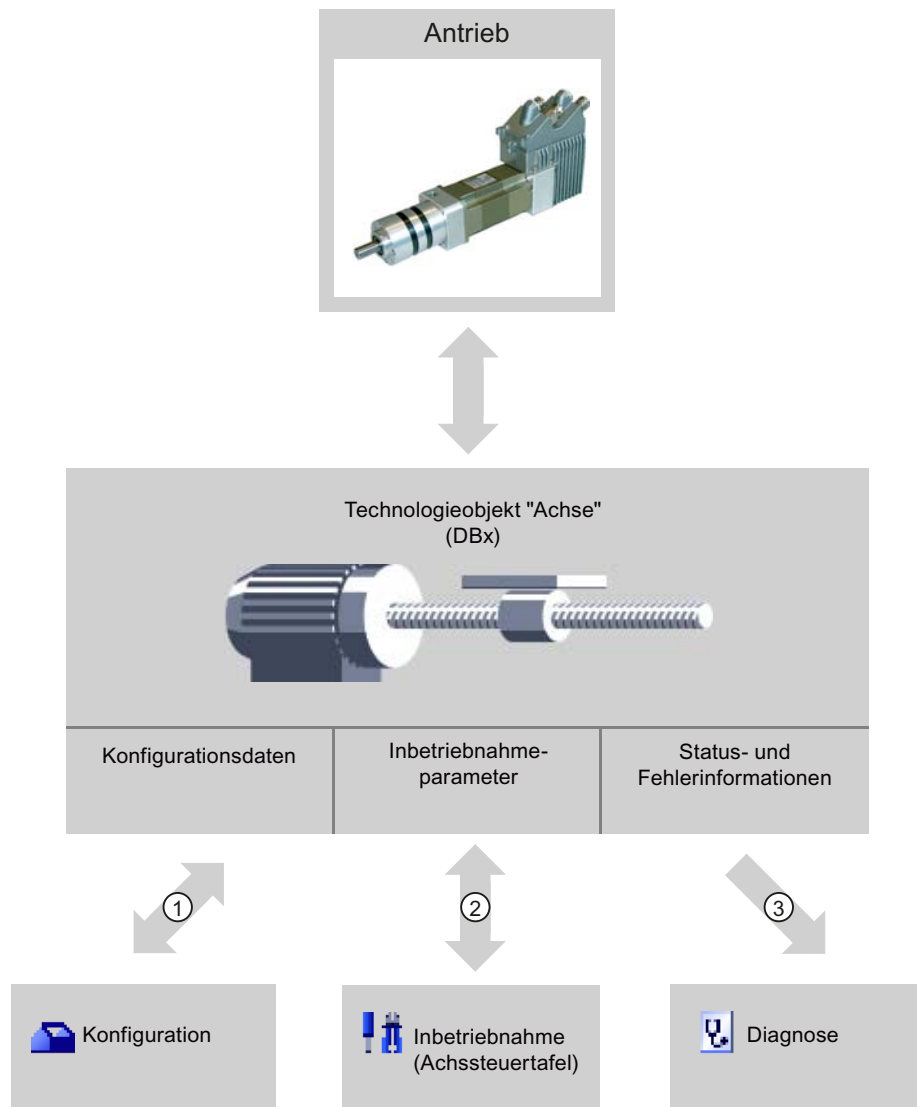
Hardware- und Software-Endschalter (Seite 4135)

Referenzieren (Seite 4137)

Variable des Technologieobjekts Achse (Seite 4242)

### 11.2.5.2 Werkzeuge des Technologieobjekts Achse

Für das Technologieobjekt "Achse" stehen Ihnen im TIA-Portal die Werkzeuge "Konfiguration", "Inbetriebnahme" und "Diagnose" zur Verfügung. Die nachfolgende Darstellung zeigt das Zusammenwirken der drei Werkzeuge mit dem Technologieobjekt und dem Antrieb:



|   |  |
|---|--|
| ① | Schreiben und Lesen der Konfigurationsdaten des Technologieobjekts                                       |
| ② | Steuern des Antriebs über das Technologieobjekt. Lesen des Achsstatus zur Anzeige in der Achssteuertafel |
| ③ | Auslesen der aktuellen Status- und Fehlerinformationen des Technologieobjekts                            |

## Konfiguration

Konfigurieren Sie mit dem Werkzeug "Konfiguration" folgende Eigenschaften des Technologieobjekts "Achse":

- Auswahl des zu verwendenden PTOs und Konfiguration der Antriebsschnittstelle
- Eigenschaften der Mechanik und Getriebeübersetzung, des Antriebs (bzw. der Maschine oder Anlage)
- Eigenschaften zur Positionsüberwachung, zu Dynamikparametern und zum Referenzieren

Die Konfiguration wird im Datenbaustein des Technologieobjekts gespeichert.

## **Inbetriebnahme**

Testen Sie mit dem Werkzeug "Inbetriebnahme" die Funktion Ihrer Achse, ohne dass ein Anwenderprogramm angelegt sein muss. Mit dem Start des Werkzeugs öffnet sich die Achssteuertafel. In der Achssteuertafel stehen Ihnen folgende Befehle zur Verfügung:

- Freigeben und Sperren der Achse
- Verfahren der Achse im Tipp-Betrieb
- Absolutes und relatives Positionieren der Achse
- Referenzieren der Achse
- Quittieren von Fehlern

Für die Bewegungsbefehle können die Dynamikwerte entsprechend angepasst werden. Die Achssteuertafel zeigt zusätzlich den aktuellen Achsstatus an.

## **Diagnose**

Kontrollieren Sie mit dem Werkzeug "Diagnose" die aktuellen Status- und Fehlerinformationen von Achse und Antrieb.

## **Siehe auch**

Für Motion Control relevante Ausgänge der CPU (Seite 4131)

Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang (Seite 4134)

Einbindung des Technologieobjekts Achse (Seite 4141)

Hardware- und Software-Endschalter (Seite 4135)

Referenzieren (Seite 4137)

Technologieobjekt Achse konfigurieren (Seite 4146)

Inbetriebnahme der Achse - Achssteuertafel (Seite 4191)

Achse - Diagnose (Seite 4214)

### **11.2.5.3 Technologieobjekt Achse hinzufügen**

Um ein Technologieobjekt "Achse" im Projektnavigator hinzuzufügen, gehen Sie folgendermaßen vor:

#### **Voraussetzung**

Ein Projekt mit einer CPU S7-1200 ist angelegt.

#### **Vorgehen**

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner Technologieobjekte.

3. Doppelklicken Sie auf "Neues Objekt hinzufügen".  
Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Wählen Sie die Technologie "Motion".
5. Öffnen Sie den Ordner "Motion Control".
6. Öffnen Sie den Ordner "S7-1200 Motion Control".
7. Klicken Sie auf den Eintrag zur Version und wählen Sie eine alternative Version der Technologie, falls Sie eine Achse einer älteren Version hinzufügen möchten.
8. Wählen Sie das Objekt "TO\_Axis\_PTO".
9. Passen Sie im Eingabefeld "Name" den Namen der Achse ihren Bedürfnissen an.
10. Wählen Sie die Option "Manuell", falls Sie die vorgeschlagene Datenbausteinnummer ändern möchten.
11. Klicken Sie auf "Weitere Informationen", wenn Sie eigene Informationen zum Technologieobjekt ergänzen möchten.
12. Klicken Sie auf die Schaltfläche "OK", wenn Sie das Technologieobjekt hinzufügen möchten.  
Klicken Sie auf die Schaltfläche "Abbrechen", wenn Sie die Eingaben verwerfen möchten.

## Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt.

## Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 4138)

### 11.2.5.4 Technologieobjekt Achse konfigurieren

#### Arbeiten mit dem Konfigurationsdialog

Die Eigenschaften des Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Um das Konfigurationsfenster des Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Projektnavigator die Gruppe des gewünschten Technologieobjekts.
2. Doppelklicken Sie auf das Objekt "Konfiguration".





Die Konfiguration ist in folgende Kategorien aufgeteilt:

- **Grundparameter**  
Die Grundparameter enthalten alle Parameter, die für eine funktionsfähige Achse konfiguriert werden müssen.
- **Erweiterte Parameter**  
Die erweiterten Parameter enthalten Parameter zur Anpassung an Ihren Antrieb, bzw. an Ihre Anlage.



## Symbole des Konfigurationsfensters

Symbole in der Bereichsnavigation der Konfiguration zeigen weitere Details zum Status der Konfiguration:

|   |   |
|---|---|
|  | <b>Die Konfiguration enthält Voreinstellungswerte und ist vollständig.</b><br>Die Konfiguration enthält ausschließlich voreingestellte Werte. Mit diesen voreingestellten Werten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.                                      |
|  | <b>Die Konfiguration enthält vom Anwender definierte Werte und ist vollständig</b><br>Alle Eingabefelder der Konfiguration enthalten gültige Werte, und mindestens ein voreingestellter Wert wurde geändert.  |
|  | <b>Die Konfiguration ist unvollständig oder fehlerhaft</b><br>Mindestens ein Eingabefeld oder eine Klappliste beinhaltet einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an. |
|  | <b>Die Konfiguration ist gültig, enthält jedoch Warnhinweise</b><br>Nur ein Hardware-Endschalter ist konfiguriert. Je nach Anlage kann die fehlende Konfiguration eines Hardware-Endschalters zu einer Gefährdung führen. Das entsprechende Feld oder die Klappliste wird gelb hinterlegt.  |

## Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 4138)  
Grundparameter (Seite 4147)  
Erweiterte Parameter (Seite 4150)

## Grundparameter

### Konfiguration - Allgemein

Konfigurieren Sie im Konfigurationsfenster "Allgemein" die grundlegenden Eigenschaften des Technologieobjekts "Achse".

### Achsname:

Definieren Sie in diesem Feld den Namen der Achse, bzw. den Namen des Technologieobjekts "Achse". Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet.

### Hardware - Schnittstelle

Die Impulse werden über fest zugeordnete digitale Ausgänge an das Leistungsteil des Antriebs ausgegeben.

Bei CPUs mit Relais-Ausgängen kann das Impulssignal nicht auf diesen Ausgängen ausgegeben werden, da die Relais die erforderlichen Schaltfrequenzen nicht unterstützen.

Um auf diesen CPUs den PTO (Pulse Train Output) nutzen zu können, muss ein Signalboard mit digitalen Ausgängen verwendet werden.

### Hinweis

Der PTO benötigt intern die Funktionalität eines schnellen Zählers (HSC). Der entsprechende schnelle Zähler kann somit nicht anderweitig verwendet werden. Der Zählerstand kann nicht über seine Eingangsadresse ausgewertet werden.

Die Zuordnung zwischen PTO und HSC ist fest. Bei anwenderseitiger Aktivierung des PTO1 wird dieser mit dem HSC1 verbunden. Wird der PTO2 aktiviert, so wird dieser mit dem HSC2 verbunden.

Wählen Sie in der Klappliste "Auswahl Impulsgenerator" den PTO (Pulse Train Output), über welchen die Impulse für die Ansteuerung der Schrittmotoren oder Servomotoren mit Impulsschnittstelle zur Verfügung gestellt werden sollen. Wurden in der Gerätekonfiguration die Impulsgeneratoren und die schnellen Zähler nicht anderweitig verwendet, so kann die Hardware-Schnittstelle automatisch konfiguriert werden. In diesem Fall wird der gewählte PTO in der Klappliste weiß hinterlegt angezeigt. In den Ausgabefeldern "Ausgangsquelle", "Impulsausgang", "Richtungsausgang" und "Zugewiesener schneller Zähler" werden die verwendeten Schnittstellen aufgelistet.

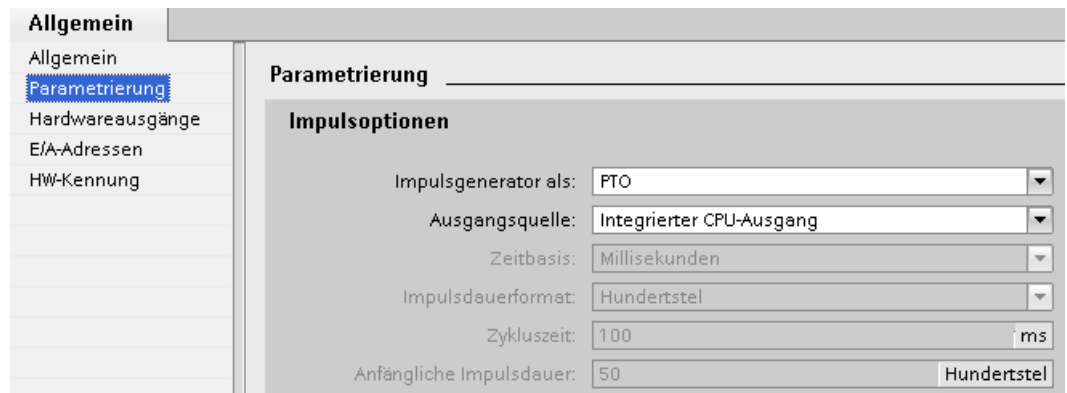
Gehen Sie wie folgt vor, wenn Sie die Schnittstellen ändern möchten, oder der PTO nicht automatisch konfiguriert werden konnte (Eintrag in der Klappliste "Auswahl Impulsgenerator" ist rot hinterlegt):

1. Klicken Sie auf die Schaltfläche "Gerätekonfiguration".  
Die Gerätekonfiguration des Impulsgenerators öffnet sich.  
Vergrößern Sie den Eigenschaftsbereich der Gerätekonfiguration, falls die Konfiguration des Impulsgenerators nicht sichtbar ist.

The screenshot shows a software configuration window. On the left, there is a navigation pane with a tree view containing the following items: 'Allgemein', 'Parametrierung', 'Hardwareausgänge', 'E/A-Adressen', and 'HW-Kennung'. The 'Allgemein' item is selected and highlighted. The main content area is titled 'Allgemein' and is divided into two sections. The first section, 'Aktivieren', contains a checkbox labeled 'Diesen Impulsgenerator aktivieren' which is checked. The second section, 'Projektinformationen', contains a text input field for 'Name' with the value 'Pulse\_1' and a text area for 'Kommentar' which is currently empty. There are small up and down arrow icons next to the 'Kommentar' field.

2. Aktivieren Sie das Optionskästchen "Diesen Impulsgenerator aktivieren".

3. Wählen Sie im Bereichsnavigator den Eintrag "Parametrierung". Die "Parametrierung" öffnet sich.



4. Wählen Sie in der Klappliste "Impulsgenerator als:" den Eintrag "PTO".
5. Wählen Sie in der Klappliste "Ausgangsquelle:" den Eintrag "Integrierter CPU-Ausgang" oder "Ausgang des Signalboards". Der Eintrag "Ausgang des Signalboards" ist je nach gestecktem Signalboard, nur für PTO1 oder für PTO1 und PTO2 auswählbar. Genauere Informationen entnehmen Sie dem Kapitel: Für Motion Control relevante Ausgänge der CPU (Seite 4131)
6. Kehren Sie in die Konfiguration der Achse zurück.  
Wurde der korrespondierende schnelle Zähler nicht bereits anderweitig verwendet, so sind die PTO-Felder der Achskonfiguration "Allgemein" ohne rote Hinterlegung. Korrigieren Sie die Konfiguration anhand der Fehlermeldungen falls dies nicht der Fall ist.

## Anwendereinheit

Wählen Sie in der Klappliste die gewünschte Einheit für das Maßsystem der Achse aus. Die gewählte Einheit wird für die weitere Konfiguration des Technologieobjekts "Achse" und für die Anzeige der aktuellen Achsdaten verwendet.

Die Werte an den Eingangsparametern (Position, Distance, Velocity, ...) der Motion Control-Anweisungen beziehen sich ebenfalls auf diese Einheit.

---

### Hinweis

Eine nachträgliche Änderung des Maßsystems kann u. U. nicht in allen Konfigurationsfenstern des Technologieobjekts korrekt umgerechnet werden. Kontrollieren Sie in diesem Fall die Konfiguration aller Achsparameter.

Gegebenenfalls müssen im Anwenderprogramm Werte an den Eingangsparametern der Motion Control-Anweisungen an die neue Maßeinheit angepasst werden.

---

## Erweiterte Parameter

### Konfiguration - Antriebssignale

Konfigurieren Sie im Konfigurationsfenster "Antriebssignale" den Ausgang für die Antriebsfreigabe und den Eingang für die Rückmeldung "Antrieb Bereit" des Antriebs.

Die Antriebsfreigabe wird von der Motion Control-Anweisung "MC\_Power" gesteuert und erteilt dem Antrieb die Leistungsfreigabe. Das Signal wird dem Antrieb über den zu konfigurierenden Ausgang zur Verfügung gestellt.

Ist der Antrieb nach Erhalt der Antriebsfreigabe bereit Bewegungen auszuführen, so meldet er "Antrieb Bereit" an die CPU. Das Signal "Antrieb Bereit" wird der CPU über den zu konfigurierenden Eingang zurückgemeldet.

Verfügt der Antrieb über keine derartigen Schnittstellen, so müssen die Parameter nicht konfiguriert werden. Wählen Sie in diesem Fall für den Bereit-Eingang den Wert TRUE.

### Siehe auch

Konfiguration - Mechanik (Seite 4150)

Positionsüberwachung (Seite 4151)

Dynamik (Seite 4156)

Referenzieren (Technologieobjekt "Achse" ab V2.0) (Seite 4163)

### Konfiguration - Mechanik

Konfigurieren Sie im Konfigurationsfenster "Mechanik" die mechanischen Eigenschaften Ihres Antriebs.

### Impulse pro Motorumdrehung

Konfigurieren Sie in diesem Feld, wie viele Impulse der Motor für eine Motorumdrehung benötigt.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- $0 < \text{Impulse pro Motorumdrehung} \leq 2147483647$

### Weg pro Motorumdrehung

Konfigurieren Sie in diesem Feld, welche Wegstrecke die Mechanik Ihrer Anlage pro Motorumdrehung zurücklegt.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- $0.0 < \text{Weg pro Motorumdrehung} \leq 1.0e12$

### Richtungssinn invertieren

Über das Optionskästchen "Richtungssinn invertieren" können Sie den Richtungsausgang an die Richtungslogik des Antriebs anpassen.

- **Richtungssinn invertieren: deaktiviert**  
0 V-Pegel = negative Fahrtrichtung  
5 V- / 24 V-Pegel = positive Fahrtrichtung (die angegebene Spannung ist abhängig von der verwendeten Hardware)
- **Richtungssinn invertieren: aktiviert**  
0 V-Pegel = positive Fahrtrichtung  
5 V- / 24 V-Pegel = negative Fahrtrichtung (die angegebene Spannung ist abhängig von der verwendeten Hardware)

### Siehe auch

Konfiguration - Antriebssignale (Seite 4150)

Positionsüberwachung (Seite 4151)

Dynamik (Seite 4156)

Referenzieren (Technologieobjekt "Achse" ab V2.0) (Seite 4163)

Zusammenhang zwischen der Fahrtrichtung und dem Spannungspegel am Richtungsausgang (Seite 4134)

### Positionsüberwachung

#### Anforderungen an Hardware-Endschalter

Verwenden Sie nur Hardware-Endschalter, die nach dem Anfahren dauerhaft geschaltet bleiben. Dieser Schaltzustand darf erst nach der Rückkehr in den zulässigen Verfahrbereich zurück genommen werden.

### Siehe auch

Konfiguration - Positionsüberwachung (Seite 4151)

Verhalten der Achse beim Ansprechen der Positionsüberwachungen (Seite 4153)

Konfiguration der Positionsüberwachung im Anwenderprogramm ändern (Seite 4155)

### Konfiguration - Positionsüberwachung

Konfigurieren Sie im Konfigurationsfenster "Positionsüberwachung" die Hardware- und Software-Endschalter der Achse.

## Aktiviere HW-Endschalter

Aktivieren Sie mit diesem Optionskästchen die Funktion des unteren und oberen Hardware-Endschalters. Während einer aktiven Referenzpunktfahrt können die Hardware-Endschalter zur Richtungsumkehr eingesetzt werden. Details finden Sie in der Konfigurationsbeschreibung zum Referenzieren.

## Eingang unterer / oberer HW-Endschalter

Wählen Sie in der Klappliste den digitalen Eingang für den unteren, bzw. für den oberen Hardware-Endschalter aus. Der Eingang muss alarmfähig sein. Als Eingänge für die HW-Endschalter stehen die digitalen On-board CPU-Eingänge und die digitalen Eingänge eines gesteckten Signalboards zur Auswahl.



### VORSICHT

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt. Bei der Verwendung als HW-Endschalter kann dies zu unerwünschten Verzögerungen führen. Verringern Sie in diesem Fall für die entsprechenden Digitaleingänge die Filterzeit.

Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfiler" eingestellt werden.

## Pegelauswahl

Wählen Sie in der Klappliste den, bei angefahrenem Hardware-Endschalter an der CPU anstehenden, Signalpegel aus.

- Auswahl "Unterer Pegel"  
0 V (FALSE) am CPU-Eingang entspricht Hardware-Endschalter angefahren
- Auswahl "Oberer Pegel"  
5 V / 24 V (TRUE) am CPU-Eingang entspricht Hardware-Endschalter angefahren (die angegebene Spannung ist abhängig von der verwendeten Hardware)

## Aktiviere SW-Endschalter

Aktivieren Sie mit diesem Optionskästchen die Funktion des unteren und oberen Software-Endschalters.

---

### Hinweis

Aktivierte Software-Endschalter wirken nur bei referenzierter Achse.

---

## Position unterer / oberer SW-Endschalter

Legen Sie in diesen Feldern den Positionswert des unteren und oberen Software-Endschalters fest.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- $-1.0e12 \leq \text{Unterer SW-Endschalter} \leq 1.0e12$
- $-1.0e12 \leq \text{Oberer SW-Endschalter} \leq 1.0e12$

Der Wert des oberen Software-Endschalters muss größer gleich dem Wert des unteren Software-Endschalters sein.

## Siehe auch

Anforderungen an Hardware-Endschalter (Seite 4151)

Verhalten der Achse beim Ansprechen der Positionsüberwachungen (Seite 4153)

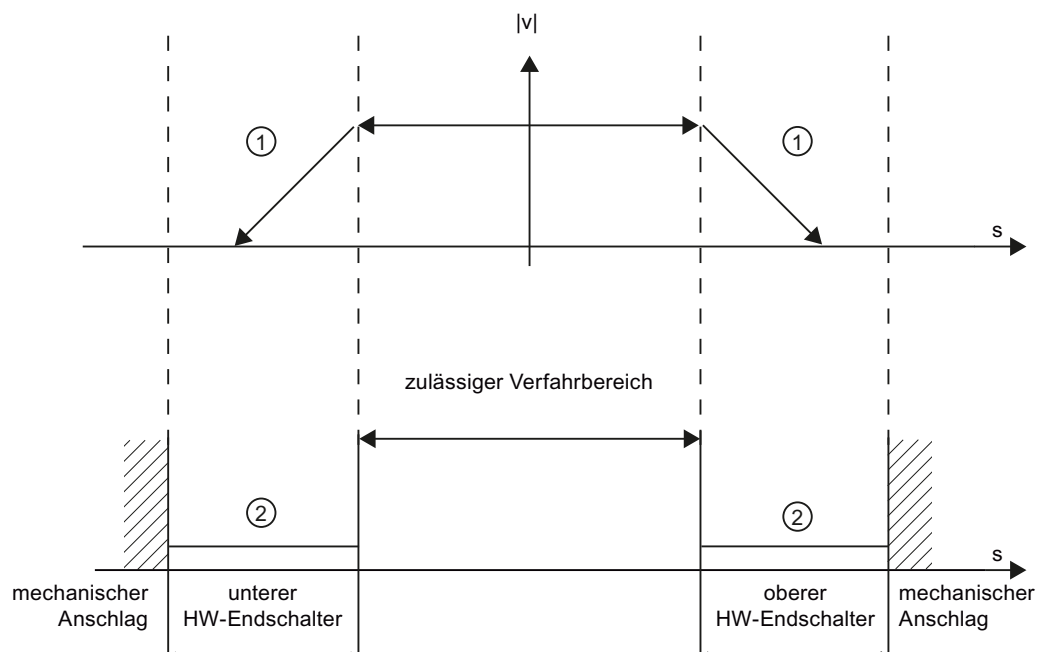
Konfiguration der Positionsüberwachung im Anwenderprogramm ändern (Seite 4155)

Konfiguration - Referenzieren - Aktiv (Seite 4165)

## Verhalten der Achse beim Ansprechen der Positionsüberwachungen

### Verhalten der Achse beim Anfahren der Hardware-Endschalter

Beim Anfahren der Hardware-Endschalter bremst die Achse mit der konfigurierten Notstopp-Verzögerung bis auf Stillstand ab. Die Notstopp-Verzögerung muss ausreichend groß gewählt werden, so dass die Achse sicher vor dem mechanischen Anschlag zum Stehen kommt. Das Verhalten der Achse nach dem Anfahren der Hardware-Endschalter entnehmen Sie der nachfolgenden Darstellung:



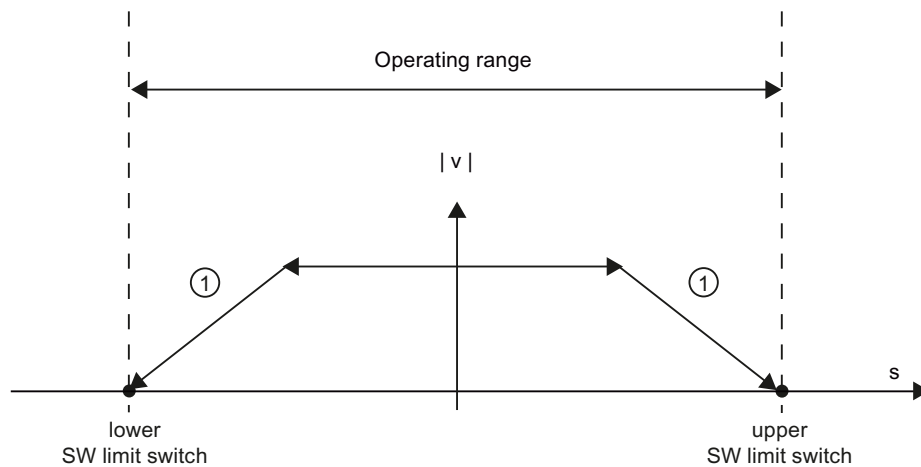
|   |  |
|---|--|
| ① | Die Achse bremsst mit der konfigurierten Notstopp-Verzögerung bis auf Stillstand ab. |
| ② | Bereich in dem die HW-Endschalter den Status "angefahren" melden.                    |

An der auslösenden Motion Control-Anweisung, am "MC\_Power" und in den Variablen des Technologieobjekts wird der Fehler "HW-Endschalter angefahren" angezeigt. Eine Anleitung zur Beseitigung des Fehlers finden Sie im Anhang unter "Liste der ErrorIDs und ErrorInfos".

### Verhalten der Achse beim Erreichen der Software-Endschalter

Eine laufende Bewegung kommt bei aktiven Software-Endschaltern auf der Position des Software-Endschalters zum Stehen. Die Achse wird mit der konfigurierten Verzögerung abgebremst.

Das Verhalten der Achse bis zum Erreichen der Software-Endschalter entnehmen Sie der nachfolgenden Darstellung:



|   |   |
|---|---|
| ① | Die Achse bremsst mit der konfigurierten Verzögerung bis auf Stillstand ab. |
|---|---|

An der auslösenden Motion Control-Anweisung, am "MC\_Power" und in den Variablen des Technologieobjekts wird der Fehler "SW-Endschalter erreicht" angezeigt. Eine Anleitung zur Beseitigung des Fehlers finden Sie im Anhang unter "Liste der ErrorIDs und ErrorInfos".

Unter welchen Umständen der Fehler "SW-Endschalter überschritten" angezeigt wird, können Sie den Topics "Software-Endschalter im Zusammenhang mit einem Referenzvorgang (Seite 4226)" und "Software-Endschalter im Zusammenhang mit Dynamikänderungen (Seite 4231)" entnehmen.

Verwenden Sie zusätzlich Hardware-Endschalter, wenn sich hinter den Software-Endschaltern ein mechanischer Anschlag befindet und die Gefahr eines mechanischen Schadens besteht.



## Siehe auch

Anforderungen an Hardware-Endschalter (Seite 4151)

Konfiguration - Positionsüberwachung (Seite 4151)

Konfiguration der Positionsüberwachung im Anwenderprogramm ändern (Seite 4155)

## Konfiguration der Positionsüberwachung im Anwenderprogramm ändern

Folgende Konfigurationsparameter können Sie zur Laufzeit des Anwenderprogramms in der CPU ändern:

### Hardware-Endschalter

Sie können die Hardware-Endschalter auch zur Laufzeit des Anwenderprogramms aktivieren und deaktivieren. Verwenden Sie hierzu folgende Variable des Technologieobjekts:

- <Achsenname>.Config.PositionLimits\_HW.Active

Wann Änderungen des Konfigurationsparameters wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

### Software-Endschalter

Sie können die Software-Endschalter auch zur Laufzeit des Anwenderprogramms aktivieren und deaktivieren, bzw. deren Positionswerte ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Achsenname>.Config.PositionLimits\_SW.Active  
zum Aktivieren und Deaktivieren der Software-Endschalter
- <Achsenname>.Config.PositionLimits\_SW.MinPosition  
zum Ändern der Position des unteren Software-Endschalters
- <Achsenname>.Config.PositionLimits\_SW.MaxPosition  
zum Ändern der Position des oberen Software-Endschalters

Wann Änderungen der Konfigurationsparameter wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

## Siehe auch

MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

Anforderungen an Hardware-Endschalter (Seite 4151)

Konfiguration - Positionsüberwachung (Seite 4151)

Verhalten der Achse beim Ansprechen der Positionsüberwachungen (Seite 4153)

## Dynamik

### Konfiguration - Dynamik Allgemein

Konfigurieren Sie im Konfigurationsfenster "Dynamik Allgemein" die maximale Geschwindigkeit, die Start/Stopp-Geschwindigkeit, die Beschleunigung und Verzögerung als auch die Ruckbegrenzung (ab Technologieobjekt "Achse" V2.0) der Achse.

### Einheit der Geschwindigkeitsbegrenzung

Wählen Sie in der Klappliste die physikalische Einheit mit der Sie die Geschwindigkeitsgrenzen festlegen möchten. Die hier eingestellte Einheit ist unabhängig von der unter "Konfiguration - Allgemein" eingestellten Maßeinheit und dient nur zur vereinfachten Eingabe.

### Maximale Geschwindigkeit / Start/Stopp-Geschwindigkeit

Definieren Sie in diesen Feldern die maximale zugelassene Geschwindigkeit und die Start/Stopp-Geschwindigkeit der Achse. Die Start/Stopp-Geschwindigkeit ist die minimal zulässige Geschwindigkeit der Achse.

Grenzwerte:

Die nachfolgend genannten Grenzwerte sind auf die Maßeinheit "Impulse/s" bezogen:

- **Technologieobjekt Achse V2.0**

- $2 \leq \text{Start/Stopp-Geschwindigkeit} \leq 20000$  (Signalboard 20kHz)  
 $2 \leq \text{Start/Stopp-Geschwindigkeit} \leq 200000$  (Signalboard 200kHz)  
 $2 \leq \text{Start/Stopp-Geschwindigkeit} \leq 100000$  (On-board CPU-Ausgänge)
- $2 \leq \text{Maximale Geschwindigkeit} \leq 20000$  (Signalboard 20kHz)  
 $2 \leq \text{Maximale Geschwindigkeit} \leq 200000$  (Signalboard 200kHz)  
 $2 \leq \text{Maximale Geschwindigkeit} \leq 100000$  (On-board CPU-Ausgänge)

- **Technologieobjekt Achse V1.0**

- $2 \leq \text{Start/Stopp-Geschwindigkeit} \leq 20000$  (Signalboard 20kHz)  
 $2 \leq \text{Start/Stopp-Geschwindigkeit} \leq 100000$  (Signalboard 200kHz)  
 $2 \leq \text{Start/Stopp-Geschwindigkeit} \leq 100000$  (On-board CPU-Ausgänge)
- $2 \leq \text{Maximale Geschwindigkeit} \leq 20000$  (Signalboard 20kHz)  
 $2 \leq \text{Maximale Geschwindigkeit} \leq 100000$  (Signalboard 200kHz)  
 $2 \leq \text{Maximale Geschwindigkeit} \leq 100000$  (On-board CPU-Ausgänge)

Der Wert der maximalen Geschwindigkeit muss größer gleich dem Wert der Start/Stopp-Geschwindigkeit sein.

Die Grenzwerte für andere Maßeinheiten müssen anwenderseitig entsprechend der gegebenen Mechanik umgerechnet werden.

### Beschleunigung / Verzögerung - Hochlaufzeit / Rücklaufzeit

Stellen Sie die gewünschte Beschleunigung in den Feldern "Hochlaufzeit" oder "Beschleunigung" ein. Die gewünschte Verzögerung kann in den Feldern "Rücklaufzeit" oder "Verzögerung" eingestellt werden.

Den Zusammenhang zwischen Hochlaufzeit und Beschleunigung, bzw. Rücklaufzeit und Verzögerung können Sie den folgenden Gleichungen entnehmen:

$$\text{Hochlaufzeit} = \frac{\text{Maximale Geschwindigkeit} - \text{Start/Stopp-Geschwindigkeit}}{\text{Beschleunigung}}$$

$$\text{Rücklaufzeit} = \frac{\text{Maximale Geschwindigkeit} - \text{Start/Stopp-Geschwindigkeit}}{\text{Verzögerung}}$$

Im Anwenderprogramm angestoßene Verfahrtaufträge werden mit der gewählten Beschleunigung / Verzögerung ausgeführt.

Grenzwerte:

Die nachfolgend genannten Grenzwerte sind auf die Maßeinheit "Impulse/s<sup>2</sup>" bezogen.

- $0.28 \leq \text{Beschleunigung} \leq 9.5e9$
- $0.28 \leq \text{Verzögerung} \leq 9.5e9$

Die Grenzwerte für andere Maßeinheiten müssen entsprechend der gegebenen Mechanik umgerechnet werden.

---

#### **Hinweis**

Änderungen der Geschwindigkeitsgrenzen ("Start/Stopp-Geschwindigkeit", sowie "Maximale Geschwindigkeit") beeinflussen die Beschleunigungs- und Verzögerungswerte der Achse. Die Hochlauf- und Rücklaufzeiten bleiben erhalten.

---

### **Aktiviere Ruckbegrenzung (ab Technologieobjekt Achse V2.0)**

Aktivieren Sie mit diesem Optionskästchen die Ruckbegrenzung.

---

#### **Hinweis**

Im Fehlerfall verzögert die Achse mit der konfigurierten Notstopp-Verzögerung. Eine aktivierte Ruckbegrenzung wird hierbei nicht berücksichtigt.

---

## Verrundungszeit / Ruck (ab Technologieobjekt Achse V2.0)

Die Parameter der Ruckbegrenzung können Sie im Feld "Verrundungszeit" oder alternativ im Feld "Ruck" eingeben:

- Stellen Sie den gewünschten Ruck für die Beschleunigungs- und Verzögerungsrampe im Feld "Ruck" ein.
- Stellen Sie die gewünschte Verrundungszeit für die Beschleunigungsrampe im Feld "Verrundungszeit" ein.

---

### Hinweis

Die eingestellte, und in der Konfiguration sichtbare Verrundungszeit gilt nur für die Beschleunigungsrampe.

Im Falle, dass sich die Werte von Beschleunigung und Verzögerung unterscheiden, wird die Verrundungszeit der Verzögerungsrampe entsprechend dem Ruck der Beschleunigungsrampe berechnet und verwendet. (siehe auch Verhalten der Achse beim Verwenden der Ruckbegrenzung (Seite 4160))

Die Verrundungszeit der Verzögerung wird wie folgt angepasst:

- **Beschleunigung > Verzögerung**  
Bei der Verzögerungsrampe wird eine kleinere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
  - **Beschleunigung < Verzögerung**  
Bei der Verzögerungsrampe wird eine größere Verrundungszeit als bei der Beschleunigungsrampe eingesetzt.
  - **Beschleunigung = Verzögerung**  
Die Verrundungszeiten der Beschleunigungsrampe und der Verzögerungsrampe sind gleich.
- 

Den Zusammenhang zwischen den Verrundungszeiten und dem Ruck können Sie den folgenden Gleichungen entnehmen:

$$\text{Verrundungszeit (Beschleunigungsrampe)} = \frac{\text{Beschleunigung}}{\text{Ruck}}$$

$$\text{Verrundungszeit (Verzögerungsrampe)} = \frac{\text{Verzögerung}}{\text{Ruck}}$$

Im Anwenderprogramm angestoßene Verfahrtaufträge werden mit dem gewählten Ruck ausgeführt.

Grenzwerte:

Die nachfolgend genannten Grenzwerte sind auf die Maßeinheit Impulse/s<sup>3</sup> bezogen.

- $0.04 \leq \text{Ruck} \leq 1.5e8$

Die Grenzwerte für andere Maßeinheiten müssen entsprechend der gegebenen Mechanik umgerechnet werden.

## Siehe auch

- Verhalten der Achse beim Verwenden der Ruckbegrenzung (Seite 4160)
- Konfiguration - Dynamik Notstopp (Seite 4159)
- Konfiguration der Dynamikwerte im Anwenderprogramm ändern (Seite 4162)

## Konfiguration - Dynamik Notstopp

Konfigurieren Sie im Konfigurationsfenster "Dynamik Notstopp" die Notstopp-Verzögerung der Achse. Im Fehlerfall und beim Sperren der Achse mit der Motion-Control-Anweisung "MC\_Power" (Eingangsparameter StopMode = 0) wird die Achse mit dieser Verzögerung zum Stillstand gebracht.

## Geschwindigkeitsgrenzen

In diesem Bereich werden die im Konfigurationsfenster "Dynamik Allgemein" parametrisierten Geschwindigkeitswerte zur Veranschaulichung erneut angezeigt.

## Verzögerung

Stellen Sie den Verzögerungswert für Notstopp in den Feldern "Notstopp-Verzögerung" oder "Notstopp-Rücklaufzeit" ein.

Den Zusammenhang zwischen Notstopp-Rücklaufzeit und Notstopp-Verzögerung können Sie der folgenden Gleichung entnehmen:

$$\text{Notstopp-Rücklaufzeit} = \frac{\text{Maximale Geschwindigkeit} - \text{Start/Stopp-Geschwindigkeit}}{\text{Notstopp-Verzögerung}}$$

Die Notstopp-Verzögerung muss ausreichend groß gewählt werden, um die Achse im Notfall rechtzeitig zum Stillstand zu bringen (z. B. beim Anfahren der Hardware-Endschalter, vor dem Erreichen des mechanischen Anschlags).

Bei der Wahl der Notstopp-Verzögerung muss die konfigurierte maximale Geschwindigkeit der Achse zugrunde gelegt werden.

Grenzwerte:

Die nachfolgend genannten Grenzwerte sind auf die Maßeinheit "Impulse/s<sup>2</sup>" bezogen.

- $0.28 \leq \text{Notstopp-Verzögerung} \leq 9.5e9$

Die Grenzwerte für andere Maßeinheiten müssen entsprechend der gegebenen Mechanik umgerechnet werden.

**Siehe auch**

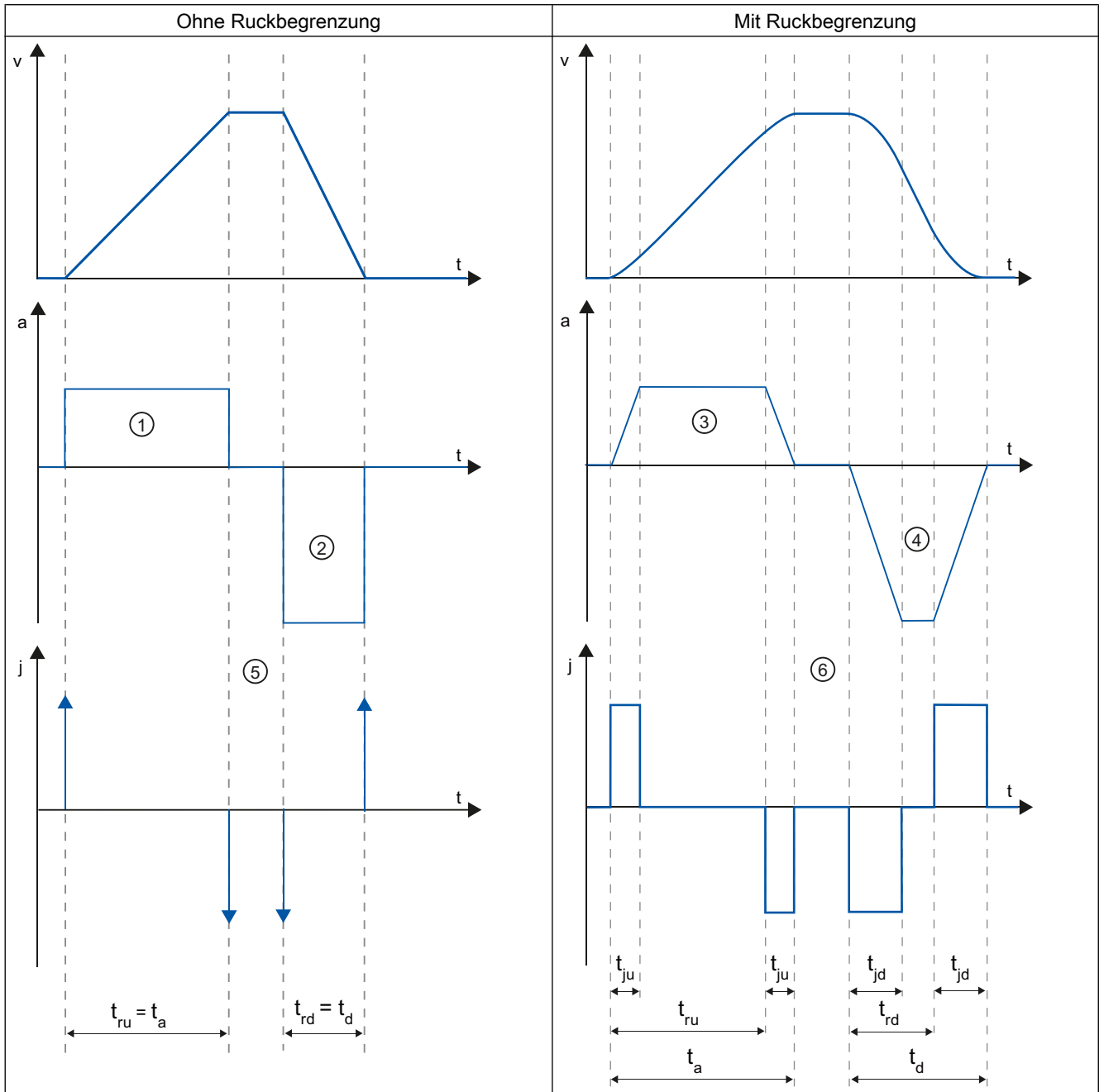
Konfiguration - Dynamik Allgemein (Seite 4156)

Konfiguration der Dynamikwerte im Anwenderprogramm ändern (Seite 4162)

**Verhalten der Achse beim Verwenden der Ruckbegrenzung**

Mit der Aktivierung der Ruckbegrenzung wird die Beschleunigung und Verzögerung der Achse nicht abrupt geändert, sondern entsprechend des eingestellten Rucks, bzw. der eingestellten

Verrundungszeit sanft angeglichen. Die nachfolgende Darstellung zeigt das Verhalten der Achse mit und ohne aktivierter Ruckbegrenzung im Detail.



|          |                 |
|----------|-----------------|
| t        | Zeitachse       |
| v        | Geschwindigkeit |
| a        | Beschleunigung  |
| j        | Ruck            |
| $t_{ru}$ | Hochlaufzeit    |

|          |  |
|----------|--|
| $t_a$    | Zeit zum Beschleunigen der Achse         |
| $t_{rd}$ | Rücklaufzeit                             |
| $t_d$    | Zeit zum Verzögern der Achse             |
| $t_{ju}$ | Verrundungszeit der Beschleunigungsrampe |
| $t_{jd}$ | Verrundungszeit der Verzögerungsrampe    |

Im dargestellten Beispiel wird eine Verfahrbewegung dargestellt, bei der der Wert der Verzögerung ② doppelt so groß wie der der Beschleunigung ① ist. Hieraus ergibt sich eine Rücklaufzeit  $t_{rd}$ , die nur halb so groß wie die Hochlaufzeit  $t_{ru}$  ist.

Ohne Ruckbegrenzung wird die Beschleunigung ① und Verzögerung ② abrupt verändert. Mit aktivierter Ruckbegrenzung wird die Beschleunigung ③ und Verzögerung ④ sanft verändert. Da der Ruck für die gesamte Bewegung gilt, ergibt sich für die Zunahme und Abnahme der Beschleunigung und Verzögerung die gleiche Steigung.

Ohne Ruckbegrenzung werden die Werte für Ruck  $j$  im Moment der Änderung unendlich groß ⑤. Mit Ruckbegrenzung wird der Ruck auf den konfigurierten Wert ⑥ begrenzt.

Die in der Konfiguration angegebene Verrundungszeit  $t_{ju}$  gilt für die Beschleunigungsrampe. Die Verrundungszeit der Verzögerungsrampe  $t_{jd}$  wird aus dem konfigurierten Wert für Ruck und der konfigurierten Verzögerung berechnet.

## Siehe auch

Konfiguration - Dynamik Allgemein (Seite 4156)

## Konfiguration der Dynamikwerte im Anwenderprogramm ändern

Folgende Konfigurationsparameter können Sie zur Laufzeit des Anwenderprogramms in der CPU ändern:

## Beschleunigung und Verzögerung

Sie können die Werte für Beschleunigung und Verzögerung auch zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- `<Achsnamen>.Config.DynamicDefaults.Acceleration`  
zum Ändern der Beschleunigung
- `<Achsnamen>.Config.DynamicDefaults.Deceleration`  
zum Ändern der Verzögerung

Wann Änderungen der Konfigurationsparameter wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.



## Notstopp-Verzögerung

Sie können den Wert für die Notstopp-Verzögerung auch zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variable des Technologieobjekts:

- <Achse>.Config.DynamicDefaults.EmergencyDeceleration

Wann Änderungen des Konfigurationsparameters wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

|  |
|--|
|  <b>WARNUNG</b> |
|--|

|   |
|---|
| Nach dem Ändern dieses Parameters müssen u. U. die Positionen der Hardware-Endschalter sowie weitere sicherheitsrelevante Einstellungen angepasst werden. |
|---|

## Ruckbegrenzung (ab Technologieobjekt Achse V2.0)

Sie können die Ruckbegrenzung auch zur Laufzeit des Anwenderprogramms aktivieren und deaktivieren und den Wert für den Ruck ändern. Verwenden Sie hierzu folgende Variable des Technologieobjekts:

- <Achse>.Config.DynamicDefaults.JerkActive  
zum Aktivieren und Deaktivieren der Ruckbegrenzung
- <Achse>.Config.DynamicDefaults.Jerk  
zum Ändern des Rucks

Wann Änderungen des Konfigurationsparameters wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

## Siehe auch

MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

Konfiguration - Dynamik Allgemein (Seite 4156)

Konfiguration - Dynamik Notstopp (Seite 4159)

## Referenzieren (Technologieobjekt "Achse" ab V2.0)

### Konfiguration - Referenzieren - Allgemein

Konfigurieren Sie im Konfigurationsfenster "Referenzieren - Allgemein" den Eingang des Referenzpunktschalters für das aktive und das passive Referenzieren.

## Eingang Referenzpunktschalter

Wählen Sie in der Klappliste den digitalen Eingang für den Referenzpunktschalter aus. Der Eingang muss alarmfähig sein. Als Eingänge für den Referenzpunktschalter stehen die On-board CPU-Eingänge und Eingänge eines gesteckten Signalboards zur Auswahl.

---

### Hinweis

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt.

Beim Einsatz als Referenzpunktschalter kann dies zu unerwünschten Verzögerungen und damit zu Ungenauigkeiten führen. Je nach Referenziergeschwindigkeit und Ausmaß des Referenzpunktschalters kann u. U. der Referenzpunkt nicht erkannt werden. Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfiler" eingestellt werden.

Die Filterzeit muss kleiner als die Dauer des Eingangssignal am Referenzpunktschalter gewählt werden.

---

## Siehe auch

Ablauf - Aktives Referenzieren (Seite 4168)

## Konfiguration - Referenzieren - Passiv

Konfigurieren Sie im Konfigurationsfenster "Referenzieren - Passiv" die für das passive Referenzieren notwendigen Parameter.

Zum passiven Referenzieren muss die Bewegung anwenderseitig (z. B. durch einen Fahrauftrag der Achse) angestoßen werden. Das passive Referenzieren wird über die Motion Control-Anweisung "MC\_Home" mit verwendetem Eingangsparameter "Mode" = 2 gestartet.

## Seite des Referenzpunktschalters

Wählen Sie hier, ob die Achse an der unteren oder der oberen Seite des Referenzpunktschalters referenziert werden soll.

## Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC\_Home" parametrisierte Position verwendet.

---

### Hinweis

Erfolgt das passive Referenzieren ohne Fahrauftrag der Achse (Achse im Stillstand), so wird mit der nächsten steigenden oder fallenden Flanke am Referenzpunktschalter referenziert.

---

## Konfiguration - Referenzieren - Aktiv

Konfigurieren Sie im Konfigurationsfenster "Referenzieren Aktiv" die für das aktive Referenzieren notwendigen Parameter. Das aktive Referenzieren wird über die Motion Control-Anweisung "MC\_Home" mit verwendetem Eingangsparameter "Mode" = 3 gestartet.

## Erlaube Richtungsumkehr am HW-Endschalter

Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Für die Richtungsumkehr müssen die Hardware-Endschalter aktiviert sein (mindestens der Hardware-Endschalter in Anfahrrichtung muss konfiguriert sein).

Wird der Hardware-Endschalter während des aktiven Referenzierens erreicht, bremst die Achse mit der konfigurierten Verzögerung (nicht mit der Notstopp-Verzögerung) ab und führt eine Richtungsumkehr durch. Der Referenzpunktschalter wird anschließend in umgekehrter Richtung gesucht.

Ist die Richtungsumkehr nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird die Referenzpunktfahrt mit Fehler abgebrochen und die Achse mit der Notstopp-Verzögerung abgebremst.

---

### Hinweis

Stellen Sie nach Möglichkeit durch eine der folgenden Maßnahmen sicher, dass die Maschine bei einer Richtungsumkehr nicht auf einen mechanischen Anschlag fährt:

- Halten Sie die Anfahrgeschwindigkeit gering
  - Vergrößern Sie die konfigurierte Beschleunigung / Verzögerung
  - Vergrößern Sie den Abstand zwischen Hardware-Endschalter und Hardwareanschlag
- 

## Anfahr- / Referenzierrichtung

Mit der gewählten Richtung bestimmen Sie beim aktiven Referenzieren die Anfahrrichtung zur Suche des Referenzpunktschalters, als auch die Referenzierrichtung. Die Referenzierrichtung legt fest mit welcher Fahrrichtung die Achse die konfigurierte Seite des Referenzpunktschalters anfährt, um den Referenzvorgang durchzuführen.

## Seite des Referenzpunktschalters

Wählen Sie hier ob die Achse an der unteren oder der oberen Seite des Referenzpunktschalters referenziert werden soll.

## Geschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher der Referenzpunktschalter während der Referenzpunktfahrt gesucht werden soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stop-Geschwindigkeit  $\leq$  Anfahrsgeschwindigkeit  $\leq$  Maximale Geschwindigkeit

### Referenziergeschwindigkeit

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher in den Referenzpunktschalter zum Referenzieren eingefahren soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stop-Geschwindigkeit  $\leq$  Referenziergeschwindigkeit  $\leq$  Maximale Geschwindigkeit

### Referenzpunktverschiebung

Weicht die gewünschte Referenzposition von der Position des Referenzpunktschalters ab, so kann in diesem Feld die Referenzpunktverschiebung angegeben werden.

Ist der Wert ungleich 0, so führt die Achse nach dem Referenzieren am Referenzpunktschalter folgende Aktionen aus:

1. Verfahren der Achse mit Referenziergeschwindigkeit um den Wert der Referenzpunktverschiebung
2. Nach dem Herausfahren der Referenzpunktverschiebung befindet sich die Achse auf der Referenzpunktposition, welche am Eingangsparameter "Position" der Motion Control-Anweisung "MC\_Home" angegeben wurde.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- $-1.0e12 \leq$  Referenzpunktverschiebung  $\leq 1.0e12$

### Referenzpunktposition

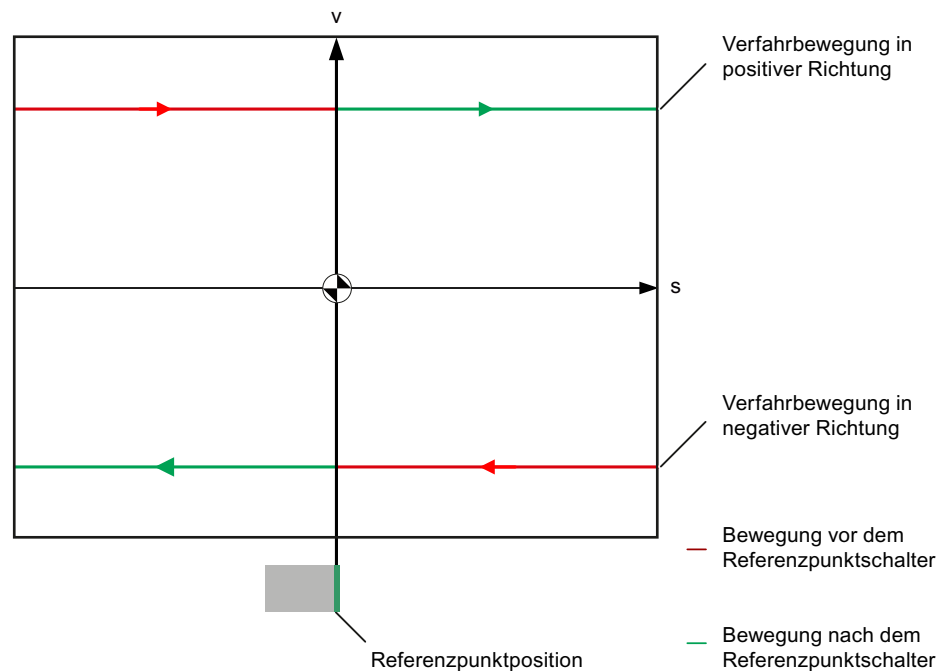
Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC\_Home" parametrisierte Position verwendet.

### Ablauf - Passives Referenzieren

Das Passive Referenzieren starten Sie mit der Motion Control-Anweisung "MC\_Home" (Eingangsparameter Mode = 2). Der Eingangsparameter "Position" gibt die absolute Referenzpunktposition an.

Die folgende Darstellung zeigt beispielhaft den Verlauf eines passiven Referenziervorgangs mit dem folgenden Konfigurationsparameter:

- "Seite des Referenzpunktschalters" = "Obere Seite"



### Bewegung vor dem Referenzpunktschalter (roter Kurvenabschnitt)

Mit dem Start des passiven Referenzierens führt die Motion Control-Anweisung "MC\_Home" selbst keine Referenzierbewegung durch. Die zum Erreichen des Referenzpunktschalters notwendige Verfahrbewegung muss anwenderseitig über andere Motion Control-Anweisungen wie z. B. "MC\_MoveRelative" realisiert werden. War die Achse zuvor schon referenziert bleibt die Variable <Achse>.StatusBits.HomingDone während des passiven Referenzierens TRUE.

### Referenzieren der Achse (Übergang vom roten zum grünen Kurvenabschnitt)

Mit dem Erreichen der konfigurierten Seite des Referenzpunktschalters wird die Achse referenziert. Die aktuelle Position der Achse wird auf die Referenzpunktposition gesetzt. Diese wird am Parameter "Position" der Motion Control-Anweisung "MC\_Home" angegeben. War die Achse zuvor noch nicht referenziert, so wird die Variable <Achse>.StatusBits.HomingDone auf TRUE gesetzt. Die zuvor gestartete Verfahrbewegung wird nicht abgebrochen.

### Bewegung nach dem Referenzpunktschalter (grüner Kurvenabschnitt)

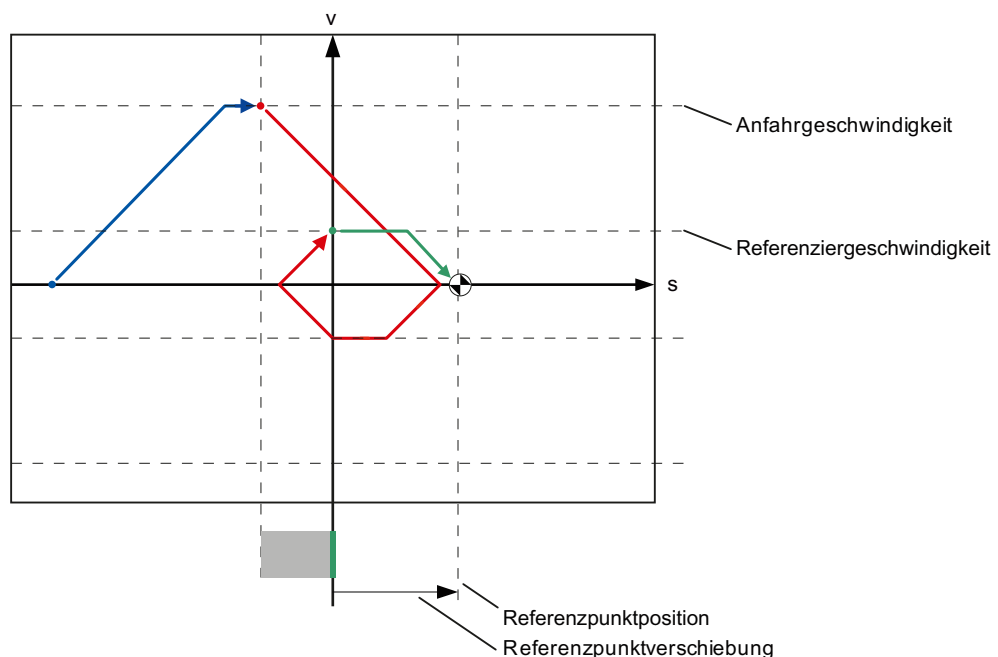
Nach dem Referenzieren am Referenzpunktschalter führt die Achse die zuvor gestartete Verfahrbewegung mit der korrigierten Achsposition bis zum Abschluss fort.

### Ablauf - Aktives Referenzieren

Das Aktive Referenzieren starten Sie mit der Motion Control-Anweisung "MC\_Home" (Eingangsparameter Mode = 3). Der Eingangsparameter "Position" gibt hierbei die absolute Referenzpunktposition an. Das aktive Referenzieren können Sie für Testzwecke alternativ in der Achssteuertafel starten.

Die folgende Darstellung zeigt beispielhaft den Verlauf einer aktiven Referenzpunktfahrt mit folgenden Konfigurationsparametern:

- "Anfahr- / Referenzierrichtung" = "Positive Richtung"
- "Seite des Referenzpunktschalters" = "Obere Seite"
- Wert der "Referenzpunktverschiebung" > 0



#### Suchen des Referenzpunktschalters (blauer Kurvenabschnitt)

Mit dem Start des Aktiven Referenzieren beschleunigt die Achse auf die konfigurierte "Anfahr- / Referenzierrichtung" und sucht mit dieser den Referenzpunktschalter. Die Variable <Achsname>.StatusBits.HomingDone wird auf FALSE gesetzt.

#### Referenzpunktfahrt (roter Kurvenabschnitt)

Mit dem Erkennen des Referenzpunktschalters bremst die Achse in diesem Beispiel ab, kehrt um, um mit der konfigurierten "Referenzierrichtung" auf der konfigurierten Seite des Referenzpunktschalters zu referenzieren. Mit dem Referenzieren wechselt die Variable <Achsname>.StatusBits.HomingDone nach TRUE.

## Herausfahren der Referenzpunktverschiebung (grüner Kurvenabschnitt)

Nach dem Referenzieren verfährt die Achse mit der Referenziergeschwindigkeit die Wegstrecke der Referenzpunktverschiebung. Dort angekommen befindet sich die Achse auf der Referenzpunktposition die am Eingangsparameter "Position" der Motion Control-Anweisung "MC\_Home" angegeben wurde.

### Siehe auch

Konfiguration - Referenzieren - Allgemein (Seite 4163)

## Konfiguration zum Referenzieren im Anwenderprogramm ändern

Ab dem Technologieobjekt "Achse" V2.0 können Sie folgende Konfigurationsparameter zur Laufzeit des Anwenderprogramms in der CPU ändern:

### Passives Referenzieren

Sie können die Seite des Referenzpunktschalters für das passive Referenzieren zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variable des Technologieobjekts:

- <Achsnamen>.Config.Homing.SidePassiveHoming  
zum Ändern der Seite des Referenzpunktschalters

Wann die Änderung des Konfigurationsparameters wirksam wird, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

### Aktives Referenzieren

Sie können die Anfahrriechtung, die Seite des Referenzpunktschalters, die Anfahrgechwindigkeit, die Referenziergeschwindigkeit sowie die Referenzpunktverschiebung für das aktive Referenzieren zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Achsnamen>.Config.Homing.AutoReversal  
zum Ändern der "Richtungsumkehr am HW-Endschalter"
- <Achsnamen>.Config.Homing.Direction  
zum Ändern der "Anfahr- / Referenzierrichtung"
- <Achsnamen>.Config.Homing.SideActiveHoming  
zum Ändern der "Seite des Referenzpunktschalters"
- <Achsnamen>.Config.Homing.FastVelocity  
zum Ändern der "Geschwindigkeit"
- <Achsnamen>.Config.Homing.SlowVelocity  
zum Ändern der "Referenziergeschwindigkeit"
- <Achsnamen>.Config.Homing.Offset  
zum Ändern der "Referenzpunktverschiebung"

Wann die Änderung des Konfigurationsparameters wirksam wird, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

**Siehe auch**

MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

## **11.2.6 Technologieobjekt Auftragstabelle**

### **11.2.6.1 Verwendung des Technologieobjekts Auftragstabelle**

Verwenden Sie das Technologieobjekt "Auftragstabelle", wenn Sie mehrere Einzelaufträge zur Steuerung einer Achse zu Bewegungssequenzen zusammenfassen möchten. Das Technologieobjekt kann ab der Technologieversion V2.0 eingesetzt werden.

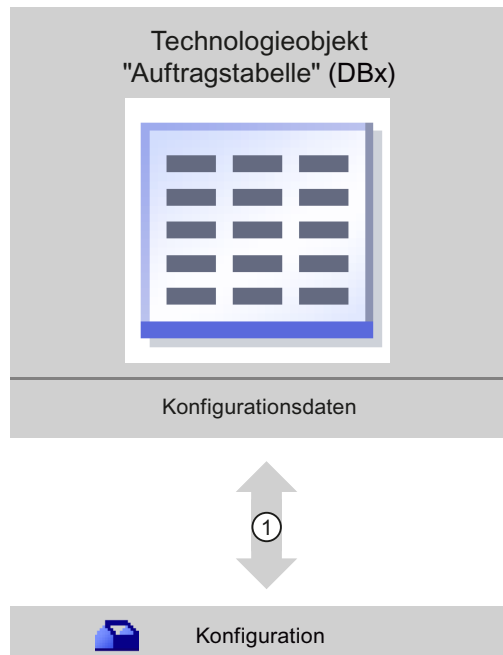
Die Bewegungssequenz konfigurieren Sie in Tabellenform in einem Konfigurationsdialog.

Das Bewegungsprofil der Bewegungssequenz kann bereits vor dem Laden des Projekts in die CPU grafisch überprüft werden. Die erstellten Auftragstabellen können anschließend im Anwenderprogramm über die Motion Control-Anweisung "MC\_CommandTable" mit einer Achse verbunden und verwendet werden. Die Auftragstabelle kann komplett oder nur zum Teil abgearbeitet werden.



### 11.2.6.2 Werkzeuge des Technologieobjekts Auftrags-tabelle

Für das Technologieobjekt "Auftrags-tabelle" steht Ihnen im TIA-Portal das Werkzeug "Konfiguration" zur Verfügung. Die nachfolgende Darstellung zeigt das Zusammenwirken des Werkzeugs mit dem Technologieobjekt:



① Schreiben und Lesen der Konfiguration des Technologieobjekts

### Konfiguration

Konfigurieren Sie mit dem Werkzeug "Konfiguration" folgende Eigenschaften des Technologieobjekts "Auftrags-tabelle":

- Erstellen Sie eine oder mehrere Bewegungssequenzen durch Konfigurieren von Einzelaufträgen.
- Konfigurieren Sie die grafische Darstellung zur Überprüfung Ihrer Bewegungssequenz anhand einer bereits konfigurierten Achse oder einer konfigurierbaren Beispielachse.

Die Daten der Bewegungssequenz werden im Datenbaustein des Technologieobjekts gespeichert.

### 11.2.6.3 Technologieobjekt Auftrags-tabelle hinzufügen

Um ein Technologieobjekt "Auftrags-tabelle" im Projektnavigator hinzuzufügen, gehen Sie folgendermaßen vor:

### Voraussetzungen

- Ein Projekt mit einer CPU S7-1200 ist angelegt.
- Die Firmwareversion der CPU ist größer oder gleich V2.1

### Vorgehen

1. Öffnen Sie in der Projektnavigation den Ordner der CPU.
2. Öffnen Sie den Ordner Technologieobjekte.
3. Doppelklicken Sie auf "Neues Objekt hinzufügen".  
Der Dialog "Neues Objekt hinzufügen" wird geöffnet.
4. Wählen Sie die Technologie "Motion".
5. Öffnen Sie den Ordner "Motion Control".
6. Öffnen Sie den Ordner "S7-1200 Motion Control"
7. Wählen Sie die Version "V2.0" des Ordners "S7-1200 Motion Control" (klicken Sie hierzu auf den Eintrag der Version).
8. Wählen Sie das Objekt "TO\_CommandTable".
9. Passen Sie im Eingabefeld "Name" den Namen der Auftragsstabelle ihren Bedürfnissen an.
10. Wählen Sie die Option "Manuell", falls Sie die vorgeschlagene Datenbausteinnummer ändern möchten.
11. Klicken Sie auf "Weitere Informationen", wenn Sie eigene Informationen zum Technologieobjekt ergänzen möchten.
12. Klicken Sie auf die Schaltfläche "OK", wenn Sie das Technologieobjekt hinzufügen möchten.  
Klicken Sie auf die Schaltfläche "Abbrechen", wenn Sie die Eingaben verwerfen möchten.

### Ergebnis

Das neue Technologieobjekt wird erzeugt und in der Projektnavigation im Ordner "Technologieobjekte" abgelegt.

#### 11.2.6.4 Technologieobjekt Auftragsstabelle konfigurieren

##### Arbeiten mit dem Konfigurationsdialog

Die Eigenschaften des Technologieobjekts konfigurieren Sie im Konfigurationsfenster. Um das Konfigurationsfenster des Technologieobjekts zu öffnen, gehen Sie folgendermaßen vor:





1. Öffnen Sie im Projektnavigator die Gruppe des gewünschten Technologieobjekts.
2. Doppelklicken Sie auf das Objekt "Konfiguration".

Die Konfiguration ist in folgende Kategorien aufgeteilt:

- **Grundparameter**  
Die Grundparameter enthalten alle Parameter, die für eine funktionsfähige Auftragstabelle konfiguriert werden müssen.
- **Erweiterte Parameter**  
Die Erweiterten Parameter enthalten die Parameter der Beispielsachse, bzw. zeigen die Parameterwerte der gewählten Achse an.

## Symbole des Konfigurationsfensters

Symbole in der Bereichsnavigation der Konfiguration zeigen weitere Details zum Status der Konfiguration:

|   |   |
|---|---|
|    | <b>Die Konfiguration enthält Voreinstellungswerte und ist vollständig.</b><br>Die Konfiguration enthält ausschließlich voreingestellte Werte. Mit diesen voreingestellten Werten ist der Einsatz des Technologieobjekts ohne weitere Änderung möglich.                                      |
|    | <b>Die Konfiguration enthält vom Anwender definierte Werte und ist vollständig</b><br>Alle Eingabefelder der Konfiguration enthalten gültige Werte, und mindestens ein voreingestellter Wert wurde geändert.  |
|    | <b>Die Konfiguration ist unvollständig oder fehlerhaft</b><br>Mindestens ein Eingabefeld oder eine Klappliste beinhaltet einen ungültigen Wert. Das entsprechende Feld oder die Klappliste wird rot hinterlegt. Beim Anklicken zeigt Ihnen die Roll-out-Fehlermeldung die Fehlerursache an. |
|  | <b>Die Konfiguration enthält nicht miteinander vereinbare Parameterwerte</b><br>Die Konfiguration enthält Parameterwerte, die sich entweder in ihrer Größe oder der Logik widersprechen. Das entsprechende Feld oder die Klappliste wird gelb hinterlegt.                                   |

## Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 4138)

Grundparameter (Seite 4173)

Erweiterte Parameter (Seite 4187)

## Grundparameter

### Konfiguration - Allgemein

Konfigurieren Sie im Konfigurationsfenster "Allgemein" den Namen des Technologieobjekts.

#### Name

Definieren Sie in diesem Feld den Namen der Auftragstabelle, bzw. den Namen des Technologieobjekts "Auftragstabelle". Das Technologieobjekt wird unter diesem Namen in der Projektnavigation aufgelistet.

## Siehe auch

Konfiguration - Auftragstabelle (Seite 4174)

Kontextmenübefehle - Auftragstabelle (Seite 4177)

Arbeiten mit dem Kurvendiagramm (Seite 4179)

Kontextmenübefehle - Kurvendiagramm (Seite 4183)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 4184)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 4186)

## Konfiguration - Auftragstabelle

Erstellen Sie im Konfigurationsfenster "Auftragstabelle" die gewünschte Bewegungssequenz und überprüfen Sie das Ergebnis anhand der grafischen Darstellung im Kurvendiagramm.

---

### Hinweis

Geringe Abweichungen in der Darstellung des Zeitverhaltens und der Position im Kurvenverlauf zur realen Bewegung der Achse sind möglich. Bewegungsverläufe als Reaktion auf das Erreichen von SW-Endschaltern werden nicht dargestellt.

---

## Aktiviere Warnungen

Aktivieren Sie mit diesem Optionskästchen die Anzeige von Warnungen in der Auftragstabelle.

## Verwende Achsparameter von

Wählen Sie in der Klappliste welche Achsparameter als Grundlage für die grafische Darstellung und die Überprüfung der Bewegungssequenz dienen sollen. Wählen Sie "Beispielachse", wenn Sie im Ordner "Technologieobjekte" noch keine Achse eingefügt haben oder Werte, die in keiner vorhandenen Achse konfiguriert wurden, verwenden möchten. Die Eigenschaften der Beispielachse konfigurieren Sie in unter "Erweiterte Parameter".

Für die Abarbeitung der Auftragstabelle im Anwenderprogramm werden die Achsparameter der am Parameter "Axis" gewählten Achse verwendet.

## Spalte: Schritt

Zeigt die Schrittnummer des Auftrags an.

## Spalte: Auftragsstyp

Wählen Sie in dieser Spalte die Auftragsarten aus, welche bei der Abarbeitung der Auftragstabelle ausgeführt werden sollen. Es können bis zu 32 Aufträge eingetragen werden.

Die Aufträge werden sequenziell abgearbeitet. Wählen Sie zwischen folgenden Einträgen und Auftragstypen:

- **Empty**  
Der Eintrag dient als Platzhalter für evtl. einzufügende Aufträge. Beim Abarbeiten der Auftragsstabelle wird der Empty-Eintrag ignoriert.
- **Halt**  
Achse anhalten  
(der Auftrag wirkt nur nach einem "Velocity set point"-Auftrag)
- **Positioning Relative**  
Achse relativ positionieren
- **Positioning Absolute**  
Achse absolut positionieren
- **Velocity set point**  
Achse mit Geschwindigkeitsvorgabe verfahren
- **Wait**  
Wartet, bis die angegebene Dauer abgelaufen ist. Wait stoppt keine laufenden Verfahrbewegungen.
- **Separator**  
Fügt oberhalb der markierten Zeile eine Separator-Zeile ein. Die Separator-Zeile dient als Bereichsgrenze für die grafische Darstellung des Kurvendiagramms.  
Verwenden Sie Separator-Zeilen, wenn Sie Teile der Auftragsstabelle abarbeiten möchten.

### Spalte: Position

Geben Sie in dieser Spalte die Position, bzw. den Verfahrweg für den gewählten Auftrag an:

- **Auftrag "Positioning Relative"**  
Der Auftrag verfährt die Achse um den angegebenen Verfahrweg.
- **Auftrag "Positioning Absolute"**  
Der Auftrag fährt die Achse auf die angegebene Position.
- **Separator**  
Der angegebene Wert gibt die Startposition für die grafische Darstellung an.

Grenzwerte (unabhängig von der gewählten Anwindereinheit):

- $-1.0e12 \leq \text{Position} / \text{Verfahrweg} \leq -1.0e-12$
- $1.0e-12 \leq \text{Position} / \text{Verfahrweg} \leq 1.0e12$
- $\text{Position} / \text{Verfahrweg} = 0.0$

### Spalte: Geschwindigkeit

Geben Sie in dieser Spalte die Geschwindigkeit für den gewählten Auftrag an:

- **Auftrag "Positioning Relative"**  
Der Auftrag verfährt die Achse mit der angegebenen Geschwindigkeit.  
Ist der gewählte Verfahrensweg nicht groß genug, so wird die angegebene Geschwindigkeit nicht erreicht.
- **Auftrag "Positioning Absolute"**  
Der Auftrag verfährt die Achse mit der angegebenen Geschwindigkeit.  
Liegt die Zielposition zu nahe an der Startposition, so wird die angegebene Geschwindigkeit nicht erreicht.
- **Auftrag " Velocity set point "**  
Der Auftrag verfährt die Achse mit der angegebenen Geschwindigkeit.  
Wird eine zu kurze Laufzeit gewählt, so wird die angegebene Geschwindigkeit während der Dauer des Auftrags nicht erreicht.

Grenzwerte (unabhängig von der gewählten Anwindereinheit):

- Für die Aufträge: "Positioning Relative" und "Positioning Absolute"
  - $1.0e-12 \leq \text{Geschwindigkeit} \leq 1.0e12$
- Für den Auftrag: "Velocity set point"
  - $-1.0e12 \leq \text{Geschwindigkeit} \leq 1.0e12$
  - $1.0e-12 \leq \text{Geschwindigkeit} \leq 1.0e12$
  - Geschwindigkeit = 0.0

### Spalte: Dauer

Geben Sie in dieser Spalte die Dauer für den gewählten Auftrag an:

- **Auftrag " Velocity set point "**  
Der Auftrag verfährt die Achse über die angegebene Dauer. Zur Dauer gehört sowohl die Beschleunigungsphase, als auch die Konstantfahrphase. Nach Ablauf der Dauer wird zum nächsten Auftrag übergegangen.
- **Auftrag "Wait"**  
Wartet, bis die angegebene Dauer abgelaufen ist.

Grenzwerte (unabhängig von der gewählten Anwindereinheit):

- $0.001s \leq \text{Dauer} \leq 64800s$

### Spalte: Nächster Schritt

Wählen Sie in der Klappliste den Übergangsmodus zum nächsten Schritt:

- **Auftrag abschließen**  
Der Auftrag wird abgeschlossen. Der nächste Auftrag wird ohne zeitliche Lücke angehängt.
- **Bewegung überschleifen**  
Die Bewegung des aktuellen Auftrags wird mit der Bewegung des nachfolgenden Auftrags überschleifen. Der Übergangsmodus "Bewegung überschleifen" ist beim Auftragsstyp "Positioning Relative" und "Positioning Absolute" auswählbar.  
Die Bewegung wird mit Bewegungen folgender Auftragsstypen überschleifen:
  - Positioning Relative
  - Positioning Absolute
  - Velocity set point

Mit anderen Auftragsstypen wird nicht überschleifen.

Das detaillierte Verhalten der Achse beim Anhängen bzw. Überschleifen eines Auftrags entnehmen Sie dem Kapitel: Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 4184)

### Spalte: Schrittcode

Geben Sie in dieser Spalte einen Zahlenwert/Bitmuster ein, die am Ausgangsparameter "StepCode" der Motion Control-Anweisung "MC\_CommandTable" während der Bearbeitung des Auftrags ausgegeben werden soll.

Grenzwerte:

- $0 \leq \text{Kennziffer} \leq 65535$

### Siehe auch

Konfiguration - Allgemein (Seite 4173)

Kontextmenübefehle - Auftragstabelle (Seite 4177)

Arbeiten mit dem Kurvendiagramm (Seite 4179)

Kontextmenübefehle - Kurvendiagramm (Seite 4183)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 4184)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 4186)

### Kontextmenübefehle - Auftragstabelle

In der Auftragstabelle stehen folgende Kontextmenübefehle zur Verfügung:

### Empty-Zeile einfügen

Fügt oberhalb der markierten Zeile eine Empty-Zeile ein.

Der Kontextmenübefehl ist nur ausführbar, wenn am Ende der Auftragstabelle ausreichend Empty-Zeilen zur Verfügung stehen.

### Empty-Zeile hinzufügen

Fügt unterhalb der markierten Zeile eine Empty-Zeile ein.

Der Kontextmenübefehl ist nur ausführbar, wenn am Ende der Auftragstabelle ausreichend Empty-Zeilen zur Verfügung stehen.

### Separator-Zeile einfügen

Fügt oberhalb der markierten Zeile eine Separator-Zeile ein.

Zwei aufeinander folgende Separator-Zeilen sind nicht möglich.

### Separator-Zeile hinzufügen

Fügt unterhalb der markierten Zeile eine Separator-Zeile ein.

Zwei aufeinander folgende Separator-Zeilen sind nicht möglich; das Hinzufügen eine Separator-Zeile an das Ende der Auftragstabelle ist ebenfalls nicht möglich.

### Ausschneiden

Entfernt die markierten Zeilen oder den Inhalt der markierten Zelle und legt sie in die Zwischenablage.

Markierte Zeilen werden entfernt; nachfolgende Zeilen der Auftragstabelle werden nach oben verschoben.

### Kopieren

Kopiert die markierten Zeilen oder den Inhalt der markierten Zelle und legt sie in die Zwischenablage.

### Einfügen

- Markierte Zeilen:  
Fügt die Zeilen der Zwischenablage oberhalb der markierten Zeile ein.
- Markierte Zelle:  
Fügt den Inhalt der Zwischenablage in die markierte Zeile ein.

Der Kontextmenübefehl ist nur ausführbar, wenn am Ende der Auftragstabelle ausreichend Empty-Zeilen zur Verfügung stehen.



## Ersetzen

Ersetzt die markierten Zeilen durch die Zeilen der Zwischenablage.

## Löschen

Löscht die markierten Zeilen. Nachfolgende Zeilen der Auftragstabelle werden nach oben verschoben.

## Siehe auch

Konfiguration - Allgemein (Seite 4173)

Konfiguration - Auftragstabelle (Seite 4174)

Arbeiten mit dem Kurvendiagramm (Seite 4179)

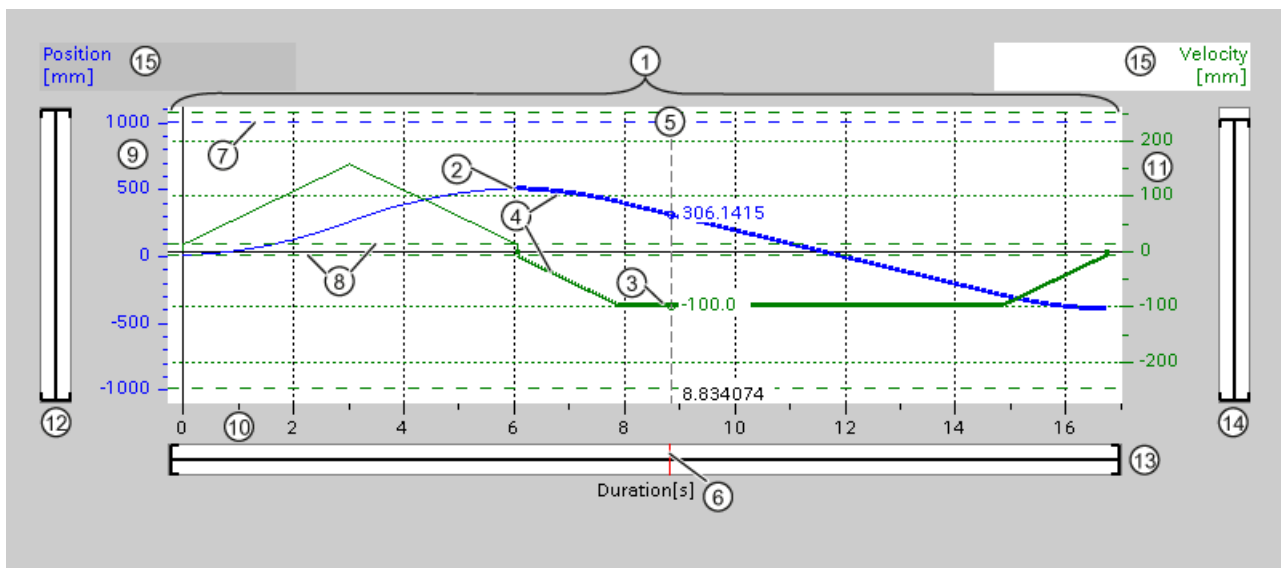
Kontextmenübefehle - Kurvendiagramm (Seite 4183)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 4184)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 4186)

## Arbeiten mit dem Kurvendiagramm

### Ansicht und Komponenten des Kurvendiagramms



|   |  |
|---|--|
| ① | Kurvenfenster                                |
| ② | Positionskurve                               |
| ③ | Geschwindigkeitskurve                        |
| ④ | Kurvenabschnitt eines selektierten Auftrags  |
| ⑤ | Messlineal                                   |
| ⑥ | Positionsmarkierung des Messlineals          |
| ⑦ | Position der SW-Endschalter                  |
| ⑧ | Start/Stop-Geschwindigkeit                   |
| ⑨ | Skalenbereich Positionsbereich               |
| ⑩ | Skalenbereich Zeitachse                      |
| ⑪ | Skalenbereich Geschwindigkeitsachse          |
| ⑫ | Verschiebepfeilbereich Positionsbereich      |
| ⑬ | Verschiebepfeilbereich Zeitachse             |
| ⑭ | Verschiebepfeilbereich Geschwindigkeitsachse |
| ⑮ | Wahl des Gitternetzes                        |

### Separator-Abschnitte auswählen

Besteht die Auftragstabelle aus mehreren durch Separator getrennten Abschnitten, so können die Abschnitte im Kurvendiagramm durch Auswahl eines Auftrags innerhalb des Abschnitts ausgewählt werden.

### Aufträge auswählen

Aufträge können im Kurvendiagramm und in der Auftragstabelle ausgewählt werden:

- Klicken Sie im Kurvendiagramm auf einen Punkt der Geschwindigkeits- oder Positionskurve. Der dazugehörige Auftrag wird in der Auftragstabelle markiert.
- Markieren Sie in der Auftragstabelle einen Auftrag.  
Die entsprechenden Kurvenbereiche werden im Kurvendiagramm hervorgehoben.

### Sichtbaren Bereich des Kurvendiagramms auswählen

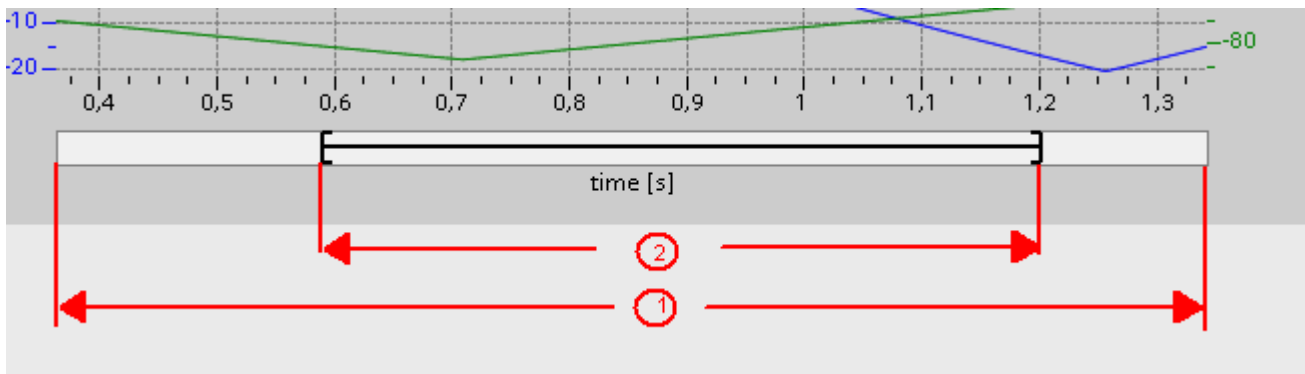
Gehen Sie wie nachfolgend beschrieben vor, um den darzustellenden Ausschnitt des Kurvendiagramms anzupassen:

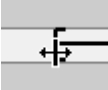

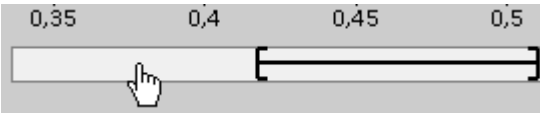
Auswahl der Skalierung im Kontextmenü:

- Skalierung auf Kurven:  
Skaliert die Achsen so, dass die Positions- und die Geschwindigkeitskurve sichtbar ist.
- Skalierung auf Kurven und Grenzen:  
Skaliert die Achsen so, dass die Positions- und die Geschwindigkeitskurve, die Positionen der aktivierten Software-Endschalter und die minimale und maximale Geschwindigkeitsgrenze sichtbar ist.

Die jeweils ausgewählte Ansicht wird im Kontextmenü mit einem Haken markiert.

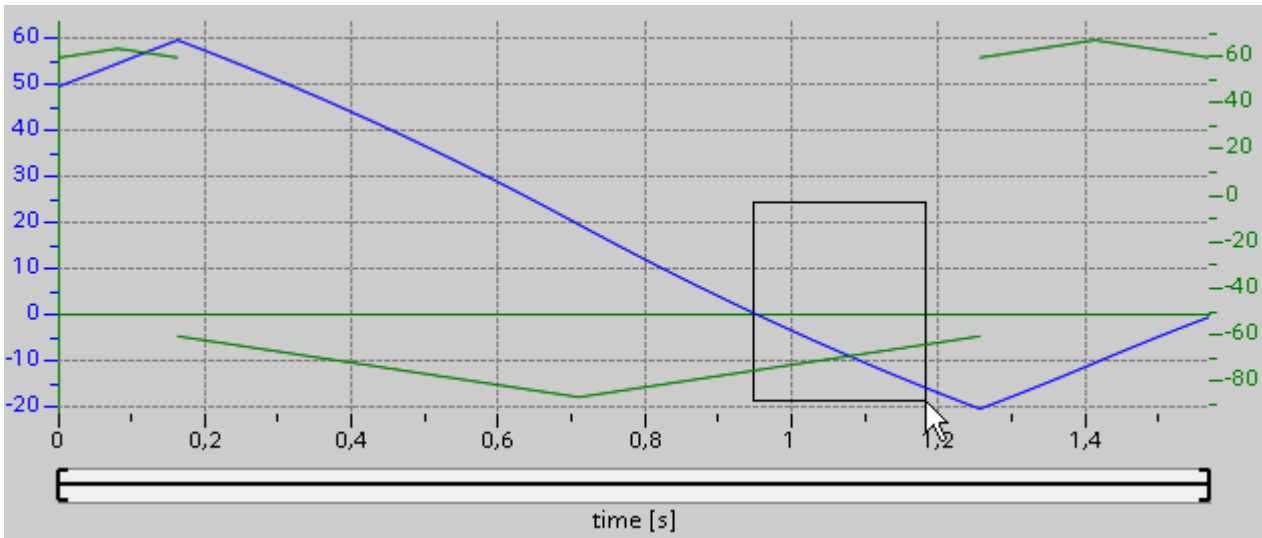
Darzustellenden Ausschnitt im Skalenbereich wählen:



|   |  |
|---|--|
|   |  |
| ① | Bereich in dem sich Kurvenwerte und / oder Grenzwerte befinden. (siehe Auswahl im Kontextmenü)   |
| ② | <p>Ausgewählter Bereich, der im Kurvenfenster dargestellt wird.<br/>                 Den Auswahlbereich stellen Sie am rechten und linken Rand mit dem Randcursor ein.</p>  <p>Die Lage innerhalb des Bereichs ① stellen Sie mit dem Verschiebecursor ein.</p>  <p>Die Lage kann auch durch Klick in Bereich ① bestimmt werden.</p>  |

Darzustellenden Ausschnitt mit der Maus wählen:

Ziehen Sie mit der Maus durch klicken und ziehen einen Bereich im Kurvendiagramm auf. Nach dem Loslassen der Maus wird die Darstellung des Kurvenbereichs auf die Auswahl vergrößert.



Letzte Änderung des Ausschnittes zurücknehmen:

Wählen Sie den Kontextmenübefehl "Zoom rückgängig", um die letzte Änderung des Ausschnittes zurück zu nehmen.

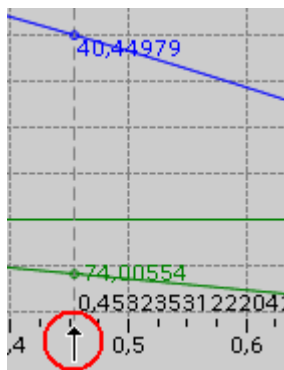
### Gitternetz synchronisieren

Wählen Sie durch Klicken der Achsskalen, ob das Gitternetz mit der Positionssachse oder mit der Geschwindigkeitsachse synchronisiert werden soll.

### Kurvenwerte am Messlineal auslesen

Aktivieren Sie das Messlineal über den Kontextmenübefehl "Messlineal einblenden".

Mit dem Messlinealcursor bewegen Sie das Messlineal auf beliebige Stellen Ihrer Kurven.



## **Siehe auch**

- Konfiguration - Allgemein (Seite 4173)
- Konfiguration - Auftragstabelle (Seite 4174)
- Kontextmenübefehle - Auftragstabelle (Seite 4177)
- Kontextmenübefehle - Kurvendiagramm (Seite 4183)
- Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 4184)
- Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 4186)

## **Kontextmenübefehle - Kurvendiagramm**

Im Bereich des Kurvenfensters stehen folgende Kontextmenübefehle zur Verfügung:

### **Zoom 100%**

Wählt einen Zoomfaktor mit dem 100% der Kurvenwerte und / oder Grenzwerte sichtbar sind.

### **Zoom rückgängig**

Nimmt die letzte Zoomänderung zurück.

### **Skalierung auf Kurven**

Skaliert die Achsen so, dass die Positions- und die Geschwindigkeitskurve sichtbar ist.

### **Skalierung auf Kurven und Grenzen**

Skaliert die Achsen so, dass die Positions- und die Geschwindigkeitskurve, die Positionen der aktivierten Software-Endschalter und die minimale und maximale Geschwindigkeitsgrenze sichtbar ist.

### **Geschwindigkeitsgrenzen einblenden**

Blendet die Linien der Geschwindigkeitsgrenzen ein.

### **Software-Endschalter einblenden**

Blendet die Linien der Software-Endschalter ein.

### **Messlineal einblenden**

Blendet das Messlineal ein / aus.

Verwenden Sie das Messlineal, wenn Sie einzelne Werte der Kurven betrachten möchten.

**Siehe auch**

Konfiguration - Allgemein (Seite 4173)

Konfiguration - Auftragstabelle (Seite 4174)

Kontextmenübefehle - Auftragstabelle (Seite 4177)

Arbeiten mit dem Kurvendiagramm (Seite 4179)

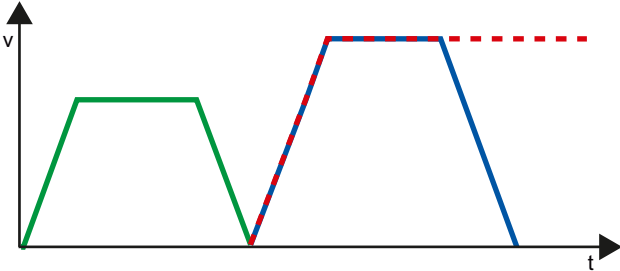
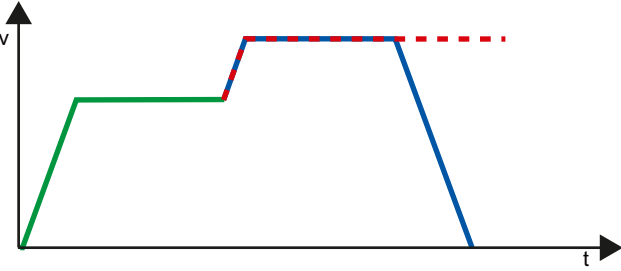
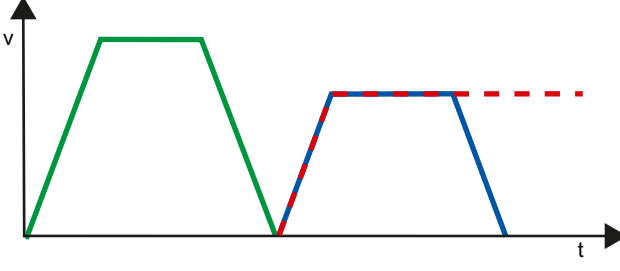
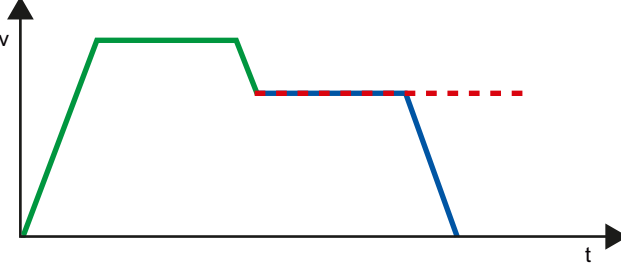
Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 4184)

Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 4186)

**Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen"**

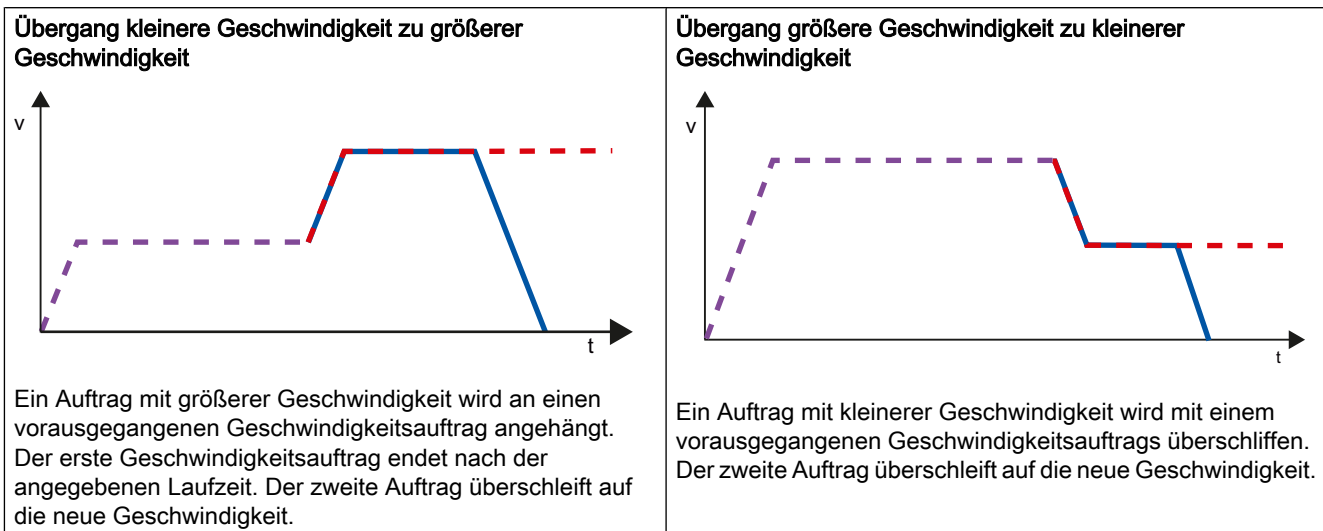
Die nachfolgenden Diagramme zeigen den Bewegungsübergang bei unterschiedlich gewählten Übergangsmodi in der Spalte "Nächster Schritt":

### Bewegungsübergang bei vorausgehenden Positionieraufträgen

| Auftrag abschließen   | Bewegung überschleifen   |
|---|--|
| <p><b>Übergang kleinere Geschwindigkeit zu größerer Geschwindigkeit</b></p>  <p>Ein Auftrag mit größerer Geschwindigkeit wird an einen vorherigen Positionierauftrag angehängt. Der erste Positionierauftrag endet an seiner Zielposition mit Geschwindigkeit "0". Der zweite Auftrag startet aus dem Stillstand.</p>  | <p><b>Übergang kleinere Geschwindigkeit zu größerer Geschwindigkeit</b></p>  <p>Ein Auftrag mit größerer Geschwindigkeit wird mit einem vorherigen Positionierauftrag überschliffen. Der erste Positionierauftrag endet ohne Stillstand an der Zielposition. Der zweite Auftrag überschleift auf die neue Geschwindigkeit.</p>   |
| <p><b>Übergang größere Geschwindigkeit zu kleinerer Geschwindigkeit</b></p>  <p>Ein Auftrag mit kleinerer Geschwindigkeit wird an ein vorherigen Positionierauftrag angehängt. Der erste Positionierauftrag endet an seiner Zielposition mit Geschwindigkeit "0". Der zweite Auftrag startet aus dem Stillstand.</p> | <p><b>Übergang größere Geschwindigkeit zu kleinerer Geschwindigkeit</b></p>  <p>Ein Auftrag mit kleinerer Geschwindigkeit wird mit einem vorherigen Positionierauftrag überschliffen. Der erste Positionierauftrag endet ohne Stillstand an der Zielposition. Der erste Auftrag überschleift auf die neue Geschwindigkeit.</p> |

|  |   |
|--|---|
|  | 1. Auftrag "Positioning Relative" oder "Positioning Absolute" |
|  | 2. Auftrag "Velocity set point"                               |
|  | 2. Auftrag "Positioning Relative" oder "Positioning Absolute" |

## Bewegungsübergang bei vorausgehenden Geschwindigkeitsaufträgen



|     |   |
|-----|---|
| —   | 1. Auftrag "Velocity set point"                               |
| - - | 2. Auftrag "Velocity set point"                               |
| —   | 2. Auftrag "Positioning Relative" oder "Positioning Absolute" |

### Siehe auch

- Konfiguration - Allgemein (Seite 4173)
- Konfiguration - Auftragstabelle (Seite 4174)
- Kontextmenübefehle - Auftragstabelle (Seite 4177)
- Arbeiten mit dem Kurvendiagramm (Seite 4179)
- Kontextmenübefehle - Kurvendiagramm (Seite 4183)
- Konfiguration der Auftragstabelle im Anwenderprogramm ändern (Seite 4186)

### Konfiguration der Auftragstabelle im Anwenderprogramm ändern

Folgende Konfigurationsparameter können Sie zur Laufzeit des Anwenderprogramms in der CPU ändern:

### Aufträge und zugehörige Werte

Sie können die Parameter der Auftragstabelle auch zur Laufzeit des Anwenderprogramms ändern. Verwenden Sie hierzu folgende Variablen des Technologieobjekts:

- <Tabellenname>.Config.Commands[1..32].Command  
zum Ändern des jeweiligen Auftragstyps
- <Tabellenname>.Config.Commands[1..32].Position  
zum Ändern der Position / des Fahrwegs



- <Tabellenname>.Config.Commands[1..32].Velocity  
zum Ändern der Geschwindigkeit
- <Tabellenname>.Config.Commands[1..32].Duration  
zum Ändern der Dauer
- <Tabellenname>.Config.Commands[1..32].BufferMode  
zum Ändern des Parameters "Nächster Schritt"
- <Tabellenname>.Config.Commands[1..32].Code  
zum Ändern des Schrittcodes

Wann Änderungen der Konfigurationsparameter wirksam werden, entnehmen Sie der Beschreibung der Variablen des Technologieobjekts im Anhang.

## Siehe auch

Konfiguration - Allgemein (Seite 4173)

Konfiguration - Auftragsstabelle (Seite 4174)

Kontextmenübefehle - Auftragsstabelle (Seite 4177)

Arbeiten mit dem Kurvendiagramm (Seite 4179)

Kontextmenübefehle - Kurvendiagramm (Seite 4183)

Auftragsübergang "Auftrag abschließen" / "Bewegung überschleifen" (Seite 4184)

## Erweiterte Parameter

## Diagrammparameter

### Konfiguration - Allgemein

Konfigurieren Sie im Konfigurationsfenster "Allgemein" die grundlegenden Eigenschaften der Diagrammdarstellung des Technologieobjekts "Auftragsstabelle".

---

#### Hinweis

Wurde unter "Verwende Achsparameter von" die Beispielachse gewählt, so kann die Maßeinheit editiert werden. Wurde eine konfigurierte Achse gewählt, so wird der Wert der Achse angezeigt.

---

### Verwende Achsparameter von

Wählen Sie in der Klappliste welche Achsparameter als Grundlage für die grafische Darstellung und die Überprüfung der Bewegungssequenz dienen sollen. Wählen Sie "Beispielachse", wenn Sie im Ordner "Technologieobjekte" noch keine Achse eingefügt haben oder Werte, die in keiner vorhandenen Achse konfiguriert wurden, verwenden möchten.

Für die Abarbeitung der Auftragsstabelle im Anwenderprogramm werden die Achsparameter der am Parameter "Axis" gewählten Achse verwendet.

## Maßeinheit

Tragen Sie in diesem Feld die Maßeinheit für die Beispielachse ein. Wurde unter "Verwende Achsparameter von" eine bereits konfigurierte Achse gewählt, so wird die dort konfigurierte Maßeinheit angezeigt.

## Konfiguration - Dynamik

Konfigurieren Sie im Konfigurationsfenster "Dynamik" die Beschleunigung und Verzögerung als auch die Ruckbegrenzung der Beispielachse.

---

### Hinweis

Wurde unter "Verwende Achsparameter von" die Beispielachse gewählt, so können die nachfolgend beschriebenen Felder editiert werden. Wurde eine konfigurierte Achse gewählt, so werden die Werte der Achse angezeigt.

---

## Beschleunigung / Verzögerung

Stellen Sie die gewünschte Beschleunigung der Beispielachse im Feld "Beschleunigung" ein. Die gewünschte Verzögerung kann im Feld "Verzögerung" eingestellt werden.

Die in der Auftragstabelle konfigurierten Verfahrtaufträge werden mit der gewählten Beschleunigung / Verzögerung berechnet.

Grenzwerte:

- $1.0e-12 \leq \text{Beschleunigung} \leq 1.0e12$
- $1.0e-12 \leq \text{Verzögerung} \leq 1.0e12$

## Aktiviere Ruckbegrenzung

Aktivieren Sie mit diesem Optionskästchen die Ruckbegrenzung.

## Ruck

Stellen Sie den gewünschten Ruck für die Hochlauf- und Rücklauframpe im Feld "Ruck" ein.

Die in der Auftragstabelle konfigurierten Verfahrtaufträge werden mit dem gewählten Ruck berechnet.

Grenzwerte:

- $1.0e-12 \leq \text{Ruck} \leq 1.0e12$

### Konfiguration - Grenzwerte

Konfigurieren Sie im Konfigurationsfenster "Grenzwerte" die Maximale Geschwindigkeit, die Start/Stopp-Geschwindigkeit als auch die SW-Endschalter der Beispielachse.

---

#### Hinweis

Wurde unter "Verwende Achsparameter von" die Beispielachse gewählt, so können die nachfolgend beschriebenen Felder editiert werden. Wurde eine konfigurierte Achse gewählt, so werden die Werte der Achse angezeigt.

---

### Maximale Geschwindigkeit / Start/Stopp-Geschwindigkeit

Definieren Sie in diesen Feldern die maximale zugelassene Geschwindigkeit und die Start/Stopp-Geschwindigkeit der Beispielachse. Die Start/Stopp-Geschwindigkeit ist die minimal zulässige Geschwindigkeit der Beispielachse.

Grenzwerte:

- $1.0e-12 \leq \text{Start/Stopp-Geschwindigkeit} \leq 1.0e12$   
Start/Stopp-Geschwindigkeit = 0.0
- $1.0e-12 \leq \text{Maximale Geschwindigkeit} \leq 1.0e12$   
Maximale Geschwindigkeit = 0.0

Der Wert der maximalen Geschwindigkeit muss größer gleich dem Wert der Start/Stopp-Geschwindigkeit sein.

### Aktiviere Software-Endschalter

Aktivieren Sie mit diesem Optionskästchen die Funktion des unteren und oberen Software-Endschalters. Im Kurvendiagramm werden Bewegungsverläufe als Reaktion auf das Erreichen von SW-Endschaltern nicht dargestellt.

### Unterer / oberer SW-Endschalter

Legen Sie in diesen Feldern den Positionswert des unteren und oberen Software-Endschalters fest.

Grenzwerte:

- $-1.0e12 \leq \text{Unterer Software-Endschalter} \leq -1.0e-12$   
 $1.0e-12 \leq \text{Unterer Software-Endschalter} \leq 1.0e12$   
Unterer Software-Endschalter = 0.0
- $-1.0e12 \leq \text{Oberer Software-Endschalter} \leq -1.0e-12$   
 $1.0e-12 \leq \text{Oberer Software-Endschalter} \leq 1.0e12$   
Oberer Software-Endschalter = 0.0

Der Wert des oberen Software-Endschalters muss größer gleich dem Wert des unteren Software-Endschalters sein.


### 11.2.7 Laden in CPU

Beim Laden in die CPU S7-1200 wird immer sichergestellt, dass Projektdaten nach dem Laden online und offline konsistent sind. Das Laden einzelner Bausteine ist nicht möglich. Beim Selektieren einzelner Bausteine werden immer alle neuen und geänderten Bausteine geladen.

Folgende Objektgruppen können in die CPU geladen werden:

| Kontextmenübefehl<br>"Laden in Gerät" | Beschreibung   |
|---------------------------------------|--|
| Alles                                 | Laden aller neuen und geänderten Bausteine und einer neuen oder geänderten Hardwarekonfiguration |
| Hardwarekonfiguration                 | Laden einer neuen oder geänderten Hardwarekonfiguration  |
| Software                              | Laden aller neuen und geänderten Bausteine   |
| Software (alle Bausteine)             | Laden aller Bausteine  |

Die Daten der Motion Control Technologieobjekte werden in Datenbausteinen gespeichert. Zum Laden eines neuen oder geänderten Technologieobjekts gelten somit die Bedingungen zum Laden von "Bausteinen"

|  |
|--|
|  <b>VORSICHT</b>  |
| <p><b>Mögliche Fehlfunktionen der Achse beim Laden ohne Hardwarekonfiguration</b></p> <p>Bei folgenden Änderungen in der Konfiguration der Achse wird die Hardwarekonfiguration geändert:</p> <ul style="list-style-type: none"> <li>• Änderung des Impulsgenerators (PTO)</li> <li>• Änderung der Adresse der HW-Endschalter</li> <li>• Änderung der Adresse des Referenzpunktschalters</li> </ul> <p>Wird die geänderte Konfiguration der Achse mit den Kontextmenübefehlen "Software" oder "Software (alle Bausteine)" ohne Laden der Hardwarekonfiguration geladen, so kann dies zu Fehlfunktionen der Achse führen.</p> <p>Stellen Sie sicher, dass unter den genannten Bedingungen die aktuelle Hardwarekonfiguration in die CPU geladen wird.</p> |

#### Laden im Betriebszustand RUN der CPU S7-1200 (ab Firmwareversion V2.2)

Ab der Firmwareversion V2.2 der CPU S7-1200 wird beim Laden im Betriebszustand RUN der CPU geprüft, ob ein Laden ohne STOP der CPU möglich ist.

Beim Laden von Datenbausteinen im Betriebszustand RUN gelten folgende Bedingungen:

|                                  | Laden in Ladespeicher | Laden in Arbeitsspeicher |
|----------------------------------|-----------------------|--------------------------|
| Datenbaustein geänderte Werte    | Ja                    | Nein                     |
| Datenbaustein geänderte Struktur | Nein                  | Nein                     |
| Datenbaustein neu                | Ja                    | Ja                       |
| Datenbaustein gelöscht           | Ja                    | Ja                       |

Ab Technologieversion V3.0 können Motion Control Technologieobjekte (Datenbausteine) auch im Betriebszustand RUN der CPU geladen werden.

Technologieobjekte kleiner V3.0 können im Betriebszustand RUN der CPU nicht geladen werden.

Wählen Sie eine der nachfolgend beschriebenen Aktionen, um die geänderte Konfiguration eines Motion Control Technologieobjekts (ab Version V3.0) in den Arbeitsspeicher zu laden:

- **Technologieobjekt Achse und Auftragstabelle**  
Wechseln Sie den Betriebszustand der CPU von STOP nach RUN.
- **Technologieobjekt Achse**  
Sperren Sie die Achse und führen Sie mit der Motion Control-Anweisung "MC\_Reset" ein "Restart" aus.
- **Technologieobjekt Auftragstabelle**  
Stellen Sie sicher, dass die Auftragstabelle nicht verwendet wird. Laden Sie mit der erweiterten Anweisung "READ\_DBL" den Datenbaustein der Auftragstabelle in den Arbeitsspeicher.

---

#### Hinweis

Im Gegensatz zum Laden im Betriebszustand STOP werden im Betriebszustand RUN keine Aktualparameter überschrieben. Änderungen an den Aktualparametern erfolgen erst beim nächsten Betriebszustandwechsel von STOP nach RUN.

---

#### Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 4138)

MC\_Reset: Fehler quittieren (Seite 2641)

### 11.2.8 Inbetriebnahme der Achse - Achssteuertafel

Verwenden Sie die Achsteuertafel, um die Achse im Handbetrieb zu verfahren, die Achseinstellungen zu optimieren und um Ihre Anlage zu testen.

Die Bedienung der Achssteuertafel ist nur möglich, wenn eine Onlineverbindung zur CPU aufgebaut ist.

---

#### Hinweis

##### Reaktionszeiten der Achssteuertafel

Die Reaktionszeit während der Bedienung der Achssteuertafel ist von der Kommunikationslast der CPU abhängig. Schließen Sie alle weiteren Onlinefenster des TIA-Portals um die Reaktionszeit niedrig zu halten.

---

### Schaltfläche "Handsteuerung"

Klicken Sie die Schaltfläche "Handsteuerung", wenn Sie die Achse in Handsteuerung verfahren möchten. Entfernen Sie zuvor im Anwenderprogramm die Achsfreigabe über die Motion Control-Anweisung "MC\_Power". In der Betriebsart "Handsteuerung" übernimmt die Achssteuertafel die Steuerhoheit über die Funktionen der Achse. Das Anwenderprogramm hat bis zum Beenden der Handsteuerung keinen Einfluss auf die Funktionen der Achse.



#### **WARNUNG**

Die Handsteuerung ist nur für eine Achse aktiv. Eine zweite Achse kann evtl. noch im Automatikmodus bewegt werden und zu gefährlichen Situationen führen.

Setzen Sie in diesem Fall die zweite Achse außer Betrieb.

### Schaltfläche "Automatikbetrieb"

Klicken Sie die Schaltfläche "Automatikbetrieb", wenn Sie die Betriebsart "Handsteuerung" beenden möchten. Die Achssteuertafel gibt die Steuerhoheit wieder ab und die Achse kann vom Anwenderprogramm gesteuert werden. Im Anwenderprogramm muss die Achse erneut freigegeben, und bei Bedarf referenziert werden.

Beenden Sie vor dem Wechsel in den Automatikbetrieb laufende Verfahrbewegungen, anderenfalls wird die Achse mit der Notstopp-Verzögerung abgebremst.

### Schaltfläche "Freigabe"

Klicken Sie die Schaltfläche "Freigabe", um die Achse in der Betriebsart "Handsteuerung" freizugeben. Nach ausgeführter Freigabe können die Funktionen der Achssteuertafel genutzt werden.

Sollte die Achse aufgrund fehlender Voraussetzungen nicht freigegeben werden können, so beachten Sie die Fehlermeldung im Feld "Fehlermeldung". Informationen zur Beseitigung des Fehlers entnehmen Sie der "Liste der ErrorIDs und ErrorInfos" im Anhang. Geben Sie die Achse nach der Behebung des Fehlers erneut frei.

### Schaltfläche "Sperrern"

Klicken Sie die Schaltfläche "Sperrern", wenn Sie die Achse in der Betriebsart "Handsteuerung" vorübergehend sperren möchten.

## Bereich "Befehl"

Die Bedienung im Bereich "Befehl" ist nur mit freigegebener Achse möglich. Wählen Sie zwischen folgenden Befehlen:

- **Tippen**  
Der Befehl entspricht dem Motion Control-Auftrag "MC\_MoveJog" im Anwenderprogramm.
- **Positionieren**  
Der Befehl entspricht den Motion Control-Aufträgen "MC\_MoveAbsolute" und "MC\_MoveRelative" im Anwenderprogramm. Zum absoluten Positionieren muss die Achse referenziert sein.
- **Referenzieren**  
Der Befehl entspricht dem Motion Control-Auftrag "MC\_Home" im Anwenderprogramm.
  - Die Schaltfläche "Referenzpunkt setzen" entspricht Mode = 0 (Direktes Referenzieren Absolut)
  - Die Schaltfläche "Aktiv Referenzieren" entspricht Mode = 3 (Aktives Referenzieren)Für das aktive Referenzieren muss in der Achskonfiguration der Referenzpunktschalter konfiguriert sein.  
Die Werte für Anfah-, Referenziergeschwindigkeit und für die Referenzpunktverschiebung werden unverändert aus der Achskonfiguration übernommen.

Je nach Auswahl werden die relevanten Felder zur Eingabe der Sollwerte und Schaltflächen zum Start des Befehls eingeblendet.

## Bereich "Achsstatus"

Ist die Betriebsart "Handsteuerung" aktiviert, so wird im Bereich "Achsstatus" der aktuelle Achs- und Antriebszustand angezeigt. Die aktuelle Position und Geschwindigkeit der Achse wird unter "Aktuelle Werte" angezeigt.

Quittieren Sie anstehende Fehler nach ihrer Beseitigung durch Klicken der Schaltfläche "Quittieren".

Das Feld "Info-Meldung" zeigt erweiterte Informationen zum Status der Achse an.

## Fehlermeldung

Das Feld "Fehlermeldung" zeigt den aktuellen Fehler an. In der Betriebsart "Handsteuerung" kann, nach Beseitigung der Fehlerursache, der Fehlereintrag mit der Schaltfläche "Quittieren" gelöscht werden.

---

### Hinweis

#### Initialwerte für Geschwindigkeit, Beschleunigung / Verzögerung und Ruck

Aus Sicherheitsgründen werden beim Aktivieren der Achssteuertafel die Parameter "Geschwindigkeit" und "Beschleunigung / Verzögerung" und "Ruck" mit lediglich 10% der in der Konfiguration eingestellten Werte initialisiert. Der Parameter "Ruck" wird erst ab Technologieobjekt "Achse" V2.0 eingesetzt.

Zur Initialisierung werden die Werte in der Konfigurationssicht "Erweiterte Parameter > Dynamik > Allgemein" zu Grunde gelegt.

Der Parameter "Geschwindigkeit" der Achssteuertafel wird aus der "Maximalen Geschwindigkeit", der Parameter "Beschleunigung / Verzögerung" aus der "Beschleunigung" der Konfiguration abgeleitet."

Die Parameter "Geschwindigkeit", "Beschleunigung / Verzögerung" und "Ruck" können in der Achssteuertafel geändert werden; die Werte in der Konfiguration bleiben davon unberührt.

---

## Siehe auch

Leitfaden zum Einsatz von Motion Control (Seite 4138)

Arbeiten mit Beobachtungstabellen (Seite 4218)

## 11.2.9 Programmieren

### 11.2.9.1 Übersicht über die Motion Control-Anweisungen

Über die Motion Control-Anweisungen steuern Sie die Achse vom Anwenderprogramm aus. Die Anweisungen starten Motion Control-Aufträge, die die gewünschten Funktionen ausführen.

Den Status der Motion Control-Aufträge, sowie eventuell bei deren Abarbeitung aufgetretene Fehler, können Sie den Ausgangsparametern der Motion Control-Anweisungen entnehmen. Folgende Motion Control-Anweisungen stehen zur Auswahl:

- MC\_Power: Achsen freigeben, sperren (Seite 2636)
- MC\_Reset: Fehler quittieren (Seite 2641)
- MC\_Home: Achsen referenzieren, Referenzpunkt setzen (Seite 2643)
- MC\_Halt: Achsen anhalten (Seite 2646)
- MC\_MoveAbsolute: Achsen absolut positionieren (Seite 2650)



- MC\_MoveRelative: Achsen relativ positionieren (Seite 2653)
- MC\_MoveVelocity: Achsen mit Drehzahlvorgabe fahren (Seite 2657)
- MC\_MoveJog: Achsen in der Betriebsart Tippen bewegen (Seite 2661)
- MC\_CommandTable: Achsaufträge als Bewegungssequenz ausführen (ab Technologieobjekt "Achse" V2.0) (Seite 2665)
- MC\_ChangeDynamic: Dynamikeinstellungen der Achse ändern (ab Technologieobjekt "Achse" V2.0) (Seite 2667)

## Siehe auch

Anwenderprogramm erstellen (Seite 4195)

Programmierhinweise (Seite 4198)

Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart (Seite 4200)

Verfolgung laufender Aufträge (Seite 4200)

Fehleranzeigen der Motion Control-Anweisungen (Seite 4213)

### 11.2.9.2 Anwenderprogramm erstellen

Nachfolgend erstellen Sie im Anwenderprogramm das Grundgerüst zur Steuerung Ihrer Achse. Mit Hilfe der einzufügenden Motion Control-Anweisungen werden alle zur Verfügung stehenden Funktionen der Achse gesteuert.

## Voraussetzung

- Das Technologieobjekt wurde erzeugt und fehlerfrei konfiguriert.

Vor der Erstellung und Test des Anwenderprogramms ist es empfehlenswert, die Funktion der Achse und der entsprechenden Anlagenteile mit der Achssteuertafel zu testen.

## Vorgehen

Gehen Sie zum Erstellen des Anwenderprogramms entsprechend dem nachfolgend beschriebenen Prinzip vor:

1. Doppelklicken Sie in der Projektnavigation Ihren Codebaustein (der Codebaustein muss im zyklischen Programm aufgerufen werden).  
Der Codebaustein wird im Programmiereditor geöffnet und die zur Verfügung stehenden Anweisungen werden eingeblendet.
2. Öffnen Sie die Kategorie "Technologie" und den Ordner "Motion Control" und "S7-1200 Motion Control".
3. Ziehen Sie per Drag & Drop die Anweisung "MC\_Power" in das gewünschte Netzwerk des Codebausteins.  
Der Dialog zur Definition des Instanz-DBs wird geöffnet.

4. Wählen Sie im Dialog zwischen folgenden Alternativen:

**Einzelinstanz**

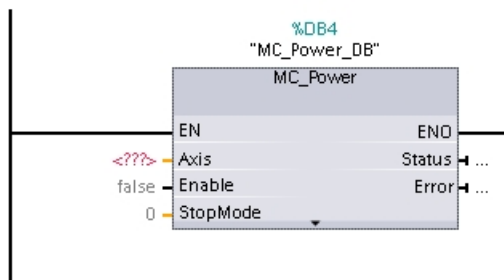
Klicken Sie die Schaltfläche "Einzelinstanz" und wählen Sie, ob Sie den Namen und die Nummer des Instanz-DBs automatisch oder manuell bestimmen möchten.

**Multiinstanz**

Klicken Sie die Schaltfläche "Multiinstanz" und wählen Sie, ob Sie den Namen der Multiinstanz automatisch oder manuell bestimmen möchten.

5. Klicken Sie die Schaltfläche "OK".

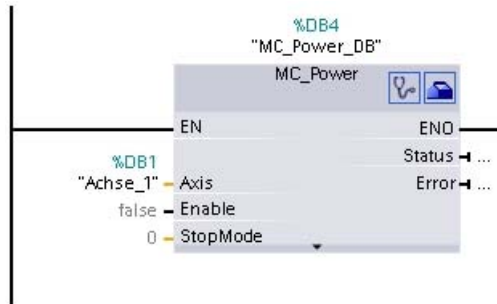
Die Motion Control-Anweisung "MC\_Power" wird in das Netzwerk eingefügt.



Mit "<???" markierte Parameter müssen versorgt werden; alle anderen Parameter sind mit Voreinstellungswerten belegt.

Schwarz dargestellte Parameter sind zum Einsatz der Motion Control-Anweisung notwendig.

6. Markieren Sie in der Projektnavigation das Technologieobjekt und ziehen es per Drag & Drop auf <???.>.



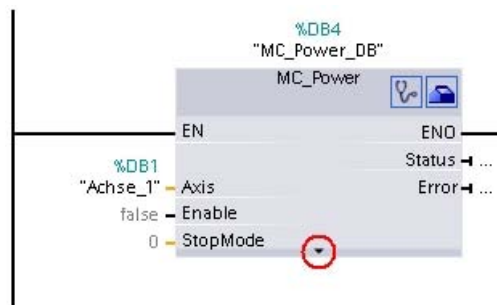
Nach der Auswahl des Datenbausteins des Technologieobjekts stehen Ihnen folgende Schaltflächen zur Verfügung:



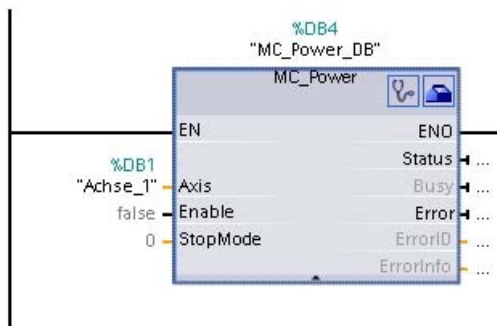
Klicken Sie auf das Stethoskopsymbol, falls Sie die Diagnose des Technologieobjekts öffnen möchten.



Klicken Sie das Werkzeugkastensymbol, falls Sie die Konfigurationssicht des Technologieobjekts öffnen möchten.



Mit Klicken des nach unten gerichteten Pfeils öffnen Sie die Sicht auf weitere Parameter der Motion Control-Anweisung.



Die nun sichtbaren, grau dargestellten Parameter können optional verwendet werden.

7. Fügen Sie entsprechend Schritt 3 bis 6, weitere von Ihnen gewünschte Motion Control-Anweisungen ein.

## Ergebnis

Das Grundgerüst zur Steuerung der Achse ist im Anwenderprogramm angelegt.

Versorgen Sie in weiteren Teilen des Anwenderprogramms die Eingangsparameter der Motion Control-Anweisungen, um die gewünschten Aufträge am Technologieobjekt "Achse" anzustoßen.

Werten Sie die Ausgangsparameter der Motion Control-Anweisungen und die Variablen des Datenbausteins der Achse aus, um die angestoßenen Aufträge und den Status der Achse zu verfolgen.

Details zu den Parametern der Motion Control-Anweisungen entnehmen Sie bitte deren Detailbeschreibung.

## Siehe auch

Übersicht über die Motion Control-Anweisungen (Seite 4194)

Programmierhinweise (Seite 4198)

Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart (Seite 4200)

Verfolgung laufender Aufträge (Seite 4200)

Fehleranzeigen der Motion Control-Anweisungen (Seite 4213)

### 11.2.9.3 Programmierhinweise

Berücksichtigen Sie bei der Erstellung Ihres Anwenderprogramms die nachfolgenden Hinweise:

- **Zyklischer Aufruf der verwendeten Motion Control-Anweisungen**  
Der aktuelle Status der Abarbeitung der Aufträge wird über die Ausgangsparameter der Motion Control-Anweisung zur Verfügung gestellt. Der Status wird mit jedem Aufruf der Motion Control-Anweisung aktualisiert. Sorgen deshalb Sie dafür, dass die verwendeten Motion Control-Anweisungen zyklisch aufgerufen werden.
- **Übernahme der Parameterwerte einer Motion Control-Anweisung**  
Die an den Eingangsparametern anstehenden Parameterwerte werden mit einer positiven Flanke am Eingangsparameter "Execute" beim Aufruf des Bausteins übernommen. Der Motion Control-Auftrag wird mit diesen Parameterwerten gestartet. Nachträglich an der Motion Control-Anweisung geänderte Parameterwerte werden erst beim nächsten Start des Motion Control-Auftrags übernommen.  
Eine Ausnahme bildet der Eingangsparameter "StopMode" der Motion Control-Anweisung "MC\_Power" und "Velocity" der Motion Control-Anweisung "MC\_MoveJog". Eine Änderung des Eingangsparameter wird auch bei "Enable" = TRUE, bzw. "JogForward" und "JogBackward" übernommen. .

- **Programmierung unter Berücksichtigung der Statusinformationen**

Achten Sie bei einer schrittweisen Abarbeitung von Motion Control-Aufträgen vor dem Start eines neuen Auftrags auf den Abschluss des laufenden Auftrags. Prüfen Sie den Abschluss des laufenden Auftrags anhand der Statusmeldungen der Motion Control-Anweisung und der Variablen "StatusBits" des Technologieobjekts.

In den nachfolgenden Beispielen muss die beschriebene Abfolge berücksichtigt werden. Wird sie nicht berücksichtigt, so wird ein Fehler der Achse oder des Auftrags angezeigt.

  - **Freigabe der Achse mit der Motion Control-Anweisung "MC\_Power"**

Bevor die Achse Aufträge zum Verfahren annehmen kann, muss die Achse freigegeben sein. Prüfen Sie die Freigabe der Achse über eine UND-Verknüpfung der Variable <Achse>.StatusBits.Enable = TRUE mit dem Ausgangsparameter Status = TRUE der Motion Control-Anweisung "MC\_Power".
  - **Fehler quittieren mit der Motion Control-Anweisung "MC\_Reset"**

Vor dem Start eines Motion Control-Auftrags müssen mit "MC\_Reset" zu quittierende Fehler quittiert werden. Beheben Sie die Fehlerursache und quittieren Sie den Fehler mit der Motion Control-Anweisung "MC\_Reset". Prüfen Sie vor dem Anstoß eines neuen Auftrags die erfolgreiche Quittierung des Fehlers. Verwenden Sie hierzu eine UND-Verknüpfung der Variable <Achse>.StatusBits.Error = FALSE mit dem Ausgangsparameter Done = TRUE der Motion Control-Anweisung "MC\_Reset".
  - **Referenzieren der Achse mit der Motion Control-Anweisung "MC\_Home"**

Bevor ein MC\_MoveAbsolute-Auftrag gestartet werden kann, muss die Achse referenziert sein. Prüfen Sie nach dem Referenzieren der Achse den erfolgreichen Abschluss über eine UND-Verknüpfung der Variable <Achse>.StatusBits.HomingDone = TRUE mit dem Ausgangsparameter Done = TRUE der Motion Control-Anweisung "MC\_Home".
- **Ablösende Bearbeitung von Motion Control-Aufträgen**

Motion Control-Aufträge zum Verfahren der Achse können auch ablösend ausgeführt werden.

Wird während eines laufenden Motion Control-Auftrags ein neuer Motion Control-Auftrag der Achse gestartet, so wird der laufende Auftrag durch den neuen Auftrag abgelöst, ohne dass der laufende Auftrag komplett abgearbeitet wird. Der abgelöste Auftrag meldet dies durch CommandAborted = TRUE an der Motion Control-Anweisung. So kann z. B. ein laufender MC\_MoveRelative-Auftrag durch einen MC\_MoveAbsolute-Auftrag abgelöst werden.
- **Mehrfachverwendung der gleichen Instanz vermeiden**

Alle relevanten Informationen eines Motion Control-Auftrags werden in seiner Instanz gespeichert.

Starten Sie unter Verwendung dieser Instanz keinen neuen Auftrag, solange Sie den Status des aktuellen Auftrags verfolgen möchten. Verwenden Sie verschiedene Instanzen, wenn Sie die Aufträge getrennt verfolgen möchten. Wird die gleiche Instanz für mehrere Motion Control-Aufträge verwendet, so werden die Status- und Fehlerinformationen der einzelnen Aufträge gegenseitig überschrieben.
- **Aufruf von Motion Control-Anweisungen in verschiedenen Prioritätsklassen (Ablaufebenen)**

Motion Control-Anweisungen mit gleicher Instanz dürfen ohne Verriegelung nicht in unterschiedlichen Prioritätsklassen aufgerufen werden. Wie Sie Motion Control-Anweisungen zur Nachverfolgung verriegelt aufrufen können, finden Sie unter "Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 4224)".

## Siehe auch

- Übersicht über die Motion Control-Anweisungen (Seite 4194)
- Anwenderprogramm erstellen (Seite 4195)
- Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart (Seite 4200)
- Verfolgung laufender Aufträge (Seite 4200)
- Fehleranzeigen der Motion Control-Anweisungen (Seite 4213)
- Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 4224)

### 11.2.9.4 Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart

Mit einem NETZ-AUS oder CPU-STOP werden alle aktiven Motion Control-Aufträge abgebrochen. Alle CPU-Ausgänge inklusive der Impuls- und Richtungsausgänge werden zurückgesetzt.

Nach einem darauffolgenden NETZ-EIN oder CPU-Neustart (CPU-RUN) werden die Technologieobjekte und die Motion Control-Aufträge neu initialisiert.

Alle Aktualdaten der Technologieobjekte sowie alle Status- und Fehlerinformationen der zuvor aktiven Motion Control-Aufträge werden auf Initialwerte zurück gesetzt.

Bevor die Achse wieder verwendet werden kann, muss sie erneut mit Hilfe der Motion Control-Anweisung "MC\_Power" freigegeben werden. Ist eine Referenzierung gewünscht, so muss die Achse mit der Motion Control-Anweisung "MC\_Home" ebenfalls erneut referenziert werden.

## Siehe auch

- Übersicht über die Motion Control-Anweisungen (Seite 4194)
- Anwenderprogramm erstellen (Seite 4195)
- Programmierhinweise (Seite 4198)
- Verfolgung laufender Aufträge (Seite 4200)
- Fehleranzeigen der Motion Control-Anweisungen (Seite 4213)

### 11.2.9.5 Verfolgung laufender Aufträge

#### Verfolgung laufender Aufträge

Bei der Verfolgung laufender Motion Control-Aufträge kann zwischen drei typische Gruppen unterschieden werden:

- **Motion Control-Anweisungen mit dem Ausgangsparameter "Done"**
- **Die Motion Control-Anweisung "MC\_MoveVelocity"**
- **Die Motion Control-Anweisung "MC\_MoveJog"**

## **Motion Control-Anweisungen mit Ausgangsparameter Done**

Motion Control-Anweisungen mit dem Ausgangsparameter "Done" werden über den Eingangsparameter "Execute" gestartet und haben einen definierten Abschluss (z. B. bei der Motion Control-Anweisung "MC\_Home": Referenzieren war erfolgreich). Damit ist der Auftrag beendet und die Achse steht still.

Die Aufträge der nachfolgenden Motion Control-Anweisungen haben einen definierten Abschluss:

- MC\_Reset
- MC\_Home
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_CommandTable (ab Technologieobjekt V2.0)
- MC\_ChangeDynamic (ab Technologieobjekt V2.0)

Der Ausgangsparameter "Done" zeigt den Wert TRUE, wenn der Auftrag erfolgreich abgeschlossen wurde.

Die Ausgangsparameter "Busy", "CommandAborted" und "Error" signalisieren, dass der Auftrag noch in Bearbeitung ist, abgebrochen wurde bzw. dass ein Fehler ansteht. Die Motion Control-Anweisungen "MC\_Reset" kann nicht abgebrochen werden und besitzt deshalb keinen Ausgangsparameter "CommandAborted". Da die Motion Control-Anweisung "MC\_ChangeDynamic" sofort abgeschlossen ist, besitzt die Anweisung keine Ausgangsparameter "Busy" und "CommandAborted".

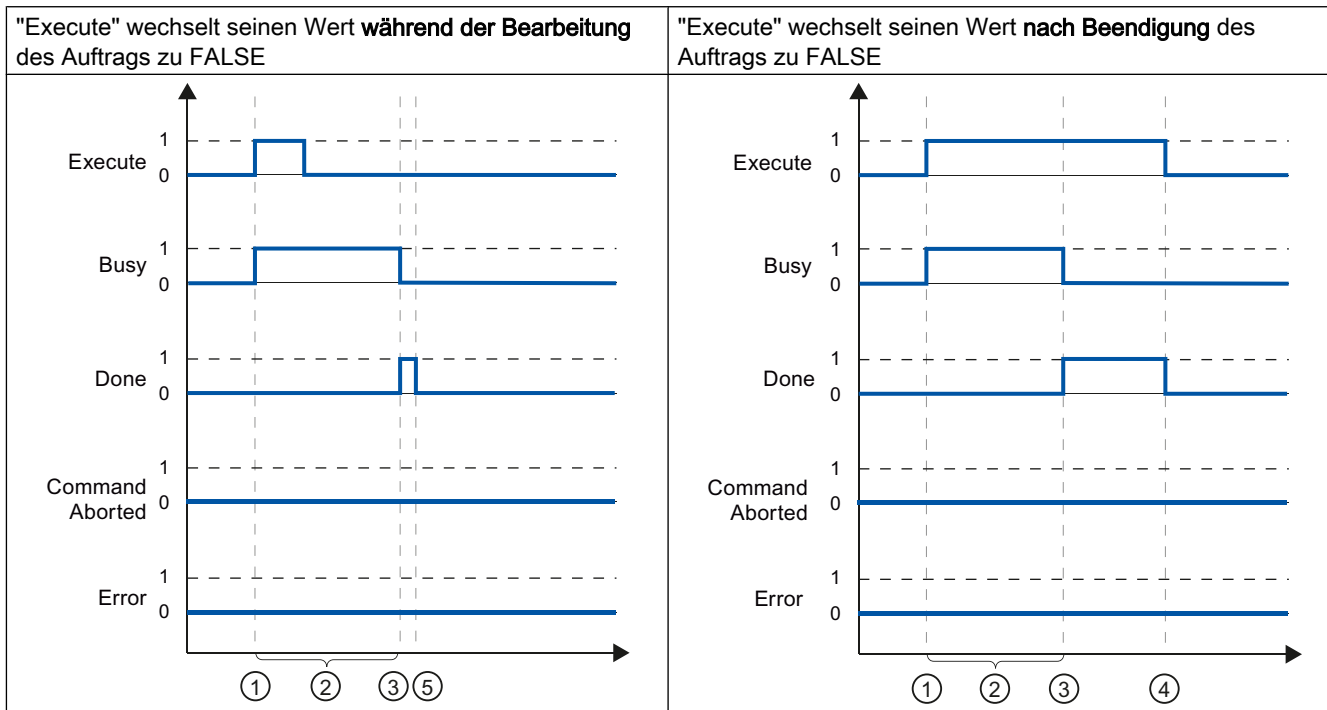
Während der Bearbeitung des Motion Control-Auftrags zeigt der Ausgangsparameter "Busy" den Wert TRUE. Wurde der Auftrag abgeschlossen, abgebrochen oder durch einen Fehler gestoppt, so wechselt der Ausgangsparameter "Busy" seinen Wert zu FALSE. Dies geschieht unabhängig vom Signal am Eingangsparameters "Execute".

Die Ausgangsparameter "Done", "CommandAborted" und "Error" zeigen den Wert TRUE für mindestens einen Zyklus an. Während der Eingangsparameter "Execute" auf TRUE gesetzt ist, werden diese Statusmeldungen speichernd angezeigt.

Nachfolgend wird das Verhalten der Statusbits beispielhaft für unterschiedliche Situationen gezeigt:

### Vollständige Abarbeitung des Auftrags

Wird der Motion Control-Auftrag bis zum Abschluss vollständig abgearbeitet, so wird dies am Ausgangsparameter "Done" mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "Done":



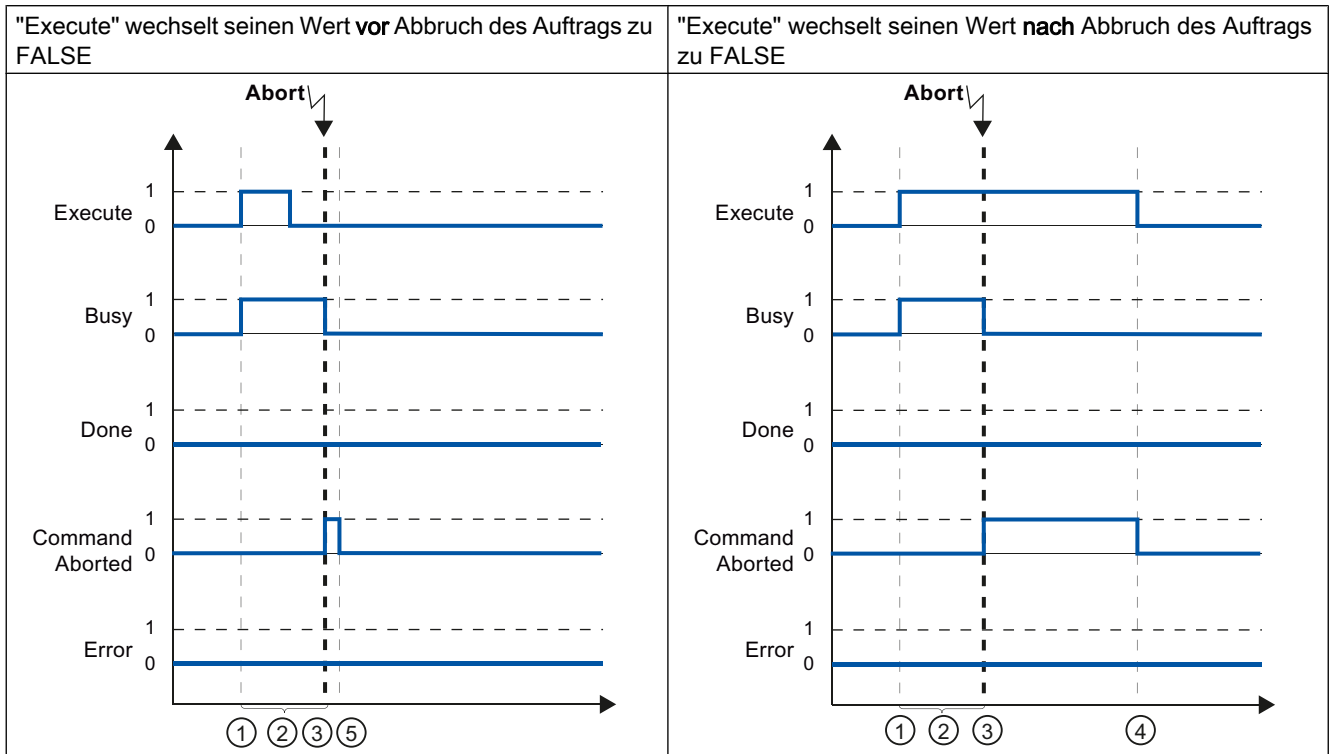
|   |  |
|---|--|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Beendigung des Auftrags beibehalten. |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.   |
| ③ | Mit dem Abschluss des Auftrags (z. B. bei der Motion Control-Anweisung "MC_Home": Referenzieren war erfolgreich) wechselt der Ausgangsparameter "Busy" nach FALSE und "Done" nach TRUE.  |
| ④ | Behält "Execute" den Wert TRUE bis nach Beendigung des Auftrags bei, so bleibt auch "Done" TRUE und wechselt seinen Wert gemeinsam mit "Execute" zu FALSE.   |
| ⑤ | Wurde "Execute" schon vor Beendigung des Auftrags auf FALSE gesetzt, so zeigt "Done" nur für einen Abarbeitungszyklus den Wert TRUE.   |

### Abbruch des Auftrags

Wird der Motion Control-Auftrag während der Bearbeitung abgebrochen, so wird dies am Ausgangsparameter "CommandAborted" mit dem Wert TRUE angezeigt. Der Signalzustand



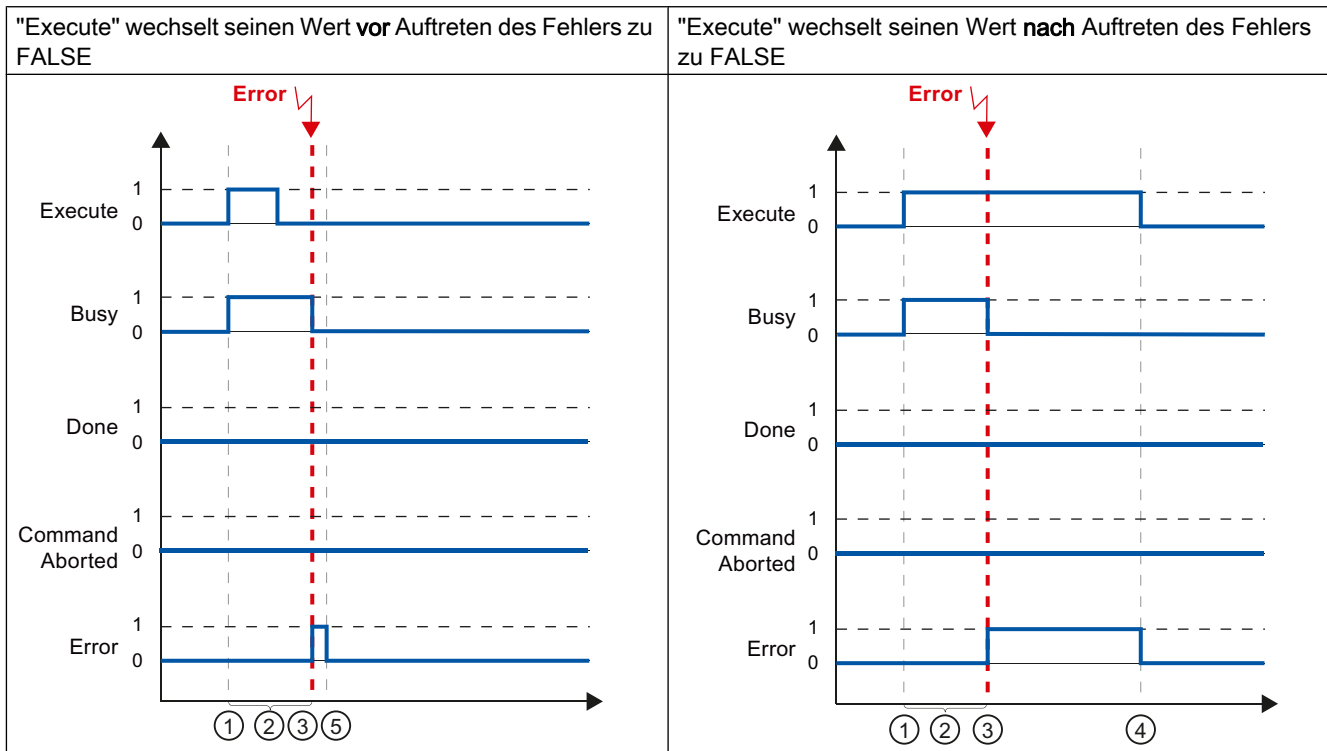
des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "CommandAborted":



|   |  |
|---|--|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Beendigung des Auftrags beibehalten. |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.   |
| ③ | Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Ausgangsparameter "Busy" nach FALSE und "CommandAborted" nach TRUE.   |
| ④ | Behält "Execute" den Wert TRUE bis nach Abbruch des Auftrags bei, so bleibt auch "CommandAborted" TRUE und wechselt seinen Wert gemeinsam mit "Execute" zu FALSE.  |
| ⑤ | Wurde "Execute" schon vor Abbruch des Auftrags auf FALSE gesetzt, so zeigt "CommandAborted" nur für einen Abarbeitungszyklus den Wert TRUE.  |

### Fehler während der Auftragsbearbeitung

Tritt während der Bearbeitung des Motion Control-Auftrags ein Fehler auf, so wird dies am Ausgangsparameter "Error" mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "Error":



|   |   |
|---|---|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Beendigung des Auftrags beibehalten |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.  |
| ③ | Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Ausgangsparameter "Busy" nach FALSE und "Error" nach TRUE.   |
| ④ | Behält "Execute" den Wert TRUE bis nach Auftreten des Fehlers bei, so bleibt "Error" TRUE und wechselt seinen Wert erst gemeinsam mit "Execute" zu FALSE.   |
| ⑤ | Wurde "Execute" schon vor Auftreten des Fehlers auf FALSE gesetzt, so zeigt "Error" nur für einen Abarbeitungszyklus den Wert TRUE.   |

### Motion Control-Anweisung MC\_MoveVelocity

Die Aufträge der Motion Control-Anweisung "MC\_MoveVelocity" haben kein definiertes Ende. Das Auftragsziel ist erfüllt, wenn die parametrisierte Geschwindigkeit erstmalig erreicht wird und die Achse mit konstanter Geschwindigkeit fährt. Das Erreichen der parametrisierten Geschwindigkeit wird am Ausgangsparameter "InVelocity" mit dem Wert TRUE angezeigt.

Der Auftrag ist abgeschlossen, wenn die parametrisierte Geschwindigkeit erreicht und der Eingangsparameter "Execute" auf den Wert FALSE gesetzt wurde. Die Bewegung der Achse ist mit dem Abschluss des Auftrags jedoch noch nicht beendet. Die Bewegung der Achse kann z. B. mit einem Motion Control-Auftrag "MC\_Halt" gestoppt werden.

Die Ausgangsparameter "Busy", "CommandAborted" und "Error" signalisieren, dass der Auftrag noch in Bearbeitung ist, abgebrochen wurde bzw. dass ein Fehler ansteht.

Während der Bearbeitung der Motion Control-Auftrag zeigt der Ausgangsparameter "Busy" den Wert TRUE. Wurde der Auftrag abgeschlossen, abgebrochen oder durch einen Fehler gestoppt, so wechselt der Ausgangsparameter "Busy" seinen Wert zu FALSE. Dies geschieht unabhängig vom Signal am Eingangsparameters "Execute".

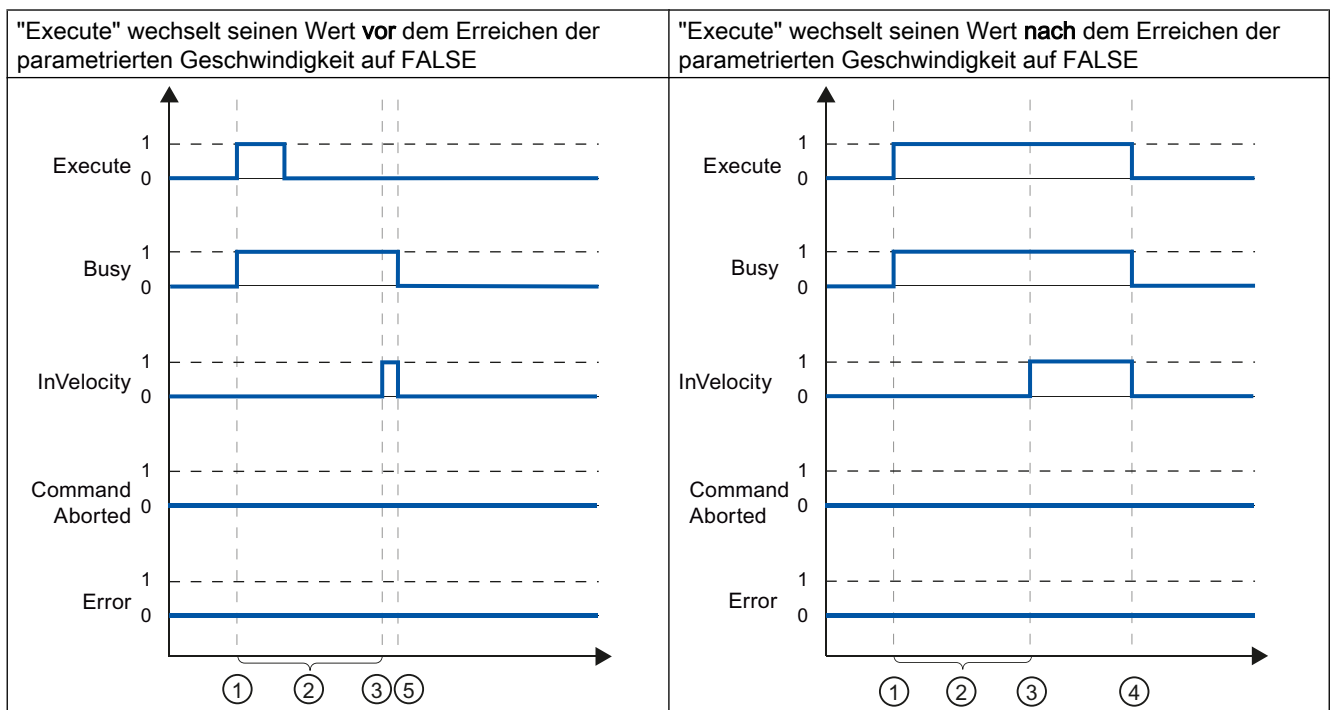
Die Ausgangsparameter "InVelocity", "CommandAborted" und "Error" zeigen bei Erfüllung den Wert TRUE für mindestens einen Zyklus an. Während der Eingangsparameter "Execute" auf TRUE gesetzt ist, werden diese Statusmeldungen speichernd angezeigt.

Nachfolgend wird das Verhalten der Statusbits beispielhaft in unterschiedlichen Situationen gezeigt:

### Die parametrisierte Geschwindigkeit wird erreicht

Wird der Motion Control-Auftrag bis zum Erreichen der parametrisierten Geschwindigkeit ausgeführt, so wird dies am Ausgangsparameter "InVelocity" mit dem Wert TRUE angezeigt.

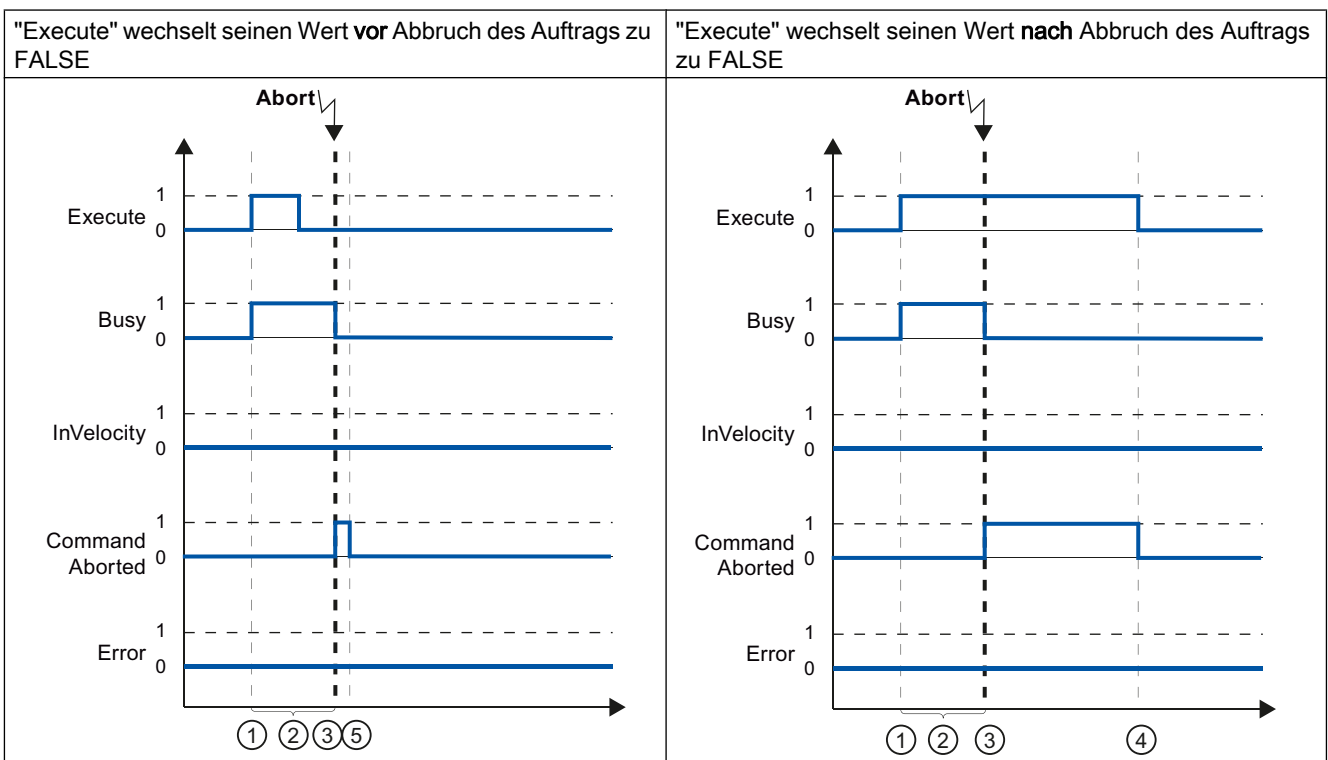
Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "InVelocity":



|   |   |
|---|---|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch vor oder erst nach Erreichen der parametrisierten Geschwindigkeit auf den Wert FALSE zurück gesetzt werden.   |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.  |
| ③ | Mit dem Erreichen der parametrisierten Geschwindigkeit wechselt der Ausgangsparameter "InVelocity" nach TRUE.   |
| ④ | Behält "Execute" den Wert TRUE auch nach Erreichen der parametrisierten Geschwindigkeit bei, so bleibt der Auftrag aktiv. "InVelocity" und "Busy" behalten den Wert TRUE und wechseln ihren Zustand erst gemeinsam mit "Execute" zu FALSE.  |
| ⑤ | Wurde "Execute" schon vor Erreichen der parametrisierten Geschwindigkeit zurück auf FALSE gesetzt, so wird der Auftrag mit dem Erreichen der parametrisierten Geschwindigkeit abgeschlossen. "InVelocity" zeigt für einen Abarbeitungszyklus den Wert TRUE und wechselt zusammen mit "Busy" zu FALSE. |

### Der Auftrag wird vor dem Erreichen der parametrisierten Geschwindigkeit abgebrochen

Wird der Motion Control-Auftrag vor dem Erreichen der parametrisierten Geschwindigkeit abgebrochen, so wird dies durch den Ausgangsparameter "CommandAborted" mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameters "CommandAborted".



|   |  |
|---|--|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Abbruch des Auftrags beibehalten |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.   |

|   |  |
|---|--|
| ③ | Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Ausgangsparameter "Busy" nach FALSE und "CommandAborted" nach TRUE. |
| ④ | Behält "Execute" den Wert TRUE bis nach Abbruch des Auftrags bei, so bleibt auch "CommandAborted" TRUE und wechselt seinen Zustand gemeinsam mit "Execute" zu FALSE.   |
| ⑤ | Wurde "Execute" schon vor Abbruch des Auftrags zurück auf FALSE gesetzt, so zeigt "CommandAborted" nur für einen Abarbeitungszyklus den Wert TRUE.   |

---

### Hinweis

Unter folgenden Bedingungen wird am Ausgangsparameter "CommandAborted" kein Abbruch angezeigt:

Die parametrisierte Geschwindigkeit wurde erreicht, der Eingangsparameter "Execute" hat den Wert FALSE und ein neuer Motion Control-Auftrag wird angestoßen.

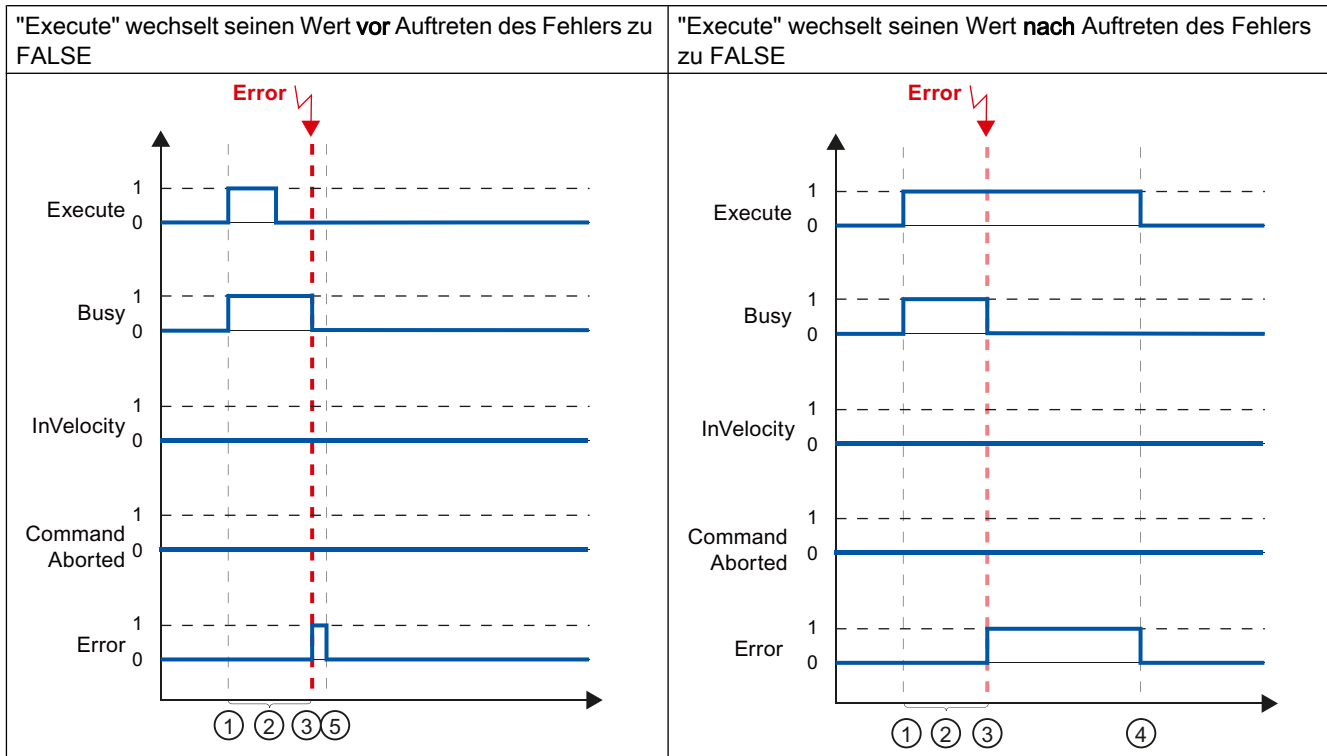
Mit dem Erreichen der parametrisierten Geschwindigkeit und dem Wert FALSE am Eingangsparameter "Execute" ist der Auftrag abgeschlossen. Der Anstoß eines neuen Auftrags wird deshalb nicht als Abbruch angezeigt.

---

### Vor dem Erreichen der parametrisierten Geschwindigkeit tritt ein Fehler auf

Tritt während der Bearbeitung des Motion Control-Auftrags vor dem Erreichen der parametrisierten Geschwindigkeit ein Fehler auf, so wird dies am Ausgangsparameter "Error"

mit dem Wert TRUE angezeigt. Der Signalzustand des Eingangsparameters "Execute" beeinflusst die Anzeigedauer am Ausgangsparameter "Error":



|   |   |
|---|---|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "Execute" gestartet. Je nach Programmierung, kann "Execute" noch während des Auftrags auf den Wert FALSE zurück gesetzt werden oder den Wert TRUE bis nach Auftreten des Fehlers beibehalten |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE..   |
| ③ | Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Ausgangsparameter "Busy" nach FALSE und "Error" nach TRUE.   |
| ④ | Behält "Execute" den Wert TRUE bis nach Auftreten des Fehlers bei, so bleibt auch "Error" TRUE und wechselt seinen Zustand erst gemeinsam mit "Execute" zu FALSE.   |
| ⑤ | Wurde "Execute" schon vor Auftreten des Fehlers zurück auf FALSE gesetzt, so zeigt "Error" lediglich für einen Abarbeitungszyklus den Wert TRUE.  |

### **Hinweis**

Unter folgenden Bedingungen wird am Ausgangsparameter "Error" kein Fehler angezeigt:

Die parametrisierte Geschwindigkeit wurde erreicht, der Eingangsparameter "Execute" hat den Wert FALSE und ein Fehler an der Achse tritt ein (z. B. Software-Endschalter wird angefahren).

Mit dem Erreichen der parametrisierten Geschwindigkeit und dem Wert FALSE am Eingangsparameter "Execute" ist der Auftrag abgeschlossen. Nach Abschluss des Auftrags wird der Fehler der Achse nur an der Motion Control-Anweisung "MC\_Power" angezeigt.

---

## **Motion Control-Anweisung MC\_MoveJog**

Die Aufträge der Motion Control-Anweisung "MC\_MoveJog" realisieren einen Tippbetrieb.

Die Motion Control-Aufträge "MC\_MoveJog" haben kein definiertes Ende. Das Auftragsziel ist erfüllt, wenn die parametrisierte Geschwindigkeit erstmalig erreicht wird und die Achse mit konstanter Geschwindigkeit fährt. Das Erreichen der parametrisierten Geschwindigkeit wird am Ausgangsparameter "InVelocity" mit dem Wert TRUE angezeigt.

Der Auftrag ist abgeschlossen, wenn der Eingangsparameter "JogForward", bzw. "JogBackward" auf den Wert FALSE gesetzt wurde und die Achse den Stillstand erreicht hat.

Die Ausgangsparameter "Busy", "CommandAborted" und "Error" signalisieren, dass der Auftrag noch in Bearbeitung ist, abgebrochen wurde bzw. dass ein Fehler ansteht.

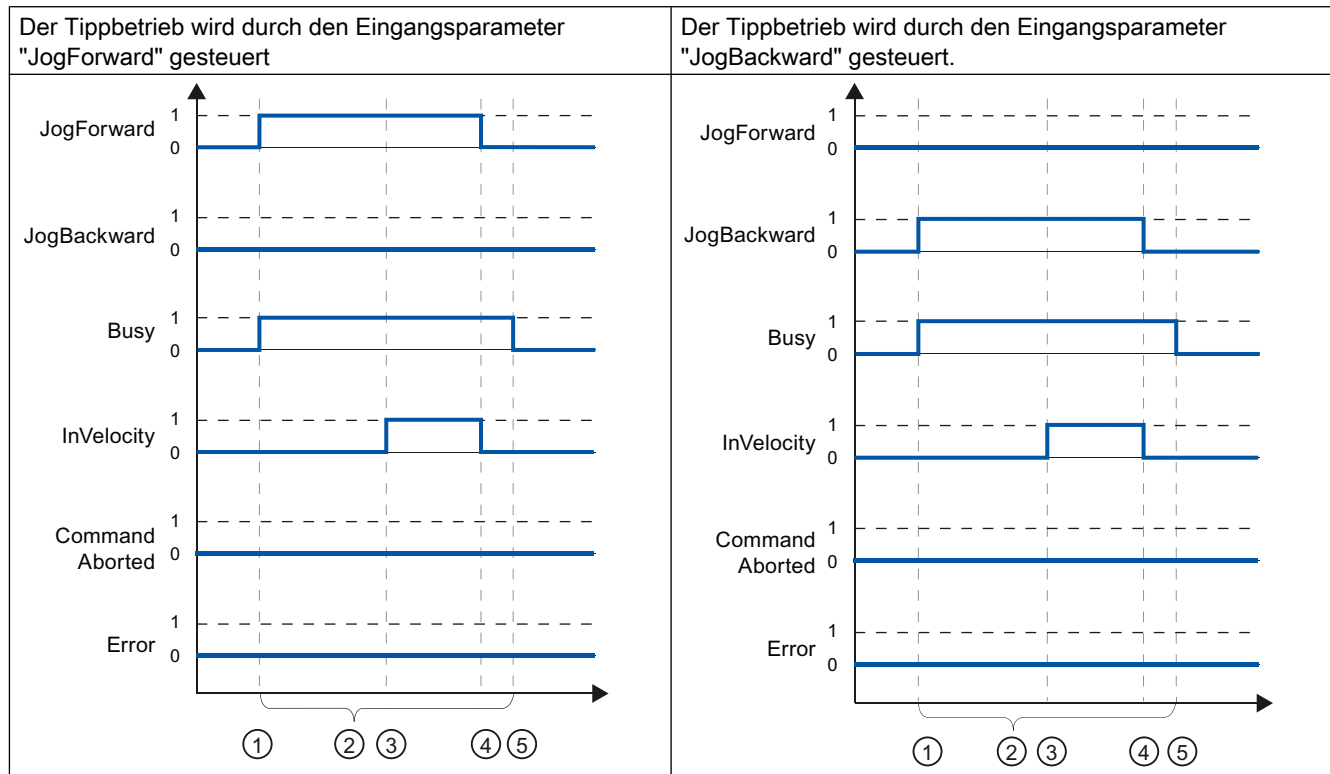
Während der Bearbeitung des Motion Control-Auftrags zeigt der Ausgangsparameter "Busy" den Wert TRUE. Wurde der Auftrag abgeschlossen, abgebrochen oder durch einen Fehler gestoppt, so wechselt der Ausgangsparameter "Busy" seinen Wert zu FALSE.

Der Ausgangsparameter "InVelocity" zeigt den Zustand TRUE, solange die Achse mit der parametrisierten Geschwindigkeit fährt. Die Ausgangsparameter "CommandAborted" und "Error" zeigen den Status für mindestens einen Zyklus an. Während einer der Eingangsparameter "JogForward" oder "JogBackward" auf TRUE gesetzt ist, werden die Statusmeldungen speichernd angezeigt.

Nachfolgend wird das Verhalten der Statusbits beispielhaft in unterschiedlichen Situationen gezeigt:

### Die parametrisierte Geschwindigkeit wird erreicht und gehalten

Wird der Motion Control-Auftrag bis zum Erreichen der parametrisierten Geschwindigkeit ausgeführt, so wird dies am Ausgangsparameter "InVelocity" mit dem Wert TRUE angezeigt.

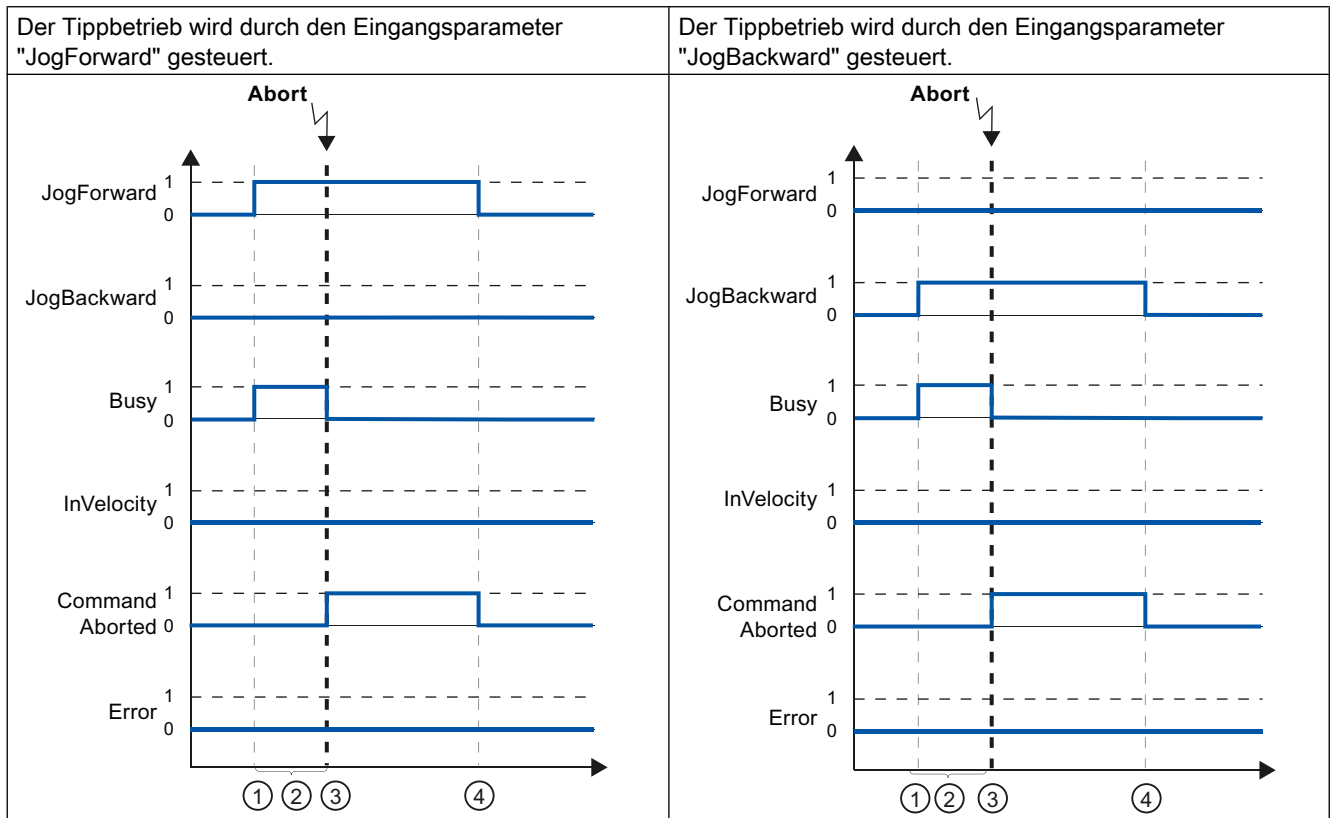


|   |  |
|---|--|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "JogForward", bzw. "JogBackward" gestartet.   |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.   |
| ③ | Mit dem Erreichen der parametrisierten Geschwindigkeit wechselt der Ausgangsparameter "InVelocity" nach TRUE.  |
| ④ | Mit dem Rücksetzen des Eingangsparameters "JogForward", bzw. "JogBackward" auf den Wert FALSE wird die Bewegung der Achse beendet. Die Achse beginnt zu verzögern. Dadurch verfährt die Achse nicht länger mit konstanter Geschwindigkeit und der Ausgangsparameter "InVelocity" wechselt seinen Zustand zu FALSE. |
| ⑤ | Ist die Achse zum Stillstand gekommen, so ist der Motion Control-Auftrag abgeschlossen und der Ausgangsparameter "Busy" wechselt seinen Wert zu FALSE.   |



### Der Auftrag wird während der Bearbeitung abgebrochen

Wird der Motion Control-Auftrag während der Bearbeitung abgebrochen, so wird dies am Ausgangsparameter "CommandAborted" mit dem Wert TRUE angezeigt. Das Verhalten ist unabhängig davon, ob die parametrierte Geschwindigkeit erreicht wurde, oder nicht.



|   |  |
|---|--|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "JogForward", bzw. "JogBackward" gestartet.   |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.   |
| ③ | Während der Auftragsbearbeitung wird der Auftrag durch einen anderen Motion Control-Auftrag abgebrochen. Mit dem Abbruch des Auftrags wechselt der Ausgangsparameter "Busy" nach FALSE und "CommandAborted" nach TRUE. |
| ④ | Mit dem Rücksetzen des Eingangsparameters "JogForward", bzw. "JogBackward" auf den Wert FALSE wechselt der Ausgangsparameter "CommandAborted" seinen Wert zu FALSE.  |

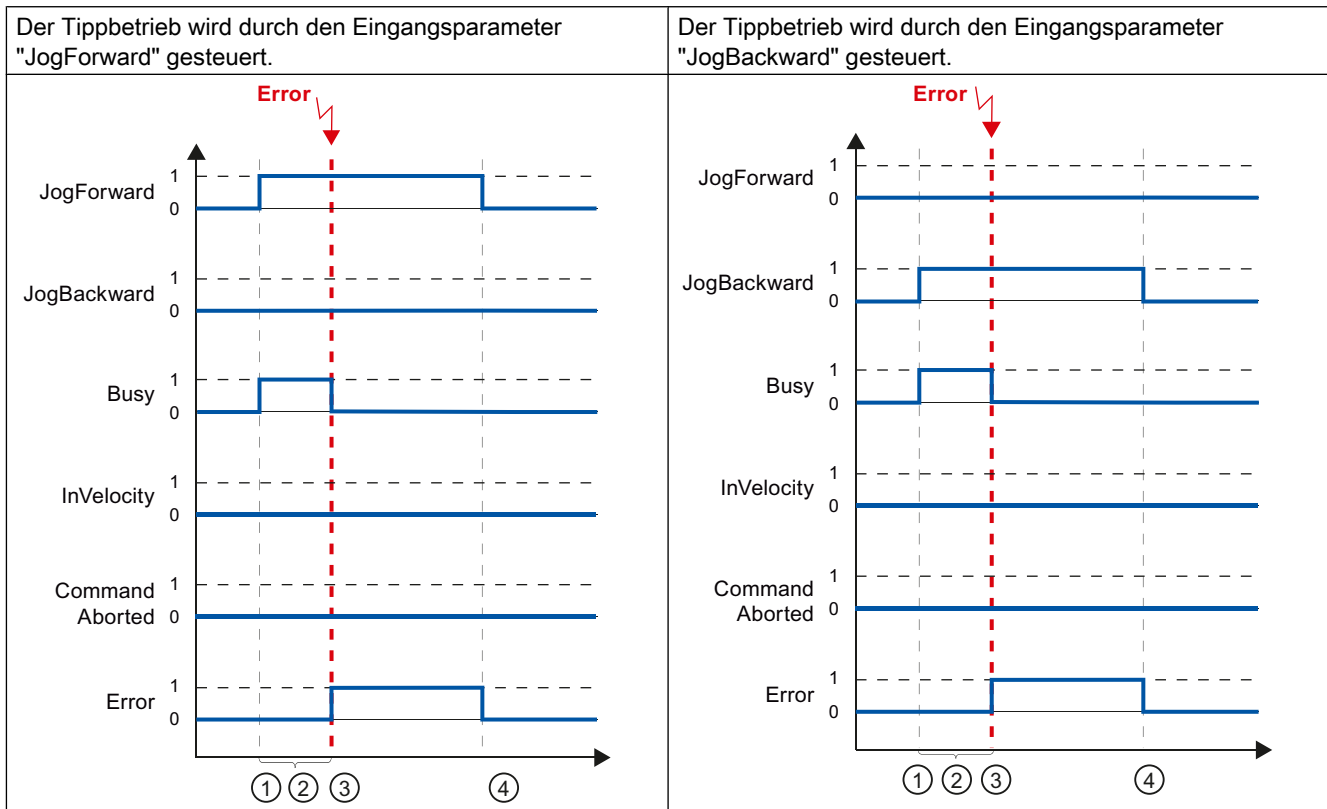
#### Hinweis

Der Auftragsabbruch wird am Ausgangsparameter "CommandAborted" nur für einen Bearbeitungszyklus angezeigt, wenn die nachfolgenden Bedingungen gemeinsam erfüllt sind:

Die Eingangsparameter "JogForward" und "JogBackward" haben den Wert FALSE (die Achse verzögert jedoch noch) und ein neuer Motion Control-Auftrag wird angestoßen.

### Während der Auftragsbearbeitung tritt ein Fehler auf

Tritt bei der Bearbeitung des Motion Control-Auftrags ein Fehler auf, so wird dies am Ausgangsparameter "Error" mit dem Wert TRUE angezeigt. Das Verhalten ist unabhängig davon, ob die parametrisierte Geschwindigkeit erreicht wurde, oder nicht.



|   |   |
|---|---|
| ① | Der Auftrag wird mit einer positiven Flanke am Eingangsparameter "JogForward", bzw. "JogBackward" gestartet.  |
| ② | Während der Auftrag aktiv ist, zeigt der Ausgangsparameter "Busy" den Wert TRUE.  |
| ③ | Während der Auftragsbearbeitung tritt ein Fehler auf. Mit dem Auftreten des Fehlers wechselt der Ausgangsparameter "Busy" nach FALSE und "Error" nach TRUE. |
| ④ | Mit dem Rücksetzen des Eingangsparameters "JogForward", bzw. "JogBackward" auf den Wert FALSE wechselt der Ausgangsparameter "Error" seinen Wert zu FALSE.  |

---

**Hinweis**

Ein aufgetretener Fehler wird am Ausgangsparameter "Error" nur für einen Bearbeitungszyklus angezeigt, wenn die nachfolgenden Bedingungen gemeinsam erfüllt sind:

Die Eingangsparameter "JogForward" und "JogBackward" haben den Wert FALSE (die Achse verzögert jedoch noch) und ein Fehler tritt ein (z. B. Software-Endschalter wird angefahren).

---

### 11.2.9.6 Fehleranzeigen der Motion Control-Anweisungen

Die Motion Control-Anweisungen zeigen etwaige Fehler der Motion Control-Aufträge und des Technologieobjekts an den Ausgangsparametern "Error", "ErrorID" und "ErrorInfo" der Motion Control-Anweisungen an.

#### Fehleranzeige an den Ausgangs-Parametern "Error", "ErrorID" und "ErrorInfo"

Zeigt der Ausgangsparameter "Error" den Wert TRUE, so konnte der Auftrag nicht oder nicht vollständig ausgeführt werden. Die Fehlerursache können Sie dem Wert am Ausgangsparameter "ErrorID" entnehmen. Eine Detailinformation zur Fehlerursache können Sie dem Wert am Ausgangs-Parameter "ErrorInfo" entnehmen. Bei der Fehleranzeige wird zwischen folgenden Fehlerklassen unterschieden:

- **Betriebsfehler mit Stopp der Achse (z. B. "HW-Endschalter wurde angefahren")**  
Betriebsfehler mit Stopp der Achse sind Fehler die zur Laufzeit des Anwenderprogramms auftreten. Befindet sich die Achse in Bewegung, so wird sie je nach Fehler mit der konfigurierten oder der Notstopp-Verzögerung gestoppt. Die Fehler werden an der fehlerauslösenden Motion Control-Anweisung und an der Motion Control-Anweisung "MC\_Power" angezeigt.
- **Betriebsfehler ohne Stopp der Achse (z. B. "Achse ist nicht referenziert")**  
Betriebsfehler ohne Stopp der Achse sind Fehler die zur Laufzeit des Anwenderprogramms auftreten. Befindet sich die Achse in Bewegung, so wird die Bewegung fortgesetzt. Die Fehler werden nur an der fehlerauslösenden Motion Control-Anweisung angezeigt.
- **Parametrierfehler der Motion Control-Anweisung (z. B. "Falscher Wert am Parameter "Velocity"")**  
Zu Parametrierfehlern kommt es bei fehlerhaften Angaben an den Eingangsparametern der Motion Control-Anweisungen. Befindet sich die Achse in Bewegung, so wird die Bewegung fortgesetzt. Die Fehler werden nur an der fehlerauslösenden Motion Control-Anweisung angezeigt.

- **Konfigurationsfehler am Technologieobjekt "Achse" (z. B. "Wert für "Beschleunigung" ist ungültig")**  
Ein Konfigurationsfehler liegt vor, wenn in der Konfiguration der Achse ein oder mehrere Parameter fehlerhaft konfiguriert oder änderbare Konfigurationsdaten während der Laufzeit des Programms fehlerhaft geändert wurden. Ist die Achse in Bewegung, so wird sie mit der konfigurierten Notstopp-Verzögerung gestoppt. Der Fehler wird an der fehlerauslösenden Motion Control-Anweisung und an der Motion Control-Anweisung "MC\_Power" angezeigt.
- **Konfigurationsfehler am Technologieobjekt "Auftragstabelle" (z. B. "Wert für "Geschwindigkeit" ist ungültig")**  
Ein Konfigurationsfehler liegt vor, wenn in der Konfiguration der Auftragstabelle ein oder mehrere Parameter fehlerhaft konfiguriert oder änderbare Konfigurationsdaten während der Laufzeit des Programms fehlerhaft geändert wurden. Befindet sich die Achse in Bewegung, so wird diese fortgesetzt. Die Fehler werden nur an der Motion Control-Anweisung "MC\_CommandTable" angezeigt.
- **Interner Fehler**  
Beim Auftreten eines internen Fehlers wird die Achse gestoppt. Die Fehler werden an der fehlerauslösenden Motion Control-Anweisung und teilweise an der Motion Control-Anweisung "MC\_Power" angezeigt.

Eine detaillierte Beschreibung der ErrorIDs und ErrorInfos als auch deren Abhilfen finden Sie im Anhang.

### Siehe auch

Übersicht über die Motion Control-Anweisungen (Seite 4194)

Anwenderprogramm erstellen (Seite 4195)

Programmierhinweise (Seite 4198)

Verhalten der Motion Control-Aufträge nach NETZ-AUS und Neustart (Seite 4200)

Verfolgung laufender Aufträge (Seite 4200)

## 11.2.10 Achse - Diagnose

### 11.2.10.1 Status- und Fehlerbits

Mit der Diagnosefunktion "Status- und Fehlerbits" überwachen Sie im TIA-Portal die wichtigsten Status- und Fehlermeldungen der Achse. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb bei aktiver Achse in den Betriebsarten "Handsteuerung" und "Automatikbetrieb" zur Verfügung. Die angezeigten Status- Fehlermeldungen haben folgende Bedeutung:

## Status der Achse

| Status                | Beschreibung  |
|-----------------------|---|
| Freigegeben           | Die Achse ist freigegeben und bereit über Motion Control-Aufträge gesteuert zu werden.<br>(Variable des Technologieobjekts: <Achsname>.StatusBits.Enable)   |
| Referenziert          | Die Achse ist referenziert und kann absolute Positionieraufträge der Motion Control-Anweisung "MC_MoveAbsolute" ausführen. Zum relativen Positionieren muss die Achse nicht referenziert sein.<br>Sonderfälle: <ul style="list-style-type: none"> <li>• Während des aktiven Referenzierens ist der Status FALSE.</li> <li>• Wird eine referenzierte Achse passiv referenziert, so ist der Status während des passiven Referenzierens TRUE.</li> </ul> (Variable des Technologieobjekts: <Achsname>.StatusBits.HomingDone) |
| Achsfehler            | Am Technologieobjekt "Achse" ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie im Automatikbetrieb den Parametern ErrorID und ErrorInfo der Motion Control-Anweisungen. Im Handbetrieb wird die detaillierte Fehlerursache im Feld "Fehlermeldung" der Achssteuertafel angezeigt.<br>(Variable des Technologieobjekts: <Achsname>.StatusBits.Error)   |
| Achssteuertafel aktiv | Die Betriebsart "Handsteuerung" wurde in der Achssteuertafel aktiviert. Die Achssteuertafel hat die Steuerhoheit über das Technologieobjekt "Achse". Die Achse kann nicht vom Anwenderprogramm gesteuert werden.<br>(Variable des Technologieobjekts: <Achsname>.StatusBits.ControlPanelActive)   |
| Restart erforderlich  | Eine geänderte Konfiguration der Achse wurde im Betriebszustand RUN der CPU in den Ladespeicher geladen. Um die geänderte Konfiguration in den Arbeitsspeicher zu laden, ist ein Restart der Achse notwendig. Verwenden Sie hierzu die Motion Control-Anweisung MC_Reset.   |

## Status des Antriebs

| Status         | Beschreibung   |
|----------------|--|
| Antrieb bereit | Der Antrieb ist betriebsbereit.<br>(Variable des Technologieobjekts: <Achsname>.StatusBits.DriveReady)   |
| Antriebsfehler | Der Antrieb hat durch Ausfall des "Antrieb bereit"-Signals einen Fehler angezeigt.<br>(Variable des Technologieobjekts: <Achsname>.ErrorBits.DriveFault) |

## Status der Achsbewegung

| Status                    | Beschreibung  |
|---------------------------|---|
| Stillstand                | Die Achse befindet sich im Stillstand.<br>(Variable des Technologieobjekts: <Achsname>.StatusBits.StandStill)                   |
| Beschleunigung            | Die Achse beschleunigt.<br>(Variable des Technologieobjekts: <Achsname>.StatusBits.Acceleration)                                |
| Konstante Geschwindigkeit | Die Achse verfährt mit konstanter Geschwindigkeit.<br>(Variable des Technologieobjekts: <Achsname>.StatusBits.ConstantVelocity) |
| Verzögerung               | Die Achse verzögert (bremst ab).<br>(Variable des Technologieobjekts: <Achsname>.StatusBits.Deceleration)                       |

### Status der Bewegungsart

| Status   | Beschreibung   |
|--|--|
| Positionieren  | Die Achse führt einen Positionierauftrag der Motion Control-Anweisung "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel aus.<br>(Variable des Technologieobjekts: <Achname>.StatusBits.PositioningCommand)       |
| Mit Geschwindigkeitsvorgabe fahren                         | Die Achse führt einen Auftrag mit Geschwindigkeitsvorgabe der Motion Control-Anweisung "MC_MoveVelocity", "MC_MoveJog" oder der Achssteuertafel aus.<br>(Variable des Technologieobjekts: <Achname>.StatusBits.SpeedCommand) |
| Referenzieren  | Die Achse führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home" oder der Achssteuertafel aus.<br>(Variable des Technologieobjekts: <Achname>.StatusBits.Homing)  |
| Auftragstabelle aktiv<br>(ab Technologieobjekt Achse V2.0) | Die Achse wird mit der Motion Control-Anweisung "MC_CommandTable" gesteuert.<br>(Variable des Technologieobjekts: <Achname>.StatusBits.CommandTableActive)   |

### Fehlermeldungen

| Fehler                                     | Beschreibung  |
|--|---|
| Unterer SW-Endschalter wurde erreicht      | Der untere Software-Endschalter wurde erreicht.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.SwLimitMinReached)   |
| Unterer SW-Endschalter wurde überschritten | Der untere Software-Endschalter wurde überschritten.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.SwLimitMinExceeded)   |
| Oberer SW-Endschalter wurde erreicht       | Der obere Software-Endschalter wurde erreicht.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.SwLimitMaxReached)  |
| Oberer SW-Endschalter wurde überschritten  | Der obere Software-Endschalter wurde überschritten.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.SwLimitMaxExceeded)  |
| Unterer HW-Endschalter wurde angefahren    | Der untere Hardware-Endschalters wurde angefahren.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.HwLimitMin)   |
| Oberer HW-Endschalter wurde angefahren     | Der obere Hardware-Endschalters wurde angefahren.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.HwLimitMax)  |
| PTO und HSC bereits in Verwendung          | Eine zweite Achse verwendet den selben PTO (Pulse Train Output) und HSC (High Speed Counter) und ist mit "MC_Power" freigegeben.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.HwUsed)   |
| Konfigurationsfehler                       | Das Technologieobjekt "Achse" wurde fehlerhaft konfiguriert oder änderbare Konfigurationsdaten wurden während der Laufzeit des Anwenderprogramms fehlerhaft geändert.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.ConfigFault) |
| Interner Fehler                            | Es ist ein interner Fehler aufgetreten.<br>(Variable des Technologieobjekts: <Achname>.ErrorBits.SystemFault)   |

**Siehe auch**

- Bewegungsstatus (Seite 4217)
- Variable StatusBits. (Seite 4252)
- Variable ErrorBits. (Seite 4255)

**11.2.10.2 Bewegungsstatus**

Mit der Diagnosefunktion "Bewegungsstatus" überwachen Sie im TIA-Portal den Bewegungsstatus der Achse. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb bei aktiver Achse in den Betriebsarten "Handsteuerung" und "Automatikbetrieb" zur Verfügung. Die angezeigten Statusinformationen haben folgende Bedeutung:

| <b>Status</b>            | <b>Beschreibung</b>   |
|--------------------------|---|
| Position                 | Das Feld "Position" zeigt die aktuelle Position der Achse an. Bei nicht referenzierter Achse zeigt der Wert den Positionswert relativ zur Freigabe-Position der Achse an.<br>(Variable des Technologieobjekts: <Achsname>.MotionStatus.Position)  |
| Geschwindigkeit          | Das Feld "Geschwindigkeit" zeigt die aktuelle Geschwindigkeit der Achse an.<br>(Variable des Technologieobjekts: <Achsname>.MotionStatus.Velocity)  |
| Zielposition             | Das Feld "Zielposition" zeigt die aktuelle Zielposition eines aktiven Positionierauftrags oder der Achssteuertafel an. Der Wert der "Zielposition" ist nur während der Ausführung eines Positionierauftrags gültig.<br>(Variable des Technologieobjekts: <Achsname>.MotionStatus.TargetPosition)                          |
| Verbleibender Verfahrweg | Das Feld "Verbleibender Verfahrweg" zeigt den aktuell verbleibenden Verfahrweg eines aktiven Positionierauftrags oder der Achssteuertafel an. Der Wert "Verbleibender Verfahrweg" ist nur während der Ausführung eines Positionierauftrags gültig.<br>(Variable des Technologieobjekts: <Achsname>.MotionStatus.Distance) |

**Siehe auch**

- Status- und Fehlerbits (Seite 4214)
- Variable MotionStatus. (Seite 4251)

**11.2.10.3 Dynamikeinstellungen**

Mit der Diagnosefunktion "Dynamikeinstellungen" kontrollieren Sie im TIA-Portal die konfigurierten Dynamikgrenzwerte der Achse. Die Anzeige der Diagnosefunktion steht im Onlinebetrieb bei aktiver Achse in den Betriebsarten "Handsteuerung" und "Automatikbetrieb" zur Verfügung. Die angezeigten Statusinformationen haben folgende Bedeutung:

| <b>Dynamische Grenze</b> | <b>Beschreibung</b>   |
|--------------------------|---|
| Beschleunigung           | Das Feld "Beschleunigung" zeigt die aktuell konfigurierte Beschleunigung der Achse an.<br>(Variable des Technologieobjekts: <Achsname>.Config.DynamicDefaults.Acceleration) |
| Verzögerung              | Das Feld "Verzögerung" zeigt die aktuell konfigurierte Verzögerung der Achse an.<br>(Variable des Technologieobjekts: <Achsname>.Config.DynamicDefaults.Deceleration)       |

| Dynamische Grenze                            | Beschreibung  |
|--|---|
| Notstopp-Verzögerung                         | Das Feld "Notstopp-Verzögerung" zeigt die aktuell konfigurierte Notstopp-Verzögerung der Achse an.<br>(Variable des Technologieobjekts:<br><Achsenname>.Config.DynamicDefaults.EmergencyDeceleration) |
| Ruck<br>(ab Technologieobjekt<br>Achse V2.0) | Das Feld zeigt den aktuell konfigurierten Ruck der Achse an.<br>(Variable des Technologieobjekts: <Achsenname>.Config.DynamicDefaults.Jerk)   |

### 11.2.11 Arbeiten mit Beobachtungstabellen

Verwenden Sie Beobachtungstabellen, wenn Sie während der Inbetriebnahme Variablen der Motion Control-Anweisungen oder des Technologieobjekts "Achse" beobachten und steuern möchten.

Für das Beobachten und Steuern von Variablen muss in einer Beobachtungstabelle der vollständige Name der Variable inklusiv Objektname und aller Strukturnamen angegeben werden.

Beispiel: <Achsenname>.Config.DynamicDefaults.Acceleration

#### Tipp:

Durch Copy & Paste können Sie die Eingabe des langen Variablennamen umgehen.

#### Vorgehen

Gehen Sie zum Einfügen der Variablennamen wie nachfolgend beschrieben vor:

1. Markieren Sie in der Projektnavigation den Instanz-Datenbaustein oder das Technologieobjekt der Achse.
2. **Parameter der Motion Control-Anweisung**
  - Wählen Sie durch Rechtsklicken den Kontextmenübefehl **Öffnen**.

**Variablen des Technologieobjekts**

  - Wählen Sie durch Rechtsklicken den Kontextmenübefehl **Öffnen im Editor**.
3. **Parameter der Motion Control-Anweisung**
  - Markieren Sie im Bereich Input, bzw. Output die Zeilen der Variablen

**Variablen des Technologieobjekts**

  - Öffnen Sie im Bereich Static die entsprechenden Strukturen und markieren Sie die Zeilen der Variablen
4. Wählen Sie den Menübefehl **Bearbeiten > Kopieren**.
5. Öffnen Sie durch Doppelklicken die Beobachtungstabelle.
6. Markieren Sie die Zeile, ab der Sie die Variablen einfügen möchten
7. Wählen Sie den Menübefehl **Bearbeiten > Einfügen**.



Die Variablen wird mit ihrem vollständigen Namen in die Beobachtungstabelle eingefügt.



**WARNUNG**

Über die Beobachtungstabelle haben Sie auch schreibenden Zugriff auf Variablen, deren Verwendung im Anwenderprogramm aus Sicherheitsgründen gesperrt ist. Ein Steuern dieser Variablen kann zur Beschädigung der aktuellen Achskonfiguration und zu undefinierten Reaktionen der Achse führen. Steuern Sie nur Variablen, deren Zugriff in der Variablenliste des Technologieobjekts mit "RW" gekennzeichnet sind.

**Siehe auch**

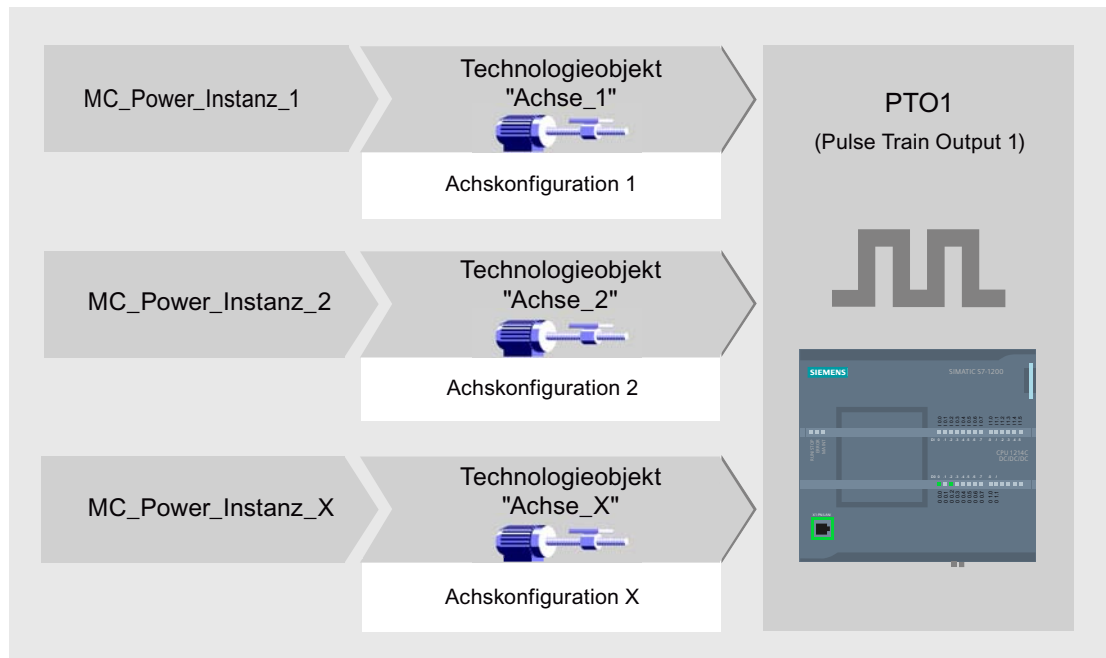
Inbetriebnahme der Achse - Achssteuertafel (Seite 4191)

## 11.2.12 Anhang

### 11.2.12.1 Mehrere Achsen mit gleichem PTO einsetzen

Nutzen Sie die Motion Control-Funktionalität der CPU S7-1200, um mehrere Technologieobjekte "Achse" mit dem gleichen PTO (Pulse Train Output) und somit mit den gleichen CPU-Ausgängen zu betreiben. Dies ist z. B. sinnvoll, wenn über einen PTO verschiedene Achskonfigurationen für unterschiedliche Produktionsabläufe genutzt werden sollen. Zwischen diesen Achskonfigurationen kann, wie nachfolgend beschrieben, beliebig

gewechselt werden. Entnehmen Sie die prinzipiellen funktionalen Zusammenhänge der nachfolgenden Darstellung:

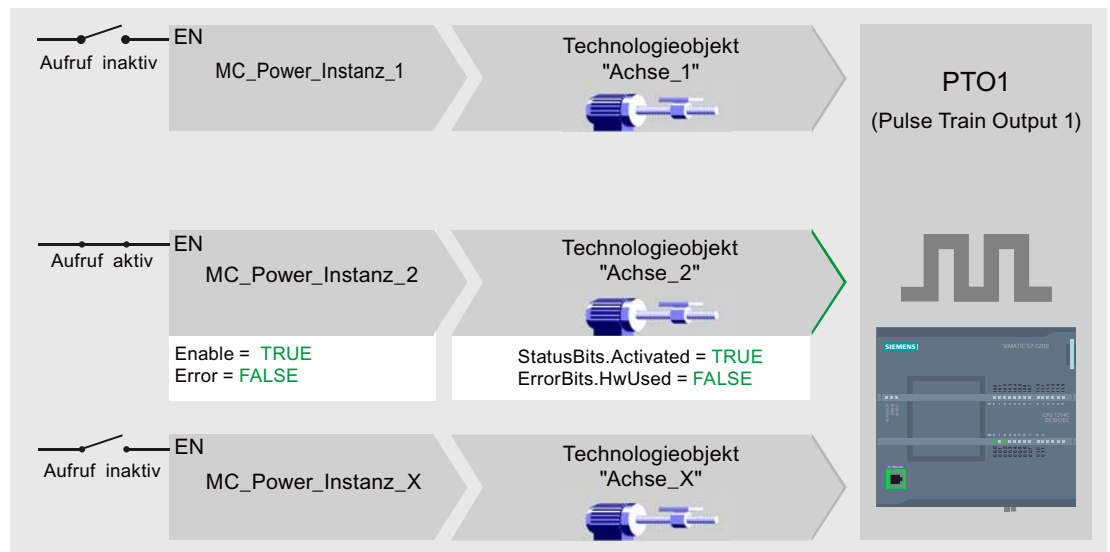


Im dargestellten Beispiel verwenden mehrere Technologieobjekte "Achse" mit jeweils eigener Achskonfiguration den gleichen PTO. Im Anwenderprogramm muss jede "Achse" mit einem eigenen Aufruf der Motion Control-Anweisung "MC\_Power" mit eigenem Instanz-Datenbaustein aufgerufen werden. Zu jedem Zeitpunkt darf nur jeweils eine "Achse" den PTO verwenden. Die Achse, die augenblicklich den PTO verwendet, zeigt dies mit der Variablen `<Achse>.StatusBits.Activated = TRUE` an.

### Wechsel des Technologieobjekts "Achse"

Wie Sie zwischen verschiedenen Technologieobjekten und somit zwischen verschiedenen Achskonfigurationen wechseln, können Sie dem nachfolgend beschriebenen Programmschema entnehmen. Um mit mehreren Achsen ohne Fehleranzeigen den selben PTO verwenden zu können, dürfen nur die Motion Control-Anweisungen der aktuell zu verwendenden Achse aufgerufen werden.

Die folgende Darstellung zeigt dies am Beispiel der Motion Control-Anweisung "MC\_Power":



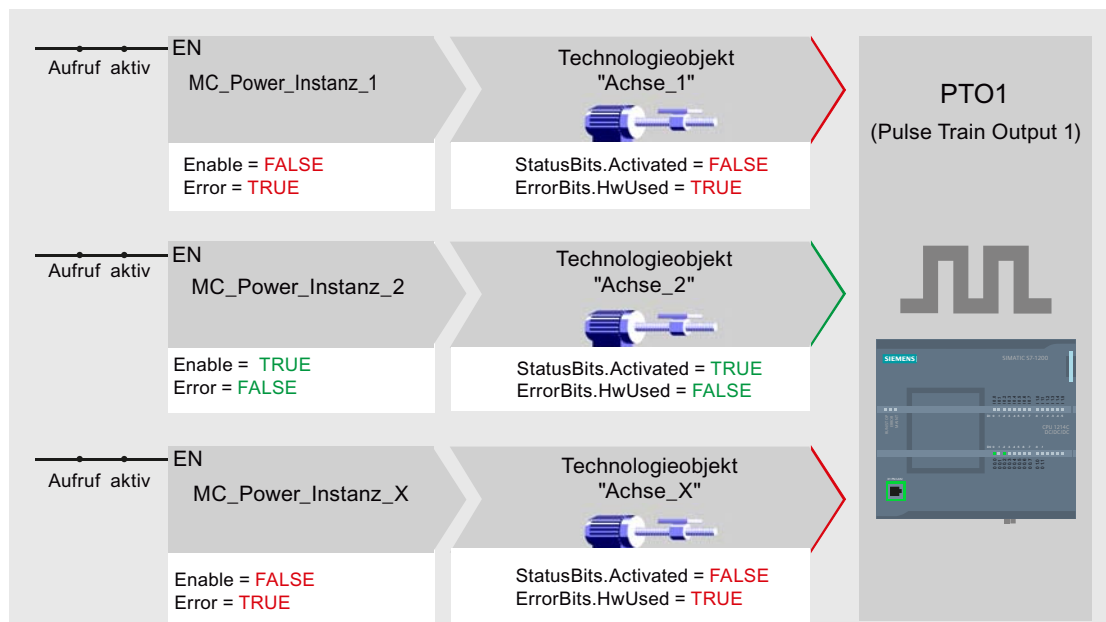
Die Variable der aktivierten Achse (hier "Achse\_2") zeigen im Anwenderprogramm folgende typischen Anzeigen:

- <Achse>.StatusBits.Activated = TRUE
- <Achse>.ErrorBits.HwUsed = FALSE

Gehen Sie wie nachfolgend beschrieben vor, um das Technologieobjekt "Achse" zu wechseln. Im Beispiel soll von "Achse\_2" zu "Achse\_1" gewechselt werden:

1. Beenden Sie evt. laufende Verfahrbewegungen der aktivierten "Achse\_2"
2. Sperren Sie "Achse\_2" mit der dazugehörigen Motion Control-Anweisung "MC\_Power" über den Eingangsparameter Enable = FALSE
3. Prüfen Sie das erfolgte Sperren der "Achse\_2" mit einer UND-Verknüpfung des Ausgangsparameters Status = FALSE der Motion Control-Anweisung "MC\_Power" und der Variable des Technologieobjekts <Achse>.StatusBits.Enable = FALSE.
4. Deaktivieren Sie den bedingten Aufruf der Motion Control-Anweisungen für "Achse\_2"
5. Aktivieren Sie den bedingten Aufruf der Motion Control-Anweisungen für "Achse\_1". Mit dem ersten Aufruf der entsprechenden Motion Control-Anweisung "MC\_Power" wird "Achse\_2" deaktiviert und "Achse\_1" aktiviert.
6. Geben Sie "Achse\_1" mit der Motion Control-Anweisung "MC\_Power" über den Eingangsparameter Enable = TRUE frei.
7. Prüfen Sie die erfolgte Freigabe der "Achse\_1" mit einer UND-Verknüpfung des Ausgangsparameters Status = TRUE der Motion Control-Anweisung "MC\_Power" und der Variable des Technologieobjekts <Achse>.StatusBits.Enable = TRUE.

Grundsätzlich ist auch der zyklische Aufruf aller Motion Control-Anweisungen aller mit einem PTO arbeitenden Achsen möglich:



Mit dem Freigeben einer Achse (hier "Achse\_2") wird diese aktiviert.

Im Gegensatz zum bedingten Aufruf zeigen die Motion Control-Anweisungen der deaktivierten Achsen (hier "Achse\_1" und "Achse\_x") Fehler an. Die Variablen dieser Achsen zeigen den Zustand `<Achse>.StatusBits.Activated = FALSE` und `<Achse>.ErrorBits.HwUsed = TRUE`.

Setzen Sie den bedingten Aufruf der Motion Control-Anweisungen ein, wenn Sie das Anwenderprogramm ohne Fehleranzeigen realisieren möchten.

## Siehe auch

Mehrere Antriebe mit gleichem PTO einsetzen (Seite 4223)

Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 4224)

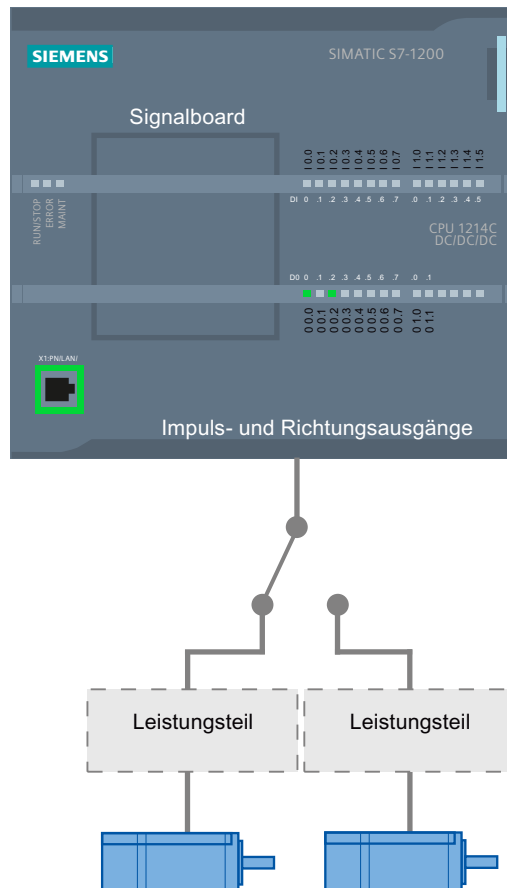
Sonderfälle beim Einsatz von Software-Endschaltern (Seite 4226)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)

Variable des Technologieobjekts Achse (Seite 4242)

### 11.2.12.2 Mehrere Antriebe mit gleichem PTO einsetzen

Sollen mehrere Antriebe alternativ genutzt werden, so können diese über eine Umschaltung an einem gemeinsamen PTO (Pulse Train Output) betrieben werden. Die nachfolgende Darstellung zeigt den prinzipiellen Schaltungsaufbau:



Die Umschaltung der Antriebe kann bei Bedarf durch das Anwenderprogramm über einen digitalen Ausgang gesteuert werden. Sind für die verschiedenen Antriebe unterschiedliche Achskonfigurationen nötig, so müssen diese für den PTO entsprechend umgeschaltet werden. Weitere Informationen hierzu finden Sie unter "Mehrere Achsen mit gleichem PTO einsetzen (Seite 4219)".

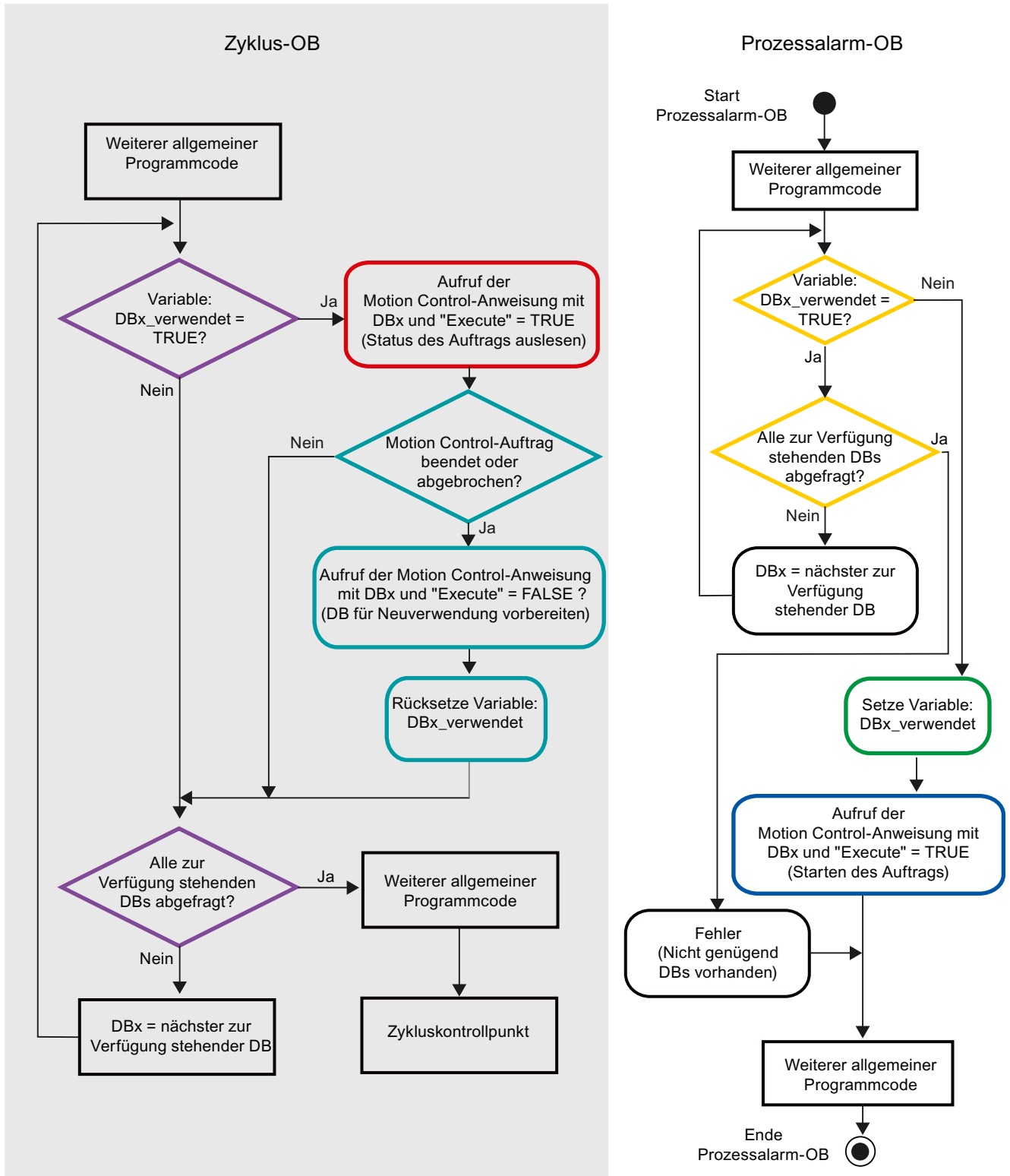
#### Siehe auch

- Mehrere Achsen mit gleichem PTO einsetzen (Seite 4219)
- Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 4224)
- Sonderfälle beim Einsatz von Software-Endschaltern (Seite 4226)
- Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)
- Variable des Technologieobjekts Achse (Seite 4242)

### 11.2.12.3 Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen

Je nach Anwendung kann es nötig sein, Motion Control-Aufträge (z. B. alarmgesteuert) in einer höheren Prioritätsklasse (Ablaufebene) zu starten.

Die Motion Control-Anweisungen müssen zur Statusverfolgung in kurzen Zeitabständen aufgerufen werden. Werden die Motion Control-Anweisungen in der höheren Prioritätsklasse nur einmalig oder in zu großen Zeitabständen aufgerufen, so können die Motion Control-Aufträge nicht ausreichend nachverfolgt werden. In diesem Fall besteht die Möglichkeit der Nachverfolgung im Zyklus-OB. Für jeden Start eines Motion Control-Auftrags in der höheren Prioritätsklasse muss ein momentan nicht verwendeter Instanz-Datenbaustein vorhanden sein. Entnehmen Sie dem nachfolgenden Flussdiagramm, wie Sie Motion Control-Aufträge in einer höheren Prioritätsklasse (z. B. Prozessalarm-OB) starten und im Zyklus-OB weiterverfolgen können:



Je nach Häufigkeit der zu startenden Motion Control-Aufträge muss eine ausreichende Anzahl von Instanz-Datenbausteinen erzeugt worden sein. Welcher Instanz-Datenbaustein

augenblicklich in Verwendung ist, muss anwenderseitig in den Variablen DBx\_verwendet verwaltet werden.

### Start des Motion Control-Auftrags im Prozessalarm-OB

Über binäre Abfragen der Variablen DBx\_verwendet (Orange) wird ein Instanz-Datenbaustein gesucht, der augenblicklich nicht in Verwendung ist. Wird ein solcher gefunden, so wird der verwendete Instanzdatenbaustein als "verwendet" gekennzeichnet (Grün) und der Motion Control-Auftrag mit diesem Instanz-Datenbaustein gestartet (Blau).

Anschließend werden evt. weitere Programmteile des Prozessalarm-OBs bearbeitet und danach in den Zyklus-OB zurückgekehrt.

### Verfolgung der gestarteten Motion-Control-Aufträge im Zyklus-OB

Im Zyklus-OB werden alle zur Verfügung stehenden Instanz-Datenbausteine auf ihre Verwendung über die Variable DBx\_verwendet geprüft (Violett).

Ist ein Instanz-Datenbaustein in Verwendung (Motion Control-Auftrag ist in Bearbeitung), so wird die Motion Control-Anweisung mit diesem Instanz-Datenbaustein und dem Eingangsparameter Execute = TRUE aufgerufen, um die Statusmeldungen auszulesen (Rot).

Ist der Auftrag abgeschlossen oder wurde er abgebrochen, so wird mit folgenden Maßnahmen fortgesetzt (Blaugrün):

- Aufruf der Motion Control-Anweisung mit dem Eingangsparameter Execute = FALSE
- Rücksetzen der Variable DBx\_verwendet

Hiermit ist die Auftragsverfolgung abgeschlossen und der Instanz-Datenbaustein steht für eine erneute Verwendung wieder zur Verfügung.

### Siehe auch

Mehrere Achsen mit gleichem PTO einsetzen (Seite 4219)

Mehrere Antriebe mit gleichem PTO einsetzen (Seite 4223)

Sonderfälle beim Einsatz von Software-Endschaltern (Seite 4226)

Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0) (Seite 4233)

Variable des Technologieobjekts Achse (Seite 4242)

## 11.2.12.4 Sonderfälle beim Einsatz von Software-Endschaltern

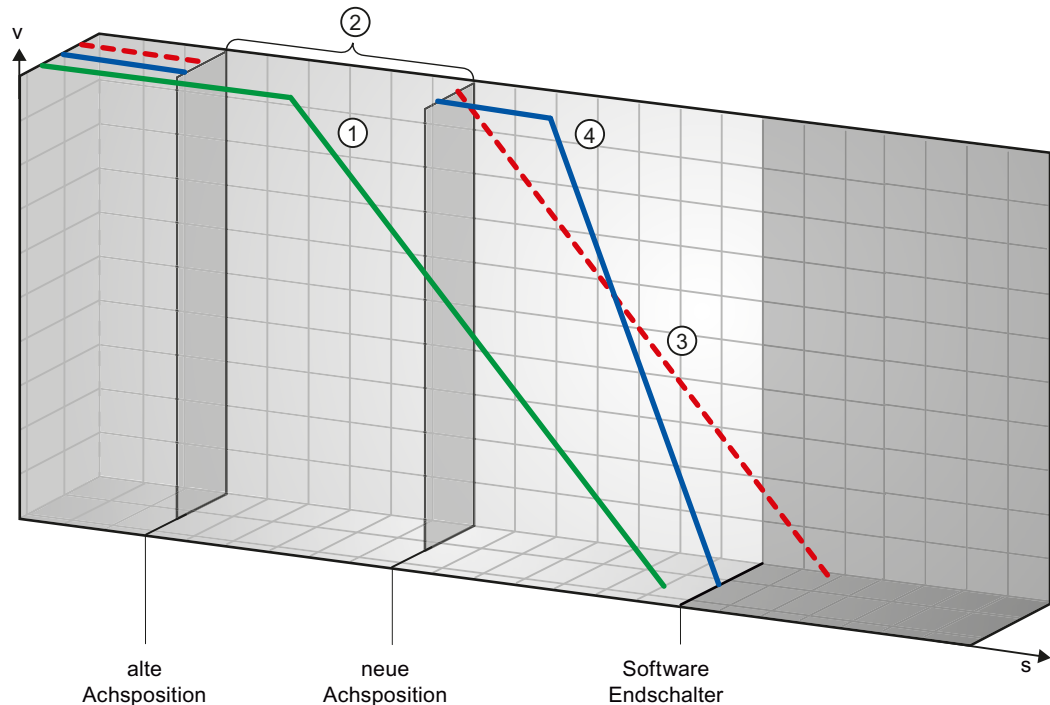
### Software-Endschalter im Zusammenhang mit einem Referenziervorgang

Durch ungünstig parametrisierte Referenzieraufträge kann der Bremsvorgang der Achse auf den Software-Endschalter beeinflusst werden. Berücksichtigen Sie die nachfolgenden Beispiele bei Ihrer Programmierung.



**Beispiel 1:**

Während eines Verfahrbefehls wird mit einem Referenzierauftrag (z. B. Referenzpunkt setzen) die aktuelle Achsposition in Richtung des Software-Endschalters verschoben. Es ist noch möglich die Achse bis zum Erreichen des Software-Endschalters zum Stillstand zu bringen:

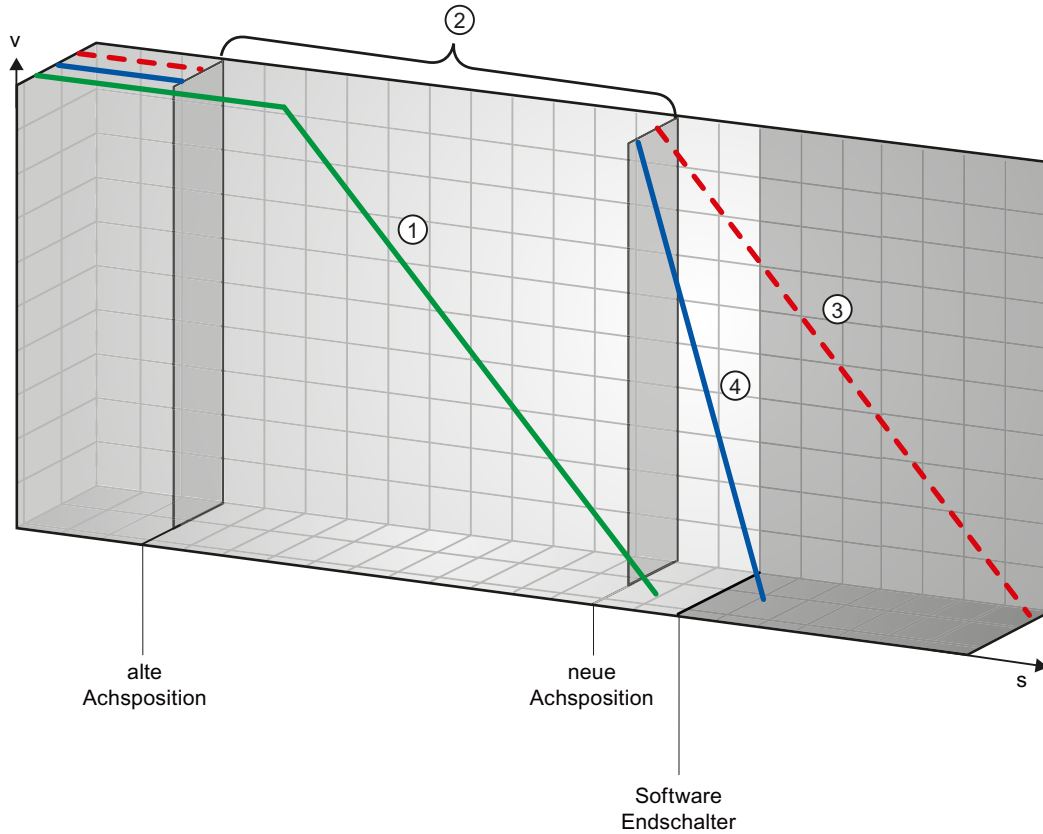


|   |  |
|---|--|
| ① | Der grüne Kurvenverlauf zeigt die Bewegung <b>ohne</b> Referenzierauftrag. Die Achse brems mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalter zum Stehen.   |
| ② | Durch den Referenzierauftrag wird eine neue Achsposition gesetzt. Der Bereich zwischen alter und neuer Achsposition wird somit "übersprungen".   |
| ③ | Bedingt durch die neue Achsposition würde die Achse mit der konfigurierten Verzögerung theoretisch hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).  |
| ④ | Da ein Abbremsen mit der konfigurierten Verzögerung nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse brems nach konstanter Fahrt mit der Notstopp-Verzögerung ab und kommt auf der Position des Software-Endschalters zum Stehen. |

**Beispiel 2:**

Während eines Verfahrbefehls wird mit einem Referenzierauftrag (z. B. Referenzpunkt setzen) die aktuelle Achsposition in Richtung des Software-Endschalters verschoben. Im Gegensatz

zu Beispiel 1 ist es nicht mehr möglich die Achse bis zum Erreichen des Software-Endschalters zum Stillstand zu bringen. Die Achse fährt über die Position des Software-Endschalters hinaus.

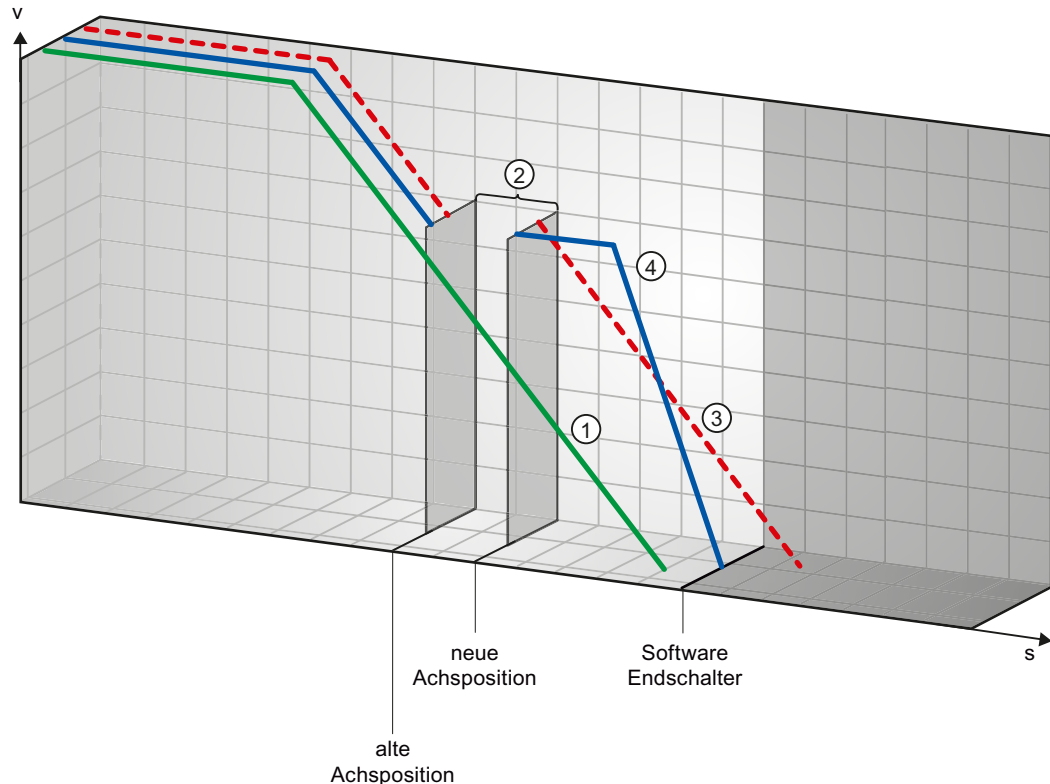


|   |  |
|---|--|
| ① | Der grüne Kurvenverlauf zeigt die Bewegung <b>ohne</b> Referenzierauftrag. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalters zum Stehen.   |
| ② | Durch den Referenzierauftrag wird eine neue Achsposition gesetzt. Der Bereich zwischen alter und neuer Achsposition wird somit "übersprungen".   |
| ③ | Bedingt durch die neue Achsposition würde die Achse mit der konfigurierten Verzögerung theoretisch weit hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).   |
| ④ | Da ein Abbremsen mit der konfigurierten Verzögerung nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst mit der Notstopp-Verzögerung ab. Jedoch reicht die Notstopp-Verzögerung nicht aus, um die Achse auf der Position des Software-Endschalters zum Stehen zu bringen. Die Position des Software-Endschalters wird überfahren. |

**Beispiel 3:**

Während eines Abbremsvorgangs wird mit einem Referenzierauftrag (z. B. Referenzpunkt setzen) die aktuelle Achsposition in Richtung des Software-Endschalters verschoben. Es ist

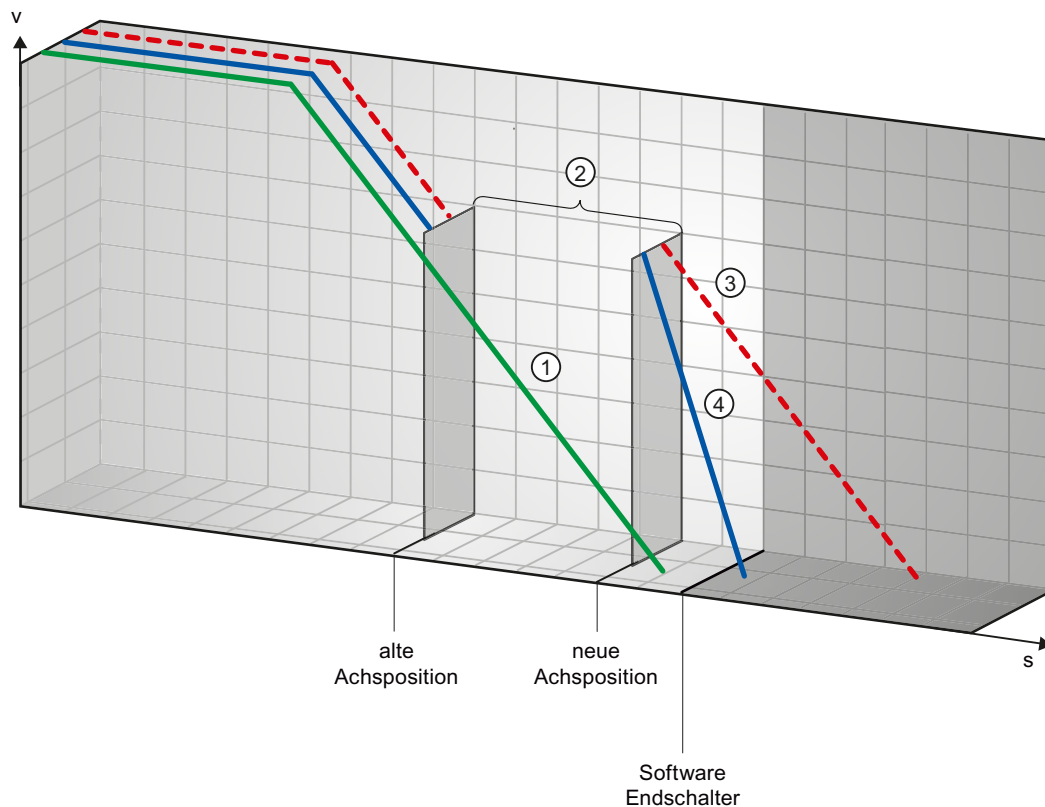
noch möglich die Achse bis zum Erreichen des Software-Endschalters zum Stillstand zu bringen:



|   |   |
|---|---|
| ① | Der grüne Kurvenverlauf zeigt die Bewegung <b>ohne</b> Referenzierauftrag. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalters zum Stehen.  |
| ② | Durch den Referenzierauftrag wird eine neue Achsposition gesetzt. Der Bereich zwischen alter und neuer Achsposition wird somit "übersprungen".  |
| ③ | Bedingt durch die neue Achsposition würde die Achse mit der konfigurierten Verzögerung theoretisch hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).   |
| ④ | Da ein Abbremsen mit der konfigurierten Verzögerung nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst nach konstanter Fahrt mit der Notstopp-Verzögerung ab und kommt auf der Position des Software-Endschalters zum Stehen. |

#### Beispiel 4:

Während eines Abbremsvorgangs wird mit einem Referenzierauftrag (z. B. Referenzpunkt setzen) die aktuelle Achsposition in Richtung des Software-Endschalters verschoben. Im Gegensatz zu Beispiel 3 ist es nicht mehr möglich die Achse bis zum Erreichen des Software-Endschalters zum Stillstand zu bringen. Die Achse fährt über die Position des Software-Endschalters hinaus.



|   |   |
|---|---|
| ① | Der grüne Kurvenverlauf zeigt die Bewegung <b>ohne</b> Referenzierauftrag. Die Achse brems mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalters zum Stehen.   |
| ② | Durch den Referenzierauftrag wird eine neue Achsposition gesetzt. Der Bereich zwischen alter und neuer Achsposition wird somit "übersprungen".  |
| ③ | Bedingt durch die neue Achsposition würde die Achse mit der konfigurierten Verzögerung theoretisch weit hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).  |
| ④ | Da ein Abbremsen mit der konfigurierten Verzögerung nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse brems mit der Notstopp-Verzögerung ab. Jedoch reicht die Notstopp-Verzögerung nicht aus, um die Achse auf der Position des Software-Endschalters zum Stehen zu bringen. Die Position des Software-Endschalters wird überfahren. |

**Siehe auch**

Software-Endschalter im Zusammenhang mit Positionsänderungen der Software-Endschalter (Seite 4231)

Software-Endschalter im Zusammenhang mit Dynamikänderungen (Seite 4231)

Verhalten der Achse beim Ansprechen der Positionsüberwachungen (Seite 4153)

## Software-Endschalter im Zusammenhang mit Positionsänderungen der Software-Endschalter

Durch eine ungünstige Änderung der Position des Software-Endschalters zur Laufzeit des Anwenderprogramms kann der Abstand zwischen aktueller Achsposition und der Position des Software-Endschalters abrupt verkürzt werden.

Die Reaktion der Achse erfolgt analog zur im Kapitel Software-Endschalter im Zusammenhang mit einem Referenziervorgang (Seite 4226) beschriebenen Reaktion.

### Siehe auch

Software-Endschalter im Zusammenhang mit einem Referenziervorgang (Seite 4226)

Software-Endschalter im Zusammenhang mit Dynamikänderungen (Seite 4231)

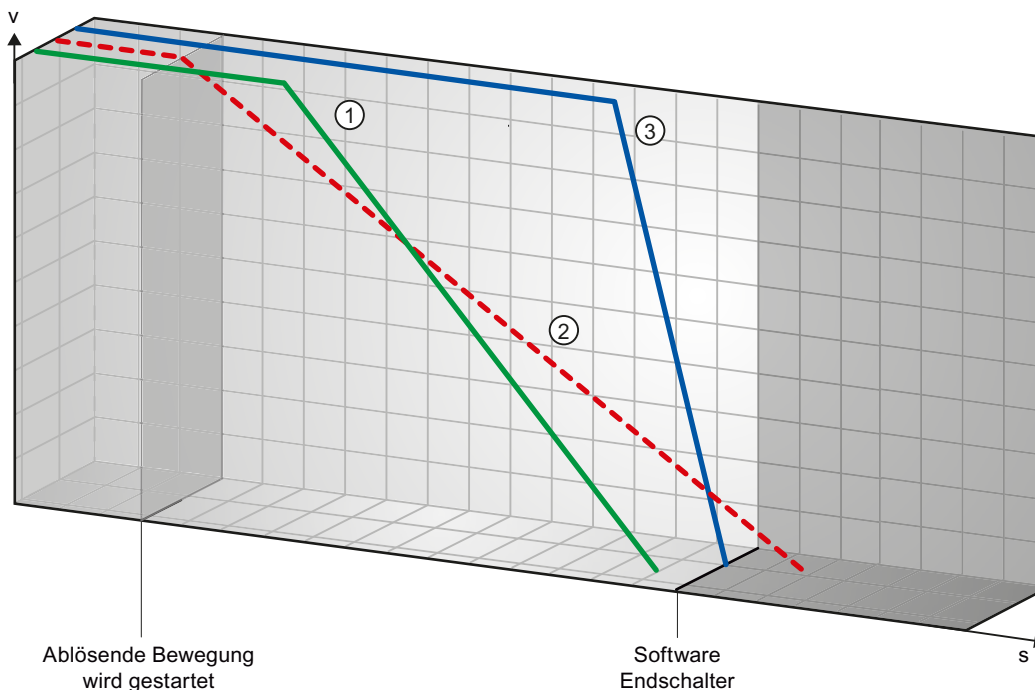
Verhalten der Achse beim Ansprechen der Positionsüberwachungen (Seite 4153)

## Software-Endschalter im Zusammenhang mit Dynamikänderungen

In Verbindung mit ablösenden Fahraufträgen kann der Verzögerungsvorgang der Achse im Bereich der Software-Endschalter beeinflusst werden. Dies gilt, wenn der ablösende Fahrauftrag mit einer geringeren Verzögerung (Variable `<Achse>.Config.DynamicDefaults.Deceleration`) gestartet wird. Berücksichtigen Sie die nachfolgenden Beispiele bei Ihrer Programmierung.

### Beispiel 1:

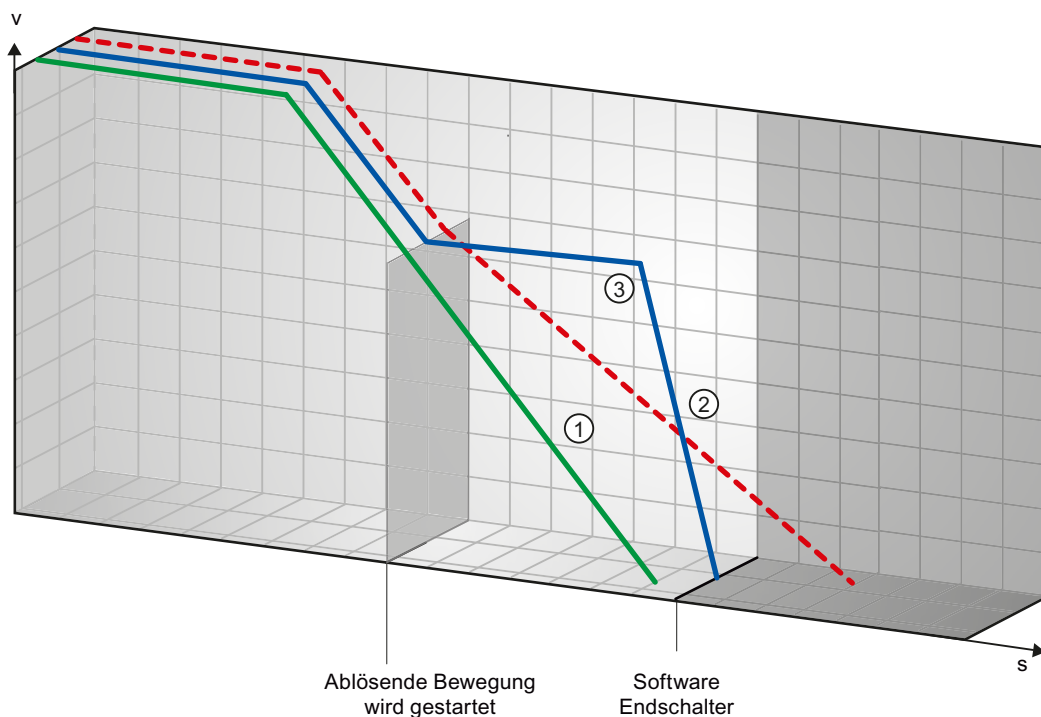
Während des Verfahrens der Achse wird ein laufender Verfahrtauftrag durch einen weiteren Verfahrtauftrag mit einer geringeren Verzögerung abgelöst:



|   |  |
|---|--|
| ① | Der grüne Kurvenverlauf zeigt die Bewegung eines laufenden Auftrags <b>ohne</b> dass dieser abgelöst wird. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalters zum Stehen.   |
| ② | Bedingt durch den ablösenden Bewegungsauftrag mit geringerer Verzögerung würde die Achse theoretisch hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).  |
| ③ | Da ein Abbremsen mit der konfigurierten Verzögerung des ablösenden Bewegungsauftrags nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst nach konstanter Fahrt mit der Notstopp-Verzögerung ab und kommt auf der Position des Software-Endschalters zum Stehen. |

**Beispiel 2:**

Während des Abbremsens der Achse wird ein laufender Verfahrtauftrag durch einen weiteren Verfahrtauftrag mit einer geringeren Verzögerung abgelöst:



|   |  |
|---|--|
| ① | Der grüne Kurvenverlauf zeigt die Bewegung eines laufenden Auftrags <b>ohne</b> dass dieser abgelöst wird. Die Achse bremst mit der konfigurierten Verzögerung ab und kommt vor der Position des Software-Endschalters zum Stehen.   |
| ② | Bedingt durch den ablösenden Bewegungsauftrag mit geringerer Verzögerung würde die Achse theoretisch weit hinter der Position des Software-Endschalters zum Stehen kommen (roter Kurvenverlauf).   |
| ③ | Da ein Abbremsen mit der konfigurierten Verzögerung des ablösenden Bewegungsauftrags nicht mehr ausreicht, folgt die Achse dem tatsächlichen blauen Kurvenverlauf. Die Achse bremst nach konstanter Fahrt mit der Notstopp-Verzögerung ab und kommt auf der Position des Software-Endschalters zum Stehen. |

## Siehe auch

Software-Endschalter im Zusammenhang mit einem Referenziervorgang (Seite 4226)

Software-Endschalter im Zusammenhang mit Positionsänderungen der Software-Endschalter (Seite 4231)

Verhalten der Achse beim Ansprechen der Positionsüberwachungen (Seite 4153)

### 11.2.12.5 Reduzierung der Geschwindigkeit bei kurzer Positionierdauer

Bei einer geplanten Positionierdauer < 2 ms kann die Geschwindigkeit des Positionierauftrags durch die CPU reduziert werden.

Der Auftrag wird über den gesamten Verlauf mit einer reduzierten Geschwindigkeit ausgeführt. Die reduzierte Geschwindigkeit (Impulse/s) errechnet sich aus der nachfolgenden Formel:

- Reduzierte Geschwindigkeit = Anzahl auszugebender Impulse \* 500Hz

Bei einer geplanten Positionierdauer >= 2 ms erfolgt **keine** Reduzierung der Geschwindigkeit.

### 11.2.12.6 Dynamische Anpassung der Start-/Stopp-Geschwindigkeit

Aufgrund der Konfiguration Ihrer Geschwindigkeitsgrenzen (Start/Stopp-Geschwindigkeit, Maximale Geschwindigkeit), der Dynamikwerte (Beschleunigung, Verzögerung, Ruck), sowie der Zielgeschwindigkeit des Verfahrtauftrags kann die Start/Stopp-Geschwindigkeit unter bestimmten Umständen von der CPU dynamisch angepasst werden.

Dies ist z. B. der Fall wenn aufgrund einer konfigurierten geringen Start/Stopp-Geschwindigkeit die benötigte Zeit für die ersten Impulse länger wäre, als für die gesamte Beschleunigung benötigt werden darf. In diesen Fällen wird der erste Impuls mit einer größeren Geschwindigkeit als der konfigurierten Start/Stopp-Geschwindigkeit ausgegeben. Die nachfolgenden Impulse werden ebenfalls dynamisch angepasst, so dass der Beschleunigungsvorgang in der vorgegebenen Zeit beendet werden kann.

Achten Sie bei eventuellen Impulsverlust darauf, dass Ihre verwendete Hardware (Antrieb) auf diese Situation abgestimmt ist, bzw. verändern Sie die Dynamikeinstellungen Ihrer Achse um die dynamische Anpassung der Start/Stopp-Geschwindigkeit zu vermeiden.

### 11.2.12.7 Liste der ErrorIDs und ErrorInfos (Technologieobjekte ab V2.0)

In den folgenden Tabellen finden Sie eine Auflistung aller ErrorIDs und ErrorInfos, die an den Motion Control-Anweisungen angezeigt werden können. Neben der Fehlerursache werden auch Abhilfen zur Beseitigung der Fehler aufgelistet:

**Betriebsfehler mit Stopp der Achse**

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe  |
|----------------|-----------|---|--|
| <b>16#8000</b> |           | <b>Antriebsfehler, "Antrieb bereit" ausgefallen</b>   |  |
|                | 16#0001   | -   | Fehler mit der Anweisung "MC_Reset" quittieren; Antriebssignal bereitstellen; Auftrag evt. erneut starten  |
| <b>16#8001</b> |           | <b>Unterer SW-Endschalter wurde ausgelöst</b>   |  |
|                | 16#000E   | Die Position des unteren SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht  | Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in positiver Richtung aus dem SW-Endschalterbereich herausfahren   |
|                | 16#000F   | Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht  |  |
|                | 16#0010   | Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten   |  |
| <b>16#8002</b> |           | <b>Oberer SW-Endschalter wurde ausgelöst</b>  |  |
|                | 16#000E   | Die Position des oberen SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht   | Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in negativer Richtung aus dem SW-Endschalterbereich herausfahren   |
|                | 16#000F   | Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht   |  |
|                | 16#0010   | Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten  |  |
| <b>16#8003</b> |           | <b>Unterer HW-Endschalter wurde angefahren</b>  |  |
|                | 16#000E   | Der untere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten.<br>(Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden) | Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrtauftrag in positiver Richtung aus dem HW-Endschalterbereich herausfahren.  |
| <b>16#8004</b> |           | <b>Oberer HW-Endschalter wurde angefahren</b>   |  |
|                | 16#000E   | Der obere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten.<br>(Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden)  | Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrtauftrag in negativer Richtung aus dem HW-Endschalterbereich herausfahren.  |
| <b>16#8005</b> |           | <b>PTO / HSC werden bereits von einer anderen Achse verwendet</b>   |  |
|                | 16#0001   | -   | <p><b>Die Achse wurde fehlerhaft konfiguriert:</b><br/>Konfiguration des PTO (Pulse Train Output) / HSC (High Speed Counter) korrigieren und in Steuerung laden</p> <p><b>Mehrere Achsen sollen mit einem PTO arbeiten:</b><br/>Eine andere Achse verwendet den PTO / HSC. Soll die aktuelle Achse die Steuerung übernehmen, so muss die andere Achse mit "MC_Power" Enable = FALSE gesperrt werden.<br/>(siehe auch Mehrere Achsen mit gleichem PTO einsetzen (Seite 4219))</p> |



| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe   |
|----------------|-----------|---|---|
| <b>16#8006</b> |           | <b>Es ist ein Kommunikationsfehler in der Achssteuertafel eingetreten</b> |   |
|                | 16#0012   | Es ist ein Zeitüberlauf eingetreten                                       | Prüfen Sie die Kabelverbindung; betätigen Sie erneut die Schaltfläche "Handsteuerung" |
| <b>16#8007</b> |           | <b>Freigabe der Achse nicht möglich</b>                                   |   |
|                | 16#0025   | Restart wird ausgeführt   | Warten Sie bis der Restart der Achse abgeschlossen ist.                               |
|                | 16#0026   | Ladevorgang im Betriebszustand RUN wird ausgeführt                        | Warten Sie bis der Ladevorgang abgeschlossen ist                                      |

### Betriebsfehler ohne Stopp der Achse

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe  |
|----------------|-----------|--|--|
| <b>16#8200</b> |           | <b>Achse ist nicht freigegeben</b>   |  |
|                | 16#0001   | -  | Achse freigegeben; Auftrag erneut starten  |
| <b>16#8201</b> |           | <b>Achse ist bereits durch eine andere "MC_Power"-Instanz freigegeben</b>  |  |
|                | 16#0001   | -  | Achse nur über eine "MC_Power"-Instanz freigegeben   |
| <b>16#8202</b> |           | <b>Maximale Anzahl gleichzeitig aktiver Motion Control-Aufträge überschritten (max. 200 Aufträge für alle Motion Control-Technologieobjekte)</b> |  |
|                | 16#0001   | -  | Anzahl der gleichzeitig aktiven Aufträge verringern;<br>Auftrag erneut starten<br><br>Ein aktiver Auftrag ist am Parameter "Busy" = TRUE der Motion Control-Anweisung zu erkennen. |
| <b>16#8203</b> |           | <b>Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben</b>  |  |
|                | 16#0001   | -  | "Handsteuerung" beenden; Auftrag erneut starten  |
| <b>16#8204</b> |           | <b>Achse ist nicht referenziert</b>  |  |
|                | 16#0001   | -  | Achse mit der Anweisung "MC_Home" referenzieren; Auftrag erneut starten  |
| <b>16#8205</b> |           | <b>Achse wird momentan durch das Anwenderprogramm gesteuert (Fehler wird nur in der Achssteuertafel angezeigt)</b>                               |  |
|                | 16#0013   | Die Achse ist im Anwenderprogramm freigegeben  | Achse mit der Anweisung "MC_Power" sperren und in der Achssteuertafel erneut "Handsteuerung" anwählen  |
| <b>16#8206</b> |           | <b>Technologieobjekt ist noch nicht aktiviert</b>  |  |
|                | 16#0001   | -  | Aktivieren Sie die Achse mit der Anweisung "MC_Power" Enable = TRUE, oder geben Sie die Achse in der Achssteuertafel frei.   |
| <b>16#8207</b> |           | <b>Auftrag abgewiesen</b>  |  |
|                | 16#0016   | Aktives Referenzieren läuft; eine andere Referenzierart kann nicht gestartet werden.   | Warten Sie den Abschluss des aktiven Referenzierens ab oder brechen Sie das aktive Referenzieren durch einen Bewegungsauftrag z. B. "MC_Halt" ab.                                  |
|                | 16#0018   | Die Achse kann nicht mit einer Auftrags-tabelle verfahren werden, während die Achse direkt oder passiv referenziert wird.                        | Warten Sie den Abschluss des direkten oder passiven Referenzierens ab.   |

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe   |
|----------------|-----------|--|---|
|                | 16#0019   | Während der Abarbeitung einer Auftragstabelle, kann die Achse nicht direkt oder passiv referenziert werden.      | Warten Sie den Abschluss der Auftragstabelle ab oder brechen Sie die Auftragstabelle durch einen Bewegungsauftrag z. B. "MC_Halt" ab. |
| <b>16#8208</b> |           | <b>Geschwindigkeitsdifferenz zwischen maximaler Geschwindigkeit und Start/Stopp-Geschwindigkeit ist ungültig</b> |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | Wert korrigieren; Auftrag erneut starten  |
|                | 16#000A   | Wert ist kleiner oder gleich 0.  |   |
| <b>16#8209</b> |           | <b>Wert der Beschleunigung des Technologieobjekts "Achse" ist ungültig</b>                                       |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | Wert korrigieren; Auftrag erneut starten  |
|                | 16#000A   | Wert ist kleiner oder gleich 0.  |   |
| <b>16#820A</b> |           | <b>Restart der Achse nicht möglich</b>   |   |
|                | 16#0013   | Die Achse ist im Anwenderprogramm freigegeben  | Achse mit der Anweisung "MC_Power" sperren; Restart erneut ausführen  |
|                | 16#0027   | Die Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben   | "Handsteuerung" beenden; Restart erneut ausführen   |
| <b>16#820B</b> |           | <b>Ausführen der Auftragstabelle nicht möglich</b>   |   |
|                | 16#0026   | Ladevorgang im Betriebszustand RUN wird ausgeführt   | Warten Sie bis der Ladevorgang abgeschlossen ist  |

### Bausteinparameterfehler

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe   |
|----------------|-----------|--|---|
| <b>16#8400</b> |           | <b>Wert am Parameter "Position" der Motion Control-Anweisung ist ungültig</b>  |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | Wert korrigieren; Auftrag erneut starten        |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )                |   |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )              |   |
| <b>16#8401</b> |           | <b>Wert am Parameter "Distance" der Motion Control-Anweisung ist ungültig</b>  |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | Wert korrigieren; Auftrag erneut starten        |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )                |   |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )              |   |
| <b>16#8402</b> |           | <b>Wert am Parameter "Velocity" der Motion Control-Anweisung ist ungültig</b>  |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | Wert korrigieren; Auftrag erneut starten        |
|                | 16#0008   | Wert ist größer als die konfigurierte maximale Geschwindigkeit                 |   |
|                | 16#0009   | Wert ist kleiner als die konfigurierte Start/Stopp-Geschwindigkeit             |   |
|                | 16#0024   | Wert ist kleiner 0   |   |
| <b>16#8403</b> |           | <b>Wert am Parameter "Direction" der Motion Control-Anweisung ist ungültig</b> |   |
|                | 16#0011   | Der Auswahlwert ist ungültig   | Auswahlwert korrigieren; Auftrag erneut starten |
| <b>16#8404</b> |           | <b>Wert am Parameter "Mode" der Motion Control-Anweisung ist ungültig</b>      |   |
|                | 16#0011   | Der Auswahlwert ist ungültig   | Auswahlwert korrigieren; Auftrag erneut starten |

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe  |
|----------------|-----------|---|--|
|                | 16#0015   | Aktives / Passives Referenzieren ist nicht konfiguriert   | Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten   |
|                | 16#0017   | Die Richtungsumkehr am HW-Endschalter ist aktiviert, obwohl die HW-Endschalter deaktiviert sind | <ul style="list-style-type: none"> <li>• Aktivieren der HW-Endschalter über die Variable &lt;Achse&gt;.Config.PositionLimits_HW.Active = TRUE, Auftrag erneut starten</li> <li>• Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten</li> </ul> |
| <b>16#8405</b> |           | <b>Wert am Parameter "StopMode" der Motion Control-Anweisung ist ungültig</b>                   |  |
|                | 16#0011   | Der Auswahlwert ist ungültig  | Auswahlwert korrigieren; Achse erneut freigeben  |
| <b>16#8406</b> |           | <b>Gleichzeitiges Tippen vorwärts und rückwärts nicht erlaubt</b>                               |  |
|                | 16#0001   | -   | Verhindern Sie den gleichzeitigen Signalzustand TRUE der Parameter "JogForward" und "JogBackward"; starten Sie den Auftrag erneut.   |
| <b>16#8407</b> |           | <b>Wechsel der Achse an der Anweisung "MC_Power" nur bei gesperrter Achse zulässig.</b>         |  |
|                | 16#0001   | -   | Aktive Achse sperren; anschließend kann die Achse gewechselt und freigegeben werden.   |
| <b>16#8408</b> |           | <b>Wert am Parameter "Axis" der Motion Control-Anweisung ist ungültig</b>                       |  |
|                | 16#001A   | Der angegebene Wert entspricht nicht der geforderten Version des Technologieobjekts             | Wert korrigieren; Auftrag erneut starten   |
|                | 16#001B   | Der angegebene Wert entspricht nicht dem geforderten Typ des Technologieobjekts                 |  |
|                | 16#001C   | Der angegebene Wert ist kein Motion Control-Technologiedatenbaustein                            |  |
| <b>16#8409</b> |           | <b>Wert am Parameter "CommandTable" der Motion Control-Anweisung ist ungültig</b>               |  |
|                | 16#001A   | Der angegebene Wert entspricht nicht der geforderten Version des Technologieobjekts             | Wert korrigieren; Auftrag erneut starten   |
|                | 16#001B   | Der angegebene Wert entspricht nicht dem geforderten Typ des Technologieobjekts                 |  |
|                | 16#001C   | Der angegebene Wert ist kein Motion Control-Technologiedatenbaustein                            |  |
| <b>16#840A</b> |           | <b>Wert am Parameter "StartStep" der Motion Control-Anweisung ist ungültig</b>                  |  |
|                | 16#000A   | Wert ist kleiner oder gleich 0.   | Wert korrigieren; Auftrag erneut starten   |
|                | 16#001D   | Der Startschritt ist größer als der Endschrift  |  |
|                | 16#001E   | Wert ist größer 32  |  |
| <b>16#840B</b> |           | <b>Wert am Parameter "EndStep" der Motion Control-Anweisung ist ungültig</b>                    |  |
|                | 16#000A   | Wert ist kleiner oder gleich 0.   | Wert korrigieren; Auftrag erneut starten   |
|                | 16#001E   | Wert ist größer 32  |  |
| <b>16#840C</b> |           | <b>Wert am Parameter "RampUpTime" der Motion Control-Anweisung ist ungültig</b>                 |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | Wert korrigieren; Auftrag erneut starten   |
|                | 16#000A   | Wert ist kleiner oder gleich 0.   |  |
| <b>16#840D</b> |           | <b>Wert am Parameter "RampDownTime" der Motion Control-Anweisung ist ungültig</b>               |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | Wert korrigieren; Auftrag erneut starten   |
|                | 16#000A   | Wert ist kleiner oder gleich 0.   |  |
| <b>16#840E</b> |           | <b>Wert am Parameter "EmergencyRampTime" der Motion Control-Anweisung ist ungültig</b>          |  |

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe                                  |
|----------------|-----------|---|--|
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | Wert korrigieren; Auftrag erneut starten |
|                | 16#000A   | Wert ist kleiner oder gleich 0.   |  |
| <b>16#840F</b> |           | <b>Wert am Parameter "JerkTime" der Motion Control-Anweisung ist ungültig</b> |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | Wert korrigieren; Auftrag erneut starten |
|                | 16#000A   | Wert ist kleiner oder gleich 0.   |  |

### Konfigurationsfehler der Achse

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe   |
|----------------|-----------|---|---|
| <b>16#8600</b> |           | <b>Parametrierung des Impulsgenerators (PTO) ist ungültig</b>               |   |
|                | 16#000B   | Die Adresse ist ungültig  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul> |
|                | 16#0014   | Die ausgewählte Hardware wird von einer anderen Anwendung verwendet         |   |
| <b>16#8601</b> |           | <b>Parametrierung des schnellen Zählers (HSC) ist ungültig</b>              |   |
|                | 16#000B   | Die Adresse ist ungültig  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul> |
|                | 16#0014   | Die ausgewählte Hardware wird von einer anderen Anwendung verwendet         |   |
| <b>16#8602</b> |           | <b>Parametrierung des "Freigabe-Ausgangs" ist ungültig</b>                  |   |
|                | 16#000B   | Die Adresse ist ungültig  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul> |
| <b>16#8603</b> |           | <b>Parametrierung des "Bereit-Eingangs" ist ungültig</b>                    |   |
|                | 16#000B   | Die Adresse ist ungültig  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul> |
| <b>16#8604</b> |           | <b>Wert für "Impulse pro Motorumdrehung" ist ungültig</b>                   |   |
|                | 16#000A   | Wert ist kleiner oder gleich Null   | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul> |
| <b>16#8605</b> |           | <b>Wert für "Weg pro Motorumdrehung" ist ungültig</b>                       |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul> |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )             |   |
|                | 16#000A   | Wert ist kleiner oder gleich Null   |   |
| <b>16#8606</b> |           | <b>Wert für "Start/Stopp-Geschwindigkeit" ist ungültig</b>                  |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul> |
|                | 16#0003   | Wert ist größer als die obere Hardwaregrenze                                |   |
|                | 16#0004   | Wert ist kleiner als die untere Hardwaregrenze                              |   |
|                | 16#0007   | Die Start/Stopp-Geschwindigkeit ist größer als die maximale Geschwindigkeit |   |
| <b>16#8607</b> |           | <b>Wert für "maximale Geschwindigkeit" ist ungültig</b>                     |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul> |
|                | 16#0003   | Wert ist größer als die obere Hardwaregrenze                                |   |
|                | 16#0004   | Wert ist kleiner als die untere Hardwaregrenze                              |   |

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe  |
|----------------|-----------|---|--|
| <b>16#8608</b> |           | <b>Wert für "Beschleunigung" ist ungültig</b>   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0003   | Wert ist größer als die obere Hardwaregrenze  |  |
|                | 16#0004   | Wert ist kleiner als die untere Hardwaregrenze  |  |
| <b>16#8609</b> |           | <b>Wert für "Verzögerung" ist ungültig</b>  |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0003   | Wert ist größer als die obere Hardwaregrenze  |  |
|                | 16#0004   | Wert ist kleiner als die untere Hardwaregrenze  |  |
| <b>16#860A</b> |           | <b>Wert für "Notstopp-Verzögerung" ist ungültig</b>   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0003   | Wert ist größer als die obere Hardwaregrenze  |  |
|                | 16#0004   | Wert ist kleiner als die untere Hardwaregrenze  |  |
| <b>16#860B</b> |           | <b>Wert für die Position des unteren SW-Endschalters ist ungültig</b>                       |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )                             |  |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )                           |  |
|                | 16#0007   | Der Positionswert des unteren SW-Endschalters ist größer als der des oberen SW-Endschalters |  |
| <b>16#860C</b> |           | <b>Wert für die Position des oberen SW-Endschalters ist ungültig</b>                        |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )                             |  |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )                           |  |
| <b>16#860D</b> |           | <b>Adresse des unteren HW-Endschalters ist ungültig</b>                                     |  |
|                | 16#000C   | Die Adresse der fallenden Flanke ist ungültig   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul>  |
|                | 16#000D   | Die Adresse der steigenden Flanke ist ungültig  |  |
| <b>16#860E</b> |           | <b>Adresse des oberen HW-Endschalter ist ungültig</b>                                       |  |
|                | 16#000C   | Die Adresse der fallenden Flanke ist ungültig   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul>  |
|                | 16#000D   | Die Adresse der steigenden Flanke ist ungültig  |  |
| <b>16#860F</b> |           | <b>Wert für "Referenzpunktverschiebung" ist ungültig</b>                                    |  |

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe  |
|----------------|-----------|--|--|
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )  |  |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )  |  |
| <b>16#8610</b> |           | <b>Wert für "Anfahrsgeschwindigkeit" ist ungültig</b>  |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0008   | Die Geschwindigkeit ist größer als die maximale Geschwindigkeit  |  |
|                | 16#0009   | Die Geschwindigkeit ist kleiner als die Start/ Stopp-Geschwindigkeit   |  |
| <b>16#8611</b> |           | <b>Wert für "Referenziergeschwindigkeit" ist ungültig</b>  |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0008   | Die Geschwindigkeit ist größer als die maximale Geschwindigkeit  |  |
|                | 16#0009   | Die Geschwindigkeit ist kleiner als die Start/ Stopp-Geschwindigkeit   |  |
| <b>16#8612</b> |           | <b>Adresse des Referenzpunktschalters ist ungültig</b>   |  |
|                | 16#000C   | Die Adresse der fallenden Flanke ist ungültig  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> </ul>  |
|                | 16#000D   | Die Adresse der steigenden Flanke ist ungültig   |  |
| <b>16#8613</b> |           | <b>Beim aktiven Referenzieren ist die Richtungsumkehr am HW-Endschalter aktiviert, obwohl die HW-Endschalter nicht konfiguriert sind</b> |  |
|                | 16#0001   | -  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
| <b>16#8614</b> |           | <b>Wert für "Ruck" ist ungültig</b>  |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | <ul style="list-style-type: none"> <li>Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#001F   | Wert ist größer als der maximal zulässige Ruck   |  |
|                | 16#0020   | Wert ist kleiner als der minimal zulässige Ruck  |  |
| <b>16#8615</b> |           | <b>Wert für "Maßeinheit" ist ungültig</b>  |  |
|                | 16#0011   | Der Auswahlwert ist ungültig   | Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben  |

## Konfigurationsfehler der Auftragstabelle

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe  |
|----------------|-----------|---|--|
| <b>16#8700</b> |           | <b>Wert für "Auftragstyp" in der Auftragstabelle ist ungültig</b>           |  |
|                | 16#0001   | -   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten</li> </ul> |
| <b>16#8701</b> |           | <b>Wert für "Position / Verfahrweg" in der Auftragstabelle ist ungültig</b> |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )             |  |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )           |  |
| <b>16#8702</b> |           | <b>Wert für "Geschwindigkeit" in der Auftragstabelle ist ungültig</b>       |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0008   | Wert ist größer als die konfigurierte maximale Geschwindigkeit              |  |
|                | 16#0009   | Wert ist kleiner als die konfigurierte Start/Stop-Geschwindigkeit           |  |
| <b>16#8703</b> |           | <b>Wert für "Dauer" in der Auftragstabelle ist ungültig</b>                 |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0021   | Wert ist größer als 64800 s   |  |
|                | 16#0022   | Wert ist kleiner als 0.001 s  |  |
| <b>16#8704</b> |           | <b>Wert für "Nächster Schritt" in der Auftragstabelle ist ungültig</b>      |  |
|                | 16#0011   | Der Auswahlwert ist ungültig  | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0023   | Der Auftragsübergang ist für diesen Auftrag nicht erlaubt                   |  |

## Interne Fehler

| ErrorID        | ErrorInfo | Beschreibung           | Abhilfe  |
|----------------|-----------|------------------------|--|
| <b>16#8FFF</b> |           | <b>Interner Fehler</b> |  |
|                | 16#F0**   | -                      | <p>NETZ-AUS und NETZ-EIN der CPU</p> <p>Sollte dies nicht zum Erfolg führen, kontaktieren Sie den Customer Support. Halten Sie folgende Information bereit:</p> <ul style="list-style-type: none"> <li>• ErrorID</li> <li>• ErrorInfo</li> <li>• Diagnosepuffereinträge</li> </ul> |

**Siehe auch**

- Mehrere Achsen mit gleichem PTO einsetzen (Seite 4219)
- Mehrere Antriebe mit gleichem PTO einsetzen (Seite 4223)
- Aufträge aus höheren Prioritätsklassen (Ablaufebenen) nachverfolgen (Seite 4224)
- Sonderfälle beim Einsatz von Software-Endschaltern (Seite 4226)
- Variable des Technologieobjekts Achse (Seite 4242)

**11.2.12.8 Variable des Technologieobjekts Achse**

**Variable Config.**

**Variable Config.General.**

**Legende**

|                  |  |   |  |  |
|------------------|--|---|--|--|
| <b>Datentyp</b>  | Datentyp der Variable  |   |  |  |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Achse überschrieben werden. |   |  |  |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:  |   |  |  |
|                  | RW   | Die Variable kann im Anwenderprogramm gelesen und geschrieben werden. |  |  |
|                  | R  | Die Variable kann im Anwenderprogramm gelesen werden.                 |  |  |
|                  | -  | Die Variable kann im Anwenderprogramm nicht verwendet werden.         |  |  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.  |   |  |  |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden.  |   |  |  |

|  |                  |                |                |            |
|--|------------------|----------------|----------------|------------|
| <b>&lt;Achse&gt;.Config.General.PTO</b>            |                  |                |                |            |
| Variable ist im Anwenderprogramm nicht auswertbar. |                  |                |                |            |
| <b>Datentyp</b>                                    | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> | <b>HMI</b> |
| DWORD  | DW#16#00000000   | -              | -              | -          |

|  |                  |                |                |            |
|--|------------------|----------------|----------------|------------|
| <b>&lt;Achse&gt;.Config.General.HSC</b>            |                  |                |                |            |
| Variable ist im Anwenderprogramm nicht auswertbar. |                  |                |                |            |
| <b>Datentyp</b>                                    | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> | <b>HMI</b> |
| DWORD  | DW#16#00000000   | -              | -              | -          |



| <b>&lt;Achse&gt;.Config.General.LengthUnit (Technologieobjekt "Achse" ab V2.0)</b>  |           |         |         |     |
|---|-----------|---------|---------|-----|
| Die in der Konfiguration gewählte Maßeinheit der Parameter:   |           |         |         |     |
| <ul style="list-style-type: none"> <li>• 1013 = "mm"</li> <li>• 1010 =: "m"</li> <li>• 1019 = "in"</li> <li>• 1018 = "ft"</li> <li>• 1005 = "°" (Grad)</li> <li>• -1 = "Impulse"</li> </ul> |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Int   | 1013      | R       | -       | X   |

### Variable Config.DriveInterface.

#### Legende

|                  |  |   |  |  |
|------------------|--|---|--|--|
| <b>Datentyp</b>  | Datentyp der Variable  |   |  |  |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Achse überschrieben werden. |   |  |  |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:  |   |  |  |
|                  | RW   | Die Variable kann im Anwenderprogramm gelesen und geschrieben werden. |  |  |
|                  | R  | Die Variable kann im Anwenderprogramm gelesen werden.                 |  |  |
|                  | -  | Die Variable kann im Anwenderprogramm nicht verwendet werden.         |  |  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.  |   |  |  |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden.  |   |  |  |

| <b>&lt;Achse&gt;.Config.DriveInterface.EnableOutput...</b> |           |         |         |     |
|--|-----------|---------|---------|-----|
| Die Variablen sind im Anwenderprogramm nicht auswertbar.   |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| -  | -         | -       | -       | -   |

| <b>&lt;Achse&gt;.Config.DriveInterface.ReadyInput...</b> |           |         |         |     |
|--|-----------|---------|---------|-----|
| Die Variablen sind im Anwenderprogramm nicht auswertbar. |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| -  | -         | -       | -       | -   |

### Variable Config.Mechanics.

#### Legende

|                  |  |
|------------------|--|
| <b>Datentyp</b>  | Datentyp der Variable  |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Achse überschrieben werden. |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:  |
|                  | RW   Die Variable kann im Anwenderprogramm gelesen und geschrieben werden.                           |
|                  | R   Die Variable kann im Anwenderprogramm gelesen werden.  |
| -                | Die Variable kann im Anwenderprogramm nicht verwendet werden.  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.  |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden.  |

| <Achse>.Config.Mechanics.PulsesPerDriveRevolution |           |         |         |     |
|---|-----------|---------|---------|-----|
| Impulse pro Motorumdrehung                        |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| DInt  | L#1000    | R       | -       | X   |

| <Achse>.Config.Mechanics.LeadScrew                               |           |         |         |     |
|--|-----------|---------|---------|-----|
| Weg pro Motorumdrehung (Angabe in der konfigurierten Maßeinheit) |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Real   | 1.0E+001  | R       | -       | X   |

| <Achse>.Config.Mechanics.InverseDirection |           |         |         |     |
|---|-----------|---------|---------|-----|
| Richtungssinn invertieren                 |           |         |         |     |
| Datentyp                                  | Startwert | Zugriff | wirksam | HMI |
| Bool                                      | FALSE     | R       | -       | X   |

### Variable Config.DynamicLimits.

#### Legende

|                  |  |
|------------------|--|
| <b>Datentyp</b>  | Datentyp der Variable  |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Achse überschrieben werden. |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:  |
|                  | RW   Die Variable kann im Anwenderprogramm gelesen und geschrieben werden.                           |
|                  | R   Die Variable kann im Anwenderprogramm gelesen werden.  |
| -                | Die Variable kann im Anwenderprogramm nicht verwendet werden.  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.  |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden.  |

| <b>&lt;Achsenname&gt;.Config.DynamicLimits.MinVelocity</b>                     |           |         |         |     |
|--|-----------|---------|---------|-----|
| Start/Stop-Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit) |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Real   | 1.0E+001  | R       | -       | X   |

| <b>&lt;Achsenname&gt;.Config.DynamicLimits.MaxVelocity</b>                   |           |         |         |     |
|--|-----------|---------|---------|-----|
| Maximale Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit) |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Real   | 2.5E+002  | R       | -       | X   |

### Variable Config.DynamicDefaults.

#### Legende

|                  |  |  |  |  |
|------------------|--|--|--|--|
| <b>Datentyp</b>  | Datentyp der Variable  |  |  |  |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Achse überschrieben werden. |  |  |  |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:  |  |  |  |
|                  | RW   | Die Variable kann im Anwenderprogramm gelesen und geschrieben werden.  |  |  |
|                  | R  | Die Variable kann im Anwenderprogramm gelesen werden.  |  |  |
|                  | -  | Die Variable kann im Anwenderprogramm nicht verwendet werden.  |  |  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.  |  |  |  |
|                  | 1  | Mit dem Aktivieren (Variable <Achsenname>.StatusBits.Activated wechselt von FALSE -> TRUE), Sperren oder Freigeben der Achse   |  |  |
|                  | 2  | Mit dem Freigeben der Achse  |  |  |
|                  | 5  | Mit dem nächsten Start eines MC_MoveAbsolute-, MC_MoveRelative-, MC_MoveVelocity-, MC_MoveJog-, MC_Halt-, MC_CommandTable- oder aktiven MC_Home-Auftrags (Mode = 3). |  |  |
|                  | 6  | Mit dem Stoppen eines MC_MoveJog-Auftrags  |  |  |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden.  |  |  |  |

| <b>&lt;Achsenname&gt;.Config.DynamicDefaults.Acceleration</b>      |           |         |               |                         |   |
|--|-----------|---------|---------------|-------------------------|---|
| Beschleunigung der Achse (Angabe in der konfigurierten Maßeinheit) |           |         |               |                         |   |
| Datentyp   | Startwert | Zugriff | wirksam       | HMI                     |   |
| Real   | 4.8E+001  | RW      | 5             | CPU Firmware V1.0       | X |
|  |           |         | 1,<br>5,<br>6 | CPU Firmware ab<br>V2.0 |   |

| <b>&lt;Achsenname&gt;.Config.DynamicDefaults.Deceleration</b>   |           |         |         |     |
|---|-----------|---------|---------|-----|
| Verzögerung der Achse (Angabe in der konfigurierten Maßeinheit) |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |

| <Achse>.Config.DynamicDefaults.Deceleration |          |    |         |                      |   |
|---|----------|----|---------|----------------------|---|
| Real  | 4.8E+001 | RW | 5, 6    | CPU Firmware V1.0    | X |
|   |          |    | 1, 5, 6 | CPU Firmware ab V2.0 |   |

| <Achse>.Config.DynamicDefaults.EmergencyDeceleration                     |           |         |         |                      |     |
|--|-----------|---------|---------|----------------------|-----|
| Notstopp-Verzögerung der Achse (Angabe in der konfigurierten Maßeinheit) |           |         |         |                      |     |
| Datentyp   | Startwert | Zugriff | wirksam |                      | HMI |
| Real   | 1.2E+002  | RW      | 2, 5, 6 | CPU Firmware V1.0    | X   |
|  |           |         | 1, 5, 6 | CPU Firmware ab V2.0 |     |

| <Achse>.Config.DynamicDefaults.JerkActive (Technologieobjekt "Achse" ab V2.0) |           |         |         |  |     |
|---|-----------|---------|---------|--|-----|
| TRUE = Die Ruckbegrenzung ist aktiviert                                       |           |         |         |  |     |
| Datentyp  | Startwert | Zugriff | wirksam |  | HMI |
| Bool  | FALSE     | RW      | 1, 5    |  | X   |

| <Achse>.Config.DynamicDefaults.Jerk (Technologieobjekt "Achse" ab V2.0)                                     |           |         |         |  |     |
|---|-----------|---------|---------|--|-----|
| Ruck während der Beschleunigungs- und Verzögerungsrampe der Achse (Angabe in der konfigurierten Maßeinheit) |           |         |         |  |     |
| Datentyp  | Startwert | Zugriff | wirksam |  | HMI |
| Real  | 1.92E+002 | RW      | 1, 5    |  | X   |

**Variable Config.PositionLimits\_SW.**

**Legende**

|                  |  |  |  |  |  |
|------------------|--|--|--|--|--|
| <b>Datentyp</b>  | Datentyp der Variable  |  |  |  |  |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Achse überschrieben werden. |  |  |  |  |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:  |  |  |  |  |
|                  | RW   | Die Variable kann im Anwenderprogramm gelesen und geschrieben werden.  |  |  |  |
|                  | R  | Die Variable kann im Anwenderprogramm gelesen werden.  |  |  |  |
|                  | -  | Die Variable kann im Anwenderprogramm nicht verwendet werden.  |  |  |  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.  |  |  |  |  |
|                  | 1  | Mit dem Aktivieren (Variable <Achse>.StatusBits.Activated wechselt von FALSE -> TRUE), Sperren oder Freigeben der Achse  |  |  |  |
|                  | 4  | Nach einem Stillstand der Achse mit dem nächsten Start eines Motion Control-Auftrags. Der Stillstand der Achse kann mit der Variable <Achse>.StatusBits.Standstill geprüft werden. |  |  |  |

|            |   |  |
|------------|---|--|
|            | 5   | Mit dem nächsten Start eines MC_MoveAbsolute-, MC_MoveRelative-, MC_MoveVelocity-, MC_MoveJog-, MC_Halt-, MC_CommandTable- oder aktiven MC_Home-Auftrags (Mode = 3). |
| <b>HMI</b> | Die Variable kann in einem HMI-System verwendet werden. |  |

| <Achsname>.Config.PositionLimits_SW.Active     |           |         |               |                      |     |
|--|-----------|---------|---------------|----------------------|-----|
| TRUE = Die Software-Endschalter sind aktiviert |           |         |               |                      |     |
| Datentyp                                       | Startwert | Zugriff | wirksam       |                      | HMI |
| Bool   | FALSE     | RW      | 4             | CPU Firmware V1.0    | X   |
|  |           |         | 1,<br>5,<br>6 | CPU Firmware ab V2.0 |     |

| <Achsname>.Config.PositionLimits_SW.MinPosition                                      |           |         |               |                      |     |
|--|-----------|---------|---------------|----------------------|-----|
| Position des unteren Software-Endschalters (Angabe in der konfigurierten Maßeinheit) |           |         |               |                      |     |
| Datentyp   | Startwert | Zugriff | wirksam       |                      | HMI |
| Real   | -1.0E+004 | RW      | 4             | CPU Firmware V1.0    | X   |
|  |           |         | 1,<br>5,<br>6 | CPU Firmware ab V2.0 |     |

| <Achsname>.Config.PositionLimits_SW.MaxPosition                                     |           |         |               |                      |     |
|---|-----------|---------|---------------|----------------------|-----|
| Position des oberen Software-Endschalters (Angabe in der konfigurierten Maßeinheit) |           |         |               |                      |     |
| Datentyp  | Startwert | Zugriff | wirksam       |                      | HMI |
| Real  | 1.0E+004  | RW      | 4             | CPU Firmware V1.0    | X   |
|   |           |         | 1,<br>5,<br>6 | CPU Firmware ab V2.0 |     |

### Variable Config.PositionLimits\_HW.

#### Legende

|                  |   |
|------------------|---|
| <b>Datentyp</b>  | Datentyp der Variable   |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Achse überschrieben werden.  |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:   |
|                  | RW Die Variable kann im Anwenderprogramm gelesen und geschrieben werden.  |
|                  | R Die Variable kann im Anwenderprogramm gelesen werden.   |
|                  | - Die Variable kann im Anwenderprogramm nicht verwendet werden.   |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.   |
|                  | 1 Mit dem Aktivieren (Variable <Achsname>.StatusBits.Activated wechselt von FALSE -> TRUE), Sperren oder Freigeben der Achse  |
|                  | 3 Nach dem Freigeben der Achse (die Achse muss zuvor im Stillstand gewesen sein). Der Stillstand der Achse kann mit der Variable <Achsname>.StatusBits.Standstill geprüft werden. |

|            |   |   |
|------------|---|---|
|            | 4   | Nach einem Stillstand der Achse mit dem nächsten Start eines Motion Control-Auftrags. Der Stillstand der Achse kann mit der Variable <Achsenname>.StatusBits.Standstill geprüft werden. |
|            | 5   | Mit dem nächsten Start eines MC_MoveAbsolute-, MC_MoveRelative-, MC_MoveVelocity-, MC_MoveJog-, MC_Halt-, MC_CommandTable- oder aktiven MC_Home-Auftrags (Mode = 3).                    |
| <b>HMI</b> | Die Variable kann in einem HMI-System verwendet werden. |   |

|   |                  |                |                |                      |            |
|---|------------------|----------------|----------------|----------------------|------------|
| <b>&lt;Achsenname&gt;.Config.PositionLimits_HW.Active</b> |                  |                |                |                      |            |
| TRUE = Die Hardware-Endschalter sind aktiv.               |                  |                |                |                      |            |
| <b>Datentyp</b>   | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> |                      | <b>HMI</b> |
| Bool  | FALSE            | RW             | 3, 4           | CPU Firmware V1.0    | X          |
|   |                  |                | 1, 5, 6        | CPU Firmware ab V2.0 |            |

|  |                  |                |                |  |            |
|--|------------------|----------------|----------------|--|------------|
| <b>&lt;Achsenname&gt;.Config.PositionLimits_HW.MinSwitchedLevel</b>  |                  |                |                |  |            |
| TRUE = 24 V am CPU-Eingang entspricht unterer Hardware-Endschalter angefahren<br>FALSE = 0 V am CPU-Eingang entspricht unterer Hardware-Endschalter angefahren |                  |                |                |  |            |
| <b>Datentyp</b>  | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> |  | <b>HMI</b> |
| Bool   | FALSE            | R              | -              |  | X          |

|  |                  |                |                |  |            |
|--|------------------|----------------|----------------|--|------------|
| <b>&lt;Achsenname&gt;.Config.PositionLimits_HW.MinFallingEvent</b> |                  |                |                |  |            |
| Variable ist im Anwenderprogramm nicht auswertbar.                 |                  |                |                |  |            |
| <b>Datentyp</b>  | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> |  | <b>HMI</b> |
| DWord  | DW#16#00000000   | -              | -              |  | -          |

|   |                  |                |                |  |            |
|---|------------------|----------------|----------------|--|------------|
| <b>&lt;Achsenname&gt;.Config.PositionLimits_HW.MinRisingEvent</b> |                  |                |                |  |            |
| Variable ist im Anwenderprogramm nicht auswertbar.                |                  |                |                |  |            |
| <b>Datentyp</b>   | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> |  | <b>HMI</b> |
| DWord   | DW#16#00000000   | -              | -              |  | -          |

|  |                  |                |                |  |            |
|--|------------------|----------------|----------------|--|------------|
| <b>&lt;Achsenname&gt;.Config.PositionLimits_HW.MaxSwitchedLevel</b>  |                  |                |                |  |            |
| TRUE = 24 V am CPU-Eingang entspricht oberer Hardware-Endschalter angefahren<br>FALSE = 0 V am CPU-Eingang entspricht oberer Hardware-Endschalter angefahren |                  |                |                |  |            |
| <b>Datentyp</b>  | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> |  | <b>HMI</b> |
| Bool   | FALSE            | R              | -              |  | X          |

|  |                  |                |                |  |            |
|--|------------------|----------------|----------------|--|------------|
| <b>&lt;Achsenname&gt;.Config.PositionLimits_HW.MaxFallingEvent</b> |                  |                |                |  |            |
| Variable ist im Anwenderprogramm nicht auswertbar.                 |                  |                |                |  |            |
| <b>Datentyp</b>  | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> |  | <b>HMI</b> |
| DWord  | DW#16#00000000   | -              | -              |  | -          |

| <b>&lt;Achsname&gt;.Config.PositionLimits_HW.MaxRisingEvent</b> |                |         |         |     |
|---|----------------|---------|---------|-----|
| Variable ist im Anwenderprogramm nicht auswertbar.              |                |         |         |     |
| Datentyp  | Startwert      | Zugriff | wirksam | HMI |
| DWord   | DW#16#00000000 | -       | -       | -   |

## Variable Config.Homing.

### Legende

|                  |  |  |  |  |
|------------------|--|--|--|--|
| <b>Datentyp</b>  | Datentyp der Variable  |  |  |  |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Achse überschrieben werden. |  |  |  |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:  |  |  |  |
|                  | RW   | Die Variable kann im Anwenderprogramm gelesen und geschrieben werden.  |  |  |
|                  | R  | Die Variable kann im Anwenderprogramm gelesen werden.  |  |  |
|                  | -  | Die Variable kann im Anwenderprogramm nicht verwendet werden.  |  |  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.  |  |  |  |
|                  | 1  | Mit dem Aktivieren (Variable <Achsname>.StatusBits.Activated wechselt von FALSE -> TRUE), Sperren oder Freigeben der Achse |  |  |
|                  | 7  | Mit dem Start eines passiven Referenzierauftrags   |  |  |
|                  | 8  | Mit dem Start eines aktiven Referenzierauftrags  |  |  |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden.  |  |  |  |

| <b>&lt;Achsname&gt;.Config.Homing.AutoReversal</b>                                  |           |         |         |                                |     |
|---|-----------|---------|---------|--------------------------------|-----|
| TRUE = Richtungsumkehr am Hardware-Endschalter aktiviert (aktives Referenzieren)    |           |         |         |                                |     |
| FALSE = Richtungsumkehr am Hardware-Endschalter deaktiviert (aktives Referenzieren) |           |         |         |                                |     |
| Datentyp  | Startwert | Zugriff | wirksam |                                | HMI |
| Bool  | TRUE      | R       | -       | Technologieobjekt "Achse" V1.0 | X   |
|   |           | RW      | 1, 8    | Technologieobjekt "Achse" V2.0 |     |

| <b>&lt;Achsname&gt;.Config.Homing.Direction</b>  |           |         |         |                                |     |
|--|-----------|---------|---------|--------------------------------|-----|
| TRUE = positive Anfahrriichtung zur Suche des Referenzpunktschalters und positive Referenzierrichtung (aktives Referenzieren)  |           |         |         |                                |     |
| FALSE = negative Anfahrriichtung zur Suche des Referenzpunktschalters und positive Referenzierrichtung (aktives Referenzieren) |           |         |         |                                |     |
| Datentyp   | Startwert | Zugriff | wirksam |                                | HMI |
| Bool   | TRUE      | R       | -       | Technologieobjekt "Achse" V1.0 | X   |
|  |           | RW      | 1, 8    | Technologieobjekt "Achse" V2.0 |     |

| <b>&lt;Achsname&gt;.Config.Homing.SideActiveHoming (Technologieobjekt "Achse" ab V2.0)</b>    |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Referenzieren an der oberen Seite des Referenzpunktschalters (aktives Referenzieren)   |           |         |         |     |
| FALSE = Referenzieren an der unteren Seite des Referenzpunktschalters (aktives Referenzieren) |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | TRUE      | RW      | 1, 8    | X   |

| <b>&lt;Achsname&gt;.Config.Homing.SidePassiveHoming (Technologieobjekt "Achse" ab V2.0)</b>    |           |         |         |     |
|--|-----------|---------|---------|-----|
| TRUE = Referenzieren an der oberen Seite des Referenzpunktschalters (passives Referenzieren)   |           |         |         |     |
| FALSE = Referenzieren an der unteren Seite des Referenzpunktschalters (passives Referenzieren) |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Bool   | TRUE      | RW      | 1, 7    | X   |

| <b>&lt;Achsname&gt;.Config.Homing.RisingEdge (Technologieobjekt "Achse" V1.0)</b>   |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Referenzieren mit der negativen Signalfanke des Referenzpunktschalters (aktives Referenzieren)                         |           |         |         |     |
| FALSE = Referenzieren mit der positiven Signalfanke des Referenzpunktschalters (aktives Referenzieren)                        |           |         |         |     |
| Den Einfluss der Variable auf das passive Referenzieren entnehmen Sie der Beschreibung unter "Konfiguration - Referenzieren". |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.Config.Homing.Offset</b>  |           |         |         |                                |
|---|-----------|---------|---------|--------------------------------|
| Referenzpunktverschiebung / Angabe in der konfigurierten Maßeinheit (aktives Referenzieren) |           |         |         |                                |
| Datentyp  | Startwert | Zugriff | wirksam | HMI                            |
| Real  | 0.0       | R       | -       | Technologieobjekt "Achse" V1.0 |
|   |           | RW      | 1, 8    |                                |

| <b>&lt;Achsname&gt;.Config.Homing.FastVelocity</b>                                       |           |         |         |                                |
|--|-----------|---------|---------|--------------------------------|
| Anfahrsgeschwindigkeit / Angabe in der konfigurierten Maßeinheit (aktives Referenzieren) |           |         |         |                                |
| Datentyp   | Startwert | Zugriff | wirksam | HMI                            |
| Real   | 2.0E+002  | R       | -       | Technologieobjekt "Achse" V1.0 |
|  |           | RW      | 1, 8    |                                |

| <b>&lt;Achsname&gt;.Config.Homing.SlowVelocity</b>  |           |         |         |     |
|---|-----------|---------|---------|-----|
| Referenziertgeschwindigkeit / Angabe in der konfigurierten Maßeinheit (aktives Referenzieren) |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |



| <b>&lt;Achsname&gt;.Config.Homing.SlowVelocity</b> |          |    |      |                                |   |
|--|----------|----|------|--------------------------------|---|
| Real   | 4.0E+001 | R  | -    | Technologieobjekt "Achse" V1.0 | X |
|  |          | RW | 1, 8 | Technologieobjekt "Achse" V2.0 |   |

| <b>&lt;Achsname&gt;.Config.Homing.FallingEvent</b> |                |         |         |     |
|--|----------------|---------|---------|-----|
| Variable ist im Anwenderprogramm nicht auswertbar. |                |         |         |     |
| Datentyp   | Startwert      | Zugriff | wirksam | HMI |
| DWord  | DW#16#00000000 | -       | -       | -   |

| <b>&lt;Achsname&gt;.Config.Homing.RisingEvent</b>  |                |         |         |     |
|--|----------------|---------|---------|-----|
| Variable ist im Anwenderprogramm nicht auswertbar. |                |         |         |     |
| Datentyp   | Startwert      | Zugriff | wirksam | HMI |
| DWord  | DW#16#00000000 | -       | -       |     |

## Variable MotionStatus.

### Legende

|                  |   |   |
|------------------|---|---|
| <b>Datentyp</b>  | Datentyp der Variable                                   |   |
| <b>Startwert</b> | Startwert der Variable                                  |   |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:           |   |
|                  | RW  | Die Variable kann im Anwenderprogramm gelesen und geschrieben werden. |
|                  | R   | Die Variable kann im Anwenderprogramm gelesen werden.                 |
|                  | -   | Die Variable kann im Anwenderprogramm nicht verwendet werden.         |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.   |   |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden. |   |

| <b>&lt;Achsname&gt;.MotionStatus.Position</b>   |           |         |         |     |
|---|-----------|---------|---------|-----|
| Aktuelle Position der Achse (Angabe in der konfigurierten Maßeinheit). Bei nicht referenzierter Achse zeigt die Variable den Positionswert relativ zur Freigabe-Position der Achse. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Real  | 0.0       | R       | -       | X   |

| <b>&lt;Achsname&gt;.MotionStatus.Velocity</b>                                 |           |         |         |     |
|---|-----------|---------|---------|-----|
| Aktuelle Geschwindigkeit der Achse (Angabe in der konfigurierten Maßeinheit). |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Real  | 0.0       | R       | -       | X   |

| <b>&lt;Achse&gt;.MotionStatus.Distance</b>   |           |         |         |     |
|--|-----------|---------|---------|-----|
| Aktueller Abstand zur Zielposition der Achse (Angabe in der konfigurierten Maßeinheit). Der Wert der Variable ist nur während der Ausführung eines Positionierauftrags mit "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel gültig. |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Real   | 0.0       | R       | -       | X   |

| <b>&lt;Achse&gt;.MotionStatus.TargetPosition</b>   |           |         |         |     |
|--|-----------|---------|---------|-----|
| Zielposition der Achse (Angabe in der konfigurierten Maßeinheit). Der Wert der Variable ist nur während der Ausführung eines Positionierauftrags mit "MC_MoveAbsolute", "MC_MoveRelative" oder der Achssteuertafel gültig. |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Real   | 0.0       | R       | -       | X   |

**Siehe auch**

Bewegungsstatus (Seite 4217)

**Variable StatusBits.**

**Legende**

|                  |   |   |  |  |
|------------------|---|---|--|--|
| <b>Datentyp</b>  | Datentyp der Variable                                   |   |  |  |
| <b>Startwert</b> | Startwert der Variable                                  |   |  |  |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:           |   |  |  |
|                  | RW  | Die Variable kann im Anwenderprogramm gelesen und geschrieben werden. |  |  |
|                  | R   | Die Variable kann im Anwenderprogramm gelesen werden.                 |  |  |
|                  | -   | Die Variable kann im Anwenderprogramm nicht verwendet werden.         |  |  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.   |   |  |  |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden. |   |  |  |

| <b>&lt;Achse&gt;.StatusBits.Activated</b>   |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse ist aktiviert. Sie ist mit dem zugeordneten PTO (Pulse Train Output) verbunden. Die Daten des Technologie-Datenbausteins werden zyklisch aktualisiert. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achse&gt;.StatusBits.Enable</b>  |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse ist freigegeben und bereit Motion Control-Aufträge anzunehmen. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.HomingDone</b>  |           |         |         |     |
|--|-----------|---------|---------|-----|
| TRUE = Die Achse ist referenziert und kann absolute Positionieraufträge ausführen. Zum relativen Positionieren muss die Achse nicht referenziert sein.           |           |         |         |     |
| Während dem aktiven Referenzieren ist der Status FALSE. Während dem passiven Referenzieren bleibt der Status TRUE wenn die Achse zuvor bereits referenziert war. |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Bool   | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.Done</b>                    |           |         |         |     |
|--|-----------|---------|---------|-----|
| TRUE = An der Achse ist kein Motion Control-Auftrag aktiv. |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Bool   | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.Error</b>   |           |         |         |     |
|--|-----------|---------|---------|-----|
| TRUE = Am Technologieobjekt Achse ist ein Fehler aufgetreten. Detaillierte Informationen zum Fehler entnehmen Sie im Automatikbetrieb den Parametern "ErrorID" und "ErrorInfo" der Motion Control-Anweisungen. Im Handbetrieb wird die detaillierte Fehlerursache im Feld "Fehlermeldung" der Achssteuertafel angezeigt. |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Bool   | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.StandStill</b> |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse befindet sich im Stillstand. |           |         |         |     |
| Datentyp                                      | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.PositioningCommand</b> |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse führt einen Positionierauftrag aus.  |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.SpeedCommand</b>                               |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse führt einen Verfahrtauftrag mit Geschwindigkeitsvorgabe aus. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.Homing</b>  |           |         |         |     |
|--|-----------|---------|---------|-----|
| TRUE = Die Achse führt einen Referenzierauftrag der Motion Control-Anweisung "MC_Home" oder der Achssteuertafel aus. |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Bool   | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.CommandTableActive</b>                               |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse wird mit der Motion Control-Anweisung "MC_CommandTable" gesteuert. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.ConstantVelocity</b>       |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse verfährt mit konstanter Geschwindigkeit. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.Acceleration</b> |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse beschleunigt.                  |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.Deceleration</b> |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Achse verzögert (bremst ab).         |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.ControlPanelActive</b>   |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Die Betriebsart "Handsteuerung" wurde in der Achssteuertafel aktiviert. Die Achssteuertafel hat die Steuerhoheit über das Technologieobjekt Achse. Die Achse kann nicht vom Anwenderprogramm gesteuert werden. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.DriveReady</b> |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Der Antrieb ist bereit.                |           |         |         |     |
| Datentyp                                      | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <b>&lt;Achsname&gt;.StatusBits.RestartRequired</b>  |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Werte wurden im Ladespeicher geändert.   |           |         |         |     |
| Um die Werte im Betriebszustand RUN der CPU in den Arbeitsspeicher zu laden, ist ein Restart der Achse notwendig. Verwenden Sie hierzu die Motion Control-Anweisung MC_Reset. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

**Siehe auch**

Status- und Fehlerbits (Seite 4214)

## Variable ErrorBits.

### Legende

|                  |  |
|------------------|--|
| <b>Datentyp</b>  | Datentyp der Variable  |
| <b>Startwert</b> | Startwert der Variable   |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:                            |
|                  | RW Die Variable kann im Anwenderprogramm gelesen und geschrieben werden. |
|                  | R Die Variable kann im Anwenderprogramm gelesen werden.                  |
| -                | Die Variable kann im Anwenderprogramm nicht verwendet werden.            |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.                    |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden.                  |

|   |                  |                |                |            |
|---|------------------|----------------|----------------|------------|
| <b>&lt;Achsname&gt;.ErrorBits.SystemFault</b> |                  |                |                |            |
| TRUE = Interner Systemfehler.                 |                  |                |                |            |
| <b>Datentyp</b>                               | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> | <b>HMI</b> |
| Bool  | FALSE            | R              | -              | X          |

|   |                  |                |                |            |
|---|------------------|----------------|----------------|------------|
| <b>&lt;Achsname&gt;.ErrorBits.ConfigFault</b> |                  |                |                |            |
| TRUE = Fehlerhafte Konfiguration der Achse.   |                  |                |                |            |
| <b>Datentyp</b>                               | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> | <b>HMI</b> |
| Bool  | FALSE            | R              | -              | X          |

|  |                  |                |                |            |
|--|------------------|----------------|----------------|------------|
| <b>&lt;Achsname&gt;.ErrorBits.DriveFault</b>   |                  |                |                |            |
| TRUE = Antrieb hat durch Ausfall des "Antrieb bereit"-Signals einen Fehler angezeigt |                  |                |                |            |
| <b>Datentyp</b>  | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> | <b>HMI</b> |
| Bool   | FALSE            | R              | -              | X          |

|  |                  |                |                |            |
|--|------------------|----------------|----------------|------------|
| <b>&lt;Achsname&gt;.ErrorBits.SwLimitMinReached</b>    |                  |                |                |            |
| TRUE = Der untere Software-Endschalters wurde erreicht |                  |                |                |            |
| <b>Datentyp</b>  | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> | <b>HMI</b> |
| Bool   | FALSE            | R              | -              | X          |

|   |                  |                |                |            |
|---|------------------|----------------|----------------|------------|
| <b>&lt;Achsname&gt;.ErrorBits.SwLimitMinExceeded</b>        |                  |                |                |            |
| TRUE = Der untere Software-Endschalters wurde überschritten |                  |                |                |            |
| <b>Datentyp</b>   | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> | <b>HMI</b> |
| Bool  | FALSE            | R              | -              | X          |

|   |                  |                |                |            |
|---|------------------|----------------|----------------|------------|
| <b>&lt;Achsname&gt;.ErrorBits.SwLimitMaxReached</b>   |                  |                |                |            |
| TRUE = Der obere Software-Endschalters wurde erreicht |                  |                |                |            |
| <b>Datentyp</b>                                       | <b>Startwert</b> | <b>Zugriff</b> | <b>wirksam</b> | <b>HMI</b> |
| Bool  | FALSE            | R              | -              | X          |

| <Achsname>.ErrorBits.SwLimitMaxExceeded                    |           |         |         |     |
|--|-----------|---------|---------|-----|
| TRUE = Der obere Software-Endschalters wurde überschritten |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Bool   | FALSE     | R       | -       | X   |

| <Achsname>.ErrorBits.HwLimitMin                           |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Der untere Hardware -Endschalters wurde angefahren |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

| <Achsname>.ErrorBits.HwLimitMax                          |           |         |         |     |
|--|-----------|---------|---------|-----|
| TRUE = Der obere Hardware -Endschalters wurde angefahren |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Bool   | FALSE     | R       | -       | X   |

| <Achsname>.ErrorBits.HwUsed   |           |         |         |     |
|---|-----------|---------|---------|-----|
| TRUE = Eine weitere Achse verwendet den selben PTO (Pulse Train Output) und ist mit "MC_Power" freigegeben. |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Bool  | FALSE     | R       | -       | X   |

### Siehe auch

Status- und Fehlerbits (Seite 4214)

### Variable Internal.

Die Variablen "Internal" beinhalten keine anwenderrelevanten Daten; im Anwenderprogramm kann auf diese Variablen nicht zugegriffen werden.

### Variable ControlPanel

Die Variablen "ControlPanel" beinhalten keine anwenderrelevanten Daten; im Anwenderprogramm kann auf diese Variablen nicht zugegriffen werden.

### Aktualisierung der Variablen des Technologieobjekts

Die in den Variablen des Technologieobjekts angezeigten Status- und Fehlerinformationen der Achse werden jeweils am Zykluskontrollpunkt aktualisiert.

Die Änderung von Werten änderbarer Konfigurations-Variablen wird nicht sofort wirksam. Unter welchem Bedingungen eine Änderung wirksam wird, entnehmen Sie der Detailbeschreibung der entsprechenden Variable.

### 11.2.12.9 Variable des Technologieobjekts Auftragstabelle

#### Variable Config.Command.Command[1 ... 32]

##### Legende

|                  |  |
|------------------|--|
| <b>Datentyp</b>  | Datentyp der Variable  |
| <b>Startwert</b> | Startwert der Variable<br>Der Startwert kann durch die Konfiguration der Auftragstabelle überschrieben werden. |
| <b>Zugriff</b>   | Zugriff auf die Variable im Anwenderprogramm:  |
|                  | RW Die Variable kann im Anwenderprogramm gelesen und geschrieben werden.                                       |
|                  | R Die Variable kann im Anwenderprogramm gelesen werden.  |
|                  | - Die Variable kann im Anwenderprogramm nicht verwendet werden.  |
| <b>wirksam</b>   | Angabe, wann eine Änderung der Variable wirksam wird.  |
| <b>HMI</b>       | Die Variable kann in einem HMI-System verwendet werden.  |

| <b>&lt;Achsenname&gt;.Config.Command.Command[x].Type</b>   |           |         |         |     |
|--|-----------|---------|---------|-----|
| Auftragstyp des Auftrags   |           |         |         |     |
| <ul style="list-style-type: none"> <li>• 0 = Auftrag "Empty"</li> <li>• 2 = Auftrag "Halt"</li> <li>• 5 = Auftrag "Positioning Relative"</li> <li>• 6 = Auftrag "Positioning Absolute"</li> <li>• 7 = Auftrag "Velocity setpoint"</li> <li>• 151 = Auftrag "Wait"</li> </ul> |           |         |         |     |
| Datentyp   | Startwert | Zugriff | wirksam | HMI |
| Int  | 0         | RW      | -       | X   |

| <b>&lt;Achsenname&gt;. Config.Command.Command[x].Position</b> |           |         |         |     |
|---|-----------|---------|---------|-----|
| Zielposition / Verfahrenweg des Auftrags                      |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Real  |           | RW      | -       | X   |

| <b>&lt;Achsenname&gt;. Config.Command.Command[x].Velocity</b> |           |         |         |     |
|---|-----------|---------|---------|-----|
| Geschwindigkeit des Auftrags                                  |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Real  | 0.0       | RW      | -       | X   |

| <Achse>. Config.Command.Command[x].Duration |           |         |         |     |
|---|-----------|---------|---------|-----|
| Dauer des Auftrags                          |           |         |         |     |
| Datentyp                                    | Startwert | Zugriff | wirksam | HMI |
| Real  | 0.0       | RW      | -       | X   |

| <Achse>. Config.Command.Command[x].BufferMode   |           |         |         |     |
|---|-----------|---------|---------|-----|
| Wert für "Nächster Schritt" des Auftrags  |           |         |         |     |
| <ul style="list-style-type: none"> <li>• 0 = "Auftrag abschließen"</li> <li>• 1 = "Bewegung überschleifen"</li> </ul> |           |         |         |     |
| Datentyp  | Startwert | Zugriff | wirksam | HMI |
| Int   | 0         | RW      | -       | X   |

| <Achse>. Config.Command.Command[x].StepCode |           |         |         |     |
|---|-----------|---------|---------|-----|
| Schrittcode des Auftrags                    |           |         |         |     |
| Datentyp                                    | Startwert | Zugriff | wirksam | HMI |
| Word  | 0         | RW      | -       | X   |

### 11.2.12.1 Dokumentation von Funktionen aus älteren Versionen

0

#### Konfiguration - Referenzieren (Technologieobjekt "Achse" V1.0)

Konfigurieren Sie im Konfigurationsfenster "Referenzieren" die Parameter für das aktive und das passive Referenzieren. Die Referenzierart wird über den Eingangsparameter "Mode" der Motion Control-Anweisung eingestellt. Hierbei entspricht Mode = 2 dem passiven und Mode = 3 dem aktiven Referenzieren.



## Eingang Referenzpunktschalter

Wählen Sie in der Klappliste den digitalen Eingang für den Referenzpunktschalter aus. Der Eingang muss alarmfähig sein. Als Eingänge für den Referenzpunktschalter stehen die On-board CPU-Eingänge und Eingänge eines gesteckten Signalboards zur Auswahl.

---

### Hinweis

Die Digitaleingänge sind standardmäßig auf eine Filterzeit von 6,4 ms eingestellt.

Beim Einsatz als Referenzpunktschalter kann dies zu unerwünschten Verzögerungen und damit zu Ungenauigkeiten führen. Je nach Referenziergeschwindigkeit und Ausmaß des Referenzpunktschalters kann u. U. der Referenzpunkt nicht erkannt werden. Die Filterzeit kann in der Gerätekonfiguration der Digitaleingänge unter "Eingangsfiler" eingestellt werden.

Die Filterzeit muss kleiner als die Dauer des Eingangssignal am Referenzpunktschalter gewählt werden.

---

## Erlaube Richtungsumkehr am HW-Endschalter (nur aktives Referenzieren)

Aktivieren Sie das Optionskästchen, wenn Sie die Hardware-Endschalter als Umkehrnocken für die Referenzpunktfahrt nutzen wollen. Für die Richtungsumkehr müssen die Hardware-Endschalter aktiviert sein. Wird die CPU-Firmware V1.0 eingesetzt, so müssen beide Hardware-Endschalter konfiguriert sein. Beim Einsatz der CPU-Firmware ab V2.0 muss nur der Hardware-Endschalter in Anfahrriechtung konfiguriert sein.

Wird der Hardware Endschalter während des aktiven Referenzierens erreicht, bremst die Achse mit der konfigurierten Verzögerung (nicht mit der Notstopp-Verzögerung) ab und führt eine Richtungsumkehr durch. Der Referenzpunktschalter wird anschließend in umgekehrter Richtung gesucht.

Ist die Richtungsumkehr nicht aktiv und erreicht die Achse während des aktiven Referenzierens den Hardware-Endschalter, dann wird die Referenzpunktfahrt mit Fehler abgebrochen und die Achse mit der Notstopp-Verzögerung abgebremst.

---

### Hinweis

Stellen Sie durch eine der folgenden Maßnahmen sicher, dass die Maschine bei einer Richtungsumkehr nicht auf einen mechanischen Anschlag fährt:

- Halten Sie die Anfahrgewindigkeit gering
  - Vergrößern Sie die konfigurierte Beschleunigung / Verzögerung
  - Vergrößern Sie den Abstand zwischen Hardware-Endschalter und Hardwareanschlag
- 

## Anfahrriechtung- / Referenzierrichtung (aktives und passives Referenzieren)

Mit der gewählten Richtung bestimmen Sie beim aktiven Referenzieren die "Anfahrriechtung" zur Suche des Referenzpunktschalters, als auch die Referenzierrichtung. Die Referenzierrichtung legt fest mit welcher Fahrriechtung die Achse die konfigurierte Seite des Referenzpunktschalters anfährt, um den Referenziervorgang durchzuführen.

Die Wirkung der eingestellten Anfahrriechtung für das passive Referenzieren entnehmen Sie der Tabelle unter "Referenzpunktschalter".

### Seite des Referenzpunktschalters (aktives und passives Referenzieren)

- **Aktives Referenzieren**  
Wählen Sie hier ob die Achse an der oberen oder der unteren Seite des Referenzpunktschalters referenziert werden soll.

---

**Hinweis**

Je nach Anfangsposition der Achse und Konfiguration der Referenzierparameter kann der Ablauf der Referenzpunktfahrt von der Grafik im Konfigurationsfenster abweichen.

---

- **Passives Referenzieren**  
Beim passiven Referenzieren müssen die Fahrbewegungen zum Referenzieren anwenderseitig durch Fahraufträge realisiert werden. An welcher Seite des Referenzpunktschalters referenziert wird, hängt von folgenden Faktoren ab:
  - Konfiguration "Anfahrriichtung"
  - Konfiguration "Referenzpunktschalter"
  - Aktuelle Fahrriichtung während des passiven Referenzierens

Details zur Wirkung der Faktoren entnehmen Sie der nachfolgenden Tabelle:

| Beeinflussende Faktoren:        |                                       |                        | Ergebnis:                              |
|---------------------------------|---------------------------------------|------------------------|--|
| Konfiguration - Anfahrriichtung | Konfiguration - Referenzpunktschalter | Aktuelle Fahrriichtung | Referenzieren am Referenzpunktschalter |
| Positiv                         | "Untere Seite"                        | Positive Richtung      | <b>Obere Seite</b>                     |
|                                 |                                       | Negative Richtung      | <b>Untere Seite</b>                    |
| Positiv                         | "Obere Seite"                         | Positive Richtung      | <b>Untere Seite</b>                    |
|                                 |                                       | Negative Richtung      | <b>Obere Seite</b>                     |
| Negativ                         | "Untere Seite"                        | Positive Richtung      | <b>Untere Seite</b>                    |
|                                 |                                       | Negative Richtung      | <b>Obere Seite</b>                     |
| Negativ                         | "Obere Seite"                         | Positive Richtung      | <b>Obere Seite</b>                     |
|                                 |                                       | Negative Richtung      | <b>Untere Seite</b>                    |

### Geschwindigkeit (nur aktives Referenzieren)

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher der Referenzpunktschalter während der Referenzpunktfahrt gesucht werden soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stop-Geschwindigkeit ≤ Anfahrgewindigkeit ≤ Maximale Geschwindigkeit

### Referenziertgeschwindigkeit (nur aktives Referenzieren)

Legen Sie in diesem Feld die Geschwindigkeit fest, mit welcher die Achse in den Referenzpunktschalter zum Referenzieren einfahren soll.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- Start/Stopp-Geschwindigkeit ≤ Referenziergeschwindigkeit ≤ Maximale Geschwindigkeit

### Referenzpunktverschiebung (nur aktives Referenzieren)

Weicht die gewünschte Referenzposition von der Position des Referenzpunktschalters ab, so kann in diesem Feld die Referenzpunktverschiebung angegeben werden.

Ist der Wert ungleich 0, so führt die Achse nach dem Referenzieren am Referenzpunktschalter folgende Aktionen aus:

1. Verfahren der Achse mit Referenziergeschwindigkeit um den Wert der Referenzpunktverschiebung
2. Nach dem Herausfahren der Referenzpunktverschiebung befindet sich die Achse auf der Referenzpunktposition, welche am Eingangsparameter "Position" der Motion Control-Anweisung "MC\_Home" angegeben wurde.

Grenzwerte (unabhängig von der gewählten Maßeinheit):

- $-1.0e12 \leq \text{Referenzpunktverschiebung} \leq 1.0e12$

### Referenzpunktposition

Als Referenzpunktposition wird die an der Motion Control-Anweisung "MC\_Home" parametrisierte Position verwendet.

### Liste der ErrorIDs und ErrorInfos (Technologieobjekte V1.0)

In den folgenden Tabellen finden Sie eine Auflistung aller ErrorIDs und ErrorInfos, die an den Motion Control-Anweisungen angezeigt werden können. Neben der Fehlerursache werden auch Abhilfen zur Beseitigung der Fehler aufgelistet:

#### Betriebsfehler mit Stopp der Achse

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe  |
|----------------|-----------|--|--|
| <b>16#8000</b> |           | <b>Antriebsfehler, "Antrieb bereit" ausgefallen</b>  |  |
|                | 16#0001   | -  | Fehler mit der Anweisung "MC_Reset" quittieren; Antriebssignal bereitstellen; Auftrag evt. erneut starten                        |
| <b>16#8001</b> |           | <b>Unterer SW-Endschalter wurde ausgelöst</b>  |  |
|                | 16#000E   | Die Position des unteren SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht | Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in positiver Richtung aus dem SW-Endschalterbereich herausfahren |
|                | 16#000F   | Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht               |  |
|                | 16#0010   | Die Position des unteren SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten          |  |
| <b>16#8002</b> |           | <b>Oberer SW-Endschalter wurde ausgelöst</b>   |  |

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe   |
|----------------|-----------|---|---|
|                | 16#000E   | Die Position des oberen SW-Endschalters wurde mit der aktuell konfigurierten Verzögerung erreicht   | Fehler mit der Anweisung "MC_Reset" quittieren; mit Fahrauftrag in negativer Richtung aus dem SW-Endschalterbereich herausfahren  |
|                | 16#000F   | Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung erreicht   |   |
|                | 16#0010   | Die Position des oberen SW-Endschalters wurde mit der Notstopp-Verzögerung überschritten  |   |
| <b>16#8003</b> |           | <b>Unterer HW-Endschalter wurde angefahren</b>  |   |
|                | 16#000E   | Der untere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten.<br>(Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden) | Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrtauftrag in positiver Richtung aus dem HW-Endschalterbereich herausfahren.   |
| <b>16#8004</b> |           | <b>Oberer HW-Endschalter wurde angefahren</b>   |   |
|                | 16#000E   | Der obere HW-Endschalter wurde angefahren. Die Achse wurde mit der Notstopp-Verzögerung angehalten.<br>(Bei einer aktiven Referenzpunktfahrt wurde der Referenzpunktschalter nicht gefunden)  | Fehler bei freigegebener Achse mit der Anweisung "MC_Reset" quittieren; mit Verfahrtauftrag in negativer Richtung aus dem HW-Endschalterbereich herausfahren.   |
| <b>16#8005</b> |           | <b>PTO / HSC werden bereits von einer anderen Achse verwendet</b>   |   |
|                | 16#0001   | -   | <b>Die Achse wurde fehlerhaft konfiguriert:</b><br>Konfiguration des PTO (Pulse Train Output) / HSC (High Speed Counter) korrigieren und in Steuerung laden<br><br><b>Mehrere Achsen sollen mit einem PTO arbeiten:</b><br>Eine andere Achse verwendet den PTO / HSC. Soll die aktuelle Achse die Steuerung übernehmen, so muss die andere Achse mit "MC_Power" Enable = FALSE gesperrt werden.<br>(siehe auch Mehrere Achsen mit gleichem PTO einsetzen (Seite 4219) ) |

### Betriebsfehler ohne Stopp der Achse

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe  |
|----------------|-----------|--|--|
| <b>16#8200</b> |           | <b>Achse ist nicht freigegeben</b>   |  |
|                | 16#0001   | -  | Achse freigeben; Auftrag erneut starten            |
| <b>16#8201</b> |           | <b>Achse ist bereits durch eine andere "MC_Power"-Instanz freigegeben</b>  |  |
|                | 16#0001   | -  | Achse nur über eine Anweisung "MC_Power" freigeben |
| <b>16#8202</b> |           | <b>Maximale Anzahl gleichzeitiger aktiver Motion Control-Aufträge überschritten (max. 200 Aufträge für alle Motion Control-Technologieobjekte)</b> |  |

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe   |
|----------------|-----------|--|---|
|                | 16#0001   | -  | Anzahl der gleichzeitig aktiven Aufträge verringern;<br>Auftrag erneut starten<br><br>Ein aktiver Auftrag ist am Parameter<br>"Busy" = TRUE der Motion Control-Anweisung zu<br>erkennen.                                    |
| <b>16#8203</b> |           | <b>Achse wird momentan in "Handsteuerung" (Achssteuertafel) betrieben</b>  |   |
|                | 16#0001   | -  | "Handsteuerung" beenden; Auftrag erneut starten   |
| <b>16#8204</b> |           | <b>Achse ist nicht referenziert</b>  |   |
|                | 16#0001   | -  | Achse mit der Anweisung "MC_Home"<br>referenzieren; Auftrag erneut starten  |
| <b>16#8205</b> |           | <b>Achse wird momentan durch das Anwenderprogramm gesteuert (Fehler wird nur in der<br/>Achssteuertafel angezeigt)</b> |   |
|                | 16#0001   | -  | Achse mit der Anweisung "MC_Power" sperren und<br>in der Achssteuertafel erneut "Handsteuerung"<br>anwählen   |
| <b>16#8206</b> |           | <b>Technologieobjekt Achse ist noch nicht aktiviert</b>  |   |
|                | 16#0001   | -  | Aktivieren Sie die Achse mit der Anweisung<br>"MC_Power" Enable = TRUE, oder geben Sie die<br>Achse in der Achssteuertafel frei.  |
| <b>16#8207</b> |           | <b>Auftrag abgewiesen</b>  |   |
|                | 16#0016   | Aktives Referenzieren läuft; eine andere<br>Referenzierart kann nicht gestartet werden.                                | Warten Sie den Abschluss des Aktiven<br>Referenzierens ab oder brechen Sie das Aktive<br>Referenzieren durch einen Bewegungsauftrag z. B.<br>"MC_Halt" ab. Anschließend kann die andere<br>Referenzierart gestartet werden. |

### Bausteinparameterfehler

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe   |
|----------------|-----------|---|---|
| <b>16#8400</b> |           | <b>Wert am Parameter "Position" der Motion Control-Anweisung ist ungültig</b> |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | Wert "Position" korrigieren; Auftrag erneut starten |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs<br>(größer als $1e^{12}$ )            |   |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs<br>(kleiner als $-1e^{12}$ )          |   |
| <b>16#8401</b> |           | <b>Wert am Parameter "Distance" der Motion Control-Anweisung ist ungültig</b> |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | Wert "Distance" korrigieren; Auftrag erneut starten |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs<br>(größer als $1e^{12}$ )            |   |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs<br>(kleiner als $-1e^{12}$ )          |   |
| <b>16#8402</b> |           | <b>Wert am Parameter "Velocity" der Motion Control-Anweisung ist ungültig</b> |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | Wert "Velocity" korrigieren; Auftrag erneut starten |
|                | 16#0008   | Geschwindigkeit ist größer als die maximale<br>Geschwindigkeit                |   |

| ErrorID        | ErrorInfo | Beschreibung  | Abhilfe  |
|----------------|-----------|---|--|
|                | 16#0009   | Geschwindigkeit ist kleiner als die Start/Stopp-Geschwindigkeit                                       |  |
| <b>16#8403</b> |           | <b>Wert am Parameter "Direction" der Motion Control-Anweisung ist ungültig</b>                        |  |
|                | 16#0011   | Auswahlwert ungültig  | Auswahlwert korrigieren; Auftrag erneut starten  |
| <b>16#8404</b> |           | <b>Wert am Parameter "Mode" der Motion Control-Anweisung ist ungültig</b>                             |  |
|                | 16#0011   | Auswahlwert ungültig  | Auswahlwert korrigieren; Auftrag erneut starten  |
|                | 16#0015   | Aktives / Passives Referenzieren ist nicht konfiguriert   | Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten   |
|                | 16#0017   | Richtungsumkehr am HW-Endschalter ist aktiviert, obwohl die HW-Endschalter deaktiviert sind           | <ul style="list-style-type: none"> <li>Aktivieren der HW-Endschalter über die Variable &lt;Achse&gt;.Config.PositionLimits_HW.Active = TRUE, Auftrag erneut starten</li> <li>Konfiguration korrigieren und in Steuerung laden; Achse freigeben und Auftrag erneut starten</li> </ul> |
| <b>16#8405</b> |           | <b>Wert am Parameter "StopMode" der Motion Control-Anweisung ist ungültig</b>                         |  |
|                | 16#0011   | Auswahlwert ungültig  | Auswahlwert korrigieren; Achse erneut freigeben  |
| <b>16#8406</b> |           | <b>Gleichzeitiges Tippen vorwärts und rückwärts nicht erlaubt</b>                                     |  |
|                | 16#0001   | -   | Verhindern Sie den gleichzeitigen Signalzustand TRUE der Parameter "JogForward" und "JogBackward"; starten Sie den Auftrag erneut.   |
| <b>16#8407</b> |           | <b>Wechsel der Achse an der Motion Control-Anweisung "MC_Power" nur bei gesperrter Achse zulässig</b> |  |
|                | 16#0001   | -   | Aktive Achse sperren; anschließend kann die Achse gewechselt und freigegeben werden.   |

## Konfigurationsfehler

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe   |
|----------------|-----------|--|---|
| <b>16#8600</b> |           | <b>Parametrierung des Impulsgenerators (PTO) ist ungültig</b>  |   |
|                | 16#000B   | Adresse ist ungültig   | Konfiguration des PTO (Pulse Train Output) korrigieren und in Steuerung laden |
| <b>16#8601</b> |           | <b>Parametrierung des schnellen Zählers (HSC) ist ungültig</b> |   |
|                | 16#000B   | Adresse ist ungültig   | Konfiguration des HSC (High Speed Counter) korrigieren und in Steuerung laden |
| <b>16#8602</b> |           | <b>Parametrierung des "Freigabe-Ausgangs" ist ungültig</b>     |   |
|                | 16#000D   | Adresse ist ungültig   | Konfiguration korrigieren und in Steuerung laden                              |
| <b>16#8603</b> |           | <b>Parametrierung des "Bereit-Eingangs" ist ungültig</b>       |   |
|                | 16#000D   | Adresse ist ungültig   | Konfiguration korrigieren und in Steuerung laden                              |
| <b>16#8604</b> |           | <b>Wert für "Impulse pro Motorumdrehung" ist ungültig</b>      |   |
|                | 16#000A   | Wert ist kleiner oder gleich Null                              | Konfiguration korrigieren und in Steuerung laden                              |
| <b>16#8605</b> |           | <b>Wert für "Weg pro Motorumdrehung" ist ungültig</b>          |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat                            | Konfiguration korrigieren und in Steuerung laden                              |
|                | 16#000A   | Wert ist kleiner oder gleich Null                              |   |
| <b>16#8606</b> |           | <b>Wert für "Start/Stopp-Geschwindigkeit" ist ungültig</b>     |   |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat                            | Konfiguration korrigieren und in Steuerung laden                              |

| ErrorID        | ErrorInfo   | Beschreibung  | Abhilfe  |
|----------------|---|---|--|
|                | 16#0003   | Wert ist größer als die Hardwaregrenze  |  |
|                | 16#0004   | Wert ist kleiner als die Hardwaregrenze   |  |
|                | 16#0007   | Start/Stopp-Geschwindigkeit ist größer als die maximale Geschwindigkeit                 |  |
| <b>16#8607</b> | <b>Wert für "Maximale Geschwindigkeit" ist ungültig</b>               |   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | Konfiguration korrigieren und in Steuerung laden   |
|                | 16#0003   | Wert ist größer als die Hardwaregrenze  |  |
|                | 16#0004   | Wert ist kleiner als die Hardwaregrenze   |  |
| <b>16#8608</b> | <b>Wert für "Beschleunigung" ist ungültig</b>                         |   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0003   | Wert ist größer als die Hardwaregrenze  |  |
|                | 16#0004   | Wert ist kleiner als die Hardwaregrenze   |  |
| <b>16#8609</b> | <b>Wert für "Verzögerung" ist ungültig</b>                            |   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0003   | Wert ist größer als die Hardwaregrenze  |  |
|                | 16#0004   | Wert ist kleiner als die Hardwaregrenze   |  |
| <b>16#860A</b> | <b>Wert für "Notstopp-Verzögerung" ist ungültig</b>                   |   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0003   | Wert ist größer als die Hardwaregrenze  |  |
|                | 16#0004   | Wert ist kleiner als die Hardwaregrenze   |  |
| <b>16#860B</b> | <b>Wert für die Position des unteren SW-Endschalters ist ungültig</b> |   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )                         |  |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )                       |  |
|                | 16#0007   | Positionswert des unteren SW-Endschalters ist größer als der des oberen SW-Endschalters |  |
| <b>16#860C</b> | <b>Wert für die Position des oberen SW-Endschalters ist ungültig</b>  |   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat   | <ul style="list-style-type: none"> <li>• Fehlerfreie Konfiguration in Steuerung laden; Achse mit der Anweisung "MC_Power" erneut freigeben</li> <li>• Fehlerhaften Wert online korrigieren, Fehler mit der Anweisung "MC_Reset" quittieren und ggf. Auftrag neu starten</li> </ul> |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )                         |  |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )                       |  |
| <b>16#860D</b> | <b>Adresse des unteren HW-Endschalters ist ungültig</b>               |   |  |
|                | 16#000C   | Adresse der fallenden Flanke ist ungültig   | Konfiguration korrigieren und in Steuerung laden   |
|                | 16#000D   | Adresse der steigenden Flanke ist ungültig  |  |

| ErrorID        | ErrorInfo | Beschreibung   | Abhilfe  |
|----------------|-----------|--|--|
| <b>16#860E</b> |           | <b>Adresse des oberen HW-Endschalter ist ungültig</b>  |  |
|                | 16#000C   | Adresse der fallenden Flanke ist ungültig  | Konfiguration korrigieren und in Steuerung laden |
|                | 16#000D   | Adresse der steigenden Flanke ist ungültig   |  |
| <b>16#860F</b> |           | <b>Wert für "Referenzpunktverschiebung" ist ungültig</b>   |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | Konfiguration korrigieren und in Steuerung laden |
|                | 16#0005   | Wert liegt außerhalb des Zahlenbereichs (größer als $1e^{12}$ )  |  |
|                | 16#0006   | Wert liegt außerhalb des Zahlenbereichs (kleiner als $-1e^{12}$ )  |  |
| <b>16#8610</b> |           | <b>Wert für "Anfahrsgeschwindigkeit" ist ungültig</b>  |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | Konfiguration korrigieren und in Steuerung laden |
|                | 16#0008   | Geschwindigkeit ist größer als die maximale Geschwindigkeit  |  |
|                | 16#0009   | Geschwindigkeit ist kleiner als die Start/Stop-Geschwindigkeit   |  |
| <b>16#8611</b> |           | <b>Wert für "Referenziergeschwindigkeit" ist ungültig</b>  |  |
|                | 16#0002   | Wert hat kein gültiges Zahlenformat  | Konfiguration korrigieren und in Steuerung laden |
|                | 16#0008   | Geschwindigkeit ist größer als die maximale Geschwindigkeit  |  |
|                | 16#0009   | Geschwindigkeit ist kleiner als die Start/Stop-Geschwindigkeit   |  |
| <b>16#8612</b> |           | <b>Adresse des Referenzpunktschalters ist ungültig</b>   |  |
|                | 16#000C   | Adresse der fallenden Flanke ist ungültig  | Konfiguration korrigieren und in Steuerung laden |
|                | 16#000D   | Adresse der steigenden Flanke ist ungültig   |  |
| <b>16#8613</b> |           | <b>Beim aktiven Referenzieren ist die Richtungsumkehr am HW-Endschalter aktiviert, obwohl die HW-Endschalter nicht konfiguriert sind</b> |  |
|                | 16#0001   | -  | Konfiguration korrigieren und in Steuerung laden |

## Interne Fehler

| ErrorID        | ErrorInfo | Beschreibung           | Abhilfe  |
|----------------|-----------|------------------------|--|
| <b>16#8FFF</b> |           | <b>Interner Fehler</b> |  |
|                | 16#F0**   | -                      | <p>NETZ-AUS und NETZ-EIN der CPU</p> <p>Sollte dies nicht zum Erfolg führen, kontaktieren Sie den Customer Support. Halten Sie folgende Information bereit:</p> <ul style="list-style-type: none"> <li>• ErrorID</li> <li>• ErrorInfo</li> <li>• Diagnosepuffereinträge</li> </ul> |

## Siehe auch

Mehrere Achsen mit gleichem PTO einsetzen (Seite 4219)



## Online- und Diagnosefunktionen nutzen

### 12.1 Erreichbare Teilnehmer anzeigen

#### Erreichbare Teilnehmer

Erreichbare Teilnehmer sind alle Geräte, die mit einer Schnittstelle des PG/PC verbunden sind und sich in angeschaltetem Zustand befinden. Insofern können auch Geräte angezeigt werden, die Sie mit den aktuell installierten Produkten nur eingeschränkt oder gar nicht projektieren können.

#### Erreichbare Teilnehmer an einer Schnittstelle des PG/PC in der Projektnavigation anzeigen

Um die erreichbaren Teilnehmer an einer einzelnen Schnittstelle des PG/PC anzuzeigen, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "Online-Zugänge" in der Projektnavigation.
2. Klicken Sie auf das Pfeilsymbol links neben der Schnittstelle, um alle Objekte sichtbar zu machen, die unterhalb der Schnittstelle angeordnet sind.
3. Doppelklicken Sie auf den Befehl "Erreichbare Teilnehmer aktualisieren" unterhalb der Schnittstelle.

Alle Teilnehmer, die über diese Schnittstelle erreichbar sind, werden in der Projektnavigation angezeigt. Bei einer großen Anzahl verbundener Geräte, kann der Aktualisierungsvorgang etwas Zeit in Anspruch nehmen. Sie sehen den Fortschritt des Aktualisierungsvorgangs in der Statusleiste. Sollten Sie das gewünschte Gerät vor Abschluss der Aktualisierung gefunden haben, können Sie die Aktualisierung der erreichbaren Teilnehmer abbrechen. Klicken Sie dazu auf das Kreuz rechts neben der Fortschrittsanzeige.

#### Erreichbare Teilnehmer in einer Liste anzeigen

Um die erreichbaren Teilnehmer an allen verfügbaren Schnittstellen in einer übersichtlichen Liste anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online" den Befehl "Erreichbare Teilnehmer".  
Der Dialog "Erreichbare Teilnehmer" wird geöffnet.
2. Wählen Sie aus der Klappliste "Typ der PG/PC-Schnittstelle" die Art der Schnittstelle.  
Abhängig von der Auswahl, werden in der Klappliste "PG/PC-Schnittstelle" nur noch die Schnittstellen des PG/PC angezeigt, die dem gewählten Schnittstellen-Typ entsprechen.
3. Wählen Sie die gewünschte Schnittstelle des PG/PC aus der Klappliste "PG/PC-Schnittstelle", z. B. eine Industrial Ethernet Karte.  
Sind an einer Schnittstelle keine Geräte erreichbar, wird die Verbindungslinie zwischen PG/PC und dem Gerät unterbrochen dargestellt. Sind Geräte erreichbar, wird die Verbindungslinie geschlossen dargestellt und die erreichbaren Teilnehmer an der gewählten Schnittstelle des PG/PC werden in einer Liste angezeigt.

4. Wenn Sie zwischenzeitlich ein neues Gerät angeschlossen haben, klicken Sie auf die Schaltfläche "Aktualisieren", um die Liste der erreichbaren Teilnehmer neu aufzubauen.
5. Um zu einem Gerät in der Projektnavigation zu springen, wählen Sie das Gerät aus der Liste der erreichbaren Teilnehmer und klicken auf die Schaltfläche "Anzeigen". Die Schnittstelle, an der das selektierte Gerät angeschlossen ist, wird in der Projektnavigation markiert.

### Zusätzliche Informationen zu den erreichbaren Teilnehmern in der Projektnavigation anzeigen

Um sich zusätzliche Informationen zu den einzelnen erreichbaren Teilnehmern in der Projektnavigation anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Pfeilsymbol links von einem der erreichbaren Teilnehmer in der Projektnavigation.  
Bei bekannten Geräten werden alle online verfügbaren Daten, wie z. B. Bausteine oder Systemdaten, angezeigt. Objekte, die Sie an dieser Stelle nicht direkt bearbeiten können, werden grau angezeigt. Wenn für ein Gerät weitere Bearbeitungsmöglichkeiten, wie z. B. das Laden über das Kontextmenü, zu Verfügung stehen, dann wird das Gerät in schwarzer Schrift angezeigt.

### Siehe auch

Gerätekonfiguration online ändern (Seite 4268)

## 12.2 Gerätekonfiguration online ändern

Manche Geräte, vorzugsweise in kleinen Hardware-Aufbauten, können Sie direkt online auf dem Gerät parametrieren. Sie müssen dazu kein Projekt anlegen und offline Daten vorhalten. So können Sie unkompliziert und schnell die Gerätekonfiguration ändern. Sie müssen die Hardware-Konfiguration nicht übersetzen und keinen Download durchführen. In Abhängigkeit vom Gerät werden entweder alle Änderungen sofort aktiv oder nach einer Bestätigung auf das Gerät geschrieben.

### Voraussetzung

- Das Gerät muss die Online-Parametrierung unterstützen. Ob Ihre jeweiligen Geräte diese Funktion unterstützen, erfahren Sie im Gerätehandbuch.
- Das Gerät muss mit dem PG/PC verbunden sein und in der Liste der erreichbaren Teilnehmer verfügbar sein.

### Vorgehen

Um die Gerätekonfiguration online zu ändern, gehen Sie folgendermaßen vor:

1. Lassen Sie sich die erreichbaren Teilnehmer an der Schnittstelle anzeigen, über die das Gerät verbunden ist. Wie Sie die erreichbaren Teilnehmer anzeigen, erfahren Sie im vorangehenden Kapitel "Erreichbare Teilnehmer anzeigen (Seite 4267)".
2. Klappen Sie das Gerät auf, um die unterlagerten Elemente anzuzeigen.

3. Doppelklicken Sie auf den Eintrag "Gerät parametrieren".  
Im Arbeitsbereich wird eine Konfigurationsseite für das Gerät geöffnet.
4. Nehmen Sie alle notwendigen Einstellungen vor.  
Bei manchen Geräten werden die geänderten Einstellungen sofort wirksam.
5. Optional in Abhängigkeit vom Gerät: Klicken Sie auf die Schaltfläche "Zum Gerät übertragen".  
Die Einstellungen werden auf das Gerät übertragen.

## 12.3 Geräte online verbinden

### 12.3.1 Allgemeine Information zum Online-Modus

#### Online-Modus

Im Online-Modus besteht eine Online-Verbindung zwischen Ihrem PG/PC und einem oder mehreren Geräten.

Eine Online-Verbindung zwischen PG/PC und Gerät ist beispielsweise für folgende Tätigkeiten erforderlich:

- Testen von Anwenderprogrammen
- Anzeigen und Ändern des Betriebszustands der CPU
- Anzeigen und Einstellen von Uhrzeit und Datum der CPU
- Anzeigen des Baugruppenzustands
- Bausteine vergleichen
- Hardware diagnostizieren

Um eine Online-Verbindung herstellen zu können, müssen PG/PC und Gerät physikalisch oder über eine Fernverbindung miteinander verbunden sein. Alternativ unterstützen einige Geräte einen Simulationsmodus. In diesem Fall wird über die virtuelle Schnittstelle PLCSIM eine Verbindung zum Gerät simuliert.

Nach dem Herstellen einer Verbindung können Sie in der Online- und Diagnosesicht oder über die Task Card "Online-Tools" auf die Daten des Geräts zugreifen. Außerdem wird der aktuelle Online-Status eines Geräts rechts neben dem Gerät in der Projektnavigation symbolisch angezeigt. Die Bedeutung der einzelnen Status-Symbole entnehmen Sie bitte dem jeweiligen Tooltip.

---

#### Hinweis

Einige Online-Funktionen sind vom Installationsumfang der Software abhängig oder davon, ob ein Projekt geöffnet ist.

---

### **Standby oder Ruhezustand des PG/PC**

Wird der PG/PC bei einer bestehenden Online-Verbindung in den Standby-Modus oder in den Ruhezustand versetzt, werden alle Online-Verbindungen getrennt. Nach dem Aufwachen des PG/PC aus dem Ruhemodus werden die Online-Verbindungen nicht automatisch wiederhergestellt.

Beachten Sie in diesem Zusammenhang, dass durch das plötzliche Trennen der Online-Verbindung Daten verloren gehen können oder ein angeschlossenes Gerät unterbricht den Programmablauf.

### **LED-Blinktest durchführen**

In vielen Online-Dialogen können Sie einen LED-Blinktest durchführen, wenn das online verbundene Gerät diese Funktion unterstützt. Aktivieren Sie das Optionskästchen "LED blinken", dann blinkt eine LED am gerade ausgewählten Teilnehmer. Diese Funktion ist beispielsweise hilfreich, wenn Sie sich nicht sicher sind, welches Gerät im Hardware-Aufbau dem gerade in der Software gewählten Teilnehmer entspricht.

Beachten Sie weitere Hinweise und mögliche Einschränkungen zum LED-Blinktest in der jeweiligen Gerätedokumentation.

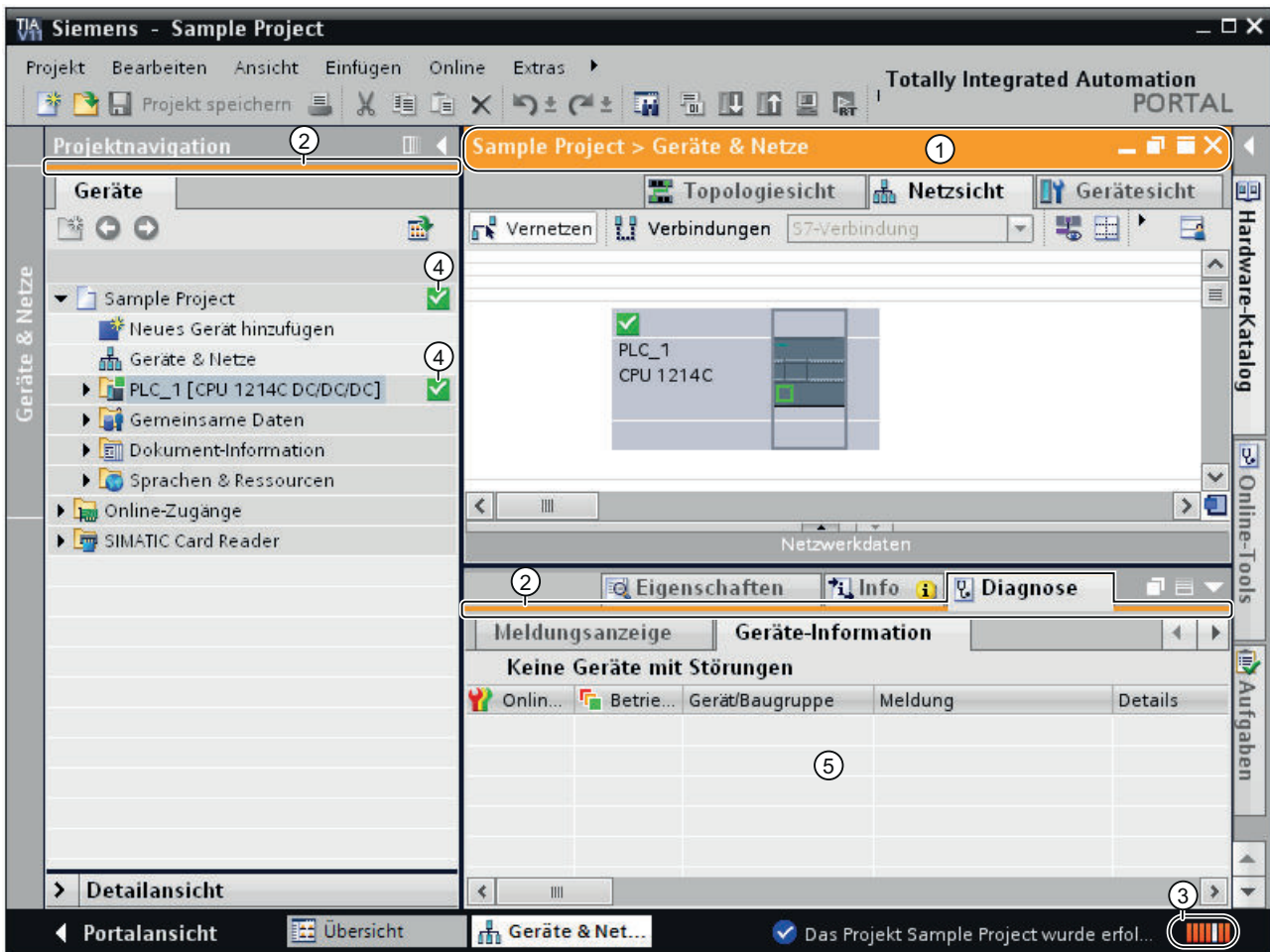
### **Siehe auch**

Ansicht im Online-Modus (Seite 4271)

### 12.3.2 Ansicht im Online-Modus

#### Online-Anzeigen

Nach erfolgreichem Aufbau der Online-Verbindung ändert sich die Bedienoberfläche. Falls ein Gerät nicht erreichbar ist, wird dies symbolisch angezeigt. Das folgende Bild zeigt ein online verbundenes Gerät und die dazugehörige Bedienoberfläche:



- ① Die Titelleiste des aktiven Fensters erhält einen orangefarbenen Hintergrund, sobald mindestens eines der Geräte, die aktuell im Editor angezeigt werden, erfolgreich online geschaltet wurde. Wenn ein oder mehrere Geräte nicht erreichbar sind, dann wird in der Titelleiste des Editors ein Symbol für eine unterbrochene Verbindung angezeigt.
- ② Die Titelleisten der inaktiven Fenster der zugehörigen Station erhalten am unteren Rand einen orangefarbenen Strich.
- ③ Am rechten Rand der Statuszeile erscheint ein orangefarbener, pulsierender Balken. Ist die Verbindung zwar zustande gekommen, aber fehlerhaft, wird statt des Balkens ein Symbol für eine unterbrochene Verbindung angezeigt. Im Inspektorfenster unter "Diagnose" erhalten Sie weitere Informationen zum Fehlerbild.

- ④ In der Projektnavigation werden bei den online verbundenen Stationen und ihren unterlagerten Objekten Betriebszustands- bzw. Diagnosesymbole angezeigt. Außerdem wird automatisch ein Vergleich zwischen dem Online- und dem Offline-Zustand durchgeführt. Unterschiede zwischen den Online- und Offline-Objekten werden ebenfalls symbolisch angezeigt.
- ⑤ Im Inspektorfenster wird der Bereich "Diagnose > Geräte-Information" in den Vordergrund gebracht.

### Abriss der Online-Verbindung

Der Online-Modus mit seiner zugehörigen Anzeige bleibt solange erhalten, wie noch mindestens ein Gerät online verbunden ist. Reißt bei einem oder mehreren Geräten die Online-Verbindung ab, bleibt das TIA-Portal im Online-Modus. Erst wenn zu keinem Gerät mehr eine Online-Verbindung besteht, wechselt die Darstellung des TIA-Portals in den Offline-Modus.

### Siehe auch

Allgemeine Information zum Online-Modus (Seite 4269)

Grundlagen zum Projektdatenvergleich (Seite 261)

## 12.3.3 Online verbinden und Online-Verbindung trennen

### Voraussetzung

Es ist mindestens eine PG/PC-Schnittstelle installiert, die physikalisch mit einem Gerät verbunden ist, z. B. mit einem Ethernet-Kabel. Alternativ ist das Aufbauen einer virtuellen Online-Verbindung über PLCSIM möglich.

### Online verbinden

Um eine Online-Verbindung herzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Projektnavigation ein oder mehrere Geräte, zu dem eine Online-Verbindung aufgebaut werden soll.
2. Wählen Sie im Menü "Online" den Befehl "Online verbinden".  
Wenn das Gerät bereits zuvor mit einer bestimmten PG/PC-Schnittstelle verbunden war, wird die Online-Verbindung automatisch mit der bisherigen PG/PC-Schnittstelle aufgebaut. In diesem Fall brauchen Sie die folgenden Schritte nicht zu beachten. Wenn vorher noch keine Verbindung bestanden hat, wird der Dialog "Online verbinden" geöffnet.
3. Wählen Sie aus der Klappliste "Typ der PG/PC-Schnittstelle" die Art der Schnittstelle. Abhängig von der Auswahl, werden in der Klappliste "PG/PC-Schnittstelle" nur noch die Schnittstellen des PG/PC angezeigt, die dem gewählten Schnittstellen-Typ entsprechen.
4. Wählen Sie die gewünschte Schnittstelle des PG/PC aus der Klappliste "PG/PC-Schnittstelle", z. B. eine Industrial Ethernet Karte.

5. Wählen Sie aus der Klappliste "Verbindung mit Subnetz" die Schnittstelle aus, über die das Gerät mit dem PG/PC verbunden ist". In diesem Fall wird zum Gerät eine direkte Verbindung aufgebaut, ohne dass ein Netzknoten, wie z. B. ein Switch, zwischengeschaltet ist. Wählen Sie alternativ das passende Subnetz für die Verbindung mit dem PG/PC aus, wenn das Gerät über einen Netzknoten erreichbar ist. Wenn Sie nicht wissen, wie das Gerät mit dem PG/PC verbunden ist, wählen Sie den Eintrag "Versuche alle Schnittstellen". Wenn Sie ein MPI- oder PROFIBUS-Subnetz gewählt haben, werden zu diesem Zeitpunkt die parametrierten Busparameter in die PG/PC-Schnittstelle übernommen.
6. Falls das Gerät über ein Gateway erreichbar ist, wählen Sie aus der Klappliste "1. Gateway" das Gateway aus, welches die betroffenen Subnetze miteinander verbindet. Sind an der Schnittstelle keine Geräte erreichbar, wird die Verbindungslinie zwischen PG/PC und dem Gerät unterbrochen dargestellt. Sind Geräte erreichbar, wird die Verbindungslinie geschlossen dargestellt und die erreichbaren Teilnehmer an der gewählten Schnittstelle des PG/PC werden in der Liste "Kompatible Teilnehmer im Zielsubnetz" angezeigt.
7. Optional: Klicken Sie auf die Schaltfläche "Aktualisieren", um die Liste "Kompatible Teilnehmer im Zielsubnetz" neu aufzubauen.
8. Optional: Aktivieren Sie in der Grafik auf der linken Seite das Optionskästchen "LED blinken", um einen LED-Blinktest durchzuführen. Auf diese Weise können Sie kontrollieren, ob Sie den richtigen Teilnehmer ausgewählt haben. Der LED-Blinktest wird nicht von allen Geräten unterstützt.
9. Selektieren Sie in der Tabelle "Kompatible Teilnehmer im Zielsubnetz" Ihr Gerät und bestätigen Sie Ihre Auswahl mit "Online verbinden". Die Online-Verbindung wird zum ausgewählten Zielgerät aufgebaut.

## Ergebnis

Nach dem Herstellen der Online-Verbindung verfärben sich die Titelleisten der Editoren orange. Außerdem werden in der Titelleiste eines Editors und in der Statusleiste jeweils ein oranger Laufbalken angezeigt. Statussymbole zeigen in der Projektnavigation den Unterschied zwischen online und offline vorhandenen Objekten an.

Der Verbindungsweg wird für zukünftige Verbindungsversuche gespeichert. Der Aufruf des Dialogs "Online verbinden" ist nicht mehr nötig, außer Sie möchten den Verbindungsweg neu bestimmen.

---

### Hinweis

Wenn kein erreichbarer Teilnehmer angezeigt wird, wählen Sie einen anderen Netzzugang für die PG/PC-Schnittstelle oder überprüfen Sie die Einstellungen der Schnittstelle.

---

## Online-Verbindung trennen

Um die bestehende Online-Verbindung zu trennen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Gerät in der Projektnavigation, für das Sie die Online-Verbindung trennen möchten.
2. Wählen Sie im Menü "Online" den Befehl "Online-Verbindung trennen".

### Siehe auch

- Online verbinden mit mehreren Geräten (Seite 4274)
- Ansicht im Online-Modus (Seite 4271)
- Zuweisen einer temporären IP-Adresse (Seite 4285)
- Einfluss von Benutzerrechten (Seite 218)

## 12.3.4 Online verbinden mit mehreren Geräten

Sie können eine Online-Verbindung zu mehreren Geräten auf einmal herstellen, ohne die einzelnen Geräte vorher in der Projektnavigation oder in der Netzsicht selektieren zu müssen.

### Voraussetzung

- Es darf kein Gerät selektiert sein
- Es ist mindestens eine PG/PC-Schnittstelle installiert, die physikalisch mit einem Gerät verbunden ist, z. B. mit einem Ethernet-Kabel. Alternativ ist das Aufbauen einer virtuellen Online-Verbindung über PLCSIM oder einer Fernverbindung möglich.

### Vorgehen

Um eine Online-Verbindung zu mehreren Geräten gleichzeitig herzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie das Projekt in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "Online verbinden".  
Der Dialog "Geräte auswählen" mit einer Tabelle aller verfügbaren Geräte wird geöffnet.
3. Selektieren Sie in der Spalte "Online verbinden" die Geräte, zu denen Sie eine Online-Verbindung aufbauen möchten.
4. Klicken Sie auf die Schaltfläche "Online verbinden".

### Ergebnis

Eine Verbindung zu allen ausgewählten Geräten wird ohne weitere Nachfrage hergestellt, wenn vorher bereits mindestens einmal eine Verbindung zu den gewählten Geräten aufgebaut war. Bestand zuvor noch keine Online-Verbindung, wird der Dialog "Online verbinden" geöffnet. Konfigurieren Sie in diesem Fall zunächst die Online-Verbindung, wie im Kapitel "Online verbinden und Online-Verbindung trennen (Seite 4272)" beschrieben.

### Siehe auch

- Online verbinden und Online-Verbindung trennen (Seite 4272)
- Zuweisen einer temporären IP-Adresse (Seite 4285)



### 12.3.5 Online-Verbindungen von mehreren Geräten trennen

Sie können die bestehenden Online-Verbindungen zu mehreren Geräten auf einmal trennen, ohne die einzelnen Geräte vorher in der Projektnavigation oder in der Netzsicht selektieren zu müssen.

#### Voraussetzung

- Es ist kein Gerät selektiert.
- Für mindestens ein Gerät besteht eine Online-Verbindung.

#### Vorgehen

Um die Online-Verbindungen zu mehreren Geräten gleichzeitig zu trennen, gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü "Online" den Befehl "Online-Verbindung trennen".  
Der Dialog "Geräte auswählen" mit einer Tabelle aller verfügbaren Geräte wird geöffnet.
2. Selektieren Sie in der Spalte "Online-Verbindung trennen" die Geräte, zu denen Sie die Online-Verbindung trennen möchten.
3. Klicken Sie auf die Schaltfläche "Online-Verbindung trennen".

#### Ergebnis

Die Online-Verbindung zu allen ausgewählten Geräten wird getrennt.

## 12.4 Software und Hardware-Konfiguration eines Geräts sichern

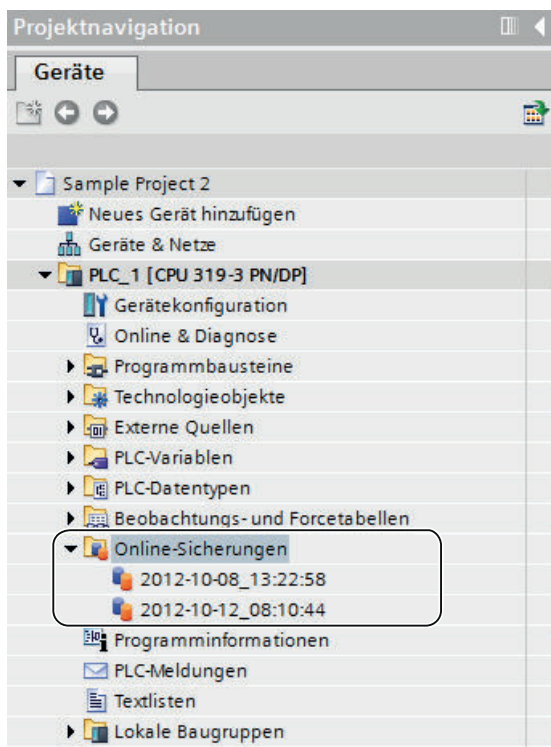
### 12.4.1 Erstellen einer Sicherung für ein Gerät

#### Software und Hardware-Konfiguration einer S7-300/400 CPU sichern

Wenn Sie bereits eine Projektierung auf eine S7-300/400 CPU geladen haben, kann es sinnvoll sein diese zu sichern. Möglicherweise haben Sie die Projektierung geändert und möchten die neue Konfiguration testen. Bevor Sie die neue Projektierung auf die CPU laden, erstellen Sie eine Sicherung des aktuellen Gerätezustands und können so zu einem späteren Zeitpunkt die aktuelle Konfiguration wiederherstellen. Die Sicherung erfolgt mit den Aktualwerten der CPU. Bei S7-400 CPUs mit Failsafe-Funktion werden die Initialwerte gesichert.

Sie können beliebig viele Sicherungen anlegen und so unterschiedliche Projektierungen für eine CPU vorhalten. Die Sicherungen werden mit dem Namen der CPU sowie Uhrzeit und Datum der Sicherung benannt. Die Sicherung finden Sie in der Projektnavigation unterhalb der CPU im Ordner "Online-Sicherungen".

Das folgende Bild zeigt eine S7-319 CPU, von der zwei Sicherungen erstellt wurden:



### Siehe auch

Software und Hardware-Konfiguration eines Geräts wiederherstellen (Seite 4277)

Sicherung der Projektierung eines Geräts (Seite 4276)

Allgemeine Informationen zum Laden (Seite 256)

## 12.4.2 Sicherung der Projektierung eines Geräts

Sie können im TIA-Portal die Projektierung einer S7-300/400 CPU sichern. So können Sie gefahrlos eine neue Projektierung auf ein Gerät laden und testen. Bei Bedarf können Sie die ursprüngliche Konfiguration der CPU wiederherstellen.

### Voraussetzung

- Die CPU muss bereits im Projekt angelegt sein.
- Es muss sich um eine S7-300/400 CPU handeln.
- Bei S7-400 CPUs muss eine Flash EPROM Memory Card gesteckt sein.
- Die CPU muss online verbunden sein. Besteht noch keine Online-Verbindung, so wird während des Backups eine Online-Verbindung hergestellt.

## Vorgehen

Um eine Sicherung der aktuellen Projektierung einer CPU zu erstellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU in der Projektnavigation.
2. Wählen Sie im Menü "Online" den Befehl "Sicherung von Online-Gerät laden".

## Ergebnis:

Eine Sicherung der gesamten Hardware-Konfiguration und der Software wird angelegt. Die Sicherung wird in der Projektnavigation im Ordner "Name der CPU > Online-Sicherungen" abgelegt. Die Sicherung trägt den Namen der CPU mit der Uhrzeit und dem Datum der Sicherung. Sie können die Sicherung umbenennen, jedoch keine Änderungen an den Inhalten der Sicherung vornehmen.

## Siehe auch

Online verbinden und Online-Verbindung trennen (Seite 4272)

Software und Hardware-Konfiguration eines Geräts wiederherstellen (Seite 4277)

Erstellen einer Sicherung für ein Gerät (Seite 4275)

### 12.4.3 Software und Hardware-Konfiguration eines Geräts wiederherstellen

Wenn Sie die Projektierung eines Geräts zu einem früheren Zeitpunkt gesichert haben, können Sie die Sicherung wieder auf das Gerät übertragen. Die gespeicherte Konfiguration wird dann auf dem Gerät wiederhergestellt.

## Voraussetzung

Sie müssen das Gerät projektiert und eine Sicherung des Geräts im Projekt gespeichert haben.

## Vorgehen

Um einen älteren Software- und Hardware-Stand auf einem Gerät wiederherzustellen, gehen Sie folgendermaßen vor:

1. Klappen Sie den Ordner des Geräts in der Projektnavigation auf, um die unterlagerten Objekte anzuzeigen.
2. Öffnen Sie den Ordner "Online-Sicherungen".
3. Selektieren Sie die Sicherung, die Sie wiederherstellen möchten.

4. Wählen Sie im Menü "Online" den Befehl "Laden in Gerät".
    - Falls Sie bereits eine Online-Verbindung hergestellt hatten, wird der Dialog "Vorschau laden" geöffnet. In diesem Dialog werden Ihnen Meldungen angezeigt und für das Laden notwendige Aktionen vorgeschlagen.
    - Falls Sie bisher noch keine Online-Verbindung hergestellt hatten, wird der Dialog "Erweitertes Laden" geöffnet und Sie müssen zunächst die Schnittstellen auswählen, über die Sie die Online-Verbindung zum Gerät herstellen möchten.  
Siehe auch: Online verbinden und Online-Verbindung trennen (Seite 4272)
  5. Kontrollieren Sie die Meldungen im Dialog "Vorschau laden" und aktivieren Sie gegebenenfalls die Aktionen in der Spalte "Aktion".
- 

#### **Hinweis**

Das Durchführen der vorgeschlagenen Aktionen bei laufendem Anlagenbetrieb kann bei Funktionsstörungen oder Programmfehlern schwere Sach- und Personenschäden verursachen!

---

6. Sobald das Laden möglich ist, wird die Schaltfläche "Laden" aktiv.
7. Klicken Sie auf die Schaltfläche "Laden".  
Die Sicherung wird auf das Gerät übertragen und das Gerät wiederhergestellt.  
Anschließend wird der Dialog "Ergebnisse laden" geöffnet. In diesem Dialog können Sie prüfen, ob der Ladevorgang erfolgreich war, und eventuell weitere Aktionen auswählen.
8. Klicken Sie auf die Schaltfläche "Fertig stellen".

#### **Siehe auch**

Erstellen einer Sicherung für ein Gerät (Seite 4275)

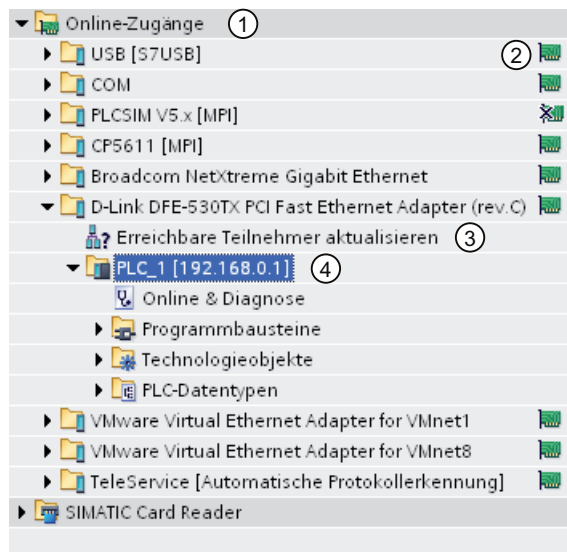
## **12.5 PG/PC-Schnittstellen konfigurieren**

### **12.5.1 Online-Zugänge**

#### **Online-Zugang des Projekts**

Im Ordner "Online-Zugänge" in der Projektnavigation finden Sie alle aktiven Schnittstellen Ihres PG/PC. An jedem Schnittstellen-Symbol erhalten Sie Informationen zum jeweiligen Status. Sie können sich die erreichbaren Teilnehmer anzeigen lassen und über das Kontextmenü die Eigenschaften einer Schnittstelle anzeigen und ändern.

Das folgende Bild zeigt den Ordner "Online-Zugänge" in der Projektnavigation:



- ① Ordner "Online-Zugänge" in der Projektnavigation  
Im Ordner "Online-Zugänge" sind sämtliche im PG/PC installierten Schnittstellen sichtbar.
- ② Statusanzeige für die Schnittstellen  
Der aktuelle Status einer Schnittstelle wird durch ein Symbol rechts neben dem Namen angezeigt. Die Bedeutung der Symbole entnehmen Sie dem Tooltip.
- ③ Aktualisieren der Liste der erreichbaren Teilnehmer.  
Diese Funktion ist für jede Hardware-Schnittstelle des PG/PC verfügbar. Software-Schnittstellen, wie z. B. eine Fernverbindung, verfügen nicht über diese Funktion.
- ④ Geräte, die über die jeweilige Schnittstelle mit dem PG/PC verbunden sind  
Der Typ des jeweiligen Geräts sowie dessen Status werden mit Hilfe des voran stehenden Symbols angezeigt.

## Erreichbare Teilnehmer anzeigen oder aktualisieren








Sie haben folgende Möglichkeiten, um sich die online erreichbaren Teilnehmer an Ihrem PG/PC anzeigen zu lassen:

- Anzeige der erreichbaren Teilnehmer an einer einzelnen Schnittstelle des PG/PC in der Projektnavigation. In der Projektnavigation können außerdem zusätzliche Informationen zu den einzelnen erreichbaren Teilnehmern angezeigt werden.
- Anzeige der erreichbaren Teilnehmer aller Schnittstellen in einer Liste.

Siehe auch: Erreichbare Teilnehmer anzeigen

## Übersicht der Symbole für erreichbare Teilnehmer

Die erreichbaren Teilnehmer werden mit einem Symbol entsprechend ihres Typs und ihres Zustandes gekennzeichnet. Im Folgenden sehen Sie eine Übersicht über alle Symbole und deren Bedeutung.

|   |  |
|---|--|
|    | Symbol für nicht identifizierte Baugruppen<br>Dieses Symbol wird immer dann angezeigt, wenn die Identifikation einer Baugruppe noch nicht vollständig abgeschlossen ist, oder wenn die Identifikation einer Baugruppe nicht erfolgreich war, beispielsweise weil die erforderlichen Online-Daten nicht gelesen werden konnten.   |
|    | Symbol für folgende Geräte-Typen: <ul style="list-style-type: none"> <li>• PLCs</li> <li>• SIMOCODE pro-Geräte</li> <li>• IE/PB Links</li> <li>• CPs von PC-Systemen</li> <li>• SCALANCE Kopfbaugruppen</li> <li>• S7-300 und S7-400 CPs</li> <li>• PROFINET IO-Devices und PROFINET CPs</li> <li>• SCALANCE-Baugruppen und Netzübergänge, die nicht identifiziert werden konnten</li> </ul> |
|    | PROFINET IO-Devices, Encoder, Switchgear, Sensoren und Identifikationssysteme, die durch ähnliche Geräte ersetzt wurden, weil diese nicht identifiziert werden konnten   |
|  | Symbol für folgende Geräte-Typen: <ul style="list-style-type: none"> <li>• HMI-Geräte</li> <li>• PROFINET IO-Devices vom Typ HMI, wenn diese nicht identifiziert werden konnten und deshalb durch ein ähnliches Gerät ersetzt wurden</li> </ul>  |
|  | PROFINET IO-Devices vom Typ Antrieb, die nicht identifiziert und daher durch ein ähnliches Gerät ersetzt wurden  |
|  | PROFINET IO-Devices vom Typ Development Kit und Network Components, die nicht identifiziert und daher durch ein ähnliches Gerät ersetzt wurden   |
|  | PROFINET IO-Devices vom Typ Teleservice Adapter, die nicht identifiziert und daher durch ein ähnliches Gerät ersetzt wurden  |

### Siehe auch

Eigenschaften einer Schnittstelle anzeigen und ändern (Seite 4281)

## 12.5.2 Grundlagen zur Parametrierung der PG/PC-Schnittstelle

### Verbindungsmöglichkeiten zu Zielsystemen

Wenn die Geräte des Projekts über unterschiedliche Subnetze verbunden sind, müssen Sie der PG/PC-Schnittstelle jeweils den passenden Netzzugang zuweisen, um Online-

Verbindungen zu den Zielsystemen aufbauen zu können. Folgende Schnittstellen werden automatisch unterstützt:

- MPI
- PROFIBUS
- Industrial Ethernet (ISO und TCP/IP)

Die Schnittstellen lassen sich in unterschiedlichem Umfang parametrieren. Welche Möglichkeiten zur Parametrierung bestehen, erfahren Sie in den folgenden Kapiteln.

---

#### **Hinweis**

Beachten Sie, dass Änderungen an Schnittstellen-Parametern direkten Einfluss auf das Betriebssystem und den PG/PC haben. Insofern sind manche Parameter nur mit ausreichend Nutzerrechten änderbar.

---

#### **Siehe auch**

Parametrieren der Industrial Ethernet-Schnittstelle (Seite 4282)

Parametrieren der MPI- und PROFIBUS-Schnittstelle (Seite 4287)

### **12.5.3 Eigenschaften einer Schnittstelle anzeigen und ändern**

#### **Einführung**

Für jede Schnittstelle können Sie Eigenschaften anzeigen und gegebenenfalls ändern, z. B. Netztyp, Adresse und Status.

#### **Vorgehen**

Um die Eigenschaften zu öffnen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation unter "Online-Zugänge" mit der rechten Maustaste auf die gewünschte Schnittstelle.
2. Wählen Sie aus dem Kontextmenü den Befehl "Eigenschaften".  
Der Dialog mit den Eigenschaften der Schnittstelle wird geöffnet. Auf der linken Seite des Dialoges finden Sie die Bereichsnavigation. Unter den einzelnen Einträgen in der Bereichsnavigation können Sie die aktuellen Parameter anschauen und bei Bedarf ändern.

### **12.5.4 Schnittstellen hinzufügen**

Sie haben die Möglichkeit nach der Installation des TIA-Portals weitere Schnittstellen zu installieren.

## Vorgehen

Um eine Schnittstelle nachträglich zu installieren und im TIA-Portal hinzuzufügen, gehen Sie folgendermaßen vor:

1. Installieren oder aktualisieren Sie die Treiber im Betriebssystem, nachdem Sie die Schnittstellen hardwareseitig installiert haben.
2. Schließen Sie das TIA-Portal, falls es noch geöffnet ist.
3. Öffnen Sie die Windows-Systemsteuerung.
4. Öffnen Sie den Eintrag "PG/PC-Schnittstelle einstellen" in der Systemsteuerung. Der Dialog "PG/PC-Schnittstelle einstellen" wird geöffnet.
5. Nehmen Sie gegebenenfalls Änderungen an der Schnittstellenkonfiguration vor und bestätigen Sie die Änderungen mit "OK". Klicken Sie auf jeden Fall auf die Schaltfläche "OK", auch wenn Sie keine Änderungen vorgenommen haben.
6. Starten Sie das TIA-Portal neu.

## Ergebnis

Die neu installierte Schnittstelle wird nun in der Projektnavigation unter dem Ordner "Online-Zugänge" angezeigt.

## 12.5.5 Ethernet Schnittstelle parametrieren

### 12.5.5.1 Parametrieren der Industrial Ethernet-Schnittstelle

#### Möglichkeiten zur Parametrierung der Industrial Ethernet-Schnittstelle

Für die Parametrierung der Industrial Ethernet-Schnittstelle haben Sie folgende Möglichkeiten:

- Vom Betriebssystem bestimmte Parameter  
Die Industrial Ethernet-Schnittstelle besitzt Parameter, die im Betriebssystem eingestellt werden und für alle angeschlossenen Geräte gültig sind. Diese Parameter werden nur angezeigt, können aber in den Netzwerkeinstellungen des Betriebssystems geändert werden.
- In der Software einstellbare Parameter

---

#### Hinweis

Beachten Sie, dass Änderungen an Schnittstellen-Parametern direkten Einfluss auf das Betriebssystem und den PG/PC haben. Insofern sind manche Parameter nur mit ausreichend Nutzerrechten änderbar.

---



## Parameter der Industrial Ethernet-Schnittstelle

Die folgende Tabelle enthält einen Überblick, welche Parameter der Industrial Ethernet-Schnittstelle vom Betriebssystem vorgegeben sind und welche vom Nutzer geändert werden können.

| Nicht änderbare Parameter         | Einstellbare Parameter                                   |
|-----------------------------------|--|
| MAC-Adresse                       | Fast Acknowledge am IE-PG-Zugang sowie für TCP/IP        |
| DHCP-Server aktiviert/deaktiviert | Timeout am IE-PG-Zugang sowie für TCP/IP                 |
| APIPA aktiviert/deaktiviert       | LLDP   |
| IP-Adresse                        | Zusätzliche, temporäre IP-Adressen für die Netzwerkkarte |
| Subnetzmaske                      | -  |
| DNS-Adressen                      | -  |
| DHCP-Adressen                     | -  |

### Siehe auch

- Grundlagen zur Parametrierung der PG/PC-Schnittstelle (Seite 4280)
- Betriebssystemparameter anzeigen (Seite 4283)
- PG/PC-Schnittstelle mit einem Subnetz verbinden (Seite 4284)
- Ethernet-Schnittstelle parametrieren (Seite 4284)
- Zuweisen einer temporären IP-Adresse (Seite 4285)
- Temporäre IP-Adressen verwalten (Seite 4286)
- Einfluss von Benutzerrechten (Seite 218)

### 12.5.5.2 Betriebssystemparameter anzeigen

Die Ethernet-Schnittstelle ist Teil des Betriebssystems. Die Parameter der Netzwerkkarte können daher komplett in den Netzwerkeinstellungen des Betriebssystems angepasst werden.

Sie können sich die folgenden Parameter in der Software anzeigen lassen:

- Physikalische Adresse der Netzwerkkarte
- Vergabe der IP-Adresse durch einen DHCP-Server aktiviert oder deaktiviert
- Vergabe einer privaten IP-Adresse durch das Betriebssystem aktiviert oder deaktiviert
- Aktuell fest zugewiesene IP-Adresse
- Zugeordnete Subnetzmaske
- DNS-Adressen
- DHCP-Adressen

Wie sie die Parameter ändern können, entnehmen Sie bitte der Dokumentation des Betriebssystems oder der Netzwerkkarte.

### Aktuelle Parameter der Ethernet-Schnittstelle anzeigen

Um sich die aktuellen Parameter der Ethernet-Schnittstelle anzeigen zu lassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie in der Bereichsnavigation "Konfigurationen > Industrial Ethernet".

### Siehe auch

Ethernet-Schnittstelle parametrieren (Seite 4284)

### 12.5.5.3 PG/PC-Schnittstelle mit einem Subnetz verbinden

Wenn Sie mehrere Subnetze angelegt haben, können Sie festlegen, mit welchem Subnetz die Ethernet-Schnittstelle verbunden werden soll.

### Vorgehen

Um das Subnetz zu wählen, mit dem die Ethernet-Schnittstelle verbunden ist, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie unter "Allgemein > Zuordnung" in der Klappliste "Verbindung mit Subnetz" das Subnetz aus, mit dem Sie die Ethernet-Schnittstelle des PG/PC verbinden möchten.
4. Schließen Sie den Dialog mit "OK".

### 12.5.5.4 Ethernet-Schnittstelle parametrieren

Einige das Netzwerkprotokoll betreffende Parameter können Sie direkt innerhalb der Software anpassen.

### Voraussetzung

Sie müssen über ausreichend Benutzerrechte verfügen.

Siehe auch: Einfluss von Benutzerrechten (Seite 218).

## Vorgehen

Um Parameter zu ändern, die das Netzwerkprotokoll betreffen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie "Konfigurationen > IE-PG-Zugang", um für die Netzwerkverwaltung relevante Protokolleinstellungen anzupassen.
  - Aktivieren Sie das Optionskästchen "Fast Acknowledge", um schnellere Reaktionszeiten bei kleineren Netzwerkpaketen zu erreichen.
  - Wählen Sie aus der Klappliste "Timeout" die Zeit aus, die bis zur Erkennung eines Netzwerkteilnehmers maximal vergehen darf.
4. Um das LLDP Protokoll zur genaueren Bestimmung der Netzwerktopologie zu aktivieren, aktivieren Sie unter "Konfigurationen > LLDP" das Optionskästchen "LLDP aktiv".
5. Wählen Sie "Konfigurationen > TCP/IP", um das TCP/IP-Protokoll für den Netzwerkverkehr während des Betriebs anzupassen.
  - Aktivieren Sie das Optionskästchen "Fast Acknowledge", um schnellere Reaktionszeiten bei kleineren Netzwerkpaketen zu erreichen.
  - Wählen Sie aus der Klappliste "Timeout" die Zeit aus, die bei der Kommunikation mit einem Netzwerkteilnehmer maximal vergehen darf bis es zu einer Zeitüberschreitung kommt.

## Siehe auch

Einfluss von Benutzerrechten (Seite 218)

Betriebssystemparameter anzeigen (Seite 4283)

### 12.5.5.5 Zuweisen einer temporären IP-Adresse

#### Hinzufügen einer temporären IP-Adresse

Befindet sich die IP-Adresse eines Geräts in einem anderen Subnetz als die IP-Adresse der Netzwerkkarte, müssen Sie der Netzwerkkarte zunächst eine zusätzliche IP-Adresse mit derselben Subnetzadresse wie der des Geräts zuweisen. Erst danach ist eine Kommunikation zwischen dem Gerät und dem PG/PC möglich.

Das Zuweisen einer zusätzlichen, temporären IP-Adresse wird gegebenenfalls automatisch vorgeschlagen, wenn Sie eine Online-Aktion durchführen möchten und sich die gegenwärtige IP-Adresse des PG/PC noch nicht im richtigen Subnetz befindet.

Eine temporär zugewiesene IP-Adresse ist bis zum nächsten Neustart des PG/PC gültig oder bis sie manuell gelöscht wird.

---

**Hinweis**

Sie müssen über ausreichend Benutzerrechte verfügen, wenn Sie eine temporäre IP-Adresse zuweisen möchten.

Siehe auch: Einfluss von Benutzerrechten (Seite 218)

---

**Siehe auch**

Temporäre IP-Adressen verwalten (Seite 4286)

### 12.5.5.6 Temporäre IP-Adressen verwalten

Befindet sich die IP-Adresse eines Geräts in einem anderen Subnetz als die aktuell fest zugewiesene IP-Adresse der Netzwerkkarte, wird der Netzwerkkarte temporär eine passende IP-Adresse aus dem Subnetz des Geräts zuweisen.

Sie können sich alle temporär zugewiesenen Adressen anzeigen lassen und diese löschen. Beachten Sie, dass IP-Adressen, welche Sie im Betriebssystem manuell zugewiesen haben, nicht im TIA-Portal angezeigt werden.

**Voraussetzung**

Für das Löschen sind ausreichend Benutzerrechte nötig.

**Vorgehen**

Um temporär zugewiesene Adressen anzuzeigen und zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie "Konfigurationen > IE-PG-Zugang".  
Eine Tabelle mit den zugewiesenen IP-Adressen wird angezeigt.
4. Klicken Sie auf die Schaltfläche "Lösche projektspezifische IP-Adressen", um alle IP-Adressen auf einmal zu löschen.

**Siehe auch**

Einfluss von Benutzerrechten (Seite 218)

### 12.5.5.7 TCP/IP Konfiguration zurücksetzen

Haben Sie Änderungen an den TCP/IP Protokolleinstellungen vorgenommen, können Sie diese auf die Standardwerte zurücksetzen.

#### Vorgehen

Um die Standwerte für die TCP/IP Konfiguration wiederherzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Ethernet-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie "Konfigurationen > TCP/IP".
4. Klicken Sie auf die Schaltfläche "Standard", um alle Einstellungen zurückzusetzen.

### 12.5.6 MPI- und PROFIBUS-Schnittstelle parametrieren

#### 12.5.6.1 Parametrieren der MPI- und PROFIBUS-Schnittstelle

#### Möglichkeiten zur Parametrierung der MPI- und PROFIBUS-Schnittstelle

Die folgenden Möglichkeiten stehen zur Parametrierung der MPI- und PROFIBUS-Schnittstelle zur Verfügung:

- Automatische Konfiguration: Sie können automatisch erkennen lassen, ob ein Gerät über PROFIBUS oder MPI mit der PG/PC-Schnittstelle verbunden ist.
- Auswahl einer voreingestellten Konfiguration für PROFIBUS- oder MPI, die nachträglich angepasst werden kann.

#### Geräte- und netzbezogene Parametriermöglichkeiten für MPI und PROFIBUS

Sie können bei MPI- und PROFIBUS-Schnittstellen gerätebezogene und netzbezogene Parameter einstellen. Gerätebezogene Parameter sind lokale Einstellungen der Schnittstelle. Netzbezogene Parameter dagegen müssen geräteübergreifend übereinstimmen.

#### Veränderbare Parameter an der MPI-Schnittstelle

Für die MPI-Schnittstelle können Sie die folgenden voreingestellten Parameter anpassen:

| Gerätebezogene Parameter | Netzbezogene Parameter |
|--------------------------|------------------------|
| Ist einziger Master      | Höchste Adresse        |
| Eigene Adresse           | Transferrate           |
| Timeout                  |                        |

## Veränderbare Parameter an der PROFIBUS-Schnittstelle

Für die PROFIBUS-Schnittstelle können Sie die folgenden voreingestellten Parameter anpassen:

| Gerätebezogene Parameter | Netzbezogene Parameter   |
|--------------------------|--------------------------|
| Ist einziger Master      | Höchste Adresse          |
| Eigene Adresse           | Transferrate             |
| Timeout                  | Profil                   |
|                          | Busparameter             |
|                          | Anzahl der Master am Bus |
|                          | Anzahl der Slaves am Bus |

### Siehe auch

Grundlagen zur Parametrierung der PG/PC-Schnittstelle (Seite 4280)

### 12.5.6.2 MPI- oder PROFIBUS-Schnittstelle automatisch parametrieren

#### Automatische Erkennung der Busparameter einrichten

Wenn Sie eine Schnittstelle mit automatischer Erkennung der Busparameter wählen, z. B. CP5611 (Auto), dann können Sie den PG/PC an MPI oder PROFIBUS anschließen ohne Busparameter einstellen zu müssen. Bei Übertragungsgeschwindigkeiten kleiner 187,5 kBit/s können allerdings Wartezeiten bis zu einer Minuten entstehen.

#### Voraussetzung

- Am Bus sind Master angeschlossen, die zyklisch Busparameter verteilen.
- Bei PROFIBUS-Netzen muss das zyklische Verteilen der Busparameter aktiviert sein.

#### Vorgehen

Um die automatische Erkennung der Busparameter zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle in der Projektnavigation.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie unter "Allgemein > Konfigurationen > Aktive Konfiguration" die Einstellung "Automatische Protokollerkennung".
4. Wählen Sie unter "Konfigurationen > Auto-Konfiguration > Lokale Einstellungen" aus der Klappliste "Eigene Adresse" die Adresse der PG/PC-Schnittstelle aus.
5. Wenn Sie anschließend die aktuellen Buseinstellungen anzeigen möchten, klicken Sie auf die Schaltfläche "Netzwerkerkennung".

## Siehe auch

MPI-Schnittstelle parametrieren (Seite 4289)

PROFIBUS-Schnittstelle parametrieren (Seite 4290)

### 12.5.6.3 MPI-Schnittstelle parametrieren

#### Parameter der MPI-Schnittstelle ändern

Für die MPI-Schnittstelle lassen sich netzbezogene Parameter und Busparameter anpassen. Zunächst sollten Sie eine Voreinstellung wählen und diese bei Bedarf genauer anpassen.

#### MPI-Schnittstelle mit Voreinstellungen parametrieren

Um die Parameter der MPI-Schnittstelle anzupassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle in der Projektnavigation.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie unter "Allgemein > Zuordnung" in der Klappliste "Verbindung mit Subnetz" das Subnetz aus, mit dem die Schnittstelle verbunden werden soll.
4. Wählen Sie unter "Allgemein > Konfigurationen" eine Voreinstellung für die geräte- und netzbezogenen Parameter. Die Voreinstellungen sind für die meisten Konfigurationen geeignet. Wählen Sie eine der folgenden Einstellungen:
  - Automatische Protokollerkennung  
Sie können das PG an MPI bzw. PROFIBUS anschließen, ohne Busparameter einstellen zu müssen. Bei Übertragungsgeschwindigkeiten kleiner 187,5 kBit/s können allerdings Wartezeiten bis zu einer Minute entstehen. Voraussetzung für die automatische Erkennung ist, dass am Bus Master angeschlossen sind, die zyklisch Busparameter verteilen. Bei PROFIBUS-Subnetzen darf das zyklische Verteilen der Busparameter nicht ausgeschaltet sein (voreingestellte PROFIBUS-Netzeinstellung).
  - MPI  
Das Übertragungsprotokoll "MPI" wird gewählt. Es werden gängige Parameter eingestellt, die für die meisten Konfigurationen passen. Sie können die Parameter jedoch bei Bedarf ändern.
  - PROFIBUS  
Das Übertragungsprotokoll "PROFIBUS" wird gewählt. Es werden gängige Parameter eingestellt, die für die meisten Konfigurationen passen. Sie können die Parameter jedoch bei Bedarf ändern.

#### Voreingestellte Parameter ändern

Um die Voreinstellungen an Ihre Bedürfnisse anzupassen, ändern Sie die entsprechenden Parameter bei Bedarf unter "Konfigurationen > MPI" ab.

Folgende gerätebezogenen Parameter können Sie einstellen:

- Ist einziger Master  
Hiermit wird eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des PG/PC an das Netz deaktiviert, weil das PG/PC der einzige Master am Bus ist.
  - Aktivieren Sie diese Option nur, wenn Sie ausschließlich Slaves an Ihr PG oder Ihren PC angeschlossen haben.
  - Wenn das Optionskästchen "Ist einziger Master" aktiviert ist, ist die Ermittlung des direkt angeschlossenen Gerätes über das Fenster "Erreichbare Teilnehmer" nicht möglich.
- Eigene Adresse  
Diese Angabe bezieht sich auf das PG beziehungsweise den PC, auf dem Sie die Parametrierung der Schnittstelle aufgerufen haben. Stellen Sie hier die lokale Teilnehmeradresse Ihres PG/PC ein.
  - Die Adresse muss netzweit eindeutig sein.
  - Unter dieser Adresse wird der PG/PC im MPI-Netz angesprochen.
- Überprüfen  
Hiermit aktivieren Sie eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des PG/PC an das Netz. Der Treiber prüft, ob die eigene Adresse bereits durch einen anderen Teilnehmer belegt ist. Hierbei werden neben den aktiven auch die passiven Teilnehmer berücksichtigt. Der Treiber hört dabei am PROFIBUS mit. Bei der automatischen Überprüfung dauert das Ankoppeln des PG/PC an das Netz länger. Um das Überprüfen zu nutzen, muss der Treiber die Funktion unterstützen. Weiterhin darf die Option "Ist einziger Master" nicht aktiviert sein.
- Timeout  
Erhöhen Sie den Wert der Verbindungsüberwachungszeit, wenn Sie z. B. Probleme mit hohen Antwortzeiten im Netz haben.

Folgende netzbezogenen Parameter können Sie einstellen:

- Höchste Adresse:  
Wählen Sie hier die projektierte höchste Teilnehmeradresse. Stellen Sie sicher, dass bei allen Teilnehmern eines MPI-Netzes die gleiche höchste Teilnehmeradresse eingestellt ist.
- Transferrate:  
Wählen Sie hier die Übertragungsgeschwindigkeit, die im MPI-Netz verwendet werden soll.

## Siehe auch

MPI- oder PROFIBUS-Schnittstelle automatisch parametrieren (Seite 4288)

### 12.5.6.4 PROFIBUS-Schnittstelle parametrieren

#### Parameter der PROFIBUS-Schnittstelle ändern

Für die PROFIBUS-Schnittstelle lassen sich netzbezogene Parameter und Busparameter genauer anpassen. Zunächst sollten Sie eine Voreinstellung wählen und diese bei Bedarf genauer anpassen.



## PROFIBUS-Schnittstelle mit Voreinstellungen parametrieren

Um die Parameter der PROFIBUS-Schnittstelle anzupassen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die Schnittstelle in der Projektnavigation.
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie unter "Allgemein > Zuordnung" in der Klappliste "Verbindung mit Subnetz" das Subnetz aus, mit dem die Schnittstelle verbunden werden soll.
4. Wählen Sie unter "Allgemein > Konfiguration" eine Voreinstellung für die geräte- und netzbezogenen Parameter. Die Voreinstellungen sind für die meisten Konfigurationen geeignet. Wählen Sie eine der folgenden Einstellungen:
  - Automatische Protokollerkennung  
Sie können das PG an MPI bzw. PROFIBUS anschließen, ohne Busparameter einstellen zu müssen. Bei Übertragungsgeschwindigkeiten kleiner 187,5 kBit/s können allerdings Wartezeiten bis zu einer Minute entstehen. Voraussetzung für die automatische Erkennung ist, dass am Bus Master angeschlossen sind, die zyklisch Busparameter verteilen. Bei PROFIBUS-Subnetzen darf das zyklische Verteilen der Busparameter nicht ausgeschaltet sein (voreingestellte PROFIBUS-Netzeinstellung).
  - MPI  
Das Übertragungsprotokoll "MPI" wird gewählt. Es werden gängige Parameter eingestellt, die für die meisten Konfigurationen passen. Sie können die Parameter jedoch bei Bedarf ändern.
  - PROFIBUS  
Das Übertragungsprotokoll "PROFIBUS" wird gewählt. Es werden gängige Parameter eingestellt, die für die meisten Konfigurationen passen. Sie können die Parameter jedoch bei Bedarf ändern.

## Voreingestellte Parameter ändern

Um die Voreinstellungen an Ihre Bedürfnisse anzupassen, ändern Sie die entsprechenden Parameter bei Bedarf unter "Konfigurationen > PROFIBUS" ab.

Folgende gerätebezogenen Parameter können Sie einstellen:

- Ist einziger Master  
Hiermit wird eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des PG/PC an das Netz deaktiviert, weil das PG/PC der einzige Master am Bus ist.
  - Aktivieren Sie diese Option nur, wenn Sie ausschließlich Slaves an Ihr PG oder Ihren PC angeschlossen haben.
  - Wenn das Optionskästchen "Ist einziger Master" aktiviert ist, ist die Ermittlung des direkt angeschlossenen Gerätes über das Fenster "Erreichbare Teilnehmer" nicht möglich.
- Eigene Adresse  
Diese Angabe bezieht sich auf das PG beziehungsweise den PC, auf dem Sie die Parametrierung der Schnittstelle aufgerufen haben. Stellen Sie hier die lokale Teilnehmeradresse Ihres PG/PC ein.
  - Die Adresse muss netzweit eindeutig sein.
  - Unter dieser Adresse wird der PG/PC im PROFIBUS-Netz angesprochen.

- **Überprüfen**  
Hiermit aktivieren Sie eine zusätzliche Sicherheitsfunktion gegen Busstörungen beim Ankoppeln des PG/PC an das Netz. Der Treiber prüft, ob die eigene Adresse bereits durch einen anderen Teilnehmer belegt ist. Hierbei werden neben den aktiven auch die passiven Teilnehmer berücksichtigt. Der Treiber hört dabei am PROFIBUS mit. Bei der automatischen Überprüfung dauert das Ankoppeln des PG/PC an das Netz länger. Um das Überprüfen zu nutzen, muss der Treiber die Funktion unterstützen. Weiterhin darf die Option "Ist einziger Master" nicht aktiviert sein.
- **Timeout**  
Erhöhen Sie den Wert der Verbindungsüberwachungszeit, wenn Sie z. B. Probleme mit hohen Antwortzeiten im Netz haben.

Folgende netzbezogenen Parameter können Sie einstellen:

- **Höchste Adresse:**  
Wählen Sie hier die projektierte höchste Teilnehmeradresse. Stellen Sie sicher, dass bei allen Teilnehmern eines PROFIBUS-Netzes die gleiche höchste Teilnehmeradresse eingestellt ist.
- **Transferrate:**  
Wählen Sie hier die Übertragungsgeschwindigkeit, die im PROFIBUS-Netz verwendet werden soll.
- **Profil:**  
Sie können zwischen vier Alternativen für die Einstellungen für den PROFIBUS wählen. "DP", "Standard" und "Universal (DP/FMS)" sind vordefinierte Einstellungen, die Sie nicht ändern können. Mit der Auswahl von "Benutzerdefiniert" können Sie die Busparameter selbst anpassen.
  - Wenn Sie "Benutzerdefiniert" gewählt haben, wechseln Sie in der Bereichsnavigation auf "Konfigurationen > PROFIBUS > Busparameter".
  - Wenn Sie eine der Voreinstellungen (DP, Standard oder Universal (DP/FMS)) ausgewählt haben, sollten Sie unter "Konfigurationen > PROFIBUS > Busparameter > Zusätzliche Parameter" das Optionskästchen "Berücksichtigen" aktivieren. Sie können dann die Anzahl der Master und Slaves am Bus einstellen. Die Busparameter werden auf diese Weise genauer berechnet und eventuelle Busstörungen können ausgeschlossen werden. Bei einem benutzerdefinierten Profil kann die Option nicht aktiviert werden.

## Siehe auch

Übersicht der Busparameter für PROFIBUS (Seite 4292)

MPI- oder PROFIBUS-Schnittstelle automatisch parametrieren (Seite 4288)

### 12.5.6.5 Übersicht der Busparameter für PROFIBUS

#### Einführung

Eine einwandfreie Funktion des PROFIBUS-Subnetzes ist nur dann gegeben, wenn die Parameter für das Busprofil aufeinander abgestimmt sind. Sie sollten die voreingestellten

Werte deshalb nur dann ändern, wenn Sie mit der Parametrierung des Busprofils für PROFIBUS vertraut sind.

Die Busparameter sind in Abhängigkeit vom Busprofil einstellbar oder nicht einstellbar. Es werden immer die Offline-Werte der Busparameter angezeigt, auch wenn Sie online mit dem Zielsystem verbunden sind.

Die angezeigten Parameter gelten für das gesamte PROFIBUS-Subnetz.

### Bedeutung der einzelnen Parameter

- Tslot: Warte-auf-Empfangszeit (slot time)  
Die Warte-auf-Empfang-Zeit (slot-time) legt fest, wie lange der Sender maximal wartet, um vom angesprochenen Partner eine Antwort zu erhalten.
- Max. Tsd: maximale Protokoll-Bearbeitungszeit ( max. station delay responder)  
Die maximale Protokoll-Bearbeitungszeit legt fest, nach welcher Zeit der antwortende Teilnehmer das Protokoll bearbeitet haben muss.
- Min. Tsd: minimale Protokoll-Bearbeitungszeit (min. station delay responder)  
Die minimale Protokoll-Bearbeitungszeit legt fest, welche Zeit der antwortende Teilnehmer mindestens benötigt, um das Protokoll zu bearbeiten.
- Tset: Auslösezeit (setup-Time)  
Die Auslösezeit ist die Zeit, die zwischen dem Empfang eines Datentelegramms und der Reaktion darauf verstreichen darf.
- Tqui: Modulator-Ausklingzeit (Quiet-Time for Modulator)  
Die Modulator-Ausklingzeit ist die Zeit, die für das Umstellen von Senden auf Empfangen benötigt wird.
- GAP-Faktor: GAP-Aktualisierungsfaktor (GAP-Faktor)  
Der GAP-Aktualisierungsfaktor legt fest, nach wieviel Token-Umläufen ein hinzugekommener Teilnehmer in den Tokenring aufgenommen wird.
- Retry Limit: maximale Anzahl von Aufruf-Wiederholungen (retry limit)  
Mit dem Parameter wird festgelegt, wieviele Versuche maximal unternommen werden, um einen Teilnehmer zu erreichen.
- Trdy: Bereitschaftszeit (ready-time)  
Die Bereitschaftszeit ist die Zeit für Quittung oder Antwort.
- Tid1: Ruhezustandszeit 1 (idle-time 1)  
Die Ruhezustandszeit 1 legt die Verzögerungszeit nach Empfang einer Antwort fest.
- Tid2: Ruhezustandszeit 2 (idle-time 2)  
Die Ruhezustandszeit 2 legt die Verzögerungszeit nach Senden eines Aufrufs ohne Antwort fest.

- **Ttr: Soll-Token-Umlaufzeit (Target-Rotation-Time)**  
Die Soll-Token-Umlaufzeit ist die maximal zur Verfügung gestellte Zeit für einen Token-Umlauf. In dieser Zeit erhalten alle aktiven Teilnehmer (Master) einmal das Senderecht (Token). Die Differenz zwischen Soll-Token-Umlaufzeit und tatsächlicher Token-Umlaufzeit bestimmt, wieviel Zeit den Mastern für das Senden von Datentelegrammen an die Slaves übrig bleibt.  
Wählen Sie als minimale Target-Rotation-Time (Ttr) = 5000 mal HSA (Highest Station Address).
- **Ansprechüberwachung: Ansprechüberwachungszeit (Watchdog)**  
Die Ansprechüberwachung legt fest, nach welcher Zeit ein Teilnehmer angesprochen werden muss.  
Wählen Sie als minimale Ansprechüberwachungszeit = 6250 mal HSA.

---

#### Hinweis

Wenn Sie ein benutzerdefiniertes Busprofil erstellen wollen, dann beachten Sie bitte, dass die minimale Target-Rotation-Time (Ttr) 5000 mal HSA (höchste PROFIBUS-Adresse) betragen sollte. Außerdem sollte die die minimale Ansprechüberwachung (Watchdog) 6250 mal HSA betragen.

---

#### Siehe auch

PROFIBUS-Schnittstelle parametrieren (Seite 4290)

#### 12.5.6.6 MPI- oder PROFIBUS Konfiguration zurücksetzen

Haben Sie Änderungen an den MPI- oder PROFIBUS-Protokolleinstellungen vorgenommen, können Sie diese auf die Standardwerte zurücksetzen.

#### Vorgehen

Um die Standardwerte für die MPI- oder PROFIBUS-Konfiguration wiederherzustellen, gehen Sie folgendermaßen vor:

1. Selektieren Sie die MPI/PROFIBUS-Schnittstelle in der Projektnavigation unter "Online-Zugänge".
2. Wählen Sie im Kontextmenü der Schnittstelle den Befehl "Eigenschaften".  
Der Dialog zur Konfiguration der Schnittstelle wird geöffnet.
3. Wählen Sie "Konfigurationen > MPI" oder "Konfigurationen > PROFIBUS", abhängig davon, welche Schnittstelleneigenschaften Sie zurücksetzen möchten.
4. Klicken Sie auf die Schaltfläche "Standard", um alle Einstellungen zurückzusetzen.

## 12.6 Trace- und Logikanalysatorfunktion nutzen

### Vorwort

#### Zweck der Dokumentation

In dieser Dokumentation werden die Diagnosemöglichkeiten mit der Trace- und Logikanalysatorfunktion beschrieben. Abhängig vom eingesetzten Gerät können sich die Aufzeichnungsmöglichkeiten unterscheiden.

#### Erforderliche Grundkenntnisse

Zum Verständnis der Dokumentation sind die folgenden Kenntnisse erforderlich:

- Allgemeine Kenntnisse auf dem Gebiet der Automatisierungstechnik
- Kenntnisse über die Verwendung von Windows-Computern
- S7-1200/1500 CPUs
  - Kenntnisse im Umgang mit dem Industrieautomatisierungssystem SIMATIC
  - Kenntnisse im Umgang mit STEP 7  $\geq$  V12.0
- SINAMICS G120
  - Kenntnisse im Umgang mit dem Antrieb

#### Gültigkeitsbereich der Dokumentation

Diese Dokumentation gilt für alle Produkte der Produktfamilie S7-1200, S7-1500 und SINAMICS G120.

#### Konventionen

Die vorliegende Dokumentation enthält Abbildungen zu den beschriebenen Geräten. Die Abbildungen können vom gelieferten Gerät in Einzelheiten abweichen.

Beachten Sie auch die folgendermaßen gekennzeichneten Hinweise:

---

#### Hinweis

Ein Hinweis enthält wichtige Informationen zum in der Dokumentation beschriebenen Produkt, zur Handhabung des Produkts oder zu dem Teil der Dokumentation, auf den besonders aufmerksam gemacht werden soll.

---

## Weitere Unterstützung

- Das Angebot an technischer Dokumentation für die einzelnen SIMATIC Produkte und Systeme finden Sie im Internet (<http://www.siemens.com/simatic-tech-doku-portal>).
- Den Online-Katalog und das Online-Bestellsystem finden Sie im Internet (<http://mall.automation.siemens.com>).

## 12.6.1 Beschreibung

### 12.6.1.1 Unterstützte Hardware

Folgende Geräte (Seite 4322) unterstützen die Trace- und Logikanalysatorfunktion:

- SIMATIC S7-1200 CPUs (ab Firmwarestand V4.x)
- SIMATIC S7-1500 CPUs
- SINAMICS G120

### 12.6.1.2 Messwertaufzeichnung mit der Tracefunktion

## Einführung

Die Trace- und Logikanalysatorfunktion rufen Sie im Projektnavigator im Ordner des Geräts unter dem Namen "Traces" (Seite 4299) auf.

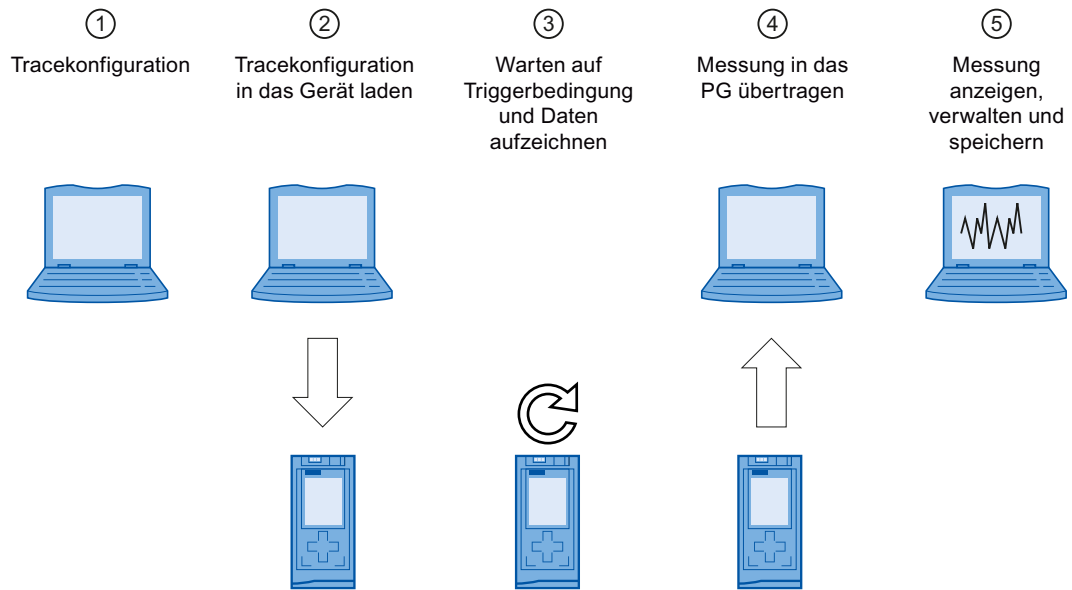
Mit der Trace- und Logikanalysatorfunktion zeichnen Sie Variablen eines Geräts auf und werten die Aufzeichnungen aus. Variablen sind z. B. Antriebsparameter oder System- und Anwendervariablen einer CPU. Die maximale Aufzeichnungsdauer ist durch die Speichergröße begrenzt. Wie viel Speicher für die Aufzeichnung zur Verfügung steht hängt von der verwendeten Hardware ab.

Die Aufzeichnungen werden auf dem Gerät gespeichert und können bei Bedarf mit dem Projektierungssystem (ES) ausgelesen und dauerhaft gespeichert werden. Somit eignet sich die Trace- und Logikanalysatorfunktion zum Beobachten hochdynamischer Vorgänge. Die aufgezeichneten Werte werden bei erneuter Aktivierung des "Traces im Gerät" überschrieben.

Abhängig vom eingesetzten Gerät (Seite 4322) können sich die Aufzeichnungsmöglichkeiten unterscheiden.

Im Kapitel Bedienen finden Sie einen Schnelleinstieg (Seite 4309) für das Arbeiten mit der Trace- und Logikanalysatorfunktion.

Das folgende Bild zeigt die Funktionsweise von "Traces":



#### ① Tracekonfiguration auf dem Programmiergerät (PG) im TIA-Portal

In der Tracekonfiguration legen Sie die aufzuzeichnenden Signale, die Dauer der Aufzeichnung und die Triggerbedingung fest. Die Tracekonfiguration ist geräteabhängig und beim jeweiligen Gerät (Seite 4322) beschrieben.

#### ② Übertragen der Tracekonfiguration vom PG auf das Gerät

Die vollständige Tracekonfiguration übertragen (Seite 4316) Sie bei bestehender Online-Verbindung in das Gerät.

#### ③ Warten auf die Aufzeichnung

Ist die Tracekonfiguration im Gerät aktiviert (Seite 4316), dann erfolgt die Aufzeichnung unabhängig vom PG. Sobald die Triggerbedingung erfüllt ist, wird die Aufzeichnung gestartet.

#### ④ Übertragen der Messung vom Gerät auf das PG

Das Speichern der Messung im Projekt (Seite 4318) legt die Messung im geöffneten Projekt des TIA-Portals ab.

#### ⑤ Messung auswerten, verwalten und speichern

Für die Auswertung der Messung stehen vielfältige Möglichkeiten im Kurvendiagramm und in der Signaltabelle (Seite 4317) zur Verfügung. Verschiedene Darstellungsformen sind möglich, z. B. für Binärsignale eine Bit-Darstellung.


Messungen können auch als Datei exportiert und importiert werden.

Mit dem Speichern des Projekts (Seite 4318) im TIA-Portal werden auch die in das Projekt übertragenen Messungen gespeichert.

### 12.6.1.3 Tracekonfiguration, Aufzeichnung und Messung

Dieses Kapitel erklärt die Bedeutung der Begriffe Tracekonfiguration, Aufzeichnung und Messung.

## Tracekonfiguration

In der Tracekonfiguration  können Sie folgende Einstellungen vornehmen:


- Aufzuzeichnende Signale
- Triggerbedingung
- Dauer der Aufzeichnung

## Aufzeichnung

Eine Aufzeichnung findet im Gerät statt. Zu jeder Tracekonfiguration im Gerät gibt es nur eine Aufzeichnung. Wenn eine Aufzeichnung neu gestartet wird, dann wird die alte Aufzeichnung überschrieben.


Eine Aufzeichnung im Gerät ist nicht remanent (geht nach Aus-/Einschalten des Geräts verloren) und kann als Messung dauerhaft im Projekt gespeichert werden.

## Traces im Gerät

Ein Trace im Gerät  besteht aus einer Tracekonfiguration und optional aus einer Aufzeichnung. Die maximale Anzahl Traces im Gerät ist geräteabhängig.

Der Status eines Trace im Gerät kann online betrachtet werden.

## Messung

Eine Messung  besteht immer aus einer Tracekonfiguration mit einer zugehörigen Aufzeichnung. Wenn ein Trace im Gerät eine Aufzeichnung enthält, dann kann dieser im Projekt als Messung gespeichert werden.

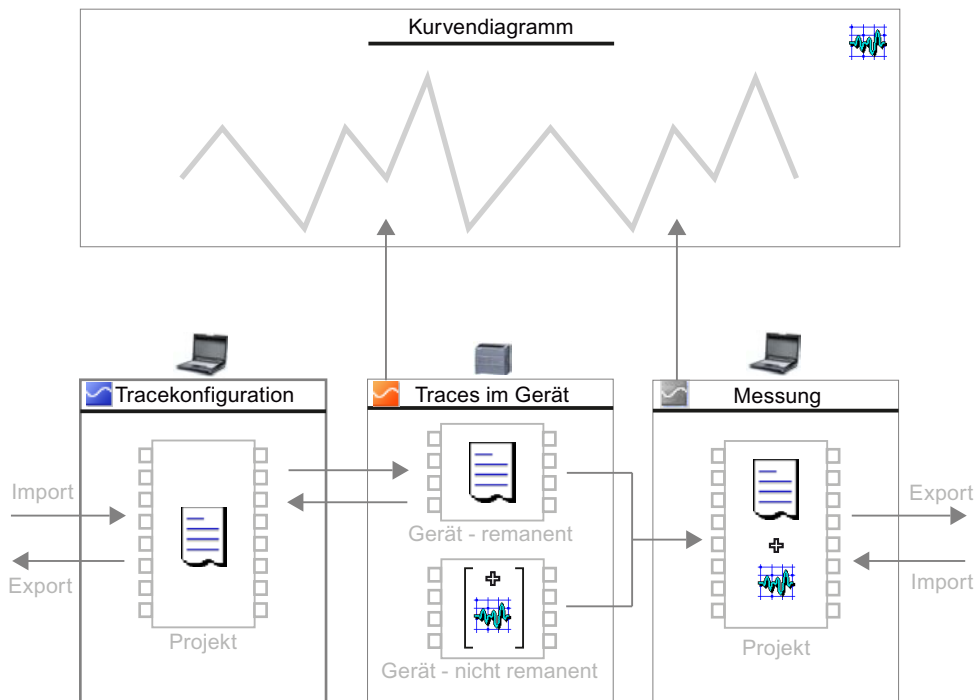
Die Aufzeichnung einer Messung kann offline betrachtet werden.

### 12.6.1.4 Datenablage

Die Traceverwaltung und das Kurvendiagramm ermöglichen unter anderem das Übertragen der Tracekonfiguration und das Betrachten der Aufzeichnung.

Das folgende Bild zeigt schematisch die Datenablage:





#### Hinweis

##### Tracekonfiguration und Messung speichern

Die Tracekonfiguration und Messung speichern Sie mit dem Projekt im TIA-Portal.

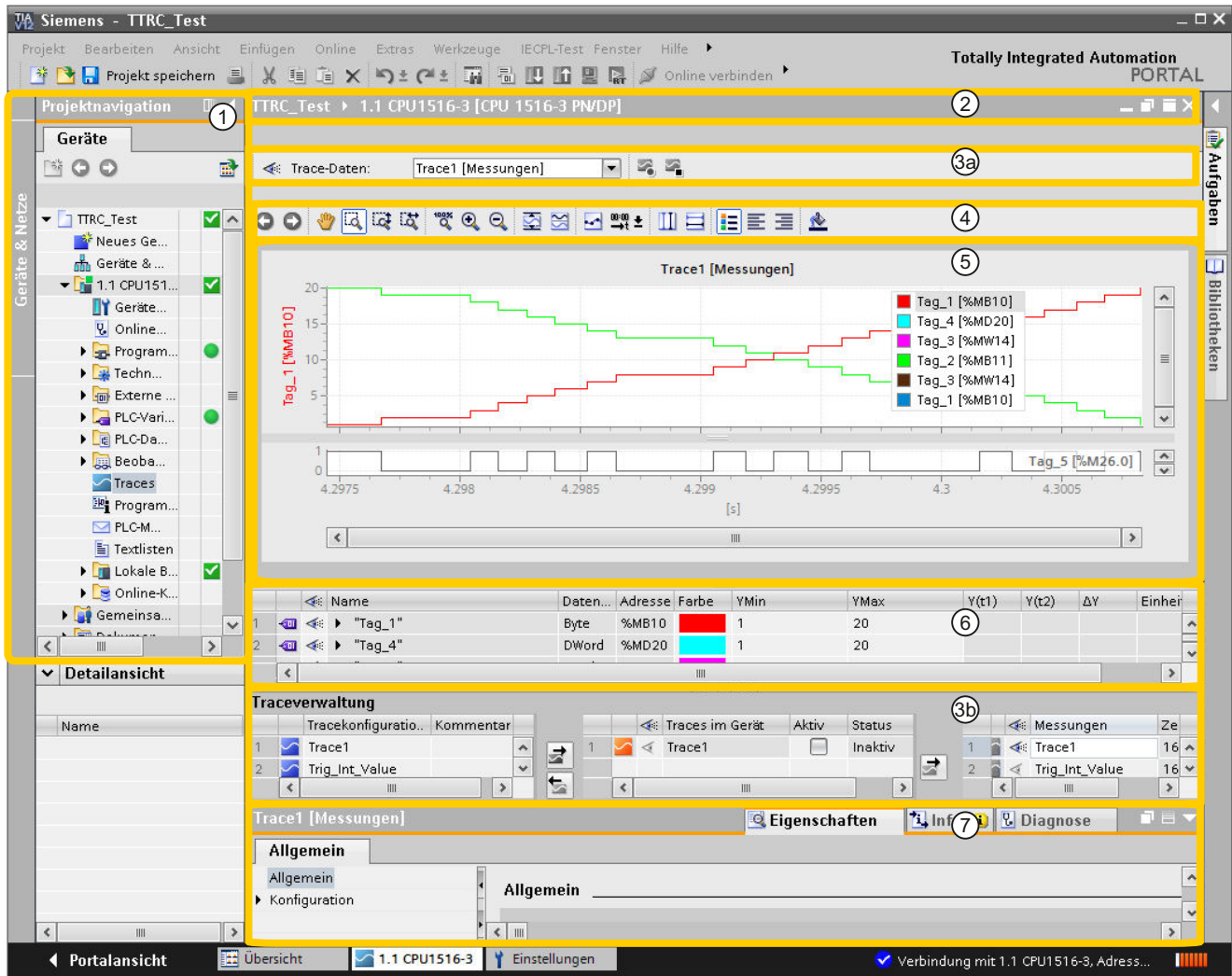
Wenn Sie das Projekt ohne Speichern schließen, werden die Tracekonfigurationen und die in das Projekt übertragenen Messungen verworfen. Der Trace-Editor kann bis zum Schließen des Projekts ohne Datenverlust geschlossen und wieder geöffnet werden.

## 12.6.2 Software-Oberfläche

### Anzeigebereiche

Die Benutzeroberfläche der Trace- und Logikanalysatorfunktion setzt sich aus mehreren Bereichen zusammen.


Das folgende Bild zeigt exemplarisch die Aufteilung der Oberfläche im TIA-Portal:




|    |   |
|----|---|
| 1  | Projektnavigator  |
| 2  | Arbeitsbereich<br>Titelleiste des Arbeitsbereichs<br>Zeigt an, zu welchem Gerät die aktuelle Darstellung gehört.  |
| 3a | Funktionsleiste der Traceverwaltung (Seite 4304)<br>Auswahl von Aufzeichnungen zur Anzeige im Kurvendiagramm und Schaltflächen zum Aktivieren/Deaktivieren von Traces im Gerät. |
| 4  | Funktionsleiste des Kurvendiagramms (Seite 4301)<br>Werkzeuge zum Bearbeiten der Messung und zur Anpassung der Darstellung im Kurvendiagramm                                    |
| 5  | Kurvendiagramm (Seite 4301)<br>Anzeige der aufgezeichneten Werte  |
| 6  | Signaltabelle (Seite 4303)<br>Signale der Messung   |

|    |  |  |
|----|--|--|
| 3b |  | Traceverwaltung (Seite 4304)<br>Anlegen neuer Tracekonfigurationen und die Verwaltung der Messungen.                               |
|    | Gerätespezifischer Bereich (Siehe Geräte (Seite 4322)) |  |
| 7  |  | Bereich Eigenschaften im Inspektorfenster<br>Konfigurationsdialoge für die Aufzeichnungsdauer, Triggerbedingung und Signalauswahl. |

### 12.6.2.1 Projektnavigators

Wenn ein Gerät die Trace- und Logikanalysatorfunktion unterstützt, wird in der Projektnavigation unter dem Gerät  "Traces" zur Auswahl angeboten.

Mit Doppelklick auf  "Traces" öffnen Sie den Trace-Editor.

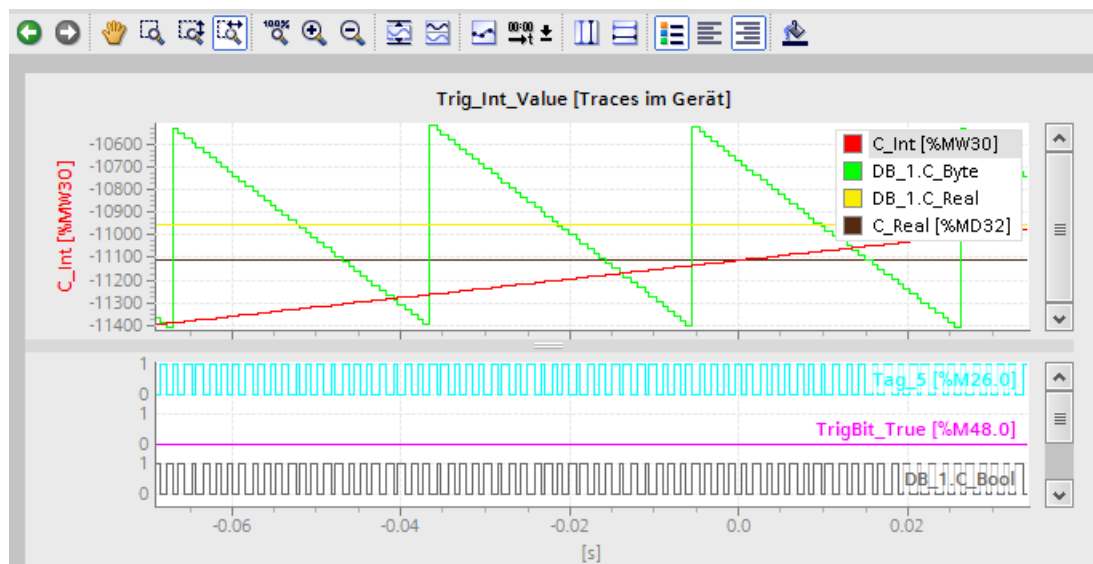
### 12.6.2.2 Arbeitsbereich

#### Oberfläche - Kurvendiagramm

Das Kurvendiagramm zeigt die ausgewählten Signale einer Aufzeichnung an. Bits werden im unteren Diagramm als Bitspur dargestellt. Die Darstellung der Signale passen Sie in der Signaltabelle (Seite 4303) und mit Hilfe der Funktionsleiste des Kurvendiagramms an.

#### Einstellmöglichkeiten und Anzeigen im Kurvendiagramm

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:



Die Skala im Diagramm gilt für das angewählte (grau hinterlegte) Signal in der Legende. Die Legende kann mit der Maus seitlich verschoben werden.

## Kontextmenübefehle












Die folgende Tabelle zeigt die Kontextmenübefehle im Kurvendiagramm:

| Kontextmenübefehl                   | Beschreibung  |
|-------------------------------------|---|
| "Diagramm als Bild speichern unter" | Exportiert die aktuelle Anzeige als Bitmap.   |
| "Bild in Zwischenablage kopieren"   | Kopiert die aktuelle Anzeige in die Zwischenablage.                                       |
| "Mess-Cursor zentrieren"            | Positioniert die aktivierten Mess-Cursor an einer zentralen Stelle der aktuellen Anzeige. |

## Funktionsleiste des Kurvendiagramms

Über Schaltflächen stehen Werkzeuge zur Anpassung der Darstellung zur Verfügung.

Die folgende Tabelle zeigt die Funktionen der Schaltflächen:

| Symbol  | Funktion               | Beschreibung  |
|---|------------------------|---|
|    | Zoom rückgängig        | Rückgängig machen der zuletzt ausgeführten Zoomfunktion.<br>Wenn mehrere Zoomfunktionen ausgeführt wurden, können diese schrittweise rückgängig gemacht werden.   |
|  | Zoom wiederholen       | Wiederholen der zuletzt rückgängig gemachten Zoomfunktion. Wenn mehrere Zoomfunktionen rückgängig gemacht wurden, können diese schrittweise wiederholt werden.    |
|  | Ansicht verschieben    | Verschieben der Anzeige mit gedrückter Maustaste.   |
|  | Zoomauswahl            | Auswahl eines beliebigen Bereichs mit gedrückter Maustaste. Die Anzeige wird auf die Bereichsauswahl skaliert.  |
|  | Zoomauswahl vertikal   | Auswahl eines vertikalen Bereichs mit gedrückter Maustaste. Die Anzeige wird auf die Bereichsauswahl skaliert.  |
|  | Zoomauswahl horizontal | Auswahl eines horizontalen Bereichs mit gedrückter Maustaste. Die Anzeige wird auf die Bereichsauswahl skaliert.  |
|  | Alles anzeigen         | Skalierung der Anzeige, dass der gesamte Zeitbereich und alle Werte angezeigt werden.   |
|  | Zoom in                | Vergrößern der Anzeige. Die Bereiche der Zeitachse und Werteachse werden mit jedem Klick auf die Schaltfläche verkleinert. Die Kurven werden größer dargestellt.  |
|  | Zoom out               | Verkleinern der Anzeige. Die Bereiche der Zeitachse und Werteachse werden mit jedem Klick auf die Schaltfläche vergrößert. Die Kurven werden kleiner dargestellt. |
|  | Automatisch skalieren  | Skalierung der Anzeige, dass für den aktuell dargestellten Zeitbereich alle Werte angezeigt werden.   |
|  | In Spuren anordnen     | Signale untereinander ohne Überschneidungen anordnen.   |

| Symbol | Funktion                           | Beschreibung   |
|--------|------------------------------------|--|
|        | Messpunkte anzeigen                | Die Messpunkte werden als kleine Kreise auf den Kurven angezeigt.  |
|        | Einheitenumschaltung der Zeitachse | Umschalten der Einheit zwischen Zeit und Zyklen.   |
|        | Vertikale Mess-Cursor anzeigen     | Anzeige der vertikalen Mess-Cursor.<br>Die vertikale Position der beiden Mess-Cursor kann mit der Maus verschoben werden. In der Signaltabelle werden entsprechend der Position die zugehörigen Messwerte und die Differenz der Mess-Cursor angezeigt. |
|        | Horizontale Mess-Cursor anzeigen   | Einblenden verschiebbarer horizontaler Mess-Cursor.  |
|        | Legende anzeigen                   | Einblenden der Legende im Kurvendiagramm.  |
|        | Legende linksbündig anzeigen       | Anzeige der Legende auf der linken Seite des Kurvendiagramms.  |
|        | Legende rechtsbündig anzeigen      | Anzeige der Legende auf der rechten Seite des Kurvendiagramms.   |
|        | Hintergrundfarbe wechseln          | Umschalten zwischen verschiedenen Hintergrundfarben.   |

## Oberfläche - Signaltabelle

Die Signaltabelle listet die Signale der ausgewählten Messung auf und bietet Einstellmöglichkeiten für einige Eigenschaften. Wenn Aufzeichnungsdaten von "Traces im Gerät" angezeigt werden und die Einstellungen in der Signaltabelle verändert werden, bleiben diese Einstellungen nur so lang erhalten, bis in den Offline-Modus gewechselt wird.




## Einstellmöglichkeiten und Anzeigen in der Signaltabelle

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

|   | Name           | Datentyp | Adresse | Farbe | YMin   | YMax   | Y(t1)  | Y(t2)  | ΔY  | Einheit | Kommentar |
|---|----------------|----------|---------|-------|--------|--------|--------|--------|-----|---------|-----------|
| 1 | "C_Int"        | Int      | %MW30   |       | -11410 | -10513 | -11257 | -11113 | 144 |         |           |
| 2 | "Tag_5"        | Bool     | %M26.0  |       | 0      | 1      | 1      | 1      | 0   |         |           |
| 3 | "TrigBit_True" | Bool     | %M48.0  |       | 0      | 1      | 0      | 0      | 0   |         |           |

Die folgende Tabelle zeigt die Einstellungen und Anzeigen der aufgezeichneten Signale:

| Spalte | Beschreibung   |
|--------|--|
|        | Statische Anzeige des Signal-Symbols   |
|        | Auswahl für die Anzeige im Kurvendiagramm  |
| "Name" | Anzeige des Signalnamen<br>Ein Klick auf den Namen eines angezeigten Signals aktualisiert die Skala im Kurvendiagramm. |

| Spalte  | Beschreibung   |
|---|--|
|  | <p>Bitauswahl öffnen</p> <p>Für folgende Datentypen können auch einzelne Bits zur Anzeige als Bitspur im unteren Kurvendiagramm ausgewählt werden:</p> <ul style="list-style-type: none"> <li>• Byte, Word, DWord, LWord</li> <li>• Sint, USInt, Int, UInt, DInt, UDInt, LInt, ULInt</li> </ul> <p>Beispiel einer geöffneten Bitauswahl für den Datentyp DWord:</p>  <p>Wählen Sie durch Klick auf das Symbol  das jeweilige Bit für die Anzeige ab oder an.</p> |
| "Datentyp"  | Anzeige des Datentyps  |
| "Adresse"   | Anzeige der Adresse (nicht bei symbolischen Variablen)   |
| "Farbe"   | Anzeige und Einstellmöglichkeit für die Farbe des Signals  |
| "YMin"  | Eingabe des minimalen Werts für die Skalierung des Signals   |
| "YMax"  | Eingabe des maximalen Werts für die Skalierung des Signals   |
| "Y(t1)"   | Anzeige des Werts an der Position des ersten Mess-Cursors  |
| "Y(t2)"   | Anzeige des Werts an der Position des zweiten Mess-Cursors   |
| "ΔY"  | Anzeige der Wertdifferenz zwischen dem ersten und dem zweiten Mess-Cursor  |
| "Einheit"   | Anzeige der Einheit (z. B. bei Technologieobjekten)  |
| "Kommentar"   | Anzeige und Eingabemöglichkeit eines Kommentars zu dem Signal  |

### Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle der Signaltabelle:

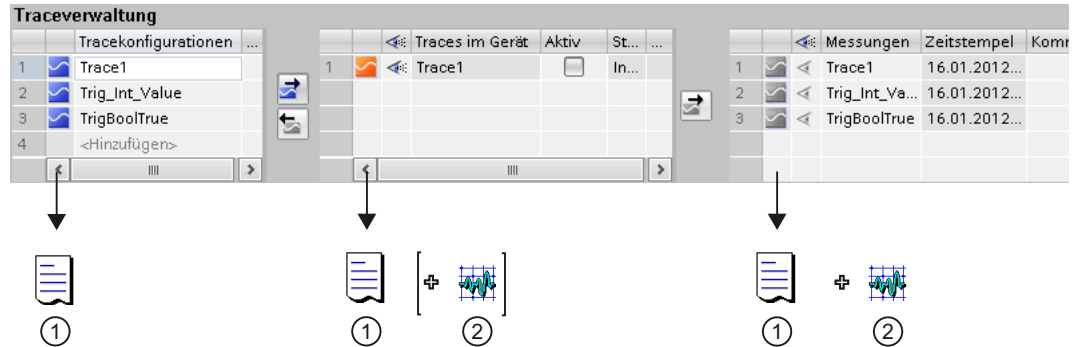
| Kontextmenübefehl        | Beschreibung   |
|--------------------------|--|
| "Ausschneiden"           | Nicht anwählbar  |
| "Kopieren"               | Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.                          |
| "Einfügen"               | Nicht anwählbar  |
| "Signal ein-/ausblenden" | Blendet das Signal im Kurvendiagramm ein- oder aus.  |
| "In den Vordergrund"     | Stellt das selektierte Signal im Vordergrund dar. Die anderen Signale werden überzeichnet. |

### Oberfläche - Traceverwaltung

Die Traceverwaltung ermöglicht die Konfiguration und Verwaltung von Tracekonfigurationen.

## Einstellmöglichkeiten und Anzeigen in der Traceverwaltung

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

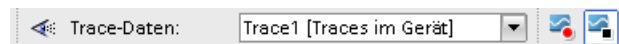


- ① Tracekonfiguration
- ② Aufzeichnung (aufgezeichnete Werte)

## Funktionsleiste der Traceverwaltung

Über eine Klappliste und Schaltflächen kann eine Aufzeichnung zur Anzeige ausgewählt und aktiviert oder deaktiviert werden.

Das folgende Bild zeigt exemplarisch die Funktionsleiste der Traceverwaltung





Die folgende Tabelle beschreibt die Funktionen der Schaltflächen:

| Symbol | Beschreibung   |
|--------|--|
|        | Auswahl für die Anzeige im Kurvendiagramm<br>Ein Klick auf das Symbol blendet die Aufzeichnung im Kurvendiagramm ein bzw. aus. |
|        | Auswahl eines "Traces im Gerät" oder einer "Messung" für die Anzeige im Kurvendiagramm   |
|        | Aktivieren der Aufzeichnung<br>Die Schaltfläche steht bei "Traces im Gerät" zur Verfügung.                                     |
|        | Deaktivieren der Aufzeichnung<br>Die Schaltfläche steht bei "Traces im Gerät" zur Verfügung.                                   |

## Tabelle "Tracekonfigurationen"

Die Tabelle enthält alle Tracekonfigurationen des Projekts, die in das Gerät übertragen werden können. Jede Tracekonfiguration enthält die aufzuzeichnenden Variablen und die erforderlichen Aufzeichnungseinstellungen. Die Daten werden im Offline- und Online-Modus angezeigt.

Die folgende Tabelle zeigt die Einstellungen und Anzeigen:

| Spalte                 | Symbol   | Beschreibung   |
|------------------------|--|--|
| -                      | <br> | Anzeige des Symbols für die Tracekonfiguration<br>Fehlerhafte Tracekonfigurationen werden mit einem roten Zusatzsymbol dargestellt und können nicht in das Gerät übertragen werden. Überprüfen Sie die gerätespezifische Konfiguration im Inspektorfenster.<br>Ein Doppelklick auf das Symbol bringt das Inspektorfenster mit den gerätespezifischen Eigenschaften in den Vordergrund. |
| "Tracekonfigurationen" | -  | Name der Tracekonfiguration<br>Das Ändern der Bezeichnung ist durch Doppelklick auf das Feld möglich.  |
| "Kommentar"            | -  | Eingabefeld für einen Kommentar  |

### Kontextmenübefehle




Die folgende Tabelle zeigt die Kontextmenübefehle der Tabelle "Tracekonfigurationen":

| Kontextmenübefehl                        | Beschreibung   |
|--|--|
| "Ausschneiden"                           | Entfernt die selektierten Zeilen und legt diese in die Zwischenablage.   |
| "Kopieren"                               | Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.  |
| "Einfügen"                               | Fügt den Inhalt der Zwischenablage ab der selektierten Zeile ein. Der bestehende Inhalt wird überschrieben.  |
| "Löschen"                                | Löscht die selektierten Zeilen aus der Tabelle.  |
| "Umbenennen"                             | Schaltet die selektierte Zelle in den Editier-Modus.   |
| "Tracekonfiguration in Gerät übertragen" | Überträgt die in der Tabelle "Tracekonfigurationen" ausgewählte Tracekonfiguration in das Gerät.<br>Die Schaltfläche ist aktiv, wenn alle erforderlichen Einstellungen in der gerätespezifischen Konfiguration im Inspektorfenster korrekt konfiguriert sind und eine Online-Verbindung besteht. |
| "Tracekonfiguration exportieren"         | Exportiert eine Tracekonfiguration als Datei mit der Dateierweiterung "*.ttcfg".   |
| "Tracekonfiguration importieren"         | Importiert eine Tracekonfiguration aus einer Datei.  |
| "Eigenschaften"                          | Bringt das Inspektorfenster mit den gerätespezifischen Eigenschaften in den Vordergrund.   |

### Tabelle "Traces im Gerät"

Die Tabelle enthält die bereits in das Gerät übertragenen Tracekonfigurationen und deren aktuellen Betriebszustand. Die Daten werden im Online-Modus angezeigt.

Die folgende Tabelle zeigt die Einstellungen und Anzeigen:

| Spalte  | Symbol  | Beschreibung  |
|---|---|---|
| -   |  | Statische Anzeige des Online-Symbols<br>Ein Doppelklick auf das Symbol bringt das Inspektorfenster mit den gerätespezifischen Eigenschaften in den Vordergrund. |
|  |  | Auswahl für die Anzeige im Kurvendiagramm<br>Ein Klick auf das Symbol blendet die Aufzeichnung im Kurvendiagramm ein bzw. aus.                                  |



| Spalte            | Symbol   | Beschreibung   |
|-------------------|--|--|
| "Traces im Gerät" | -  | Name der Tracekonfiguration  |
| "Aktiv"           | <input type="checkbox"/> <input checked="" type="checkbox"/> | Aktivieren der Aufzeichnung und Rückmeldung des Zustands<br>Das Optionskästchen wird am Ende der Aufzeichnung zurückgesetzt.   |
| "Status"          | -  | Zustandsanzeige der Tracekonfiguration<br>Folgende Zustände sind möglich: <ul style="list-style-type: none"> <li>• "Wird installiert"</li> <li>• "Warten auf Trigger"</li> <li>• "Aufzeichnung"</li> <li>• "Aufzeichnung abgeschlossen"</li> <li>• "Abgebrochen"</li> <li>• "Fehlerzustand"</li> </ul> |
| "Kommentar"       | -  | Anzeige des Kommentars der Tracekonfiguration  |

### Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle der Tabelle "Traces im Gerät":




| Kontextmenübefehl                    | Beschreibung   |
|--------------------------------------|--|
| "Ausschneiden"                       | Entfernt die selektierten Zeilen und legt diese in die Zwischenablage.                   |
| "Kopieren"                           | Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.                        |
| "Einfügen"                           | Nicht anwählbar  |
| "Löschen"                            | Löscht die selektierten Zeilen aus der Tabelle.  |
| "Umbenennen"                         | Nicht anwählbar  |
| "Zu Tracekonfigurationen hinzufügen" | Tracekonfiguration vom Gerät in die Tabelle Tracekonfigurationen übertragen              |
| "Zu Messungen hinzufügen"            | Fügt den ausgewählten "Trace im Gerät" in der Tabelle "Messungen" hinzu.                 |
| "Aktivieren"                         | Aktiviert die Aufzeichnung und zeigt den Zustand in der Tabelle an                       |
| "Im Kurvendiagramm ein-/ausblenden"  | Blendet die Aufzeichnung im Kurvendiagramm ein bzw. aus.                                 |
| "Eigenschaften"                      | Bringt das Inspektorfenster mit den gerätespezifischen Eigenschaften in den Vordergrund. |

### Tabelle "Messungen"

Die Tabelle enthält die Tracekonfigurationen mit Aufzeichnungen. Die Daten werden im Offline- und Online-Modus angezeigt.

Messungen können auch exportiert und importiert werden, siehe Messungen exportieren und importieren (Seite 4318).

Die folgende Tabelle zeigt die Einstellungen und Anzeigen:

| Spalte  | Symbol  | Beschreibung   |
|---|---|--|
| -   |  | Statische Anzeige des Symbols für eine Messung<br>Ein Doppelklick auf das Symbol bringt das Inspektorfenster mit den gerätespezifischen Eigenschaften in den Vordergrund.  |
|  |  | Auswahl für die Anzeige im Kurvendiagramm<br>Ein Klick auf das Symbol blendet die Aufzeichnung im Kurvendiagramm ein bzw. aus.   |
| "Messungen"   | -   | Name der Messung<br>Das Ändern der Bezeichnung ist durch Doppelklick auf das Feld möglich.   |
| "Zeitstempel"   | -   | Bedeutung vom Zeitstempel ist geräteabhängig<br>Beispiele: <ul style="list-style-type: none"> <li>• SIMATIC S7-1200/1500 CPUs<br/>Mit dem Start der Aufzeichnung wird die absolute Uhrzeit der Steuerung eingetragen.</li> <li>• SINAMICS G120<br/>Mit dem Übertragen der Tracekonfiguration in das Gerät wird der Übertragungszeitpunkt eingetragen.</li> </ul> |
| "Kommentar"   | -   | Eingabefeld für einen Kommentar  |




**Kontextmenübefehle**

Die folgende Tabelle zeigt die Kontextmenübefehle der Tabelle "Messungen":

| Kontextmenübefehl                    | Beschreibung   |
|--------------------------------------|--|
| "Ausschneiden"                       | Entfernt die selektierten Zeilen und legt diese in die Zwischenablage.                   |
| "Kopieren"                           | Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.                        |
| "Einfügen"                           | Nicht anwählbar  |
| "Löschen"                            | Löscht die selektierten Zeilen aus der Tabelle.  |
| "Umbenennen"                         | Schaltet die selektierte Zelle in den Editier-Modus.                                     |
| "Zu Tracekonfigurationen hinzufügen" | Tracekonfiguration der Messung in die Tabelle Tracekonfigurationen übertragen            |
| "Im Kurvendiagramm ein-/ausblenden"  | Blendet die Aufzeichnung im Kurvendiagramm ein bzw. aus.                                 |
| "Messung exportieren"                | Exportiert eine Messung als Datei mit der Dateierweiterung "*.ttrec" oder "*.csv".       |
| "Messung importieren"                | Importiert eine Messung aus einer Datei mit der Dateierweiterung "*.ttrec".              |
| "Eigenschaften"                      | Bringt das Inspektorfenster mit den gerätespezifischen Eigenschaften in den Vordergrund. |

## Schaltflächen

Die folgende Tabelle zeigt die über Schaltflächen verfügbaren Funktionen:

| Symbol  | Beschreibung   |
|---|--|
|  | Schaltfläche zum Übertragen einer ausgewählten Tracekonfiguration in das Gerät<br>Die ausgewählte Tracekonfiguration in der Tabelle "Tracekonfigurationen" wird in das Gerät übertragen.<br>Die Schaltfläche ist aktiv, wenn alle erforderlichen Einstellungen in der gerätespezifischen Konfiguration im Inspektorfenster korrekt konfiguriert sind und eine Online-Verbindung besteht. |
|  | Schaltfläche zum Übertragen einer ausgewählten Tracekonfiguration vom Gerät<br>Die in der Tabelle "Traces im Gerät" ausgewählte Tracekonfiguration wird vom Gerät in die Tabelle "Tracekonfigurationen" übertragen.  |
|  | Schaltfläche zum Übertragen einer ausgewählten Messung aus dem Gerät in das Projekt<br>Die Auswahl in der Tabelle "Traces im Gerät" wird in der Tabelle "Messungen" hinzugefügt.   |

### 12.6.2.3 Gerätespezifischer Bereich

Der gerätespezifische Bereich ermöglicht die Tracekonfiguration.

Die aufzuzeichnenden Signale und die Triggerbedingung werden hier festgelegt. Diese und weitere Konfigurationsmöglichkeiten sind bei den einzelnen Geräten im gerätespezifischen Teil (Seite 4322) beschrieben.

## 12.6.3 Bedienen

### 12.6.3.1 Schnelleinstieg

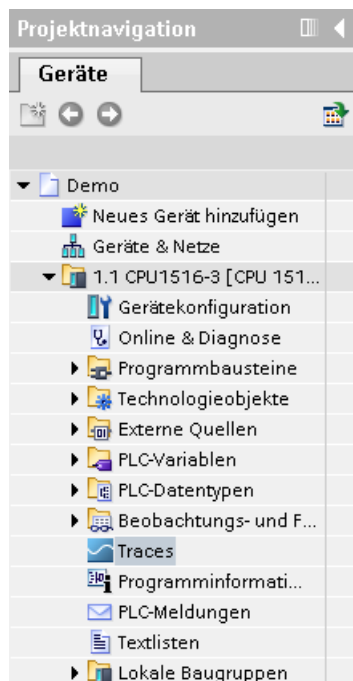
Diese Beschreibung zeigt beispielhaft für die S7-1500 CPU die Schritte für eine Aufzeichnung. Die gezeigten Einstellungen können sich geräteabhängig unterscheiden.

## Voraussetzung

Ein Gerät ist projektiert, das die Trace- und Logikanalysatorfunktion unterstützt.

### Trace-Editor aufrufen

Das folgende Bild zeigt den Projektnavigator mit "Traces" unter dem Gerät:

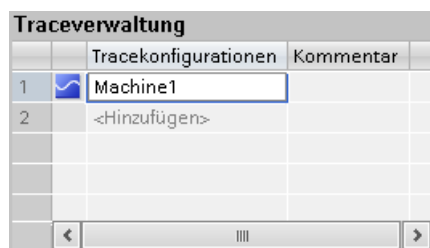


Vorgehen:

1. Öffnen Sie den Trace-Editor durch Doppelklick auf den Eintrag "Traces".

### Tracekonfiguration anlegen

Das folgende Bild zeigt die Tabelle "Tracekonfigurationen":

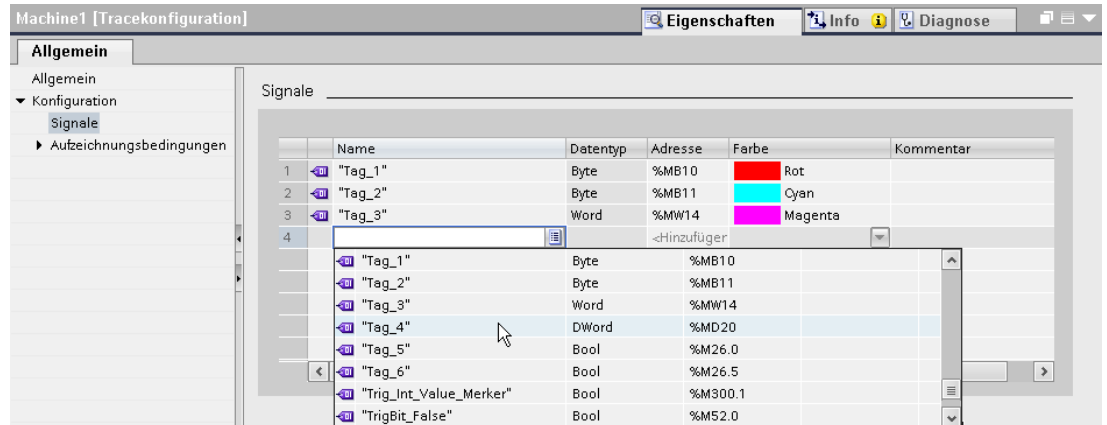


Vorgehen:

1. Geben Sie in der Tabelle "Tracekonfigurationen" den Namen der Tracekonfiguration ein.

## Signale auswählen

Das folgende Bild zeigt die Konfiguration der Signale:

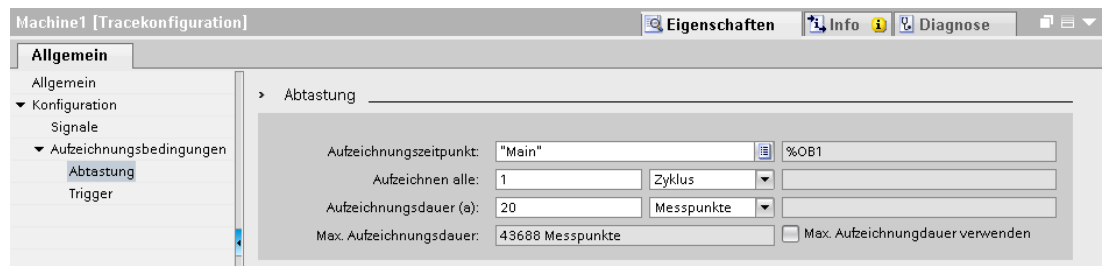


Vorgehen:

1. Wählen Sie im Bereich "Signale" des Inspektorfensters die aufzuzeichnenden Signale aus.

## Aufzeichnungstakt konfigurieren

Das folgende Bild zeigt die Konfiguration der Abtastung:

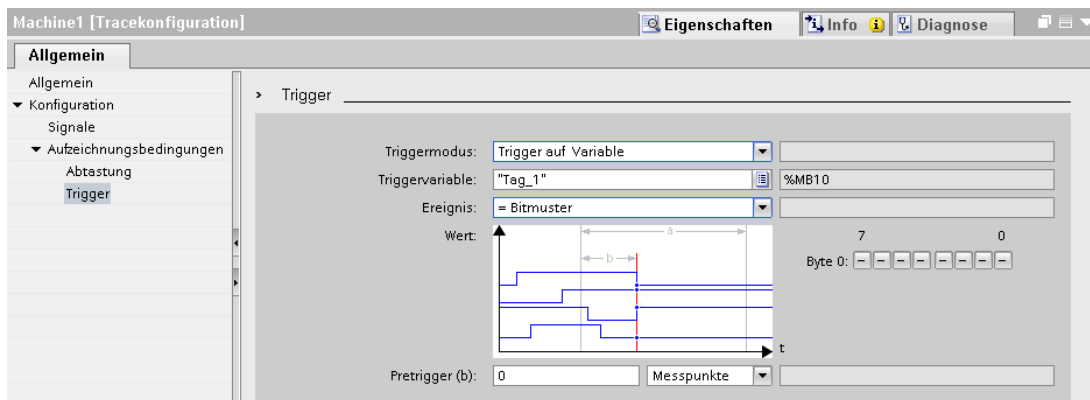


Vorgehen:

1. Konfigurieren Sie die Abtastung.

### Trigger konfigurieren

Das folgende Bild zeigt die Konfiguration des Triggers:



Vorgehen:




1. Konfigurieren Sie den Triggermodus und die Bedingung für den gewählten Trigger.

### Tracekonfiguration in das Gerät übertragen

Das folgende Bild zeigt die Traceverwaltung im Online-Modus:

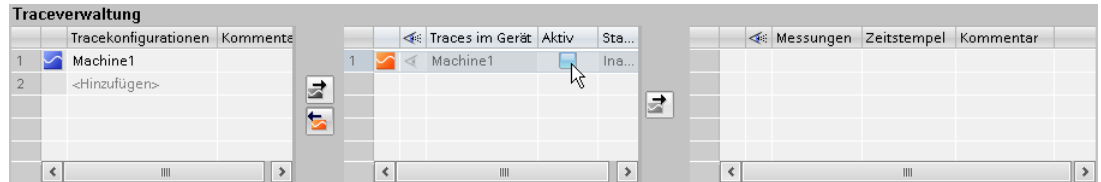


Vorgehen:

1. Prüfen Sie anhand des Symbols in der Tabelle "Tracekonfigurationen", ob die Konfiguration korrekt ist.  
Das Symbol  zeigt eine korrekte Tracekonfiguration an.  
Das Symbol  zeigt eine fehlerhafte Tracekonfiguration an.  
Nur korrekte Tracekonfigurationen können in das Gerät übertragen werden.
2. Stellen Sie eine Online-Verbindung zum Gerät her.
3. Wählen Sie die Tracekonfiguration in der Tabelle "Tracekonfigurationen" aus und übertragen Sie diese mit der Schaltfläche  in das Gerät.

### Aufzeichnung aktivieren

Das folgende Bild zeigt die Traceverwaltung mit einer Tracekonfiguration im Gerät:

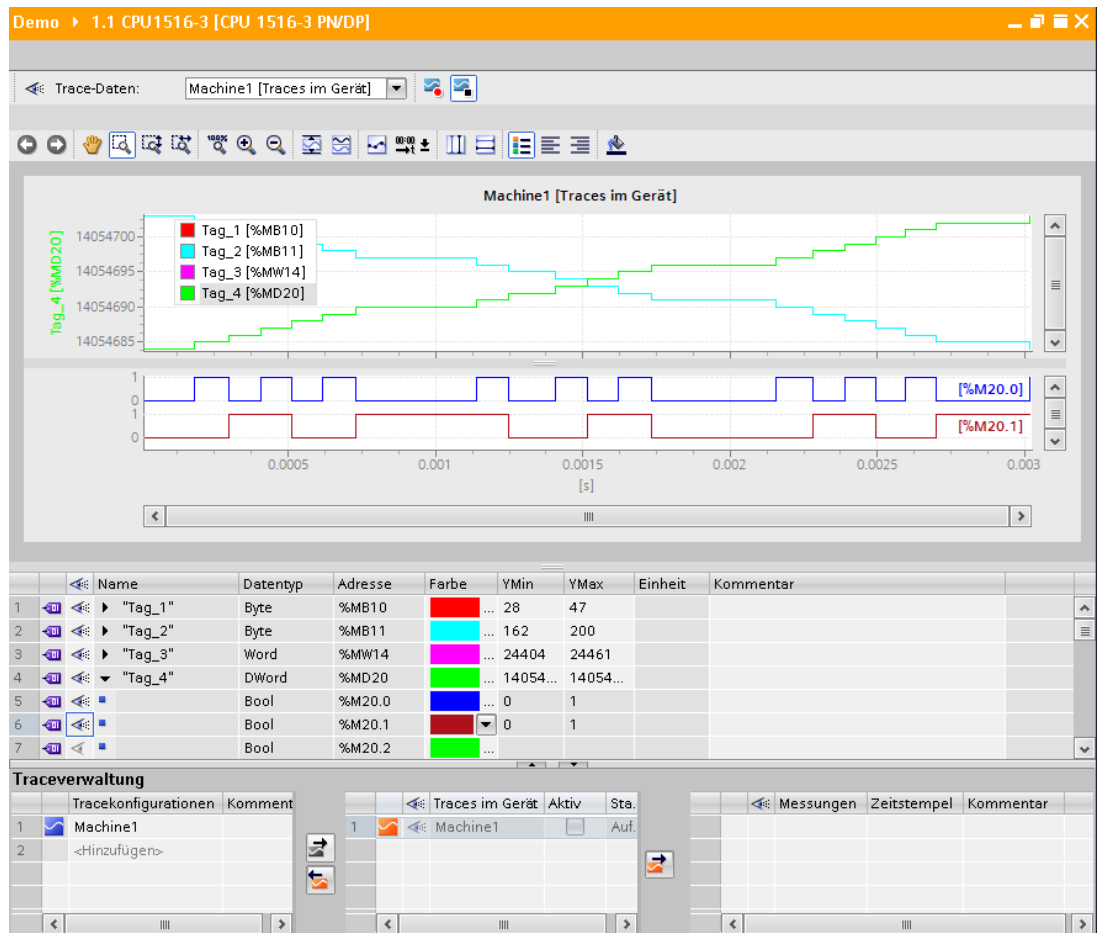


Vorgehen:




1. Klicken Sie in der Tabelle "Traces im Gerät" auf das Optionskästchen in der Spalte "Aktiv".

### Aufzeichnung anzeigen

Das folgende Bild zeigt das Kurvendiagramm mit der Aufzeichnung des "Traces im Gerät":

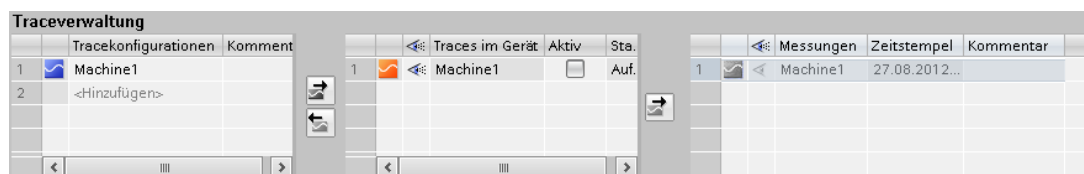


Vorgehen:


1. Warten Sie, bis in der Tabelle "Traces im Gerät" der Status "Aufzeichnung" oder "Aufzeichnung abgeschlossen" angezeigt wird.
2. Klicken Sie in der Tabelle "Traces im Gerät" auf das Symbol  um die Aufzeichnung im Kurvendiagramm anzuzeigen.
3. Klicken Sie in der Signaltabelle auf das Symbol  eines Signals.  
Es werden die einzelnen Bits des Signals für die Anzeige als Bitspur angeboten.
4. Wählen Sie in der Signaltabelle mit dem Symbol  die einzelnen Signale und Bits für die Darstellung an oder ab.

### Messung im Projekt speichern

Das folgende Bild zeigt die im Projekt gespeicherte Messung:



Vorgehen:

1. Wählen Sie die Messung in der Tabelle "Traces im Gerät" aus und übertragen Sie diese mit der Schaltfläche  in das Projekt.

### 12.6.3.2 Anwenden der Tracefunktion - Übersicht

#### Voraussetzung

Im TIA-Portal ist ein Gerät projektiert, das die Trace- und Logikanalysatorfunktion unterstützt und zu dem eine Online-Verbindung besteht.

#### Vorgehen

Die folgende Tabelle zeigt eine Handlungsübersicht mit den typischen Schritten beim Arbeiten mit der Trace- und Logikanalysatorfunktion.

| Schritt | Beschreibung  |
|---------|---|
| 1       | Trace-Editor aufrufen (Seite 4315)                      |
| 2       | Trace konfigurieren (Seite 4315)                        |
| 3       | Tracekonfiguration in das Gerät übertragen (Seite 4316) |
| 4       | Trace im Gerät aktivieren / deaktivieren (Seite 4316)   |
| 5       | Aufzeichnung beobachten (Seite 4317)                    |
| 6       | Messungen im Projekt speichern (Seite 4318)             |
| 7       | Aufzeichnung anzeigen (Seite 4317)                      |



### 12.6.3.3 Trace-Editor aufrufen

Pro Gerät kann im Projektnavigator jeweils ein Trace-Editor geöffnet werden.

Die folgende Anleitung beschreibt, wie Sie den Trace-Editor im Arbeitsbereich des TIA-Portals aufrufen.

#### Voraussetzung

Ein Gerät ist projektiert, das die Trace- und Logikanalysatorfunktion unterstützt.

#### Vorgehen

Um den Trace-Editor aufzurufen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Projektnavigator auf das Gerät.  
Der Ordner des Geräts wird geöffnet.
2. Doppelklicken Sie auf "Traces".

### 12.6.3.4 Traceverwaltung

#### Trace konfigurieren

#### Voraussetzung

Der Trace-Editor ist geöffnet.

#### Tracekonfiguration anlegen und löschen

Um eine neue Tracekonfiguration anzulegen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Traceverwaltung (Seite 4304) in der Tabelle "Tracekonfigurationen" auf den Eintrag "<Hinzufügen>".
2. Geben Sie den Namen für die Tracekonfiguration ein.  
Die neue Tracekonfiguration wird angelegt.

Um eine Tracekonfiguration zu löschen, gehen Sie folgendermaßen vor:

1. Wählen Sie durch Rechtsklicken auf eine Tracekonfiguration in der Tabelle "Tracekonfigurationen" den Kontextmenübefehl "Löschen".  
Die Tracekonfiguration wird gelöscht.

#### Trace konfigurieren

Bei der Konfiguration legen Sie die Aufzeichnungs- und Triggerbedingungen fest und wählen Sie die aufzuzeichnenden Signale aus.

Siehe Kapitel Konfigurieren unter dem jeweiligen Gerät (Seite 4322).

---

#### **Hinweis**

##### **Tracekonfiguration speichern**


Die Tracekonfiguration speichern Sie mit dem Projekt im TIA-Portal.

Wenn Sie das Projekt ohne Speichern schließen, wird die Konfiguration verworfen. Der Trace-Editor kann bis zum Schließen des Projekts ohne Datenverlust geschlossen und wieder geöffnet werden.

---



## **Tracekonfiguration in das Gerät übertragen**

### **Voraussetzung**

- Eine gültige Tracekonfiguration  befindet sich in der Tabelle "Tracekonfigurationen".
- Die maximale Anzahl "Traces im Gerät" ist noch nicht erreicht.
- Es besteht eine Online-Verbindung zum Gerät.

### **Vorgehen**

Um eine Tracekonfiguration in das Gerät zu übertragen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Tabelle "Tracekonfigurationen" eine Tracekonfiguration  aus.
2. Klicken Sie auf die Schaltfläche  zum Übertragen der Tracekonfiguration in das Gerät.

### **Ergebnis**

Die Tracekonfiguration wird in das Gerät übertragen und in der Tabelle "Traces im Gerät" angezeigt.

## **Trace im Gerät aktivieren / deaktivieren**

### **Voraussetzung**

Eine Tracekonfiguration befindet sich im Gerät und es besteht eine Online-Verbindung zum Gerät.

## Trace im Gerät aktivieren

Um die Aufzeichnung für einen Trace im Gerät zu aktivieren, gehen Sie folgendermaßen vor:

1. Aktivieren Sie das Optionskästchen der Spalte "Aktiv" in der Tabelle "Traces im Gerät". Der Trace im Gerät wird aktiviert und startet die Aufzeichnung entsprechend der konfigurierten Triggerbedingung. Die Triggerbedingung ist gerätespezifisch und im Kapitel Konfigurieren unter dem jeweiligen Gerät (Seite 4322) beschrieben.

---

### Hinweis

Wenn eine Aufzeichnung neu gestartet wird, gehen die bisher aufgezeichneten Werte verloren.

Um die aufgezeichneten Werte zu sichern speichern Sie die Messung im Projekt (Seite 4318) , bevor Sie die Aufzeichnung erneut aktivieren.

---

## Trace im Gerät deaktivieren

Um einen aktivierten Trace im Gerät zu deaktivieren, gehen Sie folgendermaßen vor:

1. Deaktivieren Sie das Optionskästchen der Spalte "Aktiv" in der Tabelle "Traces im Gerät". Der Trace im Gerät wird deaktiviert.


## Aufzeichnung anzeigen

### Voraussetzung

Eine Tracekonfiguration mit Aufzeichnung befindet sich im Gerät und es besteht eine Online-Verbindung zum Gerät oder es befindet sich ein Eintrag in der Tabelle "Messungen".

### Vorgehen

Um die Aufzeichnung anzuzeigen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Tabelle "Traces im Gerät" einen Eintrag aus.
2. Klicken Sie für den gewählten Eintrag in die Spalte .

Oder:

1. Wählen Sie in der Tabelle "Messungen" einen Eintrag aus.
2. Klicken Sie für den gewählten Eintrag in die Spalte .

## Ergebnis

Die Aufzeichnung wird im Kurvendiagramm und der Signaltabelle angezeigt.


## Messungen im Projekt speichern

### Voraussetzung

Eine Tracekonfiguration mit Aufzeichnung befindet sich im Gerät und es besteht eine Online-Verbindung zum Gerät.

### Vorgehen

Um eine Aufzeichnung im Projekt zu speichern, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Traceverwaltung in der Tabelle "Traces im Gerät" auf einen Eintrag.
2. Klicken Sie auf die Schaltfläche  zum Übertragen des selektierten Eintrags. Der selektierte Eintrag in der Tabelle "Traces im Gerät" wird in der Tabelle "Messungen" hinzugefügt.
3. Speichern Sie das Projekt im TIA-Portal.

## Messungen exportieren und importieren

### Voraussetzung

In der Tabelle "Messungen" befindet sich mindestens eine Messung.

### Messungen exportieren

Um eine Messung zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie durch Rechtsklicken auf den Eintrag in der Tabelle "Messungen" den Kontextmenübefehl "Messung exportieren".
2. Wählen Sie einen Ordner, Dateinamen und Dateityp zum Speichern der Messung.
3. Klicken Sie auf die Schaltfläche "Speichern".

### Messungen importieren

Um eine Messung zu importieren, gehen Sie folgendermaßen vor:

1. Wählen Sie durch Rechtsklicken in die Tabelle "Messungen" den Kontextmenübefehl "Messung importieren".
2. Wählen Sie die zu importierende Datei vom Dateityp "\*.ttrec" mit der Messung.
3. Klicken Sie auf die Schaltfläche "Öffnen".  
Die importierte Messung wird mit dem Dateinamen in der Tabelle "Messungen" angezeigt.


## Tracekonfiguration aus dem Gerät in das Projekt übertragen

### Voraussetzung

Eine Tracekonfiguration befindet sich im Gerät und es besteht eine Online-Verbindung zum Gerät.

### Vorgehen

Um eine Tracekonfiguration in das Projekt zu übertragen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Tabelle "Traces im Gerät" einen Eintrag.
2. Klicken Sie auf die Schaltfläche  zum Übertragen der Tracekonfiguration aus dem Gerät.

### Ergebnis

Die Konfiguration des "Trace im Gerät" wird in die Tabelle "Tracekonfigurationen" übernommen.

## Traces im Gerät löschen

### Voraussetzung

Ein Trace befindet sich im Gerät und es besteht eine Online-Verbindung zum Gerät.

### Trace im Gerät löschen

Um einen Trace im Gerät zu löschen, gehen Sie folgendermaßen vor:

1. Selektieren Sie eine oder mehrere zu löschende Zeilen in der Tabelle "Traces im Gerät".
2. Drücken Sie <Entf> um die "Traces im Gerät" zu löschen.

## 12.6.3.5 Signaltabelle


### Verwendung der Signaltabelle

Die Signaltabelle zeigt die aufgezeichneten Werte von einem Trace im Gerät oder einer Messung an. In der Tabelle können Sie einzelne Signale für die Anzeige ein- oder ausblenden und Eigenschaften für die Anzeige anpassen.

Für einige Datentypen können auch einzelne Bits ausgewählt und als Bitspur angezeigt werden.


Die folgende Handlungsanleitung beschreibt die Bedienung der Signaltabelle.

### Voraussetzung

- In den Tabellen "Traces im Gerät" oder "Messungen" ist ein Eintrag in der Spalte  angewählt.
- Für die Anzeige einzelner Bits als Bitspur:  
Mindestens ein aufgezeichnetes Signal unterstützt die Anzeige als Bitspur.



### In der Signaltabelle einzelne Signale auswählen und das Format ändern

Um die Anzeige Ihren Anforderungen anzupassen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Spalte  auf das Symbol des jeweiligen Signals, um es für die Anzeige ab- oder anzuwählen.
2. Klicken Sie für das jeweilige Signal in die Spalte "Farbe", um die voreingestellte Farbe des Signals zu ändern.
3. Klicken Sie im Kontextmenü eines Signals auf den Eintrag "In den Vordergrund", um es im Vordergrund darzustellen.

### In der Signaltabelle einzelne Bits für die Anzeige als Bitspur auswählen

Um einzelne Bits als Bitspur im unteren Kurvendiagramm anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Signaltabelle auf das Symbol  eines Signals.
2. Klicken Sie in der geöffneten Bitauswahl des Signals auf das Symbol , um einzelne Bits für die Anzeige ab- oder anzuwählen.

## 12.6.3.6 Kurvendiagramm

### Verwendung des Kurvendiagramms

Das Kurvendiagramm zeigt die in der Signaltabelle ausgewählten Signale einer Aufzeichnung an. Die Aufzeichnung wählen Sie in den Tabellen "Traces im Gerät" oder "Messungen" der Traceverwaltung aus.

Der Darstellungsbereich kann beliebig gezoomt werden. Mit Hilfe von Mess-Cursoren können einzelne Werte zur Anzeige in der Signaltabelle ausgewählt werden.


Die folgende Handlungsanleitung beschreibt den Einsatz der Mess-Cursor.

### Voraussetzung

Eine Aufzeichnung ist zur Anzeige ausgewählt.


### Auswertung eines bestimmten Zeitpunkts einer Aufzeichnung

Um die Werte für einen bestimmten Messpunkt anzuzeigen, gehen Sie folgendermaßen vor:

1. Blenden Sie die vertikalen Mess-Cursor über die Schaltfläche  ein.
2. Verschieben Sie einen Mess-Cursor mit der Maus an die gewünschte Position der Aufzeichnung.  
Die Werte der Signale werden in der Signaltabelle angezeigt. Der Zeitpunkt bzw. Messpunkt zu den Mess-Cursoren wird im unteren Bereich des Kurvendiagramms angezeigt.


### Auswertung der Differenz zwischen zwei Messpunkten

Um die Differenz anzuzeigen, gehen Sie folgendermaßen vor:

1. Blenden Sie die vertikalen Mess-Cursor über die Schaltfläche  ein.
2. Verschieben Sie die beiden Mess-Cursor mit der Maus an die gewünschten Messpunkte der Aufzeichnung.  
Die Werte der Signale und die Differenz werden in der Signaltabelle angezeigt. Der Zeitpunkt bzw. Messpunkt zu den Mess-Cursoren wird im unteren Bereich des Kurvendiagramms angezeigt.

### Horizontale Mess-Cursor verwenden

Um das Erreichen eines bestimmten Werts zu prüfen, gehen Sie folgendermaßen vor:

1. Blenden Sie die horizontalen Mess-Cursor über die Schaltfläche  ein.
2. Verschieben Sie einen Mess-Cursor mit der Maus auf den gewünschten Wert der Aufzeichnung.  
Die Werte der Mess-Cursor für das gewählte Signal werden im unteren Bereich des Kurvendiagramms angezeigt.

### Drucken einer Aufzeichnung

Das Kurvendiagramm unterstützt das Speichern der Anzeige als Bitmap und das Kopieren der Anzeige in die Zwischenablage. Nutzen Sie diese Funktionen (Seite 4301) auch zum Drucken.

## 12.6.4 Geräte

### 12.6.4.1 S7-1200/1500 CPUs

#### Erfassbare Variablen

#### Geräteabhängige Erfassung von Variablen

Die folgende Aufzählung zeigt, aus welchen Operandenbereichen Variablen aufgezeichnet werden können:

- Prozessabbild der Eingänge
- Prozessabbild der Ausgänge
- Merker
- Datenbausteine

#### Datentypen

Alle elementaren Datentypen können aufgezeichnet werden.

Die folgende Tabelle listet die elementaren Datentypen auf:

| Datentypen              | Hinweis                        |
|-------------------------|--------------------------------|
| <b>Binärzahlen</b>      |                                |
| BOOL                    | -                              |
| <b>Bitfolgen</b>        |                                |
| BYTE                    | -                              |
| WORD                    | -                              |
| DWORD                   | -                              |
| LWORD                   | Symbolischer Name erforderlich |
| <b>Ganzzahlen</b>       |                                |
| SINT                    | -                              |
| USINT                   | -                              |
| INT                     | -                              |
| UINT                    | -                              |
| DINT                    | -                              |
| UDINT                   | -                              |
| LINT                    | Symbolischer Name erforderlich |
| ULINT                   | Symbolischer Name erforderlich |
| <b>Gleitpunktzahlen</b> |                                |
| REAL                    | -                              |
| LREAL                   | Symbolischer Name erforderlich |



## Lebensdauer der Tracekonfiguration und aufgezeichneten Werte im Gerät

Tracekonfigurationen im Gerät bleiben bei NETZ-AUS erhalten. Nach dem Wiederanlauf der CPU wird die Aufzeichnung neu aktiviert.

Aufgezeichnete Werte gehen durch den Wiederanlauf verloren.

## Aufzeichnungsebenen

Die folgende Aufzählung zeigt die Ablaufebenen, die für den Aufzeichnungstakt ausgewählt werden können:

- Programmzyklus - OB 1
- Uhrzeitalarm - OB 1x
- Verzögerungsalarm - OB 2x
- Weckalarm - OB 3x
- Synchronisierte Bearbeitungszyklen - OB 6x, nicht OB 60
- Servotask - OB 91 (S7-1500)
- IPO - OB 92 (S7-1500)

---

### Hinweis

Die Messwerte werden jeweils am Ende des OBs nach der Bearbeitung des Anwenderprogramms aufgezeichnet.

---

## Mengengerüst

Die folgende Tabelle zeigt das maximale Mengengerüst, das mit der Trace- und Logikanalysatorfunktion aufgezeichnet werden kann:

| Gerät                              | Maximale Anzahl "Traces im Gerät" | Maximale Anzahl Signale pro Tracekonfiguration |
|------------------------------------|-----------------------------------|--|
| S7-1200<br>(ab Firmwarestand V4.x) | 2                                 | 16   |
| S7-1500                            | 4                                 | 16   |

## CPU-Belastung durch Traceaufzeichnung

Eine Traceaufzeichnung erhöht die Laufzeit der jeweiligen Aufzeichnungsebene, die bei hoher Auslastung der CPU zu einem Ablaufebenen-Überlauf führen kann.

### Abhilfe

- **Tracekonfiguration ändern**
  - 1) Konfigurieren Sie weniger Variablen und Signale.
  - 2) Erhöhen Sie dann die Anzahl Variablen und Signale schrittweise bis zur maximalen Signalanzahl ohne Ablaufebenen-Überlauf.
- **Langsamere Aufzeichnungsebene wählen**

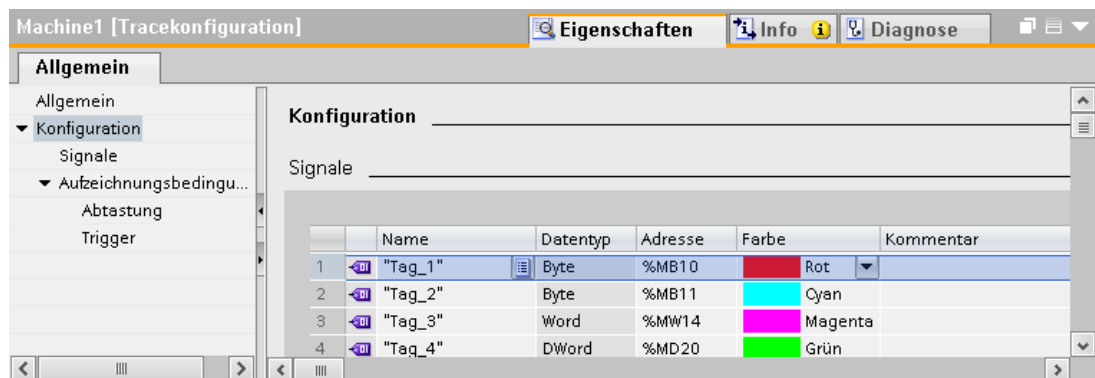
## Software-Oberfläche des gerätespezifischen Bereichs

### Aufbau der Oberfläche

### Anzeigebereiche im Eigenschaftsfenster des Inspektors

Die Einstellmöglichkeiten unterscheiden sich in Abhängigkeit vom konfigurierten Gerät.

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:



Die Bereichsnavigation bietet folgende Einträge zur Auswahl an:

- Allgemein
- Konfiguration
  - Signale (Seite 4325)
  - Aufzeichnungsbedingungen (Seite 4326)

### Anzeige der Eigenschaften einer Tracekonfiguration

Die Tracekonfiguration wird in der Traceverwaltung ausgewählt:

- Auswahl in Tabelle "Tracekonfigurationen"  
Die Eigenschaften werden im Inspektorfenster angezeigt und können geändert werden.
- Auswahl in Tabelle "Traces im Gerät" oder Tabelle "Messungen"  
Die Eigenschaften werden im Inspektorfenster schreibgeschützt angezeigt.

### Oberfläche - Allgemein

Der Bereich "Allgemein" zeigt den Namen der Tracekonfiguration und Eingabefelder für den Autor und einen Kommentar an.

## Eingabemöglichkeiten und Anzeigen unter Allgemein

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

Die folgende Tabelle zeigt die Eingabefelder und Anzeigen:

| Spalte      | Symbol | Beschreibung                    |
|-------------|--------|---------------------------------|
| "Name"      | -      | Name der Tracekonfiguration     |
| "Autor"     | -      | Autor der Tracekonfiguration    |
| "Kommentar" | -      | Eingabefeld für einen Kommentar |

## Oberfläche - Signale


Der Bereich "Signale" zeigt eine Tabelle an, in der die aufzuzeichnenden Signale für die angewählte Tracekonfiguration konfiguriert werden.

## Einstellmöglichkeiten und Anzeigen unter "Signale"

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

Die folgende Tabelle zeigt die Einstellungen und Anzeigen:

| Spalte | Symbol | Beschreibung  |
|--------|--------|---|
|        |        | Anzeige des Signalsymbols für ein ausgewähltes Signal   |
| "Name" | -      | Eingabefeld für die Bezeichnung oder Adresse des Signals<br><b>Beispiele:</b> <ul style="list-style-type: none"> <li>• M0.0</li> <li>• DB1.DBW3</li> <li>• "Datenbaustein_1".Druck</li> </ul> |

| Spalte      | Symbol  | Beschreibung  |
|-------------|---|---|
| -           |  | Schaltfläche zum Öffnen der Signalauswahl-Tabelle<br>Die Schaltfläche wird bei Auswahl der Tabellenzeile eingeblendet.<br>Mit Klick auf das Symbol öffnen Sie eine Tabelle, welche mögliche Signale zur Auswahl anbietet. Das ausgewählte Signal wird im Eingabefeld angezeigt. |
| "Datentyp"  | -   | Textfeld mit der Anzeige des Datentyps für das Signal   |
| "Adresse"   | -   | Eingabefeld für die Adresse des Signals<br>Bei rein symbolischen Signalen bleibt das Feld leer.   |
| "Farbe"     | -   | Textfeld zur Anzeige und Auswahl der Farbe<br>Klicken Sie auf das ausgewählte Feld, um die Signalfarbe darzustellen.<br>Klicken Sie auf die Signalfarbe, um den Farbauswahldialog zu öffnen.  |
| "Kommentar" | -   | Eingabefeld für einen Kommentar   |

### Kontextmenübefehle

Die folgende Tabelle zeigt die Kontextmenübefehle der Tabelle:

| Kontextmenübefehl | Beschreibung   |
|-------------------|--|
| "Ausschneiden"    | Entfernt die selektierten Zeilen und legt diese in die Zwischenablage.   |
| "Kopieren"        | Kopiert den Inhalt der selektierten Zeilen in die Zwischenablage.  |
| "Einfügen"        | Fügt den Inhalt der Zwischenablage ab der selektierten Zeile ein.<br>Der bestehende Inhalt wird überschrieben. |
| "Löschen"         | Löscht die selektierten Zeilen aus der Tabelle.  |
| "Umbenennen"      | Schaltet die selektierte Zelle in den Editier-Modus.   |

### Aufzeichnungsbedingungen

#### Unterstützte Datentypen

Die folgende Tabelle zeigt die unterstützten Datentypen für die Triggervariable:

| Speicherbedarf und Format der Zahl | Datentyp           |
|------------------------------------|--------------------|
| 1-Byte                             | BOOL               |
| 8-Bit-Ganzzahlen                   | SINT, USINT, BYTE  |
| 16-Bit-Ganzzahlen                  | INT, UINT, WORD    |
| 32-Bit-Ganzzahlen                  | DINT, UDINT, DWORD |
| 64-Bit-Ganzzahlen                  | LINT, ULINT, LWORD |
| 32-Bit-Gleitpunktzahlen            | REAL               |
| 64-Bit-Gleitpunktzahlen            | LREAL              |

## Oberfläche - Aufzeichnungsbedingungen

Der Bereich "Aufzeichnungsbedingungen" zeigt für die ausgewählte Tracekonfiguration die Triggerbedingung an und in welchem Takt, wie schnell und wie lange aufgezeichnet wird. Die Konfiguration ist nur möglich, wenn die Tracekonfiguration in der Tabelle "Tracekonfigurationen" der Traceverwaltung angewählt ist.

## Einstellmöglichkeiten und Anzeigen unter "Aufzeichnungsbedingungen"

Das folgende Bild zeigt exemplarisch die Darstellung im TIA-Portal:

Aufzeichnungsbedingungen

> Abtastung

Aufzeichnungszeitpunkt: "Main" %OB1

Aufzeichnen alle: 1 Zyklus

Aufzeichnungsdauer (a): 20 Messpunkte

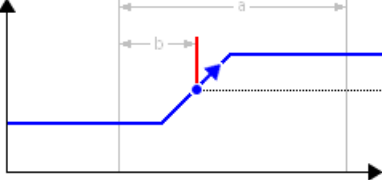
Max. Aufzeichnungsdauer: 32766 Messpunkte  Max. Aufzeichnungsdauer verwenden

> Trigger

Triggermodus: Trigger auf Variable


Triggervariable: "C\_Int" %MW30

Ereignis: Steigendes Signal

Wert:  = 0

Pretrigger (b): 0 Messpunkte

| Einstellung/Anzeige      | Beschreibung  |
|--------------------------|---|
| "Aufzeichnungszeitpunkt" |   |
| Klappliste               | Auswahl des Aufzeichnungszeitpunkts<br>Siehe Aufzeichnungsebenen (Seite 4323)   |
| Textfeld                 | Detailinfos zum ausgewählten Aufzeichnungszeitpunkt   |
| "Aufzeichnen alle"       |   |
| Eingabefeld              | Eingabe der Untersetzung bezogen auf die Zyklen   |
| Klappliste               | Auswahl des Untersetzungsfaktors<br>Folgende Einstellungen sind möglich: <ul style="list-style-type: none"> <li>"Zyklus"</li> </ul> |
| Textfeld                 | Anzeige der Abtastzeit unter Berücksichtigung der konfigurierten Untersetzung   |
| "Aufzeichnungsdauer"     |   |

| Einstellung/Anzeige       |   | Beschreibung   |
|---------------------------|---|--|
|                           | Eingabefeld   | Eingabe der Aufzeichnungsdauer bezogen auf die Messpunkte<br>Wenn das Optionskästchens "Aufzeichnungsdauer = Max. Aufzeichnungsdauer" aktiviert ist, werden Eingaben durch den in "Max. Aufzeichnungsdauer" angezeigten Wert überschrieben.  |
|                           | Klappliste  | Auswahl der Zeiteinheit für die Aufzeichnungsdauer<br>Folgende Einstellungen sind möglich: <ul style="list-style-type: none"> <li>• "Messpunkte"</li> </ul>  |
|                           | Textfeld  | Anzeige der berechneten Aufzeichnungsdauer   |
| "Max. Aufzeichnungsdauer" |   |  |
|                           | Textfeld  | Anzeige der berechneten maximalen Aufzeichnungsdauer<br>Die "Max. Aufzeichnungsdauer" ist davon abhängig, wie viele Signale aufgezeichnet werden und welchen Datentyp diese Signale haben.   |
|                           | "Max. Aufzeichnungsdauer verwenden"   | Setzen der Aufzeichnungsdauer auf die maximale Aufzeichnungsdauer<br>Mit Aktivierung des Optionskästchens wird die Aufzeichnungsdauer auf die maximal mögliche Aufzeichnungsdauer gesetzt. Die eingestellte Untersetzung im Eingabefeld "Aufzeichnen alle" wird berücksichtigt. Die Aufzeichnungsdauer wird auch angepasst, wenn weitere Signale hinzugefügt werden. |
| "Triggermodus"            |   |  |
|                           | Klappliste  | Folgende Einstellungen sind möglich: <ul style="list-style-type: none"> <li>• "Aufzeichnung sofort"<br/>Die Aufzeichnung erfolgt sofort nach der Aktivierung im Gerät.</li> <li>• "Trigger auf Variable"<br/>Die Aufzeichnung erfolgt, sobald der Trace im Gerät aktiviert ist und die konfigurierte Bedingung des Triggers erfüllt ist.</li> </ul>                  |
|                           | Textfeld  | Anzeige des Trigger-Typs   |
| "Triggervariable"         |   |  |
|                           | Eingabefeld   | Eingabe eines Signals<br><b>Beispiele:</b> <ul style="list-style-type: none"> <li>• M0.0</li> <li>• DB1.DBW3</li> <li>• "Datenbaustein_1".Temperatur</li> </ul>  |
|                           |  | Öffnen der Signalauswahl-Tabelle<br>Mit Klick auf das Symbol öffnen Sie eine Tabelle, welche mögliche Signale zur Auswahl als Triggervariable anbietet. Das ausgewählte Signal wird im Eingabefeld angezeigt.  |
|                           | Textfeld  | Anzeige der Adresse der Trigger-Variable   |
| "Ereignis"                |   |  |
|                           | Klappliste  | Ereignisauswahl, auf das die Triggervariable geprüft wird<br>Die Einträge der Klappliste sind im Kapitel Triggerereignis (Seite 4329) beschrieben.   |
|                           | Textfeld  | Anzeige der Art des Vergleichs   |

| Einstellung/Anzeige | Beschreibung   |
|---------------------|--|
| "Wert"              | Konfiguration des gewählten Ereignisses<br>Die Konfigurationsmöglichkeiten unterscheiden sich in Abhängigkeit vom Format der Triggervariable und dem gewählten Ereignis.<br>Siehe Triggerereignis (Seite 4329).  |
| "Pretrigger"        | Mit dem "Pretrigger" definieren Sie die Zeitspanne, in der die Signale bereits vor der Erfüllung der eigentlichen Triggerbedingung aufgezeichnet werden.<br>Tritt das Triggerereignis sofort oder kurz nach dem Aktivieren der Aufzeichnung ein, kann sich eine kürzerer Aufzeichnungsdauer ergeben.<br>Beispiel für "Aufzeichnungsdauer" = 20 Messpunkte und "Pretrigger" = 5 Messpunkte: <ul style="list-style-type: none"> <li>• Triggerereignis tritt 50 Messpunkte nach Aktivierung der Aufzeichnung ein<br/>Tatsächliche Aufzeichnungsdauer = 20 Messpunkte</li> <li>• Triggerereignis tritt 2 Messpunkte nach Aktivierung der Aufzeichnung ein<br/>Tatsächliche Aufzeichnungsdauer = 17 Messpunkte</li> </ul> |
| Eingabefeld         | Eingabe der Dauer bezogen auf die Auswahl in der Klappliste.   |
| Klappliste          | Auswahl der Zeiteinheit<br>Folgende Einstellungen sind möglich: <ul style="list-style-type: none"> <li>• "Messpunkte"</li> </ul>   |
| Textfeld            | Anzeige der berechneten "Pretrigger" Zeitspanne  |

### Triggerereignis

In Abhängigkeit von der Auswahl in der Klappliste unterscheiden sich die weiteren Einstellungen zu "Ereignis".

Nachfolgend sind die einzelnen Ereignisse beschrieben.

#### "=TRUE"

Unterstützte Datentypen: Bit (Seite 4326)

Der Start der Aufzeichnung erfolgt bei dem Zustand TRUE des Triggers.

#### "=FALSE"

Unterstützte Datentypen: Bit (Seite 4326)

Der Start der Aufzeichnung erfolgt bei dem Zustand FALSE des Triggers.

#### "Steigende Flanke"

Unterstützte Datentypen: Bit (Seite 4326)

Der Start der Aufzeichnung erfolgt, wenn der Trigger den Zustand von FALSE auf TRUE wechselt.

Nach Aktivieren des Traces im Gerät sind mindestens zwei Zyklen zur Erkennung der Flanke erforderlich.

### **"Steigendes Signal"**

Unterstützte Datentypen: Ganzzahlen und Gleitpunktzahlen (Seite 4326)

Der Start der Aufzeichnung erfolgt, wenn der aufsteigende Wert des Triggers den zu diesem Ereignis konfigurierten Wert erreicht oder überschreitet.

Nach Aktivieren des Traces im Gerät sind mindestens zwei Zyklen zur Erkennung der Flanke erforderlich.

### **"Fallende Flanke"**

Unterstützte Datentypen: Bit (Seite 4326)

Der Start der Aufzeichnung erfolgt, wenn der Trigger den Zustand von TRUE auf FALSE wechselt.

Nach Aktivieren des Traces im Gerät sind mindestens zwei Zyklen zur Erkennung der Flanke erforderlich.

### **"Fallendes Signal"**

Unterstützte Datentypen: Ganzzahlen und Gleitpunktzahlen (Seite 4326)

Der Start der Aufzeichnung erfolgt, wenn der absteigende Wert des Triggers den zu diesem Ereignis konfigurierten Wert erreicht oder unterschreitet.

Nach Aktivieren des Traces im Gerät sind mindestens zwei Zyklen zur Erkennung der Flanke erforderlich.

### **"Im Bereich"**

Unterstützte Datentypen: Ganzzahlen und Gleitpunktzahlen (Seite 4326)

Der Start der Aufzeichnung erfolgt, sobald sich der Wert des Triggers in dem zu diesem Ereignis konfigurierten Wertebereich befindet.

### **"Außerhalb des Bereichs"**

Unterstützte Datentypen: Ganzzahlen und Gleitpunktzahlen (Seite 4326)

Der Start der Aufzeichnung erfolgt, sobald sich der Wert des Triggers außerhalb des zu diesem Ereignis konfigurierten Wertebereichs befindet.

### **"= Wert"**

Unterstützte Datentypen: Ganzzahlen (Seite 4326)

Der Start der Aufzeichnung erfolgt, wenn der Wert des Triggers gleich dem zu diesem Ereignis konfigurierten Wert ist.



### "<> Wert"

Unterstützte Datentypen: Ganzzahlen (Seite 4326)

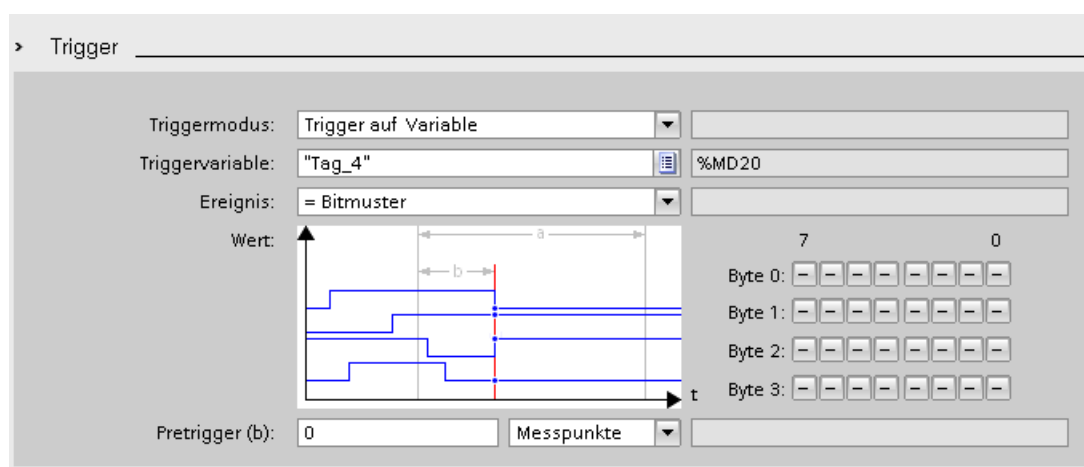
Der Start der Aufzeichnung erfolgt, wenn der Wert des Triggers ungleich dem zu diesem Ereignis konfigurierten Wert ist.

### "= Bitmuster"

Unterstützte Datentypen: Ganzzahlen (Seite 4326)

Der Start der Aufzeichnung erfolgt, wenn der Wert des Triggers mit dem zu diesem Ereignis konfigurierten Bitmuster übereinstimmt.

Das folgende Bild zeigt die Einstellmöglichkeiten für ein "Bitmuster":



Durch Klick auf die jeweilige Schaltfläche kann zwischen den Symbolen umgeschaltet werden.

Die folgende Tabelle zeigt die Symbole:

| Symbol                     | Beschreibung               |
|----------------------------|----------------------------|
| <input type="checkbox"/>   | Bit wird nicht ausgewertet |
| <input type="checkbox"/> 0 | Bit wird auf FALSE geprüft |
| <input type="checkbox"/> 1 | Bit wird auf TRUE geprüft  |

### "<> Bitmuster"

Unterstützte Datentypen: Ganzzahlen (Seite 4326)

Der Start der Aufzeichnung erfolgt, wenn der Wert des Triggers nicht mit dem zu diesem Ereignis konfigurierten Bitmuster übereinstimmt.

### Siehe auch

Triggerbedingungen konfigurieren (Seite 4334)

## Konfigurieren

### Trace konfigurieren - Übersicht

Die Konfiguration der Aufzeichnungsbedingungen und aufzuzeichnenden Signale ist gerätespezifisch.

### Voraussetzung

Eine Tracekonfiguration ist angelegt und in der Tabelle "Tracekonfigurationen" der Traceverwaltung angewählt.

### Vorgehen

Die folgende Tabelle zeigt die Vorgehensweise zum Konfigurieren.

| Schritt | Beschreibung   |
|---------|--|
| 1       | Dokumentation der Konfiguration (optional)<br>Geben Sie einen Kommentar und einen Autor zu der Konfiguration ein.  |
| 2       | Signale auswählen (Seite 4332)<br>Wählen Sie im Bereich "Signale" die aufzuzeichnenden Signale aus.  |
| 3       | Aufzeichnungstakt und Dauer konfigurieren<br>Wählen Sie im Bereich "Aufzeichnungsbedingungen" einen Aufzeichnungszeitpunkt, einen Takt und die Dauer aus.                                  |
| 4       | Triggerbedingungen konfigurieren (Seite 4334)<br>Legen Sie im Bereich "Aufzeichnungsbedingungen" fest, ob die Aufzeichnung sofort oder in Abhängigkeit von einer Triggerbedingung erfolgt. |

### Signale auswählen


### Voraussetzung

- Eine Tracekonfiguration ist angelegt und in der Tabelle "Tracekonfigurationen" der Traceverwaltung angewählt.
- Der Bereich "Signale" ist im Inspektorfenster geöffnet.

### Vorgehen

Um die aufzuzeichnenden Signale zu konfigurieren, gehen Sie folgendermaßen vor:

1. Klicken Sie in die erste leere Zeile der Tabelle.
2. Klicken Sie in der Spalte "Name" auf die erste leere Zeile

3. Wählen Sie ein Signal aus. Folgende Möglichkeiten stehen zur Verfügung:
  - Klicken Sie in der Spalte "Name" auf die Schaltfläche  und wählen Sie eine Variable aus.
  - Geben Sie in der Spalte "Name" die symbolische Bezeichnung der Variable in die Zelle ein.
  - Geben Sie in der Spalte "Adresse" direkt die Adresse ein.
4. Klicken Sie in die Spalte "Anzeigefarbe" und wählen Sie eine Farbe für die Darstellung des Signals aus.
5. Klicken Sie in die Spalte "Kommentar" und geben Sie einen Kommentar zu dem Signal ein.
6. Wiederholen Sie das Vorgehen ab Schritt 1, bis alle aufzuzeichnenden Signale in der Tabelle eingetragen sind.


## Aufzeichnungstakt und Dauer konfigurieren

### Voraussetzung

- Eine Tracekonfiguration ist angelegt und in der Tabelle "Tracekonfigurationen" der Traceverwaltung angewählt.
- Der Bereich "Aufzeichnungsbedingungen" ist im Inspektorfenster geöffnet.

### Vorgehen

Um den Takt und die Dauer einer Aufzeichnung zu konfigurieren, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die Schaltfläche  für den Aufzeichnungszeitpunkt.
2. Wählen Sie einen OB für den Aufzeichnungszeitpunkt (Seite 4323) aus.
3. Wählen Sie in der Klappliste für "Aufzeichnen alle" eine Einheit für den Untersetzungsfaktor aus.
4. Geben Sie in das Eingabefeld für "Aufzeichnen alle" den Faktor für die Untersetzung ein.
5. Wählen Sie in der Klappliste für "Aufzeichnungsdauer" eine Einheit aus.
6. Legen Sie die Aufzeichnungsdauer fest.  
Folgende Möglichkeiten stehen zur Verfügung:
  - Geben Sie in das Eingabefeld für "Aufzeichnungsdauer" einen Wert für die Dauer ein.
  - Aktivieren Sie das Optionskästchen "Max. Aufzeichnungsdauer verwenden"

## Triggerbedingungen konfigurieren

### Voraussetzung

- Eine Tracekonfiguration ist angelegt und in der Tabelle "Tracekonfigurationen" der Traceverwaltung angewählt.
- Der Bereich "Aufzeichnungsbedingungen" ist im Inspektorfenster geöffnet.


### Triggerbedingung "Aufzeichnung sofort"

Um die Aufzeichnung sofort zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste für "Triggermodus" den Eintrag "Aufzeichnung sofort" aus. Die Eingabefelder für die Triggervariable werden ausgeblendet.

### Triggerbedingung "Trigger auf Variable"

Um die Aufzeichnung in Abhängigkeit einer Bedingung zu starten, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Klappliste für "Triggermodus" den Eintrag "Trigger auf Variable" aus.
2. Wählen Sie eine Triggervariable aus. Folgende Möglichkeiten stehen zur Verfügung:
  - Klicken Sie auf die Schaltfläche  für die Triggervariable und wählen Sie eine Variable aus.
  - Geben Sie direkt die Adresse oder die symbolische Bezeichnung der Variable in das Eingabefeld für die Triggervariable ein.

Eine Klappliste mit Ereignissen und Eingabefeldern wird angezeigt. Die Anzeige ist abhängig vom Datentyp der Variable.

3. Konfigurieren Sie das Ereignis
4. Wählen Sie in der Klappliste für "Pretrigger" eine Einheit für den Pretrigger aus.
5. Um bereits eine Zeitspanne vor dem Triggerereignis aufzuzeichnen, geben Sie in das Eingabefeld für den Pretrigger einen Wert größer 0 ein.

---

#### Hinweis

Die Triggerbedingung wird unabhängig von der Einstellung in "Aufzeichnen alle" in jedem Zyklus überprüft. Zur sicheren Erkennung des Triggers muss das Triggersignal mindestens einen vollen Zyklus lang anstehen.

---

## 12.7 Fernverbindung mit TeleService aufbauen

### 12.7.1 Grundlagen für das Arbeiten mit TeleService

#### 12.7.1.1 Einführung in TeleService

##### Einführung

Mit TeleService wird Ihre Steuerung telekommunikationsfähig. Sie können dezentrale Anlagen über Fernverbindungen zentral verwalten, steuern und überwachen.

##### Funktionsumfang

Mit TeleService können Sie den Funktionsumfang des TIA-Portals über ein Telefonnetz oder eine Internetverbindung nutzen, indem Sie eine Fernverbindung zu einer entfernten Anlage aufbauen. Über die Online-Verbindung ist es möglich, auch eine entfernte Anlage wie gewohnt mit dem TIA-Portal zu bearbeiten.

##### Vorteile

Durch den Einsatz von TeleService ergeben sich für Sie folgende Vorteile:

- Sie können auch entfernte Anlagenteile gut erreichen und in ein Gesamtsystem einbeziehen.
- Bei Fehlern in einer entfernten Anlage können Sie schnelle Hilfe und Unterstützung bieten, ohne selbst vor Ort sein zu müssen.
- Sie können Ihre Ressourcen effektiv einsetzen.
- Kosten werden beträchtlich reduziert.
- Stillstandszeiten von Anlagen können wesentlich verringert werden.
- Die Wirtschaftlichkeit Ihrer Anlage steigt.

##### Siehe auch

Funktionalität von TeleService (Seite 4336)

## 12.7.1.2 Funktionalität von TeleService

### Einsatzmöglichkeiten von TeleService

TeleService bietet Ihnen folgende Einsatzmöglichkeiten:

- **Zugriff auf entfernte Anlagen (Fernwartung):**  
Sie können dezentrale Anlagen über Fernverbindungen zentral verwalten, steuern und überwachen.  
Dies ist möglich mit einer CPU S7-300/400, einer CPU S7-1200 und einer CPU S7-1500 und jeweils einem TS Adapter MPI oder einem TS Adapter IE.
- **Verbindungsaufbau von und zu entfernten Anlagen (PG-AS-Fernkopplung):**  
Mit Hilfe von PRODAVE MPI V5.0 und höher können Sie eine Fernverbindung zu einer entfernten Anlage aufbauen und mit der Kommunikationsanweisung "PG\_DIAL" können Sie eine Fernverbindung von einer entfernten Anlage aufbauen.  
Dies ist möglich mit einer CPU S7-300/400 und einem TS Adapter MPI.
- **Datenaustausch zwischen Anlagen (AS-AS-Fernkopplung):**  
Mit Hilfe der Kommunikationsanweisung "AS\_DIAL" können zwei Automatisierungssysteme über das Telefonnetz Prozessdaten miteinander austauschen.  
Dies ist möglich mit einer CPU S7-300/400 und einem TS Adapter MPI.
- **Senden einer SMS von einer Anlage:**  
Mit Hilfe der Kommunikationsanweisung "SMS\_SEND" kann ein Automatisierungssystem über ein GSM-Funkmodem eine Nachricht (SMS) versenden.  
Dies ist möglich mit einer CPU S7-300/400 und einem TS Adapter MPI.
- **Senden einer E-Mail von einer Anlage**  
Mit Hilfe der folgenden Kommunikationsanweisungen und einem TS Adapter IE kann ein Automatisierungssystem auch eine E-Mail verschicken.
  - S7-300/400er CPUs (CPU S7-31x-2PN/DP bzw. CPU 41x-3PN/D) verwenden die Anweisung "AS\_MAIL"
  - S7-1200er CPUs verwenden die Anweisung "TM\_MAIL"
  - S7-1500er CPUs verwenden die Anweisung "TMAIL\_C"

## 12.7.1.3 Telefonbuch bei TeleService

### Einführung

Durch einen Doppelklick auf den Ordner "Telefonbuch" in der Projektnavigation öffnen Sie den Telefonbucheditor, der das Telefonbuch von TeleService anzeigt.

Jede Version des TIA-Portals verfügt über ein eigenes "globales Telefonbuch". Wird in einer neuen Version des TIA-Portals ein globales Telefonbuch aus einer älteren Version des TIA-Portals gefunden, so werden sie einmalig gefragt, ob Sie dieses Telefonbuch importieren möchten.

Dies hat den Vorteil, dass Ihnen auch in der neuen Version des TIA-Portals die Anlagendaten aus der vorherigen Version zur Verfügung stehen.

### **Eigenschaften des globalen Telefonbuchs**

Das globale Telefonbuch dient in TeleService zum Verwalten der spezifischen Anlagendaten, die zum Aufbauen einer Fernverbindung benötigt werden.

Wenn Sie das Telefonbuch zum ersten Mal öffnen, wird ein leeres Telefonbuch mit allen vorhandenen Spalten angezeigt, ansonsten wird das zuletzt bearbeitete Telefonbuch angezeigt.

In einem Telefonbuch können Sie eine beliebige Anzahl von Anlagen eintragen. Anlagen enthalten die Daten, die Sie für den Aufbau einer Fernverbindung benötigen, z. B. den Namen und den Standort des Teilnehmers sowie die anzuwählende Telefonnummer mit den landesspezifischen Details. Bei VPN-Verbindungen können Sie an Stelle der Telefonnummer eine IP-Adresse bzw. einen DNS-Namen eintragen.

Die für den Verbindungsaufbau verwendeten TS Adapter sind farblich unterschiedlich dargestellt, abhängig davon, ob ein TS Adapter MPI oder ein TS Adapter IE für den Verbindungsaufbau eingesetzt wird.

### **Siehe auch**

Arbeiten mit dem Telefonbuch (Seite 4337)

## **12.7.2 Arbeiten mit dem Telefonbuch**

### **12.7.2.1 Grundlagen zum Arbeiten mit dem Telefonbuch**

#### **Arbeiten mit dem Telefonbuch**

Folgende Möglichkeiten stehen Ihnen für das Arbeiten mit einem Telefonbuch zur Verfügung:

- Telefonbuch öffnen
- Telefonbuch speichern
- Telefonbuchdaten importieren
- Telefonbuchdaten exportieren
- Telefonbuchdaten drucken
- Telefonbuchdaten zum Aufbau einer Fernverbindung verwenden

Diese Funktionen können Sie komfortabel über die in der Funktionsleiste des Telefonbuchs angezeigten Schaltflächen durchführen.

---

**Hinweis**

**Zugriff auf Telefonbücher**

Das Telefonbuch wird benutzerspezifisch in TeleService hinterlegt. Es ist jedoch nicht möglich, gleichzeitig mit mehr als einer Instanz des TIA-Portals auf das globale Telefonbuch zuzugreifen.

---

**Siehe auch**

- Telefonbuch öffnen (Seite 4340)
- Telefonbuch speichern (Seite 4341)
- Telefonbuchdaten exportieren (Seite 4343)
- Telefonbuch drucken (Seite 4343)
- Aufbau des Telefonbuchs (Seite 4338)

**12.7.2.2 Aufbau des Telefonbuchs**

**Einführung**

In TeleService dient ein globales Telefonbuch zum Verwalten der Daten, die Sie für den Aufbau einer Fernverbindung benötigen. Wenn Sie die Verbindungsdaten einmalig erstellt und im Telefonbuch hinterlegt haben, können Sie immer wieder darauf zugreifen, um eine Fernverbindung aufzubauen.

**Aufbau des Telefonbuches**

Das in TeleService integrierte globale Telefonbuch enthält folgende Spalten:

| Spaltenname                  | Erläuterung  |
|------------------------------|--|
| Anlagenname                  | Tragen Sie hier den gewünschten Namen für Ihre Anlage ein.   |
| Adapter-Typ                  | Wählen Sie aus der Klappliste den verwendeten Adapter-Typ aus: TS Adapter MPI oder TS Adapter IE.  |
| Verbindungs-Typ              | Wählen Sie den gewünschten Verbindungs-Typ aus: Wählverbindung oder VPN-Verbindung.  |
| Ortskennzahl                 | Tragen Sie hier die gewünschte Ortskennzahl ein.<br>Diese Spalte ist nur aktiv bei Wählverbindungen. Bei VPN-Verbindungen ist diese Spalte nicht editierbar.   |
| Telefonnummer/Remote-Adresse | Tragen Sie hier die gewünschten Verbindungsdaten für den Aufbau der Fernverbindung ein. Dies kann für Wählverbindungen eine Telefonnummer sein und für VPN-Verbindungen ein DNS-Name oder eine IP-Adresse. |



| Spaltenname   | Erläuterung   |
|---------------|---|
| Fingerabdruck | Tragen Sie hier den zugehörigen Fingerabdruck für den Aufbau einer VPN-Verbindung zum TS Adapter IE Advanced ein.   |
| Land          | Tragen Sie die zugehörige Landeskennzahl ein.<br>Diese Spalte ist nur aktiv bei Wählverbindungen. Bei VPN-Verbindungen ist diese Spalte nicht editierbar. |
| Benutzername  | Tragen Sie den Benutzernamen ein, unter dem Sie angemeldet sind.  |
| Kennwort      | Tragen Sie hier das zum Benutzernamen gehörige Kennwort ein.  |
| Gruppe        | Tragen Sie hier die zugehörige Gruppe ein, falls Sie eine Gruppierung vorgenommen haben.  |
| Firma         | Tragen Sie die Firma ein, die kontaktiert werden soll.  |
| Abteilung     | Tragen Sie die zugehörige Abteilung ein.  |
| Straße        | Tragen Sie die Straße ein.  |
| Ort           | Tragen Sie den Ort ein, zu dem die Fernverbindung aufgebaut werden soll.  |
| Kommentar     | Tragen Sie hier, falls gewünscht, einen Kommentar ein.  |











### Spalten ein- oder ausblenden

Einzelne Spalten können Sie ein- oder ausblenden. Selektieren Sie dazu die gewünschte Spaltenüberschrift und rufen Sie über die rechte Maustaste das Kontextmenü auf.

### 12.7.2.3 Symbole im Telefonbuch

#### Bedeutung der Symbole in TeleService

Die nachfolgende Tabelle zeigt die Bedeutung der Symbole in TeleService:

| Symbol  | Bedeutung  |
|---|--|
|  | Öffnet das globale Telefonbuch                                 |
|  | Importiert ein Telefonbuch                                     |
|  | Exportiert ein Telefonbuch                                     |
|  | Baut eine Fernverbindung auf                                   |
|  | Trennt die aktive Fernverbindung                               |
|  | Baut eine Fernverbindung auf oder trennt diese                 |
|  | Zeigt im Telefonbuch die Verbindung zu einem TS Adapter IE an  |
|  | Zeigt im Telefonbuch die Verbindung zu einem TS Adapter MPI an |
|  | Fügt im Telefonbuch eine neue Zeile hinzu                      |
|  | Fügt im Telefonbuch eine neue Zeile ein                        |

## 12.7.2.4 Telefonbuch verwalten

### Telefonbuch öffnen

#### Öffnen von Telefonbüchern

Um das Telefonbuch zu öffnen, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation unter "Online Zugänge" > "TeleService" auf den Ordner "Telefonbuch".
2. Das Telefonbuch wird geöffnet, so dass Sie die gewünschten Anlagedaten eintragen bzw. nachbearbeiten können.

### Zeilen ins Telefonbuch einfügen

#### Zeilen ins Telefonbuch einfügen

Um eine neue Zeile ins Telefonbuch einzufügen, gehen Sie folgendermaßen vor:

1. Markieren Sie die Zeile, vor der Sie eine neue Zeile einfügen möchten.
2. Klicken Sie in der Funktionsleiste auf die Schaltfläche "Zeile einfügen".

### Ergebnis

Vor der markierten Zeile wird eine neue Zeile ins Telefonbuch eingefügt.

### Spalten im Telefonbuch ein- und ausblenden

#### Ein und Ausblenden von Spalten

Um Spalten im Telefonbuch ein- oder auszublenden, gehen Sie folgendermaßen vor:

1. Klicken Sie in einen Spaltenkopf.
2. Wählen Sie im Kontextmenü den Befehl "Spalten ein-/ausblenden".  
Die Auswahl der verfügbaren Spalten wird angezeigt.
3. Um eine Spalte einzublenden, aktivieren Sie das Optionskästchen der Spalte.
4. Um eine Spalten auszublenden, deaktivieren Sie das Optionskästchen der Spalte.

### Ergebnis

Die jeweiligen Spalten werden bei der Anzeige des Telefonbuchs ein- oder ausgeblendet.

## Telefonbuch speichern

### Speichern von Telefonbüchern

Beim Verlassen des Telefonbuch-Editors oder beim Verlassen des TIA-Portals werden Sie gefragt, ob Sie das globale Telefonbuch speichern wollen.

Zum Speichern des Telefonbuchs klicken Sie auf "Ja".

## Telefonbuchdaten importieren

### Einführung

Es besteht die Möglichkeit, die Telefonbuchdaten von einer externen Datei oder aus einer älteren Version des TIA-Portals zu importieren.

### Voraussetzung

Sie haben bereits eine importfähige Telefonbuchdatei angelegt.

Sie haben mit einer älteren Version des TIA-Portals bereits ein Telefonbuch angelegt.

### Telefonbuchdaten aus Telefonbuchdatei importieren

Um Telefonbuchdaten aus einer Telefonbuchdatei zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "TeleService" unter "Online-Zugänge" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "Telefonbuch".
3. Klicken Sie im Telefonbuch in der Funktionsleiste auf die Schaltfläche "Importieren".
4. Bestätigen Sie die Abfrage, ob der aktuelle Stand des Arbeitstelefonbuchs gespeichert werden soll ggf. mit "Ja" und wählen Sie im nachfolgenden Dialog aus, wohin das Telefonbuch gespeichert werden soll.
5. Falls Sie den aktuellen Stand des Arbeitstelefonbuchs nicht speichern wollen, so wählen Sie bei der Abfrage "nein". Wählen Sie im nachfolgenden Dialog aus, welche Telefonbuchdatei in das aktuelle Telefonbuch gespeichert werden soll.
6. Verlassen Sie den Dialog mit "OK".

### Ergebnis

Die importierten Telefonbuchdaten werden im globalen Telefonbuch angezeigt.

---

**Hinweis**

**Wählregeln definieren**

Damit die Spalten "Ortskennzahl" und "Land" im importierten Telefonbuch für Wählverbindungen editierbar sind, müssen Wählregeln definiert sein.

Um Wählregeln zu definieren, folgen Sie dem im Verlauf angezeigten Link.

---

### Telefonbuchdaten aus älterer Version des TIA-Portals importieren

Um Telefonbuchdaten aus einer älteren Version des TIA-Portals zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "TeleService" unter "Online-Zugänge" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "Telefonbuch".
3. Klicken Sie im Telefonbuch in der Funktionsleiste auf die Schaltfläche "Importieren".
4. Bestätigen Sie die Abfrage, ob der aktuelle Stand des Arbeitstelefonbuchs gespeichert werden soll ggf. mit "Ja" und wählen Sie im nachfolgenden Dialog aus, wohin das Telefonbuch gespeichert werden soll.
5. Falls Sie den aktuellen Stand des Arbeitstelefonbuchs nicht speichern wollen, so wählen Sie bei der Abfrage "nein".
6. Tragen Sie im nachfolgenden Dialog folgende Pfadangabe ein: "%appdata%\siemens\automation\TeleService\GlobalTeleServicePhoneBook.tel".
7. Verlassen Sie den Dialog mit "OK".

### Ergebnis

Die aus der älteren Version des TIA-Portals importierten Telefonbuchdaten werden im globalen Telefonbuch angezeigt.

---

**Hinweis**

**Wählregeln definieren**

Damit die Spalten "Ortskennzahl" und "Land" im importierten Telefonbuch für Wählverbindungen editierbar sind, müssen Wählregeln definiert sein.

Um Wählregeln zu definieren, folgen Sie dem im Verlauf angezeigten Link.

---

### Siehe auch

Wählregeln definieren (Seite 4344)

## Telefonbuchdaten exportieren

### Einführung

Es besteht die Möglichkeit, Telefonbuchdaten in eine externe Datei zu exportieren.

### Voraussetzung

Sie haben unter TeleService bereits ein Telefonbuch mit den entsprechenden Anlagendaten erstellt.

### Vorgehen

Um die Telefonbuchdaten zu exportieren gehen Sie folgendermaßen vor:

1. Öffnen Sie auf den Ordner "TeleService" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "Telefonbuch".
3. Klicken Sie im Telefonbuch in der Funktionsleiste auf die Schaltfläche "Exportieren".
4. Wählen Sie im nachfolgenden Dialog aus, wohin das aktuelle Telefonbuch exportiert werden soll.
5. Verlassen Sie den Dialog mit "OK".

### Ergebnis

Die exportierten Telefonbuchdaten werden in der angegebenen Exportdatei gespeichert.

## Telefonbuch drucken

### Drucken von Telefonbüchern

Sie können sich alle Daten eines Telefonbuchs oder nur einen bestimmten Teil eines Telefonbuchs ausdrucken lassen.

### Gehen Sie folgendermaßen vor:

1. Öffnen Sie das Telefonbuch.
2. Wählen Sie den Befehl **Telefonbuch > Drucken** oder klicken Sie auf die entsprechende Schaltfläche in der Funktionsleiste. Daraufhin öffnet sich das Dialogfeld "Drucken".
3. Geben Sie hier an, ob Sie das gesamte Telefonbuch oder nur einen Teil des Telefonbuchs drucken möchten, und stellen Sie alle weiteren Optionen ein.
4. Starten Sie den Druckvorgang mit "OK".

## Ergebnis

Die Daten des Telefonbuchs werden auf dem voreingestellten Drucker ausgegeben. Umfasst der Ausdruck mehrere Seiten, wird am rechten unteren Blattrand nach der Seitennummer eine Folgeseitenkennung ausgegeben. Die letzte Seite des Ausdrucks enthält keine Folgeseitenkennung.

## Wählregeln definieren

### Vorgehensweise

Um standortspezifische Wählregeln zu definieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie im TIA-Portal den Ordner "Online-Zugänge" und selektieren Sie den Ordner "TeleService".
2. Rufen Sie über das Kontextmenü die Eigenschaften von "TeleService" auf.
3. Öffnen Sie im nachfolgenden Dialog das Register "Erweiterte Einstellungen" und aktivieren Sie die Option "Wählregeln benutzen".
4. Klicken Sie auf die Schaltfläche "Anpassen".
5. Tragen Sie im nachfolgenden Dialog die gewünschten Standortinformationen für den jeweiligen Verbindungsaufbau ein.
  - Geben Sie bei neuen Standorten das Land bzw. die Region und die zugehörige Ortskennzahl an.
  - Stellen Sie, falls erforderlich, die richtige Netzkennzahl und Amtskennziffer für Ortsgespräche/Ferngespräche ein.
  - Selektieren Sie das gewünschte Wählverfahren für den Standort.
6. Verlassen Sie diesen Dialog, indem Sie auf "OK" klicken.
7. Wählen Sie im nachfolgenden Dialog den Standort aus, von dem aus gewählt werden soll, und klicken Sie auf "OK".
8. Überprüfen Sie im nachfolgenden TeleService-Dialog, dass unter "Standort" der gewünschte Standort eingetragen ist.
9. Bestätigen Sie Ihre Eingaben, indem Sie auf "OK" klicken.

---

### Hinweis

#### Modembetrieb an Amtsleitung oder Nebenstelle

Wenn Sie Ihr Modem an einer Amtsleitung (Hauptanschluss) betreiben, brauchen Sie keine Amtskennziffer anzugeben. Die Felder der Amtskennziffer für Ortsgespräche und für Ferngespräche müssen leer sein.

Wenn Sie Ihr Modem an einer Nebenstelle betreiben, sollten Sie die Amtskennziffer eintragen, die gewählt werden muss, um eine Amtsleitung zu erhalten.

---

### Beispiel für die Verwendung von Wählregeln

Das nachfolgende Beispiel zeigt die Verwendung von standortspezifischen Wählregeln für die Stadt Karlsruhe in Deutschland.

- Sie tragen im Telefonbuch von TeleService für eine Anlage die Rufnummer "1234567" ein, ohne Vorwahl und ohne Landeskennzeichen.

**Ergebnis:**

Es wird immer die Rufnummer "1234567" angewählt, unabhängig vom Standort des Anrufers.

- Sie tragen im Telefonbuch von TeleService zusätzlich zur Rufnummer "1234567" noch die standortspezifischen Wahlparameter für die Vorwahl von Karlsruhe "0721" und das Landeskennzeichen "+49" für Deutschland ein.

**Ergebnis:**

Aus Karlsruhe wird die Telefonnummer "1234567" angewählt.

Aus einer anderen Stadt innerhalb Deutschlands wird die Telefonnummer "07211234567" angewählt.

Aus einem anderen Land wird die Telefonnummer "00497211234567" angewählt.

---

#### Hinweis

##### Anzeige im Telefonbuch

Im Telefonbuch von TeleService sind die Spalten "Ortskennzahl" und "Land" nur dann editierbar, wenn Sie die Wählregeln wie oben beschrieben definiert haben.

---

## 12.7.3 Fernverbindung als Wählverbindung

### 12.7.3.1 Grundlagen zum Aufbauen einer Wählverbindung

#### Einsetzen eines TS Adapters für Wählverbindungen

Für den Aufbau einer Fernverbindung als Wählverbindung mit TeleService ist der Einsatz eines TS Adapters erforderlich.

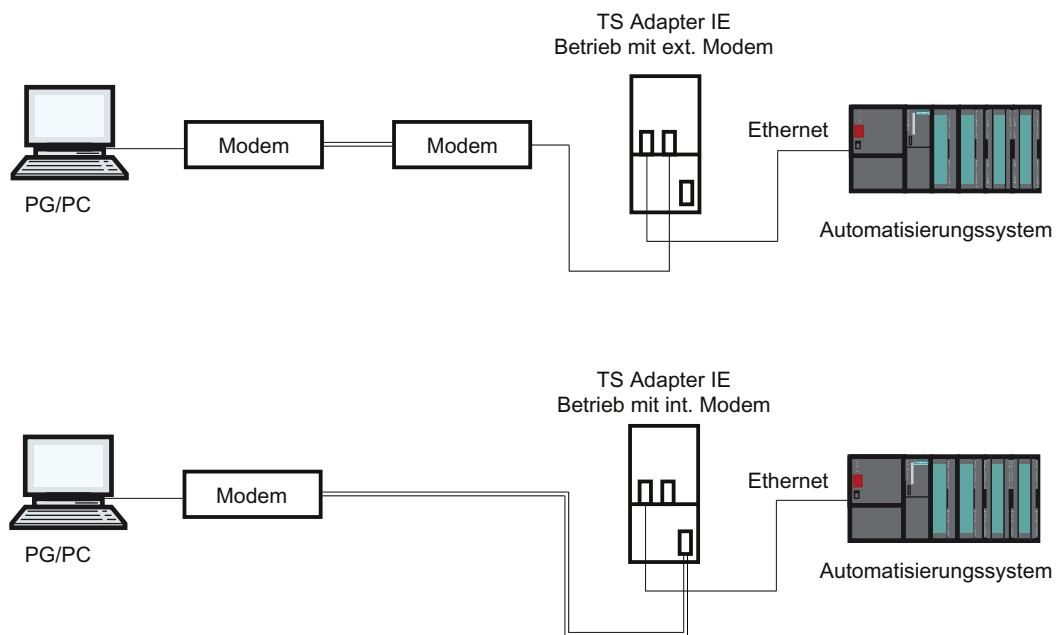
Der TS Adapter dient dazu, ein Automatisierungssystem über ein Modem an ein Telefonnetz anzuschließen, um es so für den Einsatz von TeleService vorzubereiten. Hierzu umfasst der TS Adapter einen Parameterspeicher, in dem ein Parametersatz für den TeleService-Betrieb gespeichert wird.

Durch die Funktion "Exportieren von Adapterparametern" können verschiedene Parametersätze in externen Dateien gespeichert und durch die Funktion "Importieren von Adapterparametern" wieder in den TS Adapter geladen werden.

### Aufbau der Fernverbindung als Wählverbindung

Sie haben die Auswahl zwischen verschiedenen TS Adaptern, die jeweils unterschiedliche Funktionalität und Anschlussmöglichkeiten bieten.

Das folgende Bild zeigt zwei mögliche Konfigurationen, um mit dem TS Adapter IE eine Wählverbindung zu einer Anlage aufzubauen.



### Übersicht der einsetzbaren TS Adapter:

Den TS Adapter gibt es in den folgenden Ausführungen:

- TS Adapter II (auch als "TS Adapter MPI" bezeichnet)
- TS Adapter IE Standard (auch als "TS Adapter IE" bezeichnet)
- TS Adapter IE Basic (auch als "TS Adapter IE" bezeichnet)

### Bezeichnung "TS Adapter"

Nachfolgend steht die Bezeichnung "TS Adapter" für alle Ausführungen. Bei Aussagen, die nur für eine bestimmte Ausführung Gültigkeit haben, wird die entsprechende Produktbezeichnung aufgelistet, also z. B. "TS Adapter II" oder "TS Adapter IE Standard".

---

#### Hinweis

Weitere Informationen zum von Ihnen verwendeten TS Adapter entnehmen Sie bitte der zu Ihrem TS Adapter mitgelieferten Dokumentation.

---



## Siehe auch

- Kurzbeschreibung des TS Adapter MPI (Seite 4355)
- Kurzbeschreibung TS Adapter IE (Seite 4362)
- Exportieren von Adapterparametern (Seite 4361)
- Importieren von Adapterparametern (Seite 4362)

## 12.7.3.2 Telefonnetze und Modems

### Zulässige Telefonnetze und Modems

#### Einsetzbare Telefonnetze

TeleService ist mit digitalen Netzen (ISDN), analogen Netzen und Funknetzen (mit GSM-Technologie) einsetzbar. Unterstützt wird in dieser Version eine Fernverbindung zu einem TS Adapter.

#### Modem-Unterstützung

TeleService wurde Modem-unabhängig realisiert. Somit können alle handelsüblichen Modems, die in der Systemsteuerung von Windows installierbar und dort als Modem sichtbar sind, auch von TeleService genutzt werden.

Die Auswahl der Modemtypen wird in erster Linie durch die vorhandene Hardware des PG/PC und das einzusetzende Telefonnetz bestimmt.

#### Folgende Modemtypen/Medien werden unterstützt:

- Modems (externe Modems an COM-Schnittstelle, interne Modems und PCMCIA-Karten)
- Externer ISDN-Adapter an COM-Schnittstelle bzw. USB-Schnittstelle
- Interner ISDN-Adapter mit virtueller COM-Schnittstelle (z.B. AVM CAPI-Port)
- externe ISDN-Modems (ISDN-Adapter mit integrierter Analog-Modem-Funktionalität) an COM-Schnittstelle bzw. USB-Schnittstelle
- Funknetz-Modems mit GSM-Technologie, PCMCIA-Adapterkarte bzw. Datenkabel und Mobiltelefon

#### Netzübergänge

Netzübergänge zwischen den verschiedenen Telefonnetzen sind prinzipiell möglich. Fernverbindungen von einem ISDN-Adapter zu einem Analog-Modem und umgekehrt funktionieren nur mit speziellen ISDN-Telefon-Adaptern.

## Leistungsfähigkeit in Telefonnetzen

Der Datendurchsatz einer Fernverbindung hängt von dem eingesetzten Modem, dem verwendeten Telefonnetz und von der Qualität der Telefonleitung ab.

## Installieren des lokalen Modems

### Einführung

Haben Sie unter Ihrem Betriebssystem bereits ein Modem für die Datenübertragung installiert, so können Sie dieses Modem auch für TeleService verwenden.

Ist dem Betriebssystem noch kein Modem bekannt, müssen Sie ein Modem installieren, um eine Fernverbindung mit TeleService aufbauen zu können.

### Vorgehen

Gehen Sie folgendermaßen vor:

1. Stellen Sie sicher, dass Ihr PG/PC und das Modem ausgeschaltet sind.
2. Schließen Sie das externe Modem physikalisch an eine COM- oder USB-Schnittstelle Ihres PG/PC an. Sie können auch ein internes Modem bzw. eine PCMCIA-Karte entsprechend den Angaben des Herstellers einbauen.
3. Schalten Sie nun zuerst das Modem und anschließend das PG bzw. den PC ein.

### Ergebnis

Plug&Play-fähige Modems werden vom Betriebssystem automatisch erkannt und installiert. Sie werden mit Hilfe von Dialogen durch den Installationsvorgang geführt.

---

#### Hinweis

##### Nicht Plug&Play-fähige Modems

Wenn Ihr Modem beim Einschalten nicht automatisch erkannt wird, so müssen Sie es über die Systemsteuerung selbst installieren.

Beachten Sie dabei die Hinweise in der zu Ihrem Modem mitgelieferten Dokumentation.

---

## Anschließen und Parametrieren des entfernten Modems

### Einführung

Eine Voraussetzung für das Arbeiten mit TeleService ist, dass an der entfernten Anlage ebenfalls ein Modem angeschlossen ist. Dieses Modem wird als "entferntes Modem" bezeichnet.

### **Parametrieren des entfernten Modems**

Das Modem erhält alle zum Betrieb erforderlichen Parameter von dem angeschlossenen TS Adapter. Dies sind u. a. die Daten zur Initialisierung des Modems und die Einstellungen zur seriellen Übertragung zwischen TS Adapter und Modem.

Die für das entfernte Modem benötigten Daten werden bereits beim Parametrieren des TS Adapters festgelegt.

Abhängig vom eingesetzten TS Adapter kann es sich um ein internes oder externes Modem handeln.

### **Vorgehen zum Anschließen eines TS Adapter mit internem Modem**

1. Schalten Sie den TS Adapter aus.
2. Schließen Sie den TS Adapter an das Automatisierungssystem an.
3. Schließen Sie den TS Adapter an den Telefonanschluss an.
4. Schalten Sie den TS Adapter ein.

### **Vorgehen zum Anschließen eines TS Adapter mit externem Modem**

1. Schalten Sie das Modem aus.
2. Schließen Sie den TS Adapter an das Automatisierungssystem an.
3. Schließen Sie den TS Adapter mit einem Modemkabel an das Modem an.
4. Schließen Sie das Modem an den Telefonanschluss an.
5. Schalten Sie das Modem ein.
6. Schalten Sie den TS Adapter ein.

---

#### **Hinweis**

##### **Beachten Sie die folgenden Hinweise zum Parametrieren des entfernten Modems:**

- Die im TS Adapter voreingestellten Parameter für das Modem und die serielle Schnittstelle sollten in den meisten Fällen einen erfolgreichen Betrieb gewährleisten, so dass nur in seltenen Ausnahmefällen eine Umparametrierung erforderlich ist.
  - Eine Umparametrierung des TS Adapter ist nur dann nötig, wenn keine Modemverbindung zu Stande kommt oder Werkseinstellungen angepasst oder optimiert werden sollen.
  - Der TS Adapter kann entweder über eine Direktverbindung oder eine Fernverbindung umparametriert werden.
-

### 12.7.3.3 Zugriffsschutz bei Wählverbindungen

#### Informationen zum Zugriffsschutz

##### Einführung

Beim Parametrieren Ihres TS Adapters können Sie sowohl den Zugriff auf die Parametrierung des TS Adapters selbst als auch den Zugriff auf entfernte Anlagen schützen.

##### Gültigkeit des Zugriffsschutzes

Der Zugriffsschutz besteht beim TS Adapter MPI ausschließlich für Fernverbindungen, im Direktanschluss kann man jederzeit auf die Parametrierung des TS Adapters zugreifen.

Beim TS Adapter IE besteht der Zugriffsschutz auch im Direktanschluss.

#### Informationen zum Zugriffsschutz

Bei Auslieferung des TS Adapters MPI ist werkseitig kein Zugriffsschutz aktiviert. Beim TS Adapter IE gibt es ein Default-Passwort.

Der erste Benutzer, der den Adapter parametriert, kann also den Zugriffsschutz aktivieren, indem er das Kennwort für einen Benutzer und/oder eine Rückrufnummer definiert.

Es handelt sich um einen mehrstufigen Zugriffsschutz mit mehreren Benutzern, jeweils mit oder ohne Administratorrechte. Beim TS Adapter MPI gibt es nur einen Administrator und maximal zwei Benutzer.

Nur ein Administrator darf andere Benutzer einrichten, deren Einstellungen ändern und diese bei Bedarf auch wieder löschen. Melden Sie sich als Benutzer ohne Administratorrechte an, so können Sie nur das eigene Kennwort und die eigene Rückrufnummer ändern. Beim TS Adapter MPI kann man im Direktanschluss jedoch uneingeschränkt auf die Parametrierung des TS Adapters zugreifen.

##### Vorteile

Der Zugriffsschutz bietet Ihnen folgende Vorteile:

- Ein unberechtigter Zugriff betriebsfremder Personen ist nahezu ausgeschlossen.
- Der Anlagenbetreiber trägt den Hauptteil der Telefongebühren.

#### Rückrufmöglichkeiten in TeleService

##### Rückrufvarianten

Die Kosten einer Telefonverbindung trägt normalerweise der Anrufer, d.h. derjenige, der die Wählverbindung aufbaut.

TeleService kann jedoch auch so genutzt werden, dass nach einer kurzen Erstverbindung die Modemverbindung erneut in umgekehrter Richtung, d.h. vom TS Adapter ausgehend,

aufgebaut wird (Rückruf). In diesem Fall trägt der Betreiber der Anlage die Kosten für den Rückruf.

### TeleService unterscheidet zwei Rückrufvarianten:

1. Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer.
2. Rückruf zu einer im TS Adapter hinterlegten Rufnummer.

### Informationen zu den Schutzstufen

#### Einführung

Bei Einrichten des Zugriffsschutzes für den Zugriff durch TeleService auf den TS Adapter gibt es zwei Schutzstufen, die mit unterschiedlichen Optionen verbunden sind.

#### Optionen für den Zugriffsschutz

##### Zugriffsschutzstufe 1:

Der TS Adapter ist durch den Benutzernamen und das Kennwort geschützt. Sie können auf den TS Adapter über beliebige Telefonanschlüsse zugreifen und optional beim Verbindungsaufbau eine beliebige Rückrufnummer angeben.

##### Zugriffsschutzstufe 2:

Der TS Adapter ist durch den Benutzernamen, das Kennwort und die Rückrufnummer geschützt. Sie können auf den TS Adapter nur über einen Telefonanschluss pro Benutzer zugreifen.

Die nachfolgende Tabelle zeigt die o. g. Bedingungen für die unterschiedlichen Schutzstufen:

| Zugriffsschutzstufe | Administrator-/Benutzer-Kennwort | Rückrufnummer   |
|---------------------|----------------------------------|-----------------|
| 1                   | eintragen                        | nicht eintragen |
| 2                   | eintragen                        | eintragen       |

#### Anmelden am TS Adapter

Beim Anmelden am TS Adapter geben Sie, nachdem Sie den Zugriffsschutz eingerichtet haben, den Benutzernamen, das dazugehörige Kennwort und optional eine Rückrufnummer ein:

| Zugriffsschutzstufe | Administrator-/Benutzer-Kennwort | Rückrufnummer                                |
|---------------------|----------------------------------|--|
| 1                   | eintragen                        | nicht eintragen oder beliebige Rückrufnummer |
| 2                   | eintragen                        | nicht eintragen                              |

Haben Sie beim Verbindungsaufbau eine Rückrufnummer eingetragen, (Zugriffsschutzstufe 1) oder eine Rückrufnummer im TS Adapter hinterlegt (Zugriffsschutzstufe 2), so wird die Modemverbindung getrennt und der TS Adapter ruft die angegebene Nummer zurück.

## Zugriffsschutz und Rückrufnummer für den TS Adapter einrichten

### Einführung

In TeleService können Sie beim Parametrieren des TS Adapters einen Zugriffsschutz und eine Rückrufnummer für die Parametrierung des Adapters und für die Verbindung zur entfernten Anlage einrichten. Nachfolgend ist die Parametrierung für einen TS Adapter MPI beschrieben. Die Parametrierung eines TS Adapters IE erfolgt analog. Die genaue Vorgehensweise ist in der Webhilfe dieses Adapters beschrieben.

### Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und wird unter "Erreichbare Teilnehmer" in der Projektnavigation angezeigt.

### Vorgehen

Um den Zugriffsschutz für den TS Adapter einzurichten, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation auf den Befehl "TS Adapter MPI parametrieren".
2. Öffnen Sie das Register "Zugriffsschutz".
3. Geben Sie hier für Ihren Benutzernamen ein Kennwort und/oder eine Nummer ein, die das Modem nach dem Anmelden zurückrufen soll.
  - Sind Sie als Administrator angemeldet, so können Sie alle Einstellungen für Administratoren und Benutzer ändern sowie Benutzer einrichten und löschen.
  - Sind Sie als Benutzer angemeldet, können Sie nur Ihre eigenen Einstellungen (Kennwort und Rückrufnummer) ändern.
4. Bestätigen Sie alle Eingaben vor dem Verlassen des Dialoges mit "OK".
5. Drücken Sie die Schaltfläche "Ja" zum Bestätigen der nachfolgenden Abfrage.

### Ergebnis

Die Parametrierung für den Zugriffsschutz und die Rückrufnummer wird im nichtflüchtigen Speicher des TS Adapters MPI gespeichert.

---

### **Hinweis**

#### **Beim Einrichten des Zugriffsschutzes ist unbedingt zu beachten:**

- Um die Rückruf-Funktionalität zu gewährleisten, sollten Sie sicherstellen, dass die Einstellungen im Register "Modem" den Gegebenheiten an der Anlage entsprechen.
  - Wenn Sie unter dem Benutzer "ADMIN" eine falsche Rückrufnummer eintragen, so können Sie den TS Adapter MPI über eine Fernverbindung nicht mehr erreichen!
  - Testen Sie die Rückrufnummer, bevor Sie diese unter dem Benutzer "ADMIN" eintragen, durch einen Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer (Zugriffsschutzstufe 1).
- 

## **Rückruf in TeleService durchführen**

### **Rückrufmöglichkeiten**

In TeleService besteht die Möglichkeit, zwei unterschiedliche Rückrufvarianten einzurichten. Folgende Rückrufmöglichkeiten stehen zur Verfügung:

- Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer.
- Rückruf zu einer im TS Adapter hinterlegten Rufnummer

### **Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer**

1. Klicken Sie in der Projektnavigation des TIA-Portals auf den Ordner "Online-Zugänge".
2. Klicken Sie auf den darin enthaltenen Ordner "TeleService".
3. Doppelklicken Sie auf den Eintrag "Fernverbindung aufbauen/trennen". Der Dialog "Fernverbindung zur entfernten Anlage aufbauen" wird geöffnet.
4. Wählen Sie in der Klappliste "Adapter-Typ" den von Ihnen verwendeten Adapter-Typ aus.
5. Wählen Sie unter "Verbindungs-Typ" die "Wählverbindung" aus, sofern nicht bereits ausgewählt.
6. Wählen Sie unter "Lokale Einstellungen" das von Ihnen verwendete Modem aus.
7. Tragen Sie die anzuwählende Telefonnummer in das entsprechende Eingabefeld ein oder öffnen Sie das Telefonbuch, indem Sie auf die dahinterliegende Schaltfläche klicken und übernehmen Sie die gewünschte Telefonnummer aus dem Telefonbuch.
8. Tragen Sie den Benutzernamen und das zugehörige Kennwort des TS Adapters ein.
9. Falls Sie einen "Verbindungsaufbau mit Rückruf" wünschen, selektieren Sie das entsprechende Optionsfeld.

10. Klicken Sie auf die Schaltfläche "Aufbauen", um die gewünschte Fernverbindung aufzubauen. Diese Schaltfläche wird erst dann aktiv, wenn Sie alle für den Aufbau einer Fernverbindung benötigten Parameter eingetragen haben. Unter "Status" wird angezeigt, ob bereits eine Fernverbindung besteht oder nicht.
11. Tragen Sie ggf. im nachfolgenden Dialog die gewünschte Rückrufnummer ein.

## Ergebnis

Die Fernverbindung zur gewünschten Anlage mit Rückruf wird aufgebaut.

In der Projektnavigation wird die verbundene Anlage mit dem entsprechenden Symbol angezeigt.

---

### Hinweis

Diese Vorgehensweise bietet sich an, wenn die Kosten für die Modemverbindung auf Anlagenseite auflaufen sollen und wenn die eigentliche Rückrufnummer variabel gehandhabt werden soll, d.h. wenn der Rückruf nicht immer an denselben Empfänger geleitet werden soll. Dies ist vor allem auch für solche Anwender interessant, die mobil sind.

---

## Rückruf zu einer im TS Adapter hinterlegten Rufnummer

1. Parametrieren Sie die gewünschte Rückrufnummer im TS Adapter.
2. Bauen Sie eine Verbindung zum TS Adapter auf, wie oben beschrieben und beachten Sie dabei folgende Besonderheiten:
  - Tragen Sie den Benutzernamen und das Kennwort ein, für die die Rückrufnummer im TS Adapter parametrierung ist.
  - Das Optionsfeld "Verbindungsaufbau mit Rückruf" brauchen Sie nicht zu selektieren, da die Rückrufnummer dem TS Adapter bereits bekannt ist.

## Ergebnis

Der Rückruf zu einer im TS Adapter hinterlegten Rufnummer ist eingerichtet. Wenn eine Fernverbindung aufgebaut wird, erfolgt der Rückruf von der entfernten Anlage.

---

### Hinweis

Diese Vorgehensweise stellt die höchste Stufe des Zugriffsschutzes dar. Sie birgt jedoch die Gefahr, dass bei einer im TS Adapter falsch hinterlegten Rückrufnummer der TS Adapter nicht mehr über eine Modemverbindung erreicht werden kann und nur durch eine Umparametrierung vor Ort wieder in Betrieb genommen werden kann.

---



### 12.7.3.4 TS Adapter MPI

#### Kurzbeschreibung des TS Adapter MPI

##### TS Adapter MPI:

Die Bezeichnung "TS Adapter MPI" ist ein Sammelbegriff für alle TS Adapter, die über eine MPI/DP-Schnittstelle verfügen.

Den TS Adapter MPI gibt es in folgenden Ausführungen:

- als TS Adapter I (kann nicht über das TIA-Portal parametrierbar werden)
- als TS Adapter II

Die nachfolgende Tabelle gibt eine Kurzbeschreibung über die jeweilige Funktionalität. Ausführliche Informationen zu dem von Ihnen verwendeten TS Adapter entnehmen Sie bitte der zu Ihrem TS Adapter mitgelieferten Dokumentation.

| <b>TS Adapter II:</b>  |  |
|------------------------|--|
| Direktanschluss        | Der Anschluss erfolgt über den Universal Serial Bus (USB). Die Firmware ist austauschbar.<br><br>Das Modem ist integriert oder auch als externes Modem anschließbar. Der TS Adapter II schaltet automatisch zwischen den Modems um. Solange kein eingeschaltetes externes Modem angeschlossen ist, benutzt er das interne Modem. |
| Es gibt zwei Varianten | <ul style="list-style-type: none"><li>• Mit internem Analog-Modem. Zudem kann an der RS232-Schnittstelle ein externes Modem angeschlossen werden.</li><li>• Mit internem ISDN-Adapter. Zudem kann an der RS232-Schnittstelle ein externes Modem angeschlossen werden</li></ul>   |

#### Verwendung der Bezeichnung "TS Adapter"

Bei TeleService steht die Bezeichnung "TS Adapter" als Generalisierung für alle Ausführungen. Bei Aussagen, die nur für eine bestimmte Ausführung eines TS Adapters Gültigkeit haben, wird die entsprechende Produktbezeichnung aufgelistet, also z. B. "TS Adapter II", "TS Adapter IE Standard" oder "TS Adapter IE Basic".

#### Funktionsweise des TS Adapter MPI

##### Funktionsweise des TS Adapters MPI

Je nach Konfiguration verbindet der TS Adapter MPI die serielle Schnittstelle oder die USB-Schnittstelle Ihres PG/PC (Direktanschluss) oder die serielle Schnittstelle eines Modems (Modemanschluss) mit dem MPI-/PROFIBUS-Netz Ihres Automatisierungssystems.

Der TS Adapter MPI verfügt über einen nichtflüchtigen Speicher. In diesem Speicher sind Parameter für folgende Funktionen hinterlegt:

- das MPI/PROFIBUS-Netz (Netz-Parameter)
- den Betrieb des eingesetzten Modems
- die serielle Schnittstelle zum Modem
- den Zugriffsschutz

### Standard-Parametrierung

Der TS Adapter wird mit einer Standard-Parametrierung ausgeliefert. Die Parameter können im Rahmen einer Parametriersitzung eingestellt und in den nichtflüchtigen Speicher des TS Adapters gespeichert werden.

In der Konfiguration "Direktanschluss" werden vom TS Adapter nur die Netz-Parameter für den Zugang zum MPI-/PROFIBUS-Netz verwendet.

In der Konfiguration "Modemanschluss" werden alle auf dem TS Adapter abgelegten Parameter aktiviert.

---

#### Hinweis

Weitere Hinweise zur Parametrierung Ihres TS Adapters entnehmen Sie bitte der zum TS Adapter mitgelieferten Dokumentation.

---

### TS Adapter MPI im Direktanschluss betreiben

#### Direktanschluss mit TS Adapter MPI

Der Direktanschluss dient zum Parametrieren des TS Adapters MPI. In der gleichen Konfiguration können Sie auch im TIA-Portal online gehen und somit die parametrierten MPI-/PROFIBUS-Parameter auf Busverträglichkeit überprüfen. Damit ist (wie mit einem PC-Adapter) ein Zugang zu SIMATIC S7/C7-Systemen über die MPI-/DP-Schnittstelle möglich, ohne dass ein PG/PC-Steckplatz durch eine MPI-/PROFIBUS-Baugruppe belegt wird.

Im Direktanschluss ist kein Zugriffsschutz für den TS Adapter aktiv. Somit kann die Parametrierung des TS Adapters ohne weiteres verändert werden, z.B. durch das Importieren von Adapterparametern.

---

#### Hinweis

##### Anzeige des TS Adapters MPI im TIA-Portal

Sobald Sie einen TS Adapter MPI über die USB-Schnittstelle mit dem PG/PC verbunden haben, wird im TIA-Portal innerhalb der Projektnavigation der Ordner "TS Adapter" angezeigt.

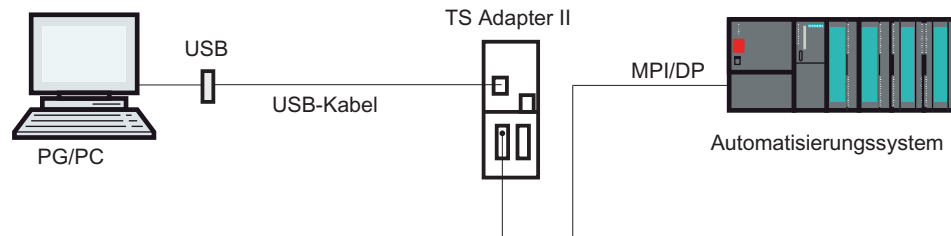
Wenn Sie den Ordner öffnen, können Sie den angeschlossenen TS Adapter MPI über die nachfolgenden Dialoge wie gewünscht parametrieren.

---

### Aufbau des Direktanschlusses für TS Adapter MPI

Beim Direktanschluss besteht über den TS Adapter MPI eine direkte Verbindung zwischen dem PG bzw. PC, auf dem TeleService installiert ist, und dem Automatisierungssystem. Ein Modem ist nicht erforderlich.

Das nachfolgende Bild zeigt den Aufbau des TS Adapters MPI im Direktanschluss.



### TS Adapter MPI im Modemanschluss betreiben

#### Einführung zum Modemanschlusses mit TS Adapter MPI

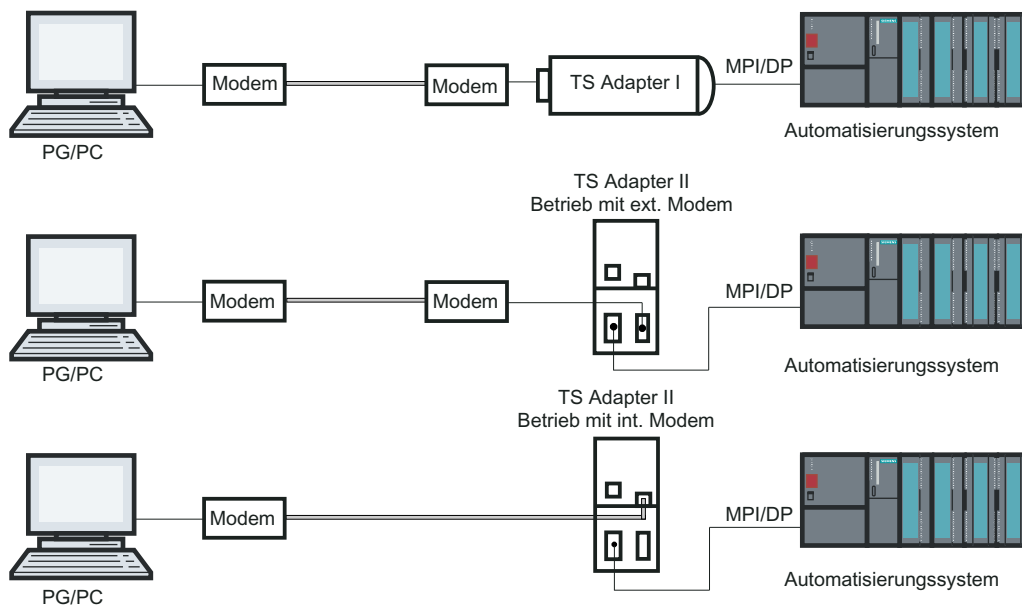
Diese Konfiguration ermöglicht es Ihnen, sich in eine entfernte Anlage ein zu wählen. Dazu bauen Sie mit TeleService über ein Telefonnetz eine Fernverbindung zu einer entfernten Anlage auf. Über die aufgebaute Modem-Verbindung können Sie dann die angewählte Anlage wie gewohnt mit dem TIA-Portal bearbeiten.

### Aufbau des Modemanschlusses mit TS Adapter MPI

Die Verbindung zwischen dem PG bzw. PC, auf dem TeleService installiert ist, und dem Automatisierungssystem, an dessen MPI-/DP-Schnittstelle der TS Adapter MPI aufgesteckt ist, wird über eine Modemstrecke aufgebaut.

Die Konfiguration schließt somit das PG bzw. den PC über das Telefonnetz und den TS Adapter MPI an die MPI-/DP-Schnittstelle des Automatisierungssystems an.

Das nachfolgende Bild zeigt den Aufbau im Modemanschluss.



---

### Hinweis

#### Parallelbetrieb zwischen Direkt- und Modemanschluss

Der TS Adapter II hat zwei Anschlüsse für die Kommunikation mit dem PG/PC, die beide gleichzeitig angeschlossen sein können. Sie verbinden die USB-Schnittstelle mit dem PG/PC und gleichzeitig die Modemschnittstelle mit dem Telefonnetz.

In dieser Konfiguration können Sie entweder den Direkt- oder den Modemanschluss nutzen.

Ein Parallelbetrieb ist **nicht** möglich!

---

## Parametriermöglichkeiten für den TS Adapter MPI

### Wissenswertes zum Parametrieren des TS Adapter MPI

Der TS Adapter MPI kann sowohl im Direktanschluss als auch über eine bestehende Fernverbindung parametrierbar werden.

Folgende Parametriermöglichkeiten stehen zur Verfügung:

- Neuparametrierung (Seite 4359)
- Wiederherstellen der Standard-Parametrierung (Seite 4360)
- Importieren von Adapterparametern (Seite 4362)
- Exportieren von Adapterparametern (Seite 4361)
- Zugriffsschutz einrichten (Seite 4352)

## Parametrierung durchführen

Parametrieren Sie Ihren TS Adapter unter Berücksichtigung der zum TS Adapter mitgelieferten Dokumentation. Dort finden Sie die benötigten Informationen zur exakten Vorgehensweise beim Parametrieren.

---

### Hinweis

#### **Beachten Sie beim Parametrieren des TS Adapters MPI:**

- Wenn Sie bei bestehender Fernverbindung die aktuellen Parameter ändern, so besteht die Gefahr, dass anschließend mit den geänderten Parametern keine Modemverbindung mehr aufgebaut werden kann. In diesem Fall kann der TS Adapter MPI nur noch im Direktanschluss parametrieren werden.
  - Das bedeutet, dass die Parametrierung mit einem PG/PC am Ort der Anlage vorgenommen werden muss oder dass der TS Adapter MPI zum Parametrieren zum Standort des lokalen PG/PC gebracht werden muss.
- 

## Positive Quittierung

Bei einer Parametrierung werden die Daten in den nichtflüchtigen Speicher des TS Adapters MPI geschrieben. Erst nachdem alle Vorkehrungen getroffen sind, dass die Parameteränderung richtig durchgeführt wurde und somit einen Spannungsausfall überlebt, wird der Parametriervorgang positiv quittiert.

## Änderungen werden beim TS Adapter MPI wie folgt wirksam:

- Die seriellen Parameter, die Modem-Parameter und die Parameter für den Zugriffsschutz werden nach dem Abbau der Fernverbindung aktiviert.
- Die geänderten Netz-Parameter werden sofort aktiviert.

## TS Adapter MPI parametrieren

### Einführung

Sie können den TS Adapter MPI sowohl im Direktanschluss als auch über eine bereits aufgebaute Fernverbindung im Modemanschluss parametrieren.

Nachfolgend wird die Vorgehensweise für das Parametrieren beschrieben.

### Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und der Ordner "TS Adapter" wird unter "Online Zugänge" in der Projektnavigation angezeigt.

## Vorgehen

Um den TS Adapter MPI im Direktanschluss zu parametrieren gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in der Projektnavigation auf den Ordner "Online Zugänge".
2. Öffnen Sie den gewünschten Ordner:
  - Bei einer Direktverbindung den Ordner "TS Adapter".
  - Bei einer bestehenden Fernverbindung den Ordner "TeleService" und dann den Ordner mit dem gewünschten Anlagennamen.
3. Wählen Sie den Befehl "TS Adapter MPI parametrieren". Daraufhin öffnet sich der Dialog "TS Adapter MPI parametrieren".
4. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
5. Bestätigen Sie Ihre Eingaben mit "OK".

## Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters MPI gespeichert. Die Parametrierung ist damit abgeschlossen.

## Standard-Parametrierung für TS Adapter MPI wieder herstellen

### Einführung

Für den TS Adapter MPI besteht die Möglichkeit, die beim Auslieferungszustand vorhandene Standardparametrierung wieder herzustellen.

### Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und wird in der Projektnavigation unter "Online Zugänge" im Ordner "TeleService" angezeigt.

## Vorgehen

Um die Standardparametrierung für den TS Adapter MPI wieder herzustellen gehen Sie folgendermaßen vor:

1. Öffnen Sie auf den Ordner "TeleService" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "TS Adapter MPI".
3. Wählen Sie den Befehl "TS Adapter MPI parametrieren". Daraufhin öffnet sich der Dialog "TS Adapter MPI parametrieren".
4. Klicken Sie unter "Allgemein" auf die Schaltfläche "Zurücksetzen".
5. Bestätigen Sie Ihre Eingaben mit "OK".

## Ergebnis

Die Standardparametrierung, mit der der TS Adapter MPI ausgeliefert wurde, ist wieder hergestellt.

## Siehe auch

Parametriermöglichkeiten für den TS Adapter MPI (Seite 4358)

## Exportieren von Adapterparametern

### Einführung

Es besteht die Möglichkeit, die Parametrierung eines TS Adapters MPI in eine externe Datei zu exportieren. Die in dieser Datei abgelegte Parametrierung können Sie wiederum in beliebig viele TS Adapter MPI importieren.

Dies bietet sich beispielsweise an, wenn Sie mehrere TS Adapter MPI identisch parametrieren möchten oder wenn Sie den Parametersatz speichern, dokumentieren oder verteilen möchten.

### Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und wird in der Projektnavigation unter "Online Zugänge" im Ordner "TeleService" angezeigt.

### Vorgehen

Um die Adapterparameter eines TS Adapters MPI zu exportieren gehen Sie folgendermaßen vor:

1. Öffnen Sie auf den Ordner "TeleService" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "TS Adapter MPI".
3. Wählen Sie den Befehl "TS Adapter MPI parametrieren". Daraufhin öffnet sich der Dialog "TS Adapter MPI parametrieren".
4. Klicken Sie auf die Schaltfläche "Exportieren".
5. Wählen Sie im nachfolgenden Fenster die Datei aus, in die Sie die Parametrierung des TS Adapters MPI exportieren möchten.
6. Bestätigen Sie mit "Speichern".

## Ergebnis

Die Parameter des TS Adapters MPI werden in der angegebenen Datei (\*.tap) gespeichert. Damit ist der Export der Adapterparameter abgeschlossen.

## Importieren von Adapterparametern

### Einführung

Es besteht die Möglichkeit, die Parametrierung eines TS Adapters MPI aus einer zuvor angelegten Export-Datei (\*.tap) zu importieren.

Die in dieser Datei abgelegte Parametrierung können Sie in beliebig viele TS Adapter importieren. Dies bietet sich beispielsweise an, wenn Sie mehrere TS Adapter MPI identisch parametrieren möchten.

Das Importieren können Sie lokal im Direktanschluss oder über eine aufgebaute Fernverbindung im Modemanschluss vornehmen.

### Voraussetzung

Ein TS Adapter MPI ist an Ihren Rechner angeschlossen und wird in der Projektnavigation unter "Online Zugänge" im Ordner "TeleService" angezeigt.

### Vorgehen

Um die Adapterparameter eines TS Adapters MPI zu importieren gehen Sie folgendermaßen vor:

1. Öffnen Sie den Ordner "TeleService" in der Projektnavigation.
2. Doppelklicken Sie auf den Ordner "TS Adapter MPI".
3. Wählen Sie den Befehl "TS Adapter MPI parametrieren". Daraufhin öffnet sich der Dialog "TS Adapter MPI parametrieren".
4. Klicken Sie auf die Schaltfläche "Importieren".
5. Wählen Sie nun die Datei aus, in die Sie die Parametrierung des TS Adapters MPI importieren möchten.
6. Bestätigen Sie den nachfolgenden Dialog mit "Ja".

### Ergebnis

Die ausgewählten Parameter werden im nichtflüchtigen Speicher des TS Adapters MPI gespeichert. Der Import der Adapterparameter ist damit abgeschlossen.

## 12.7.3.5 TS Adapter IE

### Kurzbeschreibung TS Adapter IE

#### TS Adapter IE

Die Bezeichnung "TS Adapter IE" ist ein Sammelbegriff für alle TS Adapter, die über eine Ethernet-Schnittstelle verfügen.



Den TS Adapter IE gibt es in folgenden Ausführungen:

- als TS Adapter IE Standard
- als TS Adapter IE Basic

Die nachfolgenden Tabellen geben eine Kurzbeschreibung über die jeweilige Funktionalität. Ausführliche Informationen zu dem von Ihnen verwendeten TS Adapter entnehmen Sie bitte der zu Ihrem TS Adapter mitgelieferten Dokumentation.

|  |
|--|
| <b>TS Adapter IE Standard:</b>   |
| Direktanschluss über Industrial Ethernet (IE). Firmwareupdate möglich. Modem integriert oder auch als externes Modem anschließbar. Keine automatische Modemumschaltung wie beim TS Adapter II. Die Parametrierung erfolgt über ein Web-Interface.                                  |
| <b>Es gibt 2 Varianten:</b>  |
| <ul style="list-style-type: none"> <li>• Mit internem Analog-Modem. Zudem kann an der RS232-Schnittstelle ein externes Modem angeschlossen werden.</li> <li>• Mit internem ISDN-Adapter. Zudem kann an der RS232-Schnittstelle ein externes Modem angeschlossen werden.</li> </ul> |

|   |
|---|
| <b>TS Adapter IE Basic:</b>   |
| Direktanschluss über Industrial Ethernet (IE). Firmwareupdate möglich. Steckbare Module. Die Parametrierung erfolgt über ein Web-Interface.   |
| <b>Es gibt 4 Varianten:</b>   |
| <ul style="list-style-type: none"> <li>• TS Adapter IE Basic MODEM:<br/>Grundgerät TS Adapter IE Basic mit TS Module MODEM für den Betrieb am analogen Telefonnetz.</li> <li>• TS Adapter IE Basic ISDN:<br/>Grundgerät TS Adapter IE Basic mit TS Module ISDN für den Betrieb an ISDN-Telefonanlagen.</li> <li>• TS Adapter IE Basic GSM:<br/>Grundgerät TS Adapter IE Basic mit TS Module GSM für den Betrieb am GSM-Funknetz.</li> <li>• TS Adapter IE Basic RS232:<br/>Grundgerät TS Adapter IE Basic mit TS Module RS232 zum Anschluss eines externen Modems.</li> </ul> |

### Verwendung der Bezeichnung "TS Adapter"

In der Online-Hilfe zu TeleService steht die Bezeichnung "TS Adapter" als Generalisierung für alle Ausführungen. Bei Aussagen, die nur für eine bestimmte Ausführung eines TS Adapters Gültigkeit haben, wird die entsprechende Produktbezeichnung aufgelistet, also z. B. "TS Adapter I", "TS Adapter II", "TS Adapter IE Standard" oder "TS Adapter IE Basic".

### Funktionsweise des TS Adapters IE

#### Funktionsweise des TS Adapters IE

Der TS Adapter IE verbindet das Telefonnetz oder die serielle Schnittstelle eines Modems mit dem Industrial Ethernet Ihres Automatisierungssystems.

Der TS Adapter IE verfügt über einen nichtflüchtigen Speicher. In diesem Speicher sind Parameter für folgende Funktionen hinterlegt:

- den Betrieb des eingesetzten Modems
- die serielle Schnittstelle zum Modem
- den Zugriffsschutz

### Standard-Parametrierung

Der TS Adapter IE wird mit einer Standard-Parametrierung ausgeliefert. Die Parameter können im Rahmen einer Parametriersitzung eingestellt und in den nichtflüchtigen Speicher des TS Adapters gespeichert werden.

---

#### Hinweis

Weitere Hinweise zur Parametrierung Ihres TS Adapters entnehmen Sie bitte der zum TS Adapter mitgelieferten Dokumentation.

---

### Anschlussarten

#### Anschlussarten des TS Adapter IE Basic

Die folgenden Bilder zeigen die möglichen Anschlussarten des TS Adapter IE Basic.

#### Direktanschluss

Im Direktanschluss am PG/PC können Sie den TS Adapter IE Basic über Ethernet parametrieren.

---

#### Hinweis

Der Betrieb des TS Adapter IE Basic ohne ein TS Modul ist unzulässig.

---

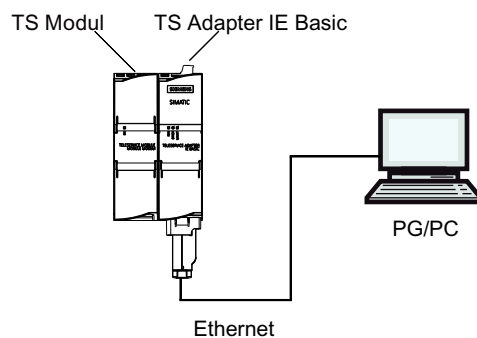


Bild 12-1 Direktanschluss

### Anschluss an das Telefonnetz

Zum direkten Anschluss an das Telefonnetz müssen Sie den TS Adapter IE Basic zusammen mit einem der folgenden TS Module betreiben:

- TS Module Modem
- TS Module ISDN

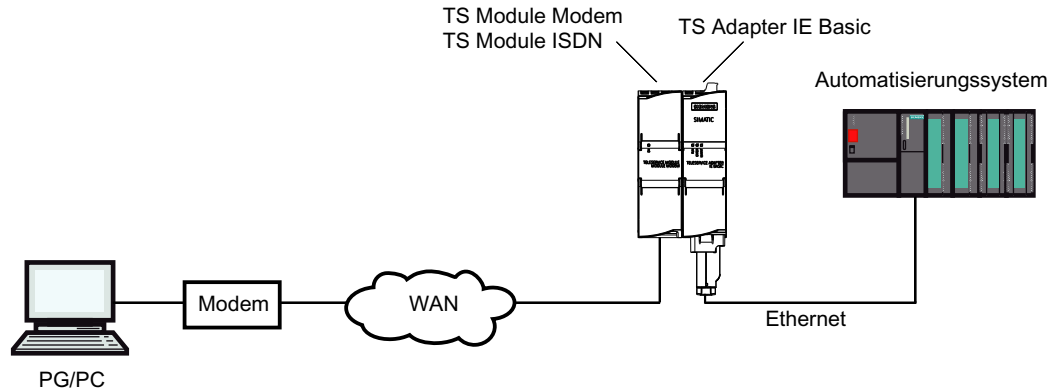


Bild 12-2 Direkter Anschluss an das Telefonnetz

Weitere Informationen zu den TS Modulen finden Sie im Handbuch *TS Adapter modular*.

### Anschluss an das GSM-Netz

Zum Anschluss an das GSM-Netz müssen Sie den TS Adapter IE Basic zusammen mit diesem TS Module betreiben:

- TS Module GSM

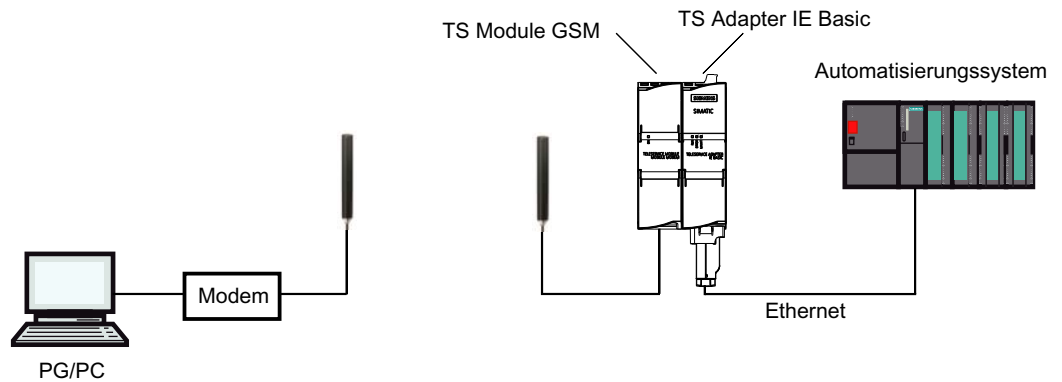


Bild 12-3 Anschluss an das GSM-Netz

Weitere Informationen zu den TS Modulen finden Sie im Handbuch *TS Adapter modular*.

### Anschluss an das Telefonnetz über ein externes Modem

Zum Anschluss an ein externes Modem müssen Sie den TS Adapter IE Basic zusammen mit diesem Modul betreiben:

- TS Module RS232

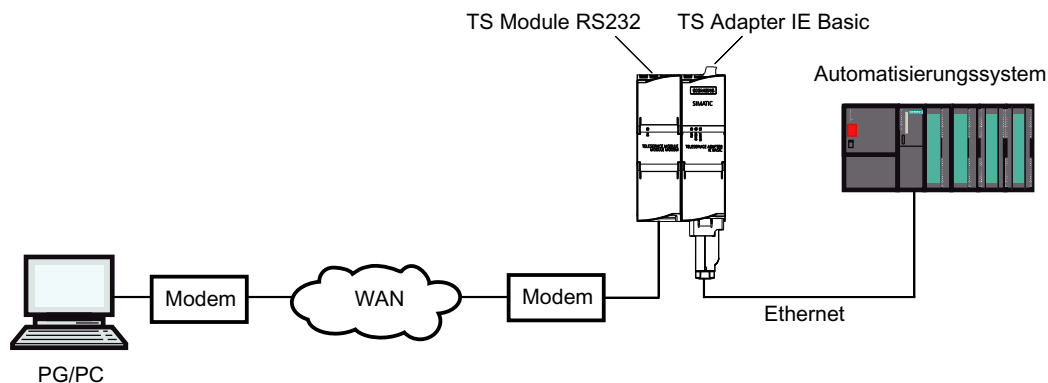


Bild 12-4 Anschluss an ein externes Modem

Weitere Informationen zu den TS Modulen finden Sie im Handbuch *TS Adapter modular*.

### Parametriermöglichkeiten für den TS Adapter IE

#### Wissenswertes zum Parametrieren des TS Adapter IE

Der TS Adapter IE wird über ein Web-Interface parametrier.

Für die Parametrierung des TS Adapter IE steht Ihnen eine dem Web-Interface zugeordnete Webhilfe zur Verfügung.

Folgende Parametriermöglichkeiten stehen zur Verfügung:

- Neuparametrierung
- Wiederherstellen der Standard-Parametrierung
- Importieren von Adapterparametern
- Exportieren von Adapterparametern

---

#### Hinweis

##### Parametrierung durchführen

Parametrieren Sie Ihren TS Adapter unter Berücksichtigung der zum TS Adapter mitgelieferten Dokumentation. Dort finden Sie die benötigten Informationen zur exakten Vorgehensweise beim Parametrieren.

---

## **TS Adapter IE parametrieren**

### **Einführung**

Sie können den TS Adapter IE sowohl im Direktanschluss als auch über eine bereits aufgebaute Fernverbindung im Modemanschluss parametrieren.

Nachfolgend werden beide Parametriermöglichkeiten beschrieben.

Spezifische Details zur Parametrierung des TS Adapters IE entnehmen Sie bitte der Dokumentation zum TS Adapter IE.

### **Parametrieren des TS Adapter IE im Direktanschluss**

#### **Voraussetzung**

Es besteht eine LAN-Verbindung zu Ihrem TS Adapter IE .

Der TS Adapter IE Basic ist an die Stromversorgung angeschlossen.

#### **Vorgehen**

Um den TS Adapter IE zu parametrieren gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation des TIA-Portals den Ordner "Online Zugänge".
2. Doppelklicken Sie hier auf die Ethernet-Schnittstelle Ihres Rechners.
3. Doppelklicken Sie auf den Befehl "Erreichbare Teilnehmer anzeigen". Daraufhin wird der TS Adapter IE angezeigt.
4. Doppelklicken Sie auf den Ordner <TS Adapter IE> und anschließend auf "Online und Diagnose" und weisen Sie dem TS Adapter IE in den nachfolgenden Dialogen die gewünschte IP-Adresse zu. Beachten Sie dabei, dass die IP-Adresse der Schnittstellenkarte des PG/PC im selben Subnetz liegt, wie die IP-Adresse, die Sie für den TS Adapter IE vergeben.
5. Aktualisieren Sie die Ansicht in der Projektnavigation für die "Erreichbaren Teilnehmer", damit der TS Adapter IE mit der neu zugewiesenen IP-Adresse angezeigt wird.
6. Öffnen Sie den Ordner <TS Adapter IE> in der Teilnehmerliste.
7. Doppelklicken Sie auf den Befehl "TS Adapter IE parametrieren". Daraufhin öffnet sich das zugeordnete Web-Interface zum Parametrieren des TS Adapters IE.
8. Führen Sie das "Logon" für das Web-Interface durch.
9. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
10. Bestätigen Sie Ihre Eingaben jeweils mit "Einstellungen speichern".

## Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters IE gespeichert. Die Parametrierung ist damit abgeschlossen.

## Parametrieren des TS Adapter IE über Fernverbindung

### Voraussetzung

Es besteht eine aufgebaute Fernverbindung zu einem TS Adapter IE .

### Vorgehen

1. Öffnen Sie in der Projektnavigation des TIA Portals den Ordner "Online Zugänge".
2. Öffnen Sie auf den Ordner "TeleService" und danach den gewünschten Anlagenordner.
3. Doppelklicken Sie auf den Befehl "TS Adapter IE parametrieren". Daraufhin öffnet sich das zugehörige Web-Interface zum Parametrieren des TS Adapters IE. Das "Logon" für das Web-Interface wird automatisch mit den Login-Daten der Fernverbindung durchgeführt.
4. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
5. Bestätigen Sie Ihre Eingaben jeweils mit "Einstellungen speichern".

## Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters IE gespeichert. Die Parametrierung ist damit abgeschlossen.

### 12.7.3.6 Aufbauen einer Wählverbindung zu einer entfernten Anlage

#### Wählverbindung aufbauen

#### Einführung zum Aufbauen einer Fernverbindung als Wählverbindung

Eine Wählverbindung kommt zustande, wenn Sie sich mit TeleService über ein Telefonnetz in eine entfernte Anlage einwählen. Hierzu ist das PG/der PC mit TeleService über ein lokales Modem an das Telefonnetz angeschlossen. Am anderen Ende ist das Automatisierungssystem über einen parametrierten TS Adapter und ein Modem mit der Telefonleitung verbunden.

#### Voraussetzungen

Ein lokales Modem ist installiert und parametriert.

In der entfernten Anlage befindet sich ein TS Adapter.

Ein entferntes Modem ist installiert und parametrier.

**Gehen Sie folgendermaßen vor:**

1. Klicken Sie in der Projektnavigation des TIA-Portals auf den Ordner "Online-Zugänge".
2. Klicken Sie auf den darin enthaltenen Ordner "TeleService".
3. Doppelklicken Sie auf den Eintrag "Fernverbindung aufbauen/trennen". Der Dialog "Fernverbindung zur entfernten Anlage aufbauen" wird geöffnet.
4. Wählen Sie in der Klappliste "Adapter-Typ" den von Ihnen verwendeten Adapter-Typ aus.
5. Wählen Sie unter "Verbindungs-Typ" die "Wählverbindung" aus.
6. Wählen Sie unter "Lokale Einstellungen" das von Ihnen verwendete Modem aus.
7. Tragen Sie die anzuwählende Telefonnummer in das entsprechende Eingabefeld ein oder öffnen Sie das Telefonbuch, indem Sie auf die dahinterliegende Schaltfläche klicken und übernehmen Sie die gewünschte Telefonnummer aus dem Telefonbuch.
8. Tragen Sie Ihren Benutzernamen und das zugehörige Kennwort ein.
9. Falls Sie einen "Verbindungsaufbau mit Rückruf" wünschen, selektieren Sie das entsprechende Optionsfeld.
10. Klicken Sie auf die Schaltfläche "Aufbauen", um die gewünschte Fernverbindung aufzubauen. Diese Schaltfläche wird erst dann aktiv, wenn Sie alle für den Aufbau einer Fernverbindung benötigten Parameter eingetragen haben. Unter "Status" wird angezeigt, ob bereits eine Fernverbindung besteht oder nicht.

**Ergebnis**

Die Wählverbindung zur gewünschten Anlage wird aufgebaut.

Sobald die Wählverbindung zustande gekommen ist, wird der Dialog geschlossen. In der Statuszeile des TIA-Portals erscheint die Meldung: "Fernverbindung ist aufgebaut". Sie können nun die Fernverbindung mit dem TIA-Portal nutzen und mit dem Automatisierungssystem kommunizieren.

**Verbindungsaufbau nicht möglich**

Ist kein Verbindungsaufbau möglich, so versuchen Sie bitte, die Fehlerursache mit Hilfe der "Hinweise zur Fehlersuche" zu ermitteln.

**Trennen der Verbindung**

Haben Sie die Bearbeitung der entfernten Anlage abgeschlossen, so beenden Sie die Fernverbindung in der Projektnavigation durch Doppelklicken auf den Eintrag "Fernverbindung aufbauen/trennen".

Durch Beenden des TIA-Portals beenden Sie ebenfalls die Fernverbindung.

## Wählverbindung trennen

### Trennen einer aktiven Wählverbindung

---

#### Hinweis

#### Trennen der Wählverbindung

Bevor Sie die Fernverbindung trennen, sollten Sie im TIA-Portal offline schalten.

---

#### Gehen Sie vor wie folgt:

1. Doppelklicken Sie im TIA-Portal auf den Eintrag "Fernverbindung aufbauen/trennen".
2. Bestätigen Sie die Abfrage im nachfolgenden Dialog mit "Ja".

#### Ergebnis

Die Verbindung wird getrennt.

## 12.7.4 Fernverbindung als VPN-Verbindung

### 12.7.4.1 Grundlagen zum Aufbauen einer VPN-Verbindung

#### Einsetzen eines TS Adapters IE Advanced für VPN-Verbindungen

Für den Aufbau einer VPN-Verbindung mit TeleService ist der Einsatz eines TS Adapter IE Advanced erforderlich.

#### Definition VPN

VPN steht für "Virtual Private Network". Damit ist ein geschlossenes privates Rechnernetz gemeint, das auf einer öffentlichen Netzwerk-Infrastruktur basiert.

VPN dient dazu, räumlich getrennte Teilnehmer eines privaten Netzes mit einer Anschlussstelle in einem anderen Netz möglichst sicher zu verbinden.

Ein VPN-Teilnehmer erhält über eine verschlüsselte Verbindung direkten Zugriff auf die Anschlussstelle im anderen Netz.



## VPN-Verbindungen

Beim Aufbau einer VPN-Verbindung mit TeleService wird zur eindeutigen Identifizierung des TS Adapters IE Advanced ein CA-Zertifikat und ein daraus generierter eindeutiger Fingerabdruck verwendet. Durch die VPN-Verbindung wird eine abhör- und manipulationssichere Kommunikation zwischen dem Remote-PC und dem TS Adapter ermöglicht.

Wenn Sie eine VPN-Verbindung zu mehreren entfernten Netzwerken aufbauen möchten, benötigen Sie für jedes Netzwerk einen eigenen TS Adapter IE Advanced als Schnittstelle. Die über den TS Adapter verbundenen Netzwerke können intern identische IP-Adressen verwenden. Lediglich die externen IP-Adressen (WAN) der jeweiligen TS Adapter müssen sich unterscheiden.

Es ist immer nur eine VPN-Verbindung zu einem TS Adapter gleichzeitig möglich.

---

### Hinweis

#### Authentifizierung

Beachten Sie bitte, dass das CA-Zertifikat ausschließlich den TS Adapter IE Advanced authentifiziert.

Ein Benutzer wird, wie bei Wählverbindungen auch, über das Login (Benutzername und Passwort) authentifiziert.

Verwenden Sie daher nur sichere Passwörter für das Login.

---

## Siehe hierzu auch

- VPN-Verbindung aufbauen (Seite 4377)
- VPN-Verbindung trennen (Seite 4378)

### 12.7.4.2 Grundlagen zu CA-Zertifikaten

#### Einführung

Um mit TeleService eine sichere VPN-Verbindung aufbauen zu können, ist es erforderlich, dass Sie bei der Konfiguration des TS Adapter IE Advanced ein CA-Zertifikat mit einem eindeutigen Fingerabdruck generieren.

Dieses CA-Zertifikat können Sie wie folgt auf jedem PC installieren, der Zugriff auf den TS Adapter IE Advanced erhalten soll:

- Über den automatischen Download im Verbindungsdialog von TeleService, indem Sie dort den zum CA-Zertifikat zugehörigen Fingerabdruck eintragen
- Über manuelle Installation mit Hilfe der Microsoft® Management Console.

### Definition CA-Zertifikat

Ein CA-Zertifikat ist ein durch eine Zertifizierungsstelle ("certificate authority" oder "certification authority", nachfolgend "CA" genannt) herausgegebenes digitales Zertifikat. Im Falls des TS Adapter IE Advanced werden selbst-signierte Zertifikate verwendet, die Zertifizierungsstelle ist in diesem Fall der TS Adapter IE Advanced selbst.

Vom CA-Zertifikat werden die Zertifikate für "SSTP" (Secure Socket Tunneling Protocol) und "HTTPS" (Hypertext Transfer Protocol Secure) abgeleitet.

CA-Zertifikate enthalten einen "Schlüssel" und Zusatzinformationen, die zur Authentifizierung sowie zur Verschlüsselung und Entschlüsselung vertraulicher Daten dienen. Als Zusatzinformationen sind zum Beispiel Gültigkeitsdauer, Verweise auf Zertifikatsperrlisten, etc. enthalten, die durch die CA mit in das Zertifikat eingebracht werden.

### Verwendung von CA-Zertifikaten bei TeleService

Um den TS Adapter IE Advanced als Verbindungspartner gegenüber dem Remote-PC eindeutig zu identifizieren, wird vom TS Adapter ein CA-Zertifikat mit einem eindeutigen Fingerabdruck generiert.

Für den Aufbau einer VPN-Verbindung ist es zwingend erforderlich, dieses CA-Zertifikat im Windows-Zertifikatsspeicher Ihres PG/PC zu hinterlegen. Wenn Sie den Web-Server im Direktanschluss aufrufen, so wird bei fehlendem CA-Zertifikat eine Sicherheitswarnung angezeigt, die akzeptiert werden kann.

---

#### Hinweis

#### Hantierung von CA-Zertifikaten

Die Hantierung von CA-Zertifikaten erfordert spezielle Kenntnisse des Betriebssystems und sollte nur von entsprechend geschultem Personal durchgeführt werden!

Zum Verwalten von CA-Zertifikaten benötigen Sie Administrator-Rechte.

---

### Definition Fingerabdruck

Der Fingerabdruck ist ein 20 Bytes langer hexadezimaler Ausdruck. Er stellt einen eindeutigen Wert für ein CA-Zertifikat dar und wird dazu verwendet, ein bestimmtes CA-Zertifikat zu identifizieren.

Der Fingerabdruck wird dynamisch mit dem SHA1-Algorithmus berechnet und ist physisch im CA-Zertifikat nicht vorhanden.

### Verwendung des Fingerabdrucks bei TeleService

Das CA-Zertifikat dient dazu, den TS Adapter IE Advanced eindeutig als Verbindungspartner zu identifizieren. Bei jeder CA-Zertifikatsgenerierung vom TS Adapter IE Advanced wird automatisch ein 20 Bytes langer, eindeutiger Fingerabdruck dieses CA-Zertifikats generiert. Er wird bei der Zertifikatsgenerierung vom TS Adapter IE Advanced automatisch berechnet. Jedes Zertifikat besitzt einen spezifischen, eindeutigen Fingerabdruck. Dieser Fingerabdruck muss an Ihren Rechner über einen sicheren Weg übergeben werden, z. B. per Telefon oder

durch eine verschlüsselte Mail. Diesen Fingerabdruck müssen Sie beim Aufbau einer VPN-Verbindung mit TeleService im Verbindungsdialog eintragen, sofern das CA-Zertifikat nicht bereits auf Ihrem PC im Windows-Zertifikatsspeicher hinterlegt ist.

Den zu Ihrem TS Adapter IE Advanced zugehörigen Fingerabdruck finden Sie im Web-Interface zum TS Adapter IE Advanced. Doppelklicken Sie zum Öffnen des Web-Interface auf den Befehl "TS Adapter IE Advanced parametrieren" in der Teilnehmerliste im TIA-Portal. Führen Sie das Web-Login durch, um den Fingerabdruck im Register "Sicherheit > Zertifikate" anzuzeigen.

### Download des CA-Zertifikats beim Verbindungsaufbau

Beim Verbindungsaufbau überprüft TeleService, ob ein passendes CA-Zertifikat im Windows-Zertifikatsspeicher Ihres PG/PC installiert ist. Wird ein passendes CA-Zertifikat gefunden, so wird die VPN-Verbindung als SSTP-Verbindung (Secure Socket Tunneling Protocol) aufgebaut.

Wird kein passendes CA-Zertifikat gefunden, so wird zunächst das CA-Zertifikat vom entsprechenden TS Adapter IE Advanced geladen. Dabei wird der TS Adapter IE Advanced über die im Verbindungsdialog eingetragene Remote-Adresse aufgerufen. Ist dieser Download des CA-Zertifikats erfolgreich, so wird der Fingerabdruck des heruntergeladenen CA-Zertifikats errechnet und mit dem im Verbindungsdialog eingetragenen Fingerabdruck verglichen. Bei Übereinstimmung beider Fingerabdrücke öffnet sich ein Dialog und Sie werden gefragt, ob das CA-Zertifikat im Windows-Zertifikatsspeicher Ihres PG/PC gespeichert werden soll. Für das Speichern des CA-Zertifikates benötigen Sie Administrator-Rechte.

Anschließend wird die VPN-Verbindung aufgebaut.

### Siehe hierzu auch

- CA-Zertifikat für VPN-Verbindung installieren (Seite 4373)
- CA-Zertifikat für VPN-Verbindung löschen (Seite 4376)

### 12.7.4.3 CA-Zertifikat für VPN-Verbindung installieren

#### Installieren von CA-Zertifikaten

Um mit TeleService eine sichere VPN-Verbindung zwischen Ihrem PG/PC und einer entfernten Anlage aufzubauen, benötigen Sie ein vom TS Adapter IE Advanced erstelltes gültiges CA-Zertifikat. Dieses muss im Windows-Zertifikatsspeicher Ihres PG/PC hinterlegt sein.

Die Installation eines CA-Zertifikates kann über einen automatischen Download oder über manuelle Installation erfolgen.

Die Verwaltung der CA-Zertifikate erfolgt über die Microsoft® Management Console.

---

### Hinweis

#### Hantierung von CA-Zertifikaten

Die Hantierung von CA-Zertifikaten erfordert spezielle Kenntnisse des Betriebssystems und sollte nur von entsprechend geschultem Personal durchgeführt werden!

Zum Verwalten von CA-Zertifikaten benötigen Sie Administrator-Rechte!

---

### Voraussetzung

Auf Ihrem Rechner ist noch kein CA-Zertifikat im Windows-Zertifikatsspeicher installiert.

### Installation von CA-Zertifikaten über automatischen Download

Gehen Sie folgendermaßen vor:

1. Melden Sie sich als Administrator am System an.
2. Übertragen Sie den zum CA-Zertifikat zugehörigen Fingerabdruck über einen "sicheren Weg", z. B. per Telefon oder über eine verschlüsselte Mail, vom TS Adapter IE Advanced auf Ihren Rechner.  
Den zu Ihrem TS Adapter IE Advanced zugehörigen Fingerabdruck finden Sie im Web-Interface zum TS Adapter IE Advanced. Doppelklicken Sie zum Öffnen des Web-Interface auf den Befehl "TS Adapter IE Advanced parametrieren" in der Teilnehmerliste im TIA-Portal. Führen Sie das Web-Login durch, um den Fingerabdruck im Register "Sicherheit > Zertifikate" anzuzeigen.
3. Klicken Sie in der Projektnavigation des TIA-Portals auf den Ordner "Online-Zugänge".
4. Klicken Sie auf den darin enthaltenen Ordner "TeleService".
5. Doppelklicken Sie auf den Eintrag "Fernverbindung aufbauen/trennen". Der Dialog "Fernverbindung zur entfernten Anlage aufbauen" wird geöffnet.
6. Wählen Sie in der Klappliste als Adapter-Typ den "TS Adapter IE" aus.
7. Wählen Sie als Verbindungs-Typ "VPN" aus.
8. Tragen Sie die IP-Adresse bzw. den DNS-Namen für den zu kontaktierenden TS Adapter IE Advanced in das entsprechende Eingabefeld ein. Alternativ können Sie evtl. bereits vorhandene Daten aus dem Telefonbuch übernehmen, indem Sie auf die dahinterliegende Schaltfläche klicken.
9. Tragen Sie Ihren Benutzernamen und das zugehörige Kennwort ein.
10. Kopieren Sie den im Web-Interface zum TS Adapter IE Advanced angezeigten Fingerabdruck in die Spalte "Fingerabdruck".
11. Klicken Sie auf die Schaltfläche "Aufbauen", um die gewünschte Fernverbindung aufzubauen. Diese Schaltfläche wird erst dann aktiv, wenn Sie alle für den Aufbau einer Fernverbindung benötigten Parameter eingetragen haben.

12. Da noch kein CA-Zertifikat auf Ihrem Rechner installiert war, wird kein gültiges CA-Zertifikat gefunden.  
Daher wird eine "normale" Verbindung (keine VPN-Verbindung) aufgebaut und das benötigte Zertifikat wird vom TS Adapter IE Advanced zu Ihrem Rechner in den Arbeitsspeicher heruntergeladen. Danach wird der Fingerabdruck berechnet (Algorithmus SHA1) und mit dem im Verbindungsdialog eingetragenen Fingerabdruck verglichen. Bei Übereinstimmung beider Fingerabdrücke öffnet sich ein Dialog und Sie werden gefragt, ob das CA-Zertifikat im Windows-Zertifikatsspeicher Ihres PG/PC gespeichert werden soll.
13. Bestätigen Sie das Speichern des CA-Zertifikats.

## Ergebnis

Die VPN-Verbindung zum gewünschten TS Adapter IE Advanced wird aufgebaut. Sobald die Verbindung aufgebaut wurde, wird der Dialog geschlossen.

## Manuelle Installation von CA-Zertifikaten

Gehen Sie folgendermaßen vor:

1. Melden Sie sich als Administrator am System an.
2. Öffnen Sie die Windows-Zertifikatsverwaltung auf Ihrem PG/PC mit Hilfe der Microsoft® Management Console.  
Klicken Sie dazu auf "Start", tragen Sie im Suchfeld "mmc" ein und drücken Sie die EINGABETASTE.  
Die Konsole öffnet sich.
3. Klicken Sie im Menü "Datei" auf den Befehl "Snap-In hinzufügen/entfernen...".  
Das Dialogfeld "Snap-In hinzufügen oder entfernen" öffnet sich.
4. Doppelklicken Sie in der Liste "Snap-In" auf "Zertifikate" und wählen Sie im anschließenden Dialog "Computerkonto" aus.
5. Wählen Sie in folgenden Dialog "Lokaler Computer" aus und klicken Sie auf "Fertig stellen" und auf "OK".  
Der Konsolenstamm öffnet sich und zeigt den Ordner "Zertifikate (Lokaler Computer)" an.
6. Öffnen Sie den angezeigten Ordner "Zertifikate (Lokaler Computer)" und klicken Sie auf "Vertrauenswürdige Stammzertifizierungsstellen".
7. Klicken Sie auf den Ordner "Zertifikate" und rufen Sie über das Kontextmenü den Befehl "Alle Aufgaben" > "Importieren..." auf.
8. Beachten Sie die im Dialog "Zertifikat - Import-Assistent" angezeigten Informationen und klicken Sie auf "Weiter".
9. Klicken Sie im nachfolgenden Dialog auf "Durchsuchen ..." und wählen Sie das gewünschte CA-Zertifikat aus.
10. Klicken Sie anschließend zwei Mal auf "Weiter" und anschließend auf "Fertig stellen", um das CA-Zertifikat zu installieren.

## Ergebnis

Das ausgewählte CA-Zertifikat wird an der angegebenen Stelle im Windows-Zertifikatsspeicher installiert.

---

### Hinweis

#### Weiterführende Informationen ...

... zur Installation von CA-Zertifikaten finden Sie über die Taste "F1" in der Online-Hilfe zu Ihrem Betriebssystem.

---

## 12.7.4.4 CA-Zertifikat für VPN-Verbindung löschen

### Löschen von CA-Zertifikaten

Gehen Sie folgendermaßen vor:

1. Melden Sie sich als Administrator am System an.
2. Öffnen Sie die Windows-Zertifikatsverwaltung auf Ihrem PG/PC mit Hilfe der Microsoft® Management Console.  
Klicken Sie dazu auf "Start", tragen Sie im Suchfeld "mmc" ein und drücken Sie die EINGABETASTE.  
Die Konsole öffnet sich.
3. Klicken Sie im Menü "Datei" auf "Snap-In hinzufügen/entfernen..."  
Das Dialogfeld zur Snap-In-Auswahl öffnet sich.
4. Doppelklicken Sie in der Liste "Snap-In" auf "Zertifikate" und wählen Sie im anschließenden Dialog "Computerkonto" aus.
5. Wählen Sie in folgenden Dialog "Lokaler Computer" aus und klicken Sie auf "Fertig stellen" und auf "OK".  
Der Konsolenstamm öffnet sich und zeigt den Ordner "Zertifikate (Lokaler Computer)" an.
6. Öffnen Sie den angezeigten Ordner "Zertifikate (Lokaler Computer)" und klicken Sie auf "Vertrauenswürdige Stammzertifizierungsstellen".
7. Öffnen Sie den Ordner "Zertifikate", wählen Sie das gewünschte CA-Zertifikat aus klicken Sie im Kontextmenü auf "Löschen".
8. Bestätigen Sie die nachfolgende Abfrage mit "Ja".

## Ergebnis

Das ausgewählte CA-Zertifikat wird aus der Liste der verfügbaren Zertifikate gelöscht.

### 12.7.4.5 Aufbauen einer VPN-Verbindung zu einer entfernten Anlage

#### VPN-Verbindung aufbauen

##### Einführung zum Aufbauen einer VPN-Verbindung

Eine VPN-Verbindung kommt zustande, wenn Sie sich mit TeleService über das Internet in eine entfernte Anlage verbinden.

Hierzu ist auf der einen Seite Ihr PG/PC mit installiertem TIA-Portal mit dem Internet verbunden. An der anderen Seite ist das Automatisierungssystem über die WAN-Schnittstelle (Wide Area Network) des parametrisierten TS Adapters IE Advanced mit dem Internet verbunden.

##### Voraussetzung

Ihr PG/PC ist mit dem Internet verbunden.

In der entfernten Anlage befindet sich ein TS Adapter IE Advanced.

Der TS Adapter IE Advanced ist parametrisiert und mit dem Internet verbunden.

Das zur Identifizierung des TS Adapters benötigte CA-Zertifikat wurde erzeugt und ist im Windows-Zertifikatsspeicher Ihres PG/PC installiert.

##### Gehen Sie folgendermaßen vor:

1. Klicken Sie in der Projektnavigation des TIA-Portals auf den Ordner "Online-Zugänge".
2. Klicken Sie auf den darin enthaltenen Ordner "TeleService".
3. Doppelklicken Sie auf den Eintrag "Fernverbindung aufbauen/trennen". Der Dialog "Fernverbindung zur entfernten Anlage aufbauen" wird geöffnet.
4. Wählen Sie in der Klappliste als "Adapter-Typ" den "TS Adapter IE" aus.
5. Tragen Sie als "Verbindungs-Typ" "VPN" ein.
6. Tragen Sie die IP-Adresse bzw. den DNS-Namen für den zu kontaktierenden TS Adapter IE Advanced in das entsprechende Eingabefeld ein. Alternativ können Sie evtl. bereits vorhandene Daten aus dem Telefonbuch übernehmen, indem Sie auf die dahinterliegende Schaltfläche klicken.
7. Tragen Sie Ihren Benutzernamen und das zugehörige Kennwort ein.
8. Klicken Sie auf die Schaltfläche "Aufbauen", um die gewünschte VPN-Verbindung aufzubauen. Diese Schaltfläche wird erst dann aktiv, wenn Sie alle für den Aufbau der Fernverbindung benötigten Parameter eingetragen haben.

##### Ergebnis

Die VPN-Verbindung zur gewünschten Anlage wird aufgebaut. Im "Status" wird der Fortschritt des Verbindungsaufbaus angezeigt. Sobald die VPN-Verbindung zustande gekommen ist, wird der Dialog geschlossen. In der Statuszeile des TIA-Portals erscheint die Meldung:

"Fernverbindung ist aufgebaut". Sie können nun die Fernverbindung mit dem TIA-Portal nutzen und mit dem Automatisierungssystem kommunizieren.

### Verbindungsaufbau nicht möglich

Ist kein Verbindungsaufbau möglich, so versuchen Sie bitte, die Fehlerursache mit Hilfe der "Hinweise zur Fehlersuche" zu ermitteln.

---

#### Hinweis

##### Regeln für IP-Adressen

- Verwenden Sie ausschließlich IP-Adressen, die im Anlagennetz noch nicht vergeben sind.
  - Falls die für den TS Adapter IE Advanced parametrisierte IP-Adresse im Anlagennetz bereits vergeben ist, so ist der TS Adapter IE Advanced nur über seine MAC-Adresse ansprechbar.
- 

### Siehe auch

CA-Zertifikat für VPN-Verbindung installieren (Seite 4373)

### VPN-Verbindung trennen

#### Trennen einer aktiven VPN-Verbindung

---

#### Hinweis

##### Trennen der VPN-Verbindung

Bevor Sie die VPN-Verbindung trennen, sollten Sie im TIA-Portal offline schalten.

---

### Gehen Sie vor wie folgt:

1. Doppelklicken Sie im TIA-Portal auf den Eintrag "Fernverbindung aufbauen/trennen".
2. Bestätigen Sie die Abfrage im nachfolgenden Dialog mit "Ja".

### Ergebnis

Die VPN-Verbindung wird getrennt.



## 12.7.4.6 TS Adapter IE Advanced

### Kurzbeschreibung TS Adapter IE Advanced

#### TS Adapter IE Advanced

Der TS Adapter IE Advanced hat folgende Eigenschaften:

- Direktanschluss über Industrial Ethernet (IE), 2 Ports
- WAN-Schnittstelle (Wide Area Network) für VPN-Verbindungen
- Firmwareupdate möglich
- Steckbare Module
- Die Parametrierung erfolgt über ein Web-Interface.

---

#### Hinweis

#### Weitere Informationen zum TS Adapter IE Advanced

Ausführliche Informationen zu dem von Ihnen verwendeten TS Adapter entnehmen Sie bitte der zu Ihrem TS Adapter mitgelieferten Dokumentation.

---

### Anschlussarten

#### Anschlussarten des TS Adapter IE Advanced

Die folgenden Bilder zeigen die möglichen Anschlussarten für den TS Adapter IE Advanced.

#### Direktanschluss

Im Direktanschluss am PG/PC können Sie den TS Adapter IE Advanced über Ethernet parametrieren.

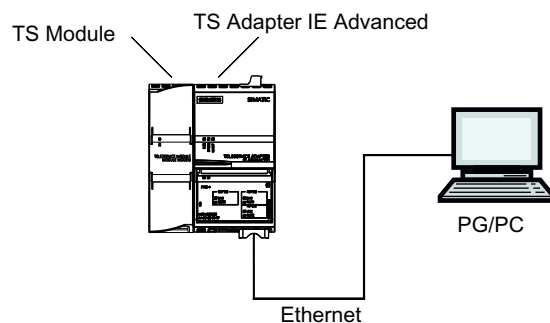


Bild 12-5 TS Adapter IE Advanced - Direktanschluss

### Anschluss an das Internet (DSL-Modem/Router)

Zum Anschluss an das Internet müssen Sie das DSL-Modem/Router am WAN-Anschluss des TS Adapter IE Advanced betreiben.

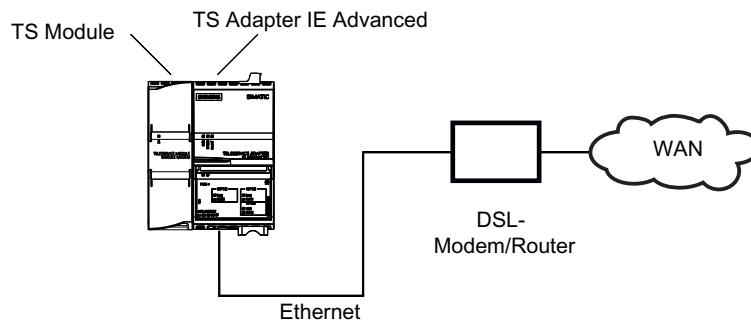


Bild 12-6 TS Adapter IE Advanced - Anschluss Internet

### Anschluss an das Firmennetz (Intranet)

Zum Anschluss an das Intranet müssen Sie das Anlagennetz (Ethernet) an einem der beiden LAN-Anschlüsse des TS Adapter IE Advanced betreiben.

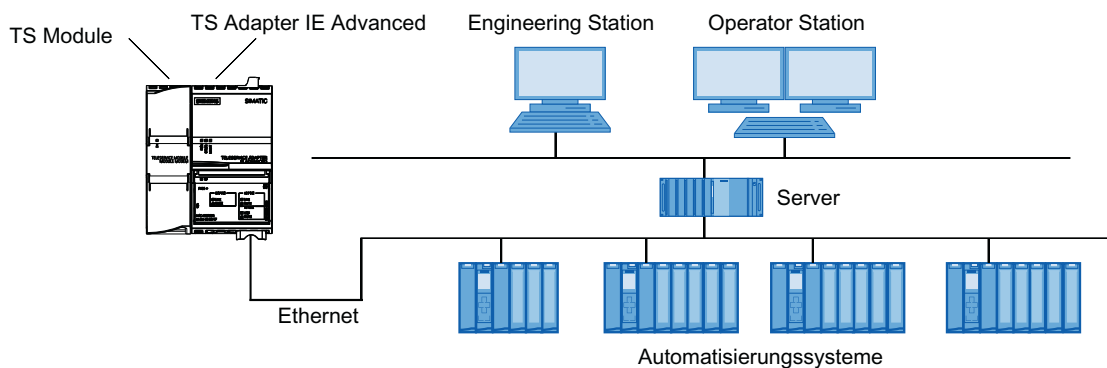


Bild 12-7 TS Adapter IE Advanced - Anschluss Firmennetz (Intranet)

### Weitere Informationen

Weitere Informationen zu den TS Modulen finden Sie in der zu Ihrem TS Adapter IE Advanced mitgelieferten Dokumentation.

### Parametriermöglichkeiten für den TS Adapter IE Advanced

#### Grundlagen zum Parametrieren des TS Adapter IE

Der TS Adapter IE Advanced wird über ein Web-Interface parametrier.

Für die Parametrierung des TS Adapter IE Advanced steht Ihnen eine dem Web-Interface zugeordnete Webhilfe zur Verfügung.

Folgende Parametriermöglichkeiten stehen z. B. zur Verfügung:

- Neuparametrierung
- Wiederherstellen der Standard-Parametrierung
- Importieren von Adapterparametern
- Exportieren von Adapterparametern

---

#### **Hinweis**

##### **Parametrierung durchführen**

Parametrieren Sie Ihren TS Adapter IE Advanced unter Berücksichtigung der mitgelieferten Dokumentation. Dort finden Sie die benötigten Informationen zur exakten Vorgehensweise beim Parametrieren.

---

## **TS Adapter IE Advanced parametrieren**

### **Einführung**

Sie können den TS Adapter IE Advanced sowohl im Direktanschluss als auch über eine bereits aufgebaute Fernverbindung parametrieren.

Nachfolgend werden beide Parametriermöglichkeiten beschrieben.

Spezifische Details zur Parametrierung des TS Adapters IE Advanced entnehmen Sie bitte der Dokumentation zum TS Adapter IE Advanced.

## **Parametrieren des TS Adapter IE Advanced im Direktanschluss**

### **Voraussetzung**

Es besteht eine LAN-Verbindung zu Ihrem TS Adapter IE Advanced.

### **Vorgehen**

Um den TS Adapter IE Advanced zu parametrieren gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation des TIA-Portals den Ordner "Online Zugänge".
2. Doppelklicken Sie hier auf die Ethernet-Schnittstelle Ihres Rechners.
3. Doppelklicken Sie auf den Befehl "Erreichbare Teilnehmer anzeigen". Daraufhin wird der TS Adapter IE Advanced angezeigt.

4. Doppelklicken Sie auf den Ordner <TS Adapter IE Advanced> und anschließend auf "Online und Diagnose" und weisen Sie dem TS Adapter in den nachfolgenden Dialogen die gewünschte IP-Adresse zu. Beachten Sie dabei, dass die IP-Adresse der Schnittstellenkarte des PG/PC im selben Subnetz liegt, wie die IP-Adresse, die Sie für den TS Adapter IE Advanced vergeben.
5. Aktualisieren Sie die Ansicht in der Projektnavigation für die "Erreichbaren Teilnehmer", damit der TS Adapter IE Advanced mit der neu zugewiesenen IP-Adresse angezeigt wird.
6. Öffnen Sie den Ordner <TS Adapter IE Advanced> in der Teilnehmerliste.
7. Doppelklicken Sie auf den Befehl "TS Adapter IE Advanced parametrieren". Daraufhin öffnet sich das zugeordnete Web-Interface zum Parametrieren des TS Adapters.
8. Führen Sie das "Logon" für das Web-Interface durch.
9. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
10. Bestätigen Sie Ihre Eingaben jeweils mit "Einstellungen speichern".

## Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters IE Advanced gespeichert. Die Parametrierung ist damit abgeschlossen.

## Parametrieren des TS Adapter IE Advanced über Fernverbindung

### Voraussetzung

Es besteht eine aufgebaute Fernverbindung zu einem TS Adapter IE Advanced.

### Vorgehen

1. Öffnen Sie in der Projektnavigation des TIA-Portals den Ordner "Online Zugänge".
2. Öffnen Sie den Ordner "TeleService" und danach den gewünschten Anlagenordner.
3. Doppelklicken Sie auf den Befehl "TS Adapter IE Advanced parametrieren". Daraufhin öffnet sich das zugehörige Web-Interface zum Parametrieren des TS Adapters IE. Das "Logon" für das Web-Interface wird automatisch mit den Login-Daten der Fernverbindung durchgeführt.
4. Stellen Sie in den einzelnen Registern des Dialoges die gewünschten Parameter ein.
5. Bestätigen Sie Ihre Eingaben jeweils mit "Einstellungen speichern".

## Ergebnis

Die eingestellten Parameter werden im nichtflüchtigen Speicher des TS Adapters IE Advanced gespeichert. Die Parametrierung ist damit abgeschlossen.

## 12.7.5 CPU-gesteuerte Fernverbindungen bei TeleService

### 12.7.5.1 Übersicht über CPU-gesteuerte Fernverbindungen

#### Einführung

TeleService bietet Ihnen, abhängig von der jeweils verwendeten CPU, unterschiedliche Möglichkeiten für den Aufbau einer Fernverbindung. Die Initiative für den Verbindungsaufbau geht dabei jeweils von der CPU aus. Für die einzelnen Verbindungsmöglichkeiten werden die nachfolgend genannten Kommunikationsanweisungen genutzt.

#### Möglichkeiten zum Verbindungsaufbau mit S7-300/400er CPUs

Folgende Kommunikationsanweisungen stehen zur Verfügung:

- Kommunikationsanweisung "PG\_DIAL": Fernverbindung zu PG/PC aufbauen
- Kommunikationsanweisung "SMS\_SEND": SMS-Nachricht senden
- Kommunikationsanweisung "AS\_DIAL": Fernverbindung zum AS aufbauen
- Kommunikationsanweisung "AS\_MAIL": E-Mail übertragen

#### Möglichkeiten zum Verbindungsaufbau mit S7-1200er CPUs

Folgende Kommunikationsanweisung steht zur Verfügung:

- Kommunikationsanweisung "TM\_MAIL": E-Mail übertragen

#### Möglichkeiten zum Verbindungsaufbau mit S7-1500er CPUs

Folgende Kommunikationsanweisung steht zur Verfügung:

- Kommunikationsanweisung "TMAIL\_C": E-Mail übertragen

---

#### Hinweis

##### Beschreibung der einzelnen Kommunikationsanweisungen

Weiterführende Informationen zu den verwendbaren Kommunikationsanweisungen finden Sie im Informationssystem des TIA-Portals im Verzeichnis "Referenzen > Kommunikation > TeleService".

---

#### Siehe auch

Parametriermöglichkeiten für den TS Adapter IE (Seite 4366)

### 12.7.5.2 Verbindungsaufbau von und zu entfernten Anlagen (PG-AS-Fernkopplung)

#### Zugriff von entfernten Anlagen auf ein PG/PC

##### Einführung

Sie können mit der Applikation TeleService und einem TS Adapter MPI eine Fernverbindung zu einer entfernten Anlage aufbauen und so mit dieser Anlage kommunizieren. Die Initiative zum Aufbau der Fernverbindung geht hierzu vom PG/PC aus.

Häufig treten jedoch Ereignisse auf einer entfernten Anlage auf, die einen schnellen Eingriff erfordern. In diesem Fall ist es dem Automatisierungssystem möglich, bei Auftreten eines asynchronen Ereignisses den Aufbau einer Fernverbindung zu einem PG/PC zu initiieren.

Das folgende Bild zeigt eine grafische Darstellung der Komponenten, die für den Verbindungsaufbau von einer Anlage zu einem PG/PC erforderlich sind.

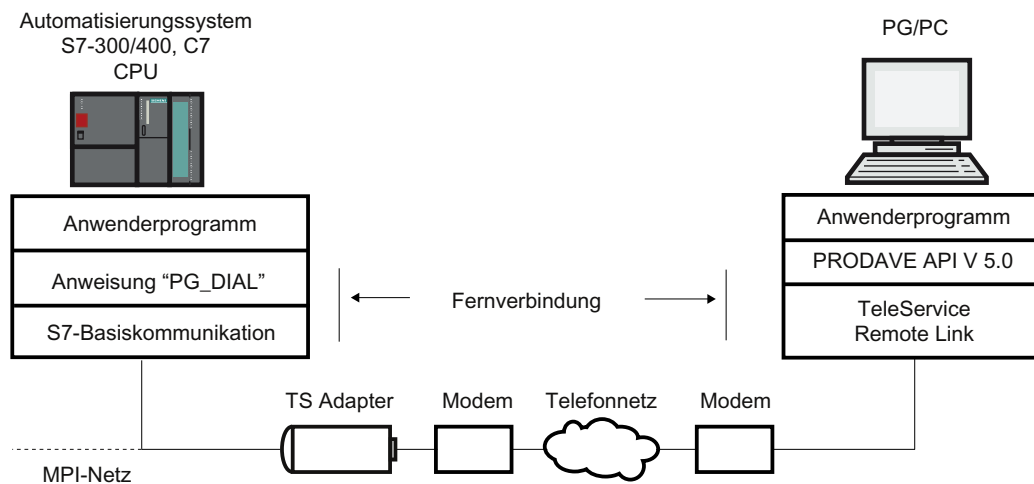


Bild 12-8 Funktion der Kommunikationsanweisung "PG\_DIAL"

#### Voraussetzungen für den Verbindungsaufbau

##### Einführung

Soll eine entfernte Anlage eine Fernverbindung zu einem PG/PC aufbauen können, müssen bestimmte Hardware- und Software-Voraussetzungen erfüllt sein. Diese Anforderungen sind nachfolgend beschrieben.

##### Hardware-Voraussetzungen:

Zum Aufbauen einer Fernverbindung von einer entfernten Anlage zu einem PG/PC wird keine weitere Hardware benötigt als die, die auch für den Zugriff des PG/PC auf die entfernte Anlage erforderlich ist.

Zum Aufbauen der Verbindung ruft Ihr Anwenderprogramm die Kommunikationsanweisung "PG\_DIAL" auf. Diese ist auf einer S7-300 oder S7-400 CPU ablauffähig, auf der die S7-Basiskommunikation realisiert ist.

Es muss ein TS Adapter I mit einer Version größer oder gleich V5.0 oder ein TS Adapter II verwendet werden.

#### **Software-Voraussetzungen auf Anlagen-Seite:**

Im Lieferumfang von TeleService ist die Kommunikationsanweisung "PG\_DIAL" enthalten, die beim Installieren des TIA-Portals mit installiert wird. Sie finden die mitinstallierten Kommunikationsanweisungen in der Task Card des Bausteineditors, im Ordner "Kommunikation > TeleService".

Soll eine entfernte Anlage eine Fernverbindung zu einem PG/PC aufbauen, muss das Anwenderprogramm der Anlage die Kommunikationsanweisung "PG\_DIAL" aufrufen.

#### **Software-Voraussetzungen auf PG/PC-Seite:**

Auf dem PG/PC benötigen Sie ein Software-Produkt, das in Verbindung mit TeleService auf den Anruf einer entfernten Anlage wartet, diesen erkennt und Ihr Anwenderprogramm informiert.

### **12.7.5.3 Datenaustausch zwischen entfernten Anlagen (AS-AS-Fernkopplung)**

#### **Grundlagen zur AS-AS-Fernkopplung**

##### **Einführung**

Mit Hilfe der AS-AS-Fernkopplung können zwei Automatisierungssysteme über das Telefonnetz Prozessdaten miteinander austauschen.

##### **Voraussetzung**

Die Kommunikationsanweisung "AS\_DIAL" steht Ihnen nur dann zur Verfügung, wenn Sie eine CPU aus der Familie S7-300/400 verwenden.

##### **Definition: Lokales und entferntes Automatisierungssystem**

- Das Automatisierungssystem, von dem die Initiative zum Aufbau der Fernverbindung ausgeht, wird als **lokal** bezeichnet.
- Das Automatisierungssystem, zu dem die Fernverbindung aufgebaut werden soll, wird als **entfernt** bezeichnet.

## Datenaustausch über AS-AS-Fernkopplung

Der Datenaustausch erfolgt über spezifische Kommunikationsanweisungen für nichtprojektierte S7-Verbindungen. Für den Aufbau einer Fernverbindung zum Automatisierungssystem nutzen Sie die Kommunikationsanweisung "AS\_DIAL".

Weiterführende Informationen zum Verbindungsaufbau finden Sie im Informationssystem im Verzeichnis "Referenzen > Kommunikation > TeleService".

Das folgende Bild zeigt eine grafische Darstellung der Komponenten, die für den Verbindungsaufbau von einem lokalen zu einem entfernten Automatisierungssystem erforderlich sind.

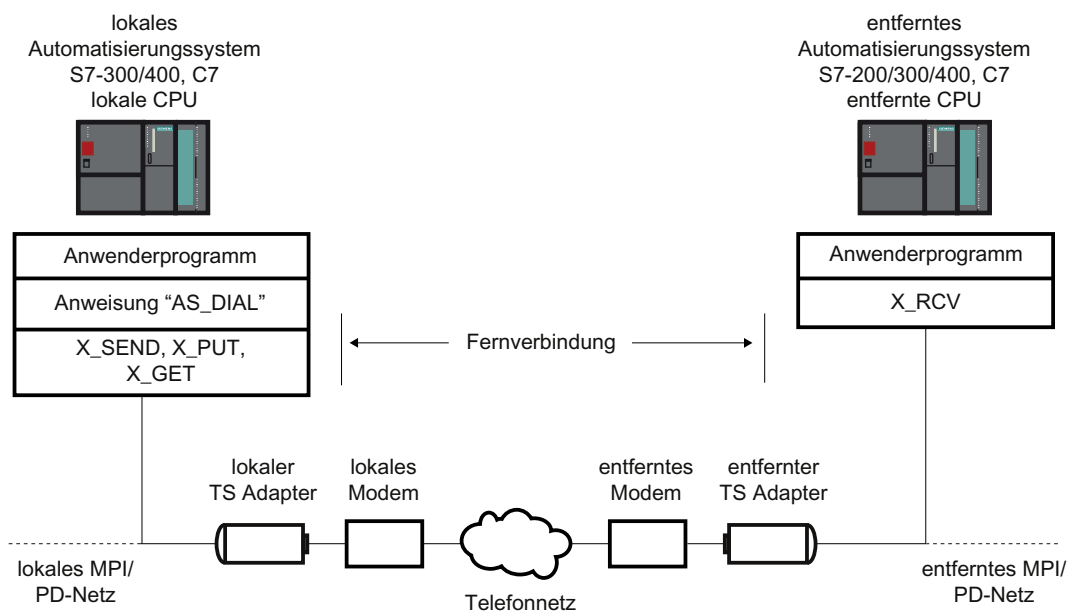


Bild 12-9 Datenaustausch über AS-AS-Fernkopplung

## Hardware- und Softwarevoraussetzungen für AS-AS-Fernkopplung

### Einführung

Soll ein lokales Automatisierungssystem eine Fernverbindung zu einem entfernten Automatisierungssystem aufbauen können, müssen bestimmte Hardware- und Softwarevoraussetzungen erfüllt sein. Diese Anforderungen sind nachfolgend beschrieben.

### Hardware-Voraussetzungen

Zur Übertragung von Prozessdaten von einem lokalen zu einem entfernten Automatisierungssystem wird keine weitere Hardware benötigt als die, die auch für den Zugriff des PG/PC auf das jeweilige Automatisierungssystem erforderlich ist.

Zum Auf- und Abbauen der Fernverbindung ruft Ihr TIA-Portal-Anwenderprogramm der lokalen CPU eine Kommunikationsanweisung auf. Diese Kommunikationsanweisung ist auf einer S7-300/400-CPU oder auch C7-CPU ablauffähig. Diese Kommunikationsanweisung setzt



voraus, dass auf der CPU die S7-Basiskommunikation realisiert ist. Ebenso muss die entfernte CPU die S7-Basiskommunikation unterstützen.

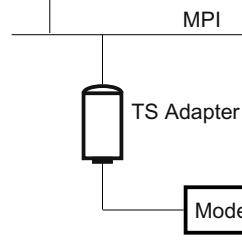
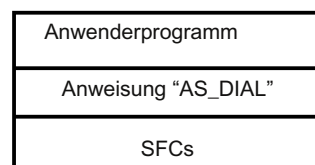
Es muss ein TS Adapter I mit einer Version größer oder gleich V5.1 oder ein TS Adapter II verwendet werden.

## Software-Voraussetzungen

Im Lieferumfang von TeleService ist die Kommunikationsanweisung "AS\_DIAL" enthalten, die beim Installieren in der Task Card im Ordner Kommunikationsanweisungen unter TeleService in die Bibliothek des TIA-Portals aufgenommen wird. Zum Auf- und Abbauen einer Fernverbindung von einem lokalen Automatisierungssystem zu einem entfernten Automatisierungssystem ist in Ihrem TIA-Portal-Anwenderprogramm der lokalen CPU die Kommunikationsanweisung "AS\_DIAL" aufzurufen

AS- AS- Fernkopplung

Automatisierungssystem  
S7 300/400, C7



Automatisierungssystem  
S7 300/400, C7

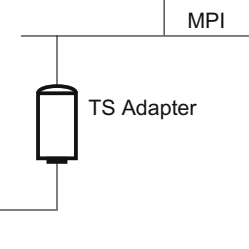
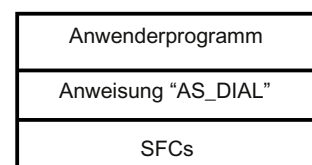


Bild 12-10 Hardware- und Softwarevoraussetzungen für AS-AS-Fernkopplung

### 12.7.5.4 Senden einer SMS von einer Anlage

#### Voraussetzungen für das Senden einer SMS

#### Einführung

Soll eine Anlage eine SMS senden, so müssen bestimmte Hardware- und Software-Voraussetzungen erfüllt sein. Diese Anforderungen sind nachfolgend beschrieben.

### Hardware-Voraussetzungen

Zum Senden einer SMS von einer Anlage benötigen Sie ein GSM-Funkmodem sowie einen TS Adapter MPI.

Es muss ein TS Adapter I mit einer Version größer oder gleich V5.2 oder ein TS Adapter II verwendet werden.

### Software-Voraussetzungen auf Anlagen-Seite

Im Lieferumfang von TeleService ist die Kommunikationsanweisung "SMS\_SEND" enthalten, die beim Installieren in der Task Card im Ordner Kommunikationsanweisungen unter TeleService in die Bibliothek des TIA-Portals aufgenommen wird. Soll eine Anlage eine SMS senden, muss das Anwenderprogramm der Anlage die Kommunikationsanweisung "SMS\_SEND" aufrufen.

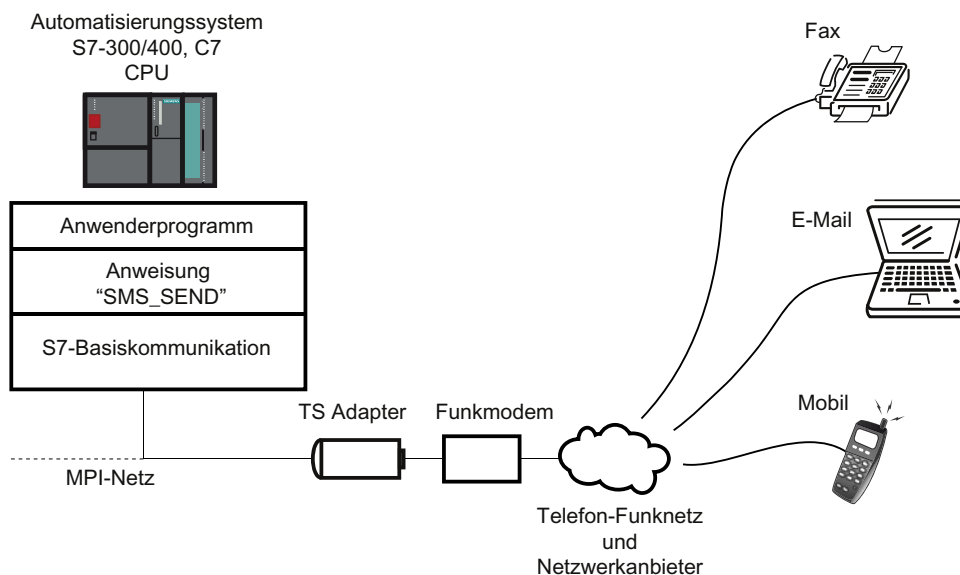


Bild 12-11 Funktionsweise der Kommunikationsanweisung "SMS\_SEND"

### Hinweis

Über Zusatzdienste des Mobilfunknetzanbieters kann eine SMS ggf. nicht nur an ein Mobiltelefon, sondern auch an eine E-Mail-Adresse oder an ein Fax-Gerät gesendet werden.

### 12.7.5.5 Senden einer E-Mail von einer Anlage

#### Voraussetzungen für das Senden einer E-Mail

##### Einführung

Soll eine Anlage eine E-Mail senden, so müssen folgende Hard- und Software-Voraussetzungen erfüllt sein:

##### Hardware-Voraussetzungen

Zum Senden einer E-Mail von einer Anlage benötigen Sie einen TS Adapter IE und eine der nachfolgenden genannten CPUs:

- eine CPU 31x-2 PN/DP ab Firmwarestand V2.5
- eine CPU 41x-3 PN/DP
- eine CPU aus der Familie S7-1200 mit Ethernet-Anschluss
- eine CPU aus der Familie S7-1500

##### Software-Voraussetzungen auf Anlagen-Seite

Im Lieferumfang von TeleService sind CPU-abhängig verschiedene Kommunikationsanweisungen enthalten, die beim Installieren in der Task Card im Ordner Kommunikationsanweisungen unter TeleService in die Bibliothek des TIA-Portals aufgenommen werden.

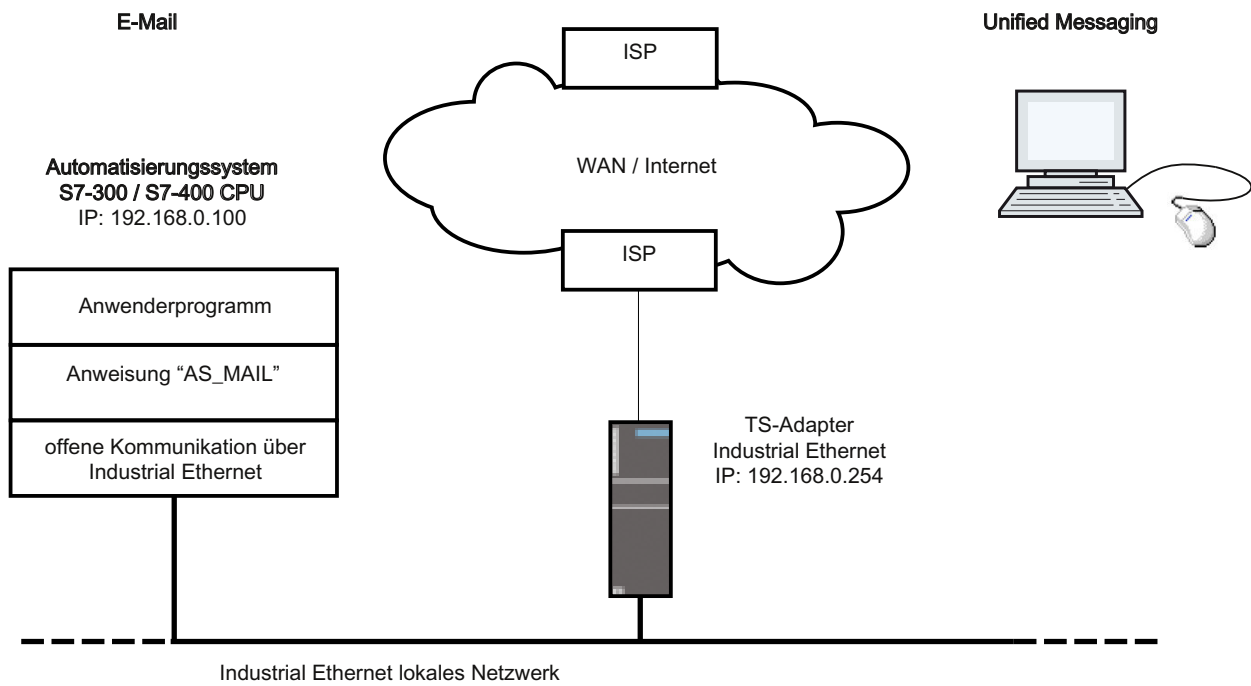
Soll eine Anlage eine E-Mail senden, so muss das Anwenderprogramm der Anlage die entsprechende CPU-abhängige Kommunikationsanweisung aufrufen.

Nachfolgende Kommunikationsanweisungen stehen zum Senden einer E-Mail zur Verfügung:

- S7-300/400er CPU: verwendet die Kommunikationsanweisung "AS\_MAIL": E-Mail übertragen
- S7-1200er CPU: verwendet die Kommunikationsanweisung "TM\_MAIL": E-Mail übertragen
- S7-1500er CPU: verwendet die Kommunikationsanweisung "TMAIL\_C": E-Mail übertragen

Die jeweilige Kommunikationsanweisung überträgt mittels Simple Mail Transfer Protocol (SMTP) mit dem Authentifizierungsverfahren "LOGIN" eine E-Mail von einer CPU zu einem Mail-Server. Bei diesem SMTP-Verfahren werden die Daten unverschlüsselt übertragen.

Das nachfolgende Bild zeigt ein Beispiel mit der Kommunikationsanweisung "AS\_MAIL":



Darüber hinaus muss bei der Konfiguration der CPU, auf der die Kommunikationsanweisung "AS\_MAIL" abläuft, für die Ethernet-Schnittstelle die Eigenschaft "Netzübergang / Router verwenden" eingestellt werden. (Zu finden in der Gerätekonfiguration unter Ethernet-Adressen und dort unter IP-Protokoll,) Als "Adresse" ist dabei die IP-Adresse der Ethernet-Schnittstelle des TS Adapters IE anzugeben.

---

### Hinweis

Weiterführende Hinweise hierzu finden Sie in der Task Card im Ordner "Kommunikationsanweisungen" unter TeleService.

---

## 12.7.6 Hinweise zur Fehlersuche

### 12.7.6.1 Allgemeine Hinweise für die Fehlersuche bei Modem-Problemen

#### Einführung

Die nachfolgenden Hinweise erleichtern es Ihnen, die Ursachen für eventuell auftretende Modem-Probleme zu finden und zu beheben:

1. Schalten Sie das "Aufzeichnen einer Protokolldatei" zum Datenverkehr zwischen PG/PC und Modem ein. Die Einträge in dieser Datei können wertvolle Hinweise bei der Ermittlung von Fehlerursachen liefern.
2. Schalten Sie an Ihrem lokalen Modem den Lautsprecher ein. Wählen Sie eine gut hörbare Lautstärke.  
Sie können dann hören, ob:
  - ein Freizeichen auf dem Anschluss vorhanden ist.
  - das angerufene Modem besetzt ist.
  - das angerufene Modem den Anruf entgegennimmt.

#### Häufige Modem-Probleme

Zu den häufigsten Modem-Problemen gehören Probleme bei der Modemverbindung:

- Modemverbindung wird nicht aufgebaut.
- Modemverbindung wird unterbrochen.

In den folgenden Topics finden Sie Tabellen, in denen mögliche Ursachen beschrieben und Hinweise zur Behebung des jeweiligen Fehlerfalls angegeben sind.

#### Siehe auch

Wählverbindung zum TS Adapter wird nicht aufgebaut (Seite 4392)

Wählverbindung vom TS Adapter wird nicht aufgebaut (Seite 4394)

Modemverbindung wird unterbrochen (Seite 4395)

Meldungen des Modems (Seite 4396)

Aufzeichnen einer Protokolldatei für das Modem (Seite 4391)

### 12.7.6.2 Aufzeichnen einer Protokolldatei für das Modem

#### Einführung

Um mögliche Fehlerursachen bei einem Modem leichter lokalisieren zu können, empfiehlt es sich, eine Protokolldatei aufzeichnen zu lassen.

**Vorgehensweise:**

Gehen Sie vor wie folgt:

1. Aktivieren Sie über die Systemsteuerung, Option "Telefon- und Modemoptionen", den Eigenschaftsdialog des verwendeten Modems.
2. Überprüfen Sie die Einstellungen zur Option "Protokoll" im Register "Diagnose" und ändern Sie, falls notwendig, die Einstellungen zur Protokolldatei, so dass diese aufgezeichnet wird.

**Ergebnis:**

In die Protokolldatei werden die Aktivitäten zwischen PG/PC und Modem eingetragen. Bei Problemen mit dem Verbindungsaufbau können Sie die Aufzeichnungen in der Protokolldatei auswerten, um Hinweise auf die Fehlerursache zu erhalten.

**12.7.6.3 Wählverbindung zum TS Adapter wird nicht aufgebaut**

**Wählverbindung zum TS Adapter wird nicht aufgebaut**

Die folgende Tabelle zeigt mögliche Ursachen und deren Behebung auf, falls keine Fernverbindung zum TS Adapter aufgebaut werden kann.

| Möglich Ursache  | Überprüfung/Behebung  |
|--|---|
| Verkabelung fehlerhaft   | <ul style="list-style-type: none"><li>• Sind alle Verbindungskabel korrekt angeschlossen?</li><li>• Sind Verbindungsstecker gelockert?</li></ul>  |
| Wahlparameter für Haupt- und Nebenanschluss falsch eingestellt | <ul style="list-style-type: none"><li>• Passen die eingestellten Eigenschaften und Wahlparameter Ihres Modems zum Telefonanschluss an Hauptanschluss oder Nebenstelle?</li><li>• Wenn Sie Ihr Modem an einer Amtsleitung (Hauptanschluss) betreiben, dürfen Sie im Dialogfeld "Wahlparameter" keine Amtskennzahl angeben. Die Felder der Amtskennzahl für Ortsgespräche und für Ferngespräche müssen leer sein.</li></ul>   |
| Wahlverfahren falsch eingestellt                               | <ul style="list-style-type: none"><li>• Ist im Dialogfeld zu den Wahlparametern Ihres Modems das korrekte Wahlverfahren (Ton/Impuls) eingestellt?</li><li>• Überprüfen Sie mit einem angeschlossenen Telefon den Anschluss, an dem Sie das Modem betreiben möchten. Bei Impulswahl sollten im Telefon Knackgeräusche während des Wählens hörbar sein, bei Tonwahl dagegen unterschiedlich hohe Töne. Stellen Sie das entsprechende Wahlverfahren bei den Wahlparametern des Modems ein.</li></ul> |

| Möglich Ursache   | Überprüfung/Behebung   |
|---|--|
| Wahlsperre aktiv  | <ul style="list-style-type: none"> <li>Die Wahlsperre ist eine länderspezifisch vorgeschriebene Modem-Eigenschaft, die je nach Modem nach einem oder mehreren fehlgeschlagenen Versuchen eines Verbindungsaufbaus wirksam wird.</li> <li>Reagiert Ihr Modem nach einigen vergeblichen Wahlversuchen nicht mehr, kann es sein, dass die Wahlsperre aktiv ist. Nach dem Wählenbefehl werden zwar noch Zeichen zum Modem geschickt, aber das Modem startet den Wahlvorgang nicht. Der Treiber bekommt eine allgemeine Fehlermeldung.</li> <li>Informieren Sie sich in der Modem-Dokumentation, wie bei Ihrem Modem die Wahlsperre realisiert ist.</li> <li>Lassen Sie die Protokolldatei (Seite 4391) (modemlog.txt) anlegen, in der die Aktivitäten zwischen PG/PC und Modem aufgezeichnet werden. Prüfen Sie dann, ob die Datei einen durch eine Wahlsperre verursachten Eintrag (z.B. DELAYED) enthält.</li> </ul> |
| Telefonanschluss defekt oder belegt   | <ul style="list-style-type: none"> <li>Schließen Sie ein Telefon an und prüfen Sie, ob auf diesem Anschluss ein Freizeichen zu hören ist.</li> <li>Ist ein analoges Telefon am gleichen Anschluss angeschlossen, so muss dieses aufgelegt sein. Wenn bereits eine Telefonverbindung besteht, kann auf diesem Anschluss keine zusätzliche Modemverbindung hergestellt werden.</li> </ul>  |
| Serielle Parameter falsch eingestellt   | <ul style="list-style-type: none"> <li>Sind im Register "Einstellungen" der Modemeigenschaften die korrekten Werte eingetragen (8 Datenbits, keine Parität, 1 Stoppbit)?<br/>Ist im Register "Allgemein" der Modemeigenschaften die richtige COM-Schnittstelle eingestellt?</li> </ul>   |
| Initialisierungsstring des TS Adapter passt nicht zum Modem.  | <ul style="list-style-type: none"> <li>Informieren Sie sich über die Anforderungen an den Modem-Initialisierungsstring und stellen Sie diesen gemäß den Anforderungen ein.<br/>Vorgehensweise zum Parametrieren des TS Adapter IE (Seite 4363)</li> </ul>  |
| Einstellung der Fehlerkorrektur zwischen Modem am TS Adapter und Modem am PC/PG sind nicht verträglich. | <ul style="list-style-type: none"> <li>Passen Sie die Modem-Einstellungen an.<br/>Wissenswertes zum Parametrieren des TS Adapter MPI (Seite 4358)<br/>Wiederherstellen der Standard-Parametrierung eines TS Adapter MPI (Seite 4360)<br/>Vorgehensweise zum Parametrieren des TS Adapters IE (Seite 4359)</li> </ul>   |

### 12.7.6.4 Wählverbindung vom TS Adapter wird nicht aufgebaut

#### Rückruf vom TS Adapter erfolgt nicht

Die folgende Tabelle zeigt mögliche Ursachen und deren Behebung auf, falls kein Rückruf vom TS Adapter erfolgt.

| Mögliche Ursache  | Überprüfung / Behebung  |
|---|---|
| Standort- oder Rufeinstellungen im TS Adapter sind fehlerhaft | Überprüfen Sie die Parametrierung des TS Adapter:<br><ul style="list-style-type: none"> <li>• Sind Wahlverfahren und Amtskennzahl entsprechend ihrem Telefonanschluss eingestellt?</li> <li>• Unterstützt das Modem am TS Adapter die für die Amtskennzahl parametrisierten Zeichen?</li> <li>• Ist bei einer Nebenstelle "Vor dem Wählen auf Freizeichen warten" deaktiviert?</li> </ul> |
| Initialisierung des Modems nicht ausreichend                  | Überprüfen Sie den String zur Initialisierung des Modems:<br><ul style="list-style-type: none"> <li>• Vielleicht benötigt das Modem zum Aufbauen einer Fernverbindung eine erweiterte Initialisierung.</li> <li>• Eigenschaften des Modem-Initialisierungsstrings beim TS Adapter MPI</li> </ul>  |
| Rückrufnummer stimmt nicht                                    | Überprüfen Sie die Parametrierung der von Ihnen vergebenen Rückrufnummer.   |

#### Anruf vom TS Adapter MPI erfolgt nicht

Die folgende Tabelle zeigt mögliche Ursachen und deren Behebung auf, falls kein Anruf vom TS Adapter MPI erfolgt.

| Mögliche Ursache                                | Überprüfung / Behebung  |
|---|---|
| Rufnummer stimmt nicht                          | Wird der Kommunikationsanweisung "PG_DIAL" die gewünschte Rufnummer übergeben?  |
| Parametrierung des TS Adapter MPI ist inkorrekt | Überprüfen Sie die Parametrierung des TS Adapter MPI:<br><ul style="list-style-type: none"> <li>• Sind Wahlverfahren und Amtskennzahl entsprechend ihrem Telefonanschluss eingestellt?</li> <li>• Unterstützt das Modem am TS Adapter MPI die für die Amtskennzahl parametrisierten Zeichen?</li> <li>• Ist bei einer Nebenstelle "Vor dem Wählen auf Freizeichen warten" deaktiviert?</li> </ul> |



### 12.7.6.5 Modemverbindung wird unterbrochen

#### Modemverbindung wird unterbrochen

Die folgende Tabelle zeigt mögliche Ursachen und deren Behebung auf, falls die Modemverbindung unterbrochen wird.

| Mögliche Ursache:   | Überprüfung / Behebung:   |
|---|---|
| Gebührenimpulse in der Leitung                              | Wenn Sie bei der Telefongesellschaft einen Gebührentakt beantragt haben, werden Gebührenimpulse erzeugt. Diese können dazu führen, dass das Modem das Trägersignal nicht mehr erkennt und abschaltet. <ul style="list-style-type: none"><li>• Stellen Sie eine längere Warte- bzw. Abschaltzeit am Modem ein.</li><li>• Lassen Sie von der Telefongesellschaft den Gebührentakt abschalten.</li></ul>                   |
| Abschirmung   | <ul style="list-style-type: none"><li>• Sind die verwendeten Verbindungskabel ausreichend abgeschirmt?</li><li>• Stellen Sie sicher, dass die Modem-Zuleitungen nicht neben Netzkabeln verlaufen und möglichst großen Abstand von Netzteilen und Bildschirmen haben.</li></ul>  |
| Protokoll-Timeout   | <ul style="list-style-type: none"><li>• Stellen Sie feste Überwachungszeiten ein.</li></ul>   |
| Automatischer Verbindungsabbau                              | <ul style="list-style-type: none"><li>• Schalten Sie die Option aus, dass sich eine bestehende Verbindung nach einer bestimmten Zeit ohne Datenübertragung selbst abschaltet ("Trennen nach Leerlauf von ...").</li></ul>   |
| Datenflusskontrolle ausgeschaltet                           | <ul style="list-style-type: none"><li>• Klicken Sie auf die Schaltfläche "Erweitert" im Register "Einstellungen" der Modemeigenschaften und aktivieren Sie im angezeigten Dialogfeld folgende Optionen (falls angeboten und noch nicht eingestellt):<ul style="list-style-type: none"><li>– Datenflusskontrolle</li><li>– Hardware (RTS/CTS)</li><li>– Datenkomprimierung</li><li>– Fehlerkontrolle</li></ul></li></ul> |
| Initialisierungsstring des TS Adapter passt nicht zum Modem | <ul style="list-style-type: none"><li>• Stellen Sie den Modem-Initialisierungsstring gemäß den Anforderungen ein. Siehe auch: Parametriermöglichkeiten für den TS Adapter IE (Seite 4366)</li></ul>   |

#### Siehe auch

Parametriermöglichkeiten für den TS Adapter MPI (Seite 4358)

### 12.7.6.6 Checkliste bei Schwierigkeiten mit dem Modem

#### Einführung

Die folgende Liste soll Ihnen bei Problemen mit dem Modem helfen, die mögliche Ursache zu finden. Wie und in welchen Dialogfeldern Sie die genannten Einstellungen definieren, erfahren Sie in den unten aufgeführten Hilfethemen.

#### Die Modemverbindung kommt nicht zustande:

- Prüfen Sie die Verkabelung und die Anschlüsse.
- Prüfen Sie, ob das Wahlverfahren (Ton/Impuls) korrekt eingestellt ist.

- Reagiert Ihr Modem nach einigen vergeblichen Wahlversuchen nicht mehr, kann es sein, dass eine Wahlsperre aktiv ist. Informieren Sie sich, wie bei Ihrem Modem die Wahlsperre implementiert ist.
- Betreiben Sie Ihr Modem an einem Hauptanschluss oder an einer Nebenstelle? Richten Sie die Eigenschaften und Wahlparameter des Modems entsprechend ein.
- Aktivieren Sie in den erweiterten Eigenschaften die Option, eine Protokolldatei aufzuzeichnen. Daraufhin wird der nächste Verbindungsaufbau in einer Datei im Windows-Verzeichnis protokolliert.
- Achten Sie darauf, dass die benutzten ISDN-TAs mit dem selben B- und D-Kanal-Protokoll arbeiten.

**Die Modemverbindung wird getrennt:**

- Gebührenimpulse können sich negativ auf eine Verbindung auswirken. Lassen Sie die Impulse von Ihrem Telekommunikationsanbieter abschalten.
- Stellen Sie feste Überwachungszeiten ein.
- Schalten Sie die Option aus, dass sich eine bestehende Verbindung nach einer bestimmten Zeit ohne Datenübertragung (Leerlauf) selbst abschaltet.
- Vergewissern Sie sich, dass Sie RTS/CTS für die Datenflusskontrolle aktiviert haben.

**12.7.6.7 Meldungen des Modems**

**Informationen in der Protokolldatei**

Die Meldungen des Modems werden in einer Protokolldatei eingetragen, sofern Sie die Aufzeichnung aktiviert haben.

Folgende Informationen sind in der Protokolldatei enthalten:

| Meldung:     | mögliche Ursache:   | Behebung:   |
|--------------|---|---|
| NO DIALTONE  | Eventuell wird auf dieser Leitung momentan ein Telefongespräch geführt.   | <ul style="list-style-type: none"><li>• Wiederholen Sie den Vorgang, wenn das Telefongespräch auf dieser Leitung beendet ist.</li></ul>   |
| NO CARRIER   | Der angewählte Teilnehmer ist nicht bereit, ist kein Modem oder kann mit der eingestellten Betriebsart keine Verbindung herstellen. | <ul style="list-style-type: none"><li>• Prüfen Sie die Rufnummern und die Einstellungen.</li></ul>  |
| BUSY         | Der angewählte Teilnehmer ist besetzt.  | <ul style="list-style-type: none"><li>• Versuchen Sie es später nochmals.</li></ul>   |
| DELAYED: ... | Wahlsperre  | <ul style="list-style-type: none"><li>• Informieren Sie sich in der Dokumentation zu Ihrem Modem, wie bei Ihrem Modem die Wahlsperre realisiert ist und entfernen Sie diese, falls gewünscht.</li></ul> |

### 12.7.6.8 Mögliche Fehlermeldungen bei VPN-Verbindungen

#### VPN-Verbindung zum TS Adapter IE Advanced wird nicht aufgebaut

Die folgende Tabelle zeigt mögliche Fehler und deren Behebung auf, falls keine VPN-Verbindung zum TS Adapter IE Advanced aufgebaut werden kann.

| Mögliche Fehler   | Überprüfung/Behebung   |
|---|--|
| Im Browser erscheint bei Aufruf des Web-Interface die Fehlermeldung "Die Webseite kann nicht angezeigt werden". | <ul style="list-style-type: none"> <li>• Hinweis: Dieses Problem tritt dann auf, wenn im Windows-Zertifikatspeicher ein CA-Zertifikat vorhanden ist, das den gleichen Namen wie das neu erzeugte CA-Zertifikat hat.</li> <li>• Falls bereits ein CA-Zertifikat für diesen TS Adapter im Windows-Zertifikatspeicher installiert ist und Sie auf dem TS Adapter ein neues CA-Zertifikat generiert haben, so entfernen Sie das alte CA-Zertifikat aus dem Windows-Zertifikatspeicher und installieren Sie das neu generierte CA-Zertifikat.</li> <li>• siehe auch: CA-Zertifikat für VPN-Verbindung installieren (Seite 4373)</li> <li>• siehe auch: CA-Zertifikat für VPN-Verbindung löschen (Seite 4376)</li> </ul> |
| Fehlermeldung "Beim Verbindungsaufbau ist eine Störung aufgetreten".  | <ul style="list-style-type: none"> <li>• Überprüfen Sie, ob in der Benutzerdatenbank des TS Adapters IE Advanced weitere Benutzer angelegt sind.</li> <li>• Überprüfen Sie, welcher Benutzer versucht, eine Fernverbindung aufzubauen. Wenn Sie als "Administrator" angemeldet sind, können Sie keine Fernverbindung aufbauen.</li> </ul>  |
| Fehlermeldung "Es konnte kein passendes CA-Zertifikat im Windows-Zertifikatspeicher gefunden werden".           | <ul style="list-style-type: none"> <li>• Nutzen Sie den automatischen Zertifikatsdownload. Tragen Sie dazu den Fingerabdruck in das entsprechende Feld im Verbindungsdialog ein.</li> <li>• siehe auch: VPN-Verbindung aufbauen (Seite 4377)<br/>Oder</li> <li>• Exportieren Sie das CA-Zertifikat aus dem Web-Interface des TS Adapter IE Advanced und installieren Sie es manuell in den Windows-Zertifikatspeicher.</li> <li>• siehe auch: CA-Zertifikat für VPN-Verbindung installieren (Seite 4373)</li> </ul>  |
| Fehlermeldung: "Die angegebene Remote-Adresse passt nicht zur Remote-Adresse des TS Adapters".                  | <ul style="list-style-type: none"> <li>• Sie müssen für den Verbindungsaufbau die Remote-Adresse verwenden, die Sie im TS Adapter IE Advanced angegeben haben.</li> <li>• Es ist nicht möglich bei eingetragenen DNS-Namen die IP-Adresse (oder umgekehrt) zu verwenden.</li> </ul>  |
| Fehlermeldung: "Die Signatur des Zertifikats konnte nicht überprüft werden".                                    | <ul style="list-style-type: none"> <li>• Falls Sie auf dem TS Adapter IE Advanced ein neues CA-Zertifikat generiert haben, müssen Sie das alte CA-Zertifikat aus dem Windows-Zertifikatspeicher löschen und das neue installieren.</li> <li>• siehe auch: CA-Zertifikat für VPN-Verbindung löschen (Seite 4376)</li> <li>• siehe auch: CA-Zertifikat für VPN-Verbindung installieren (Seite 4373)</li> </ul>   |
| Das Versenden einer E-Mail funktioniert nicht.  | <ul style="list-style-type: none"> <li>• Überprüfen Sie, ob ein Netzwerk/Internetverbindung am WAN-Port besteht.</li> <li>• Überprüfen Sie, ob der Mailserver erreichbar ist.</li> <li>• Überprüfen Sie, ob im TS Adapter IE Advanced ausgehende Verbindungen erlaubt sind.</li> <li>• Überprüfen Sie, ob der SMTP-Port in der Firewall für ausgehende Verbindungen geöffnet ist.</li> </ul>   |



## Hardware-Dokumentation

### 13.1 Allgemeines zur Hardware-Dokumentation

#### Weiterführende Informationen zur verfügbaren Hardware

Über das TIA-Portal lässt sich je nach installierten Produkten eine große Menge unterschiedlicher Hardware projektieren. Die verfügbare Hardware finden Sie im Hardware-Katalog. Alle aktuellen Handbücher, Betriebsanleitungen und FAQs sowie Updates zu Ihren Geräten finden Sie im Service- und Support-Bereich (<https://support.automation.siemens.com/>) der Siemens-Internetseite.

Um Ihnen das Auffinden der passenden Dokumente zu Ihrer Hardware im Service- und Support-Bereich zu erleichtern, finden Sie in den folgenden Hilfefkapiteln alle Baugruppen und Baugruppenfamilien, die Ihnen im aktuellen Installationsumfang des TIA-Portals zur Verfügung stehen. Zu jeder Baugruppe finden Sie einen Link, der Sie direkt zu den zugehörigen Handbüchern und Betriebsanleitungen im Service- und Support-Bereich führt.

### 13.2 HMI

#### 13.2.1 Basic Panels

##### 13.2.1.1 Basic Panels

Informationen zu Basic Panels finden Sie hier (<http://support.automation.siemens.com/WW/view/de/28426379/133300>).

#### 13.2.2 Panels

##### 13.2.2.1 70er Serie Panels

Informationen zu Panels der 70er Serie finden Sie hier (<http://support.automation.siemens.com/WW/view/de/15271786/133300>).

##### 13.2.2.2 170er Serie Panels

Informationen zu Panels der 170er Serie finden Sie hier (<http://support.automation.siemens.com/WW/view/de/10805566/133300>).

### 13.2.2.3 270er Serie Panels

Informationen zu Panels der 270er Serie finden Sie hier (<http://support.automation.siemens.com/WW/view/de/10805567/133300>).

## 13.2.3 Comfort Panels

### 13.2.3.1 Comfort Panels

Informationen zu Comfort Panels finden Sie hier (<http://support.automation.siemens.com/WW/view/de/47182890/133300>).

## 13.2.4 Multi Panels

### 13.2.4.1 170er Serie

Informationen zu Multi Panels der 170er Serie finden Sie hier. (<http://support.automation.siemens.com/WW/view/de/28421795/133300>)

### 13.2.4.2 270er Serie

Informationen zu Multi Panels der 270er Serie finden Sie hier (<http://support.automation.siemens.com/WW/view/de/10805569/133300>).

### 13.2.4.3 370er Serie

Informationen zu Multi Panels der 370er Serie finden Sie hier (<http://support.automation.siemens.com/WW/view/de/10805570/133300>).

## 13.2.5 Mobile Panels

### 13.2.5.1 170er Serie

Informationen zu Mobile Panels der 170er Serie finden Sie hier (<http://support.automation.siemens.com/WW/view/de/40104710>).

### 13.2.5.2 270er Serie

Informationen zu Mobile Panels der 270er Serie finden Sie hier (<http://support.automation.siemens.com/WW/view/de/22584001/133300>).

## 13.2.6 Key Panels

### 13.2.6.1 Key Panels

Informationen zu Key Panels finden Sie hier (<http://support.automation.siemens.com/WW/view/de/47416561>).

### 13.2.6.2 Push Button Panels

Informationen zu Push Button Panels finden Sie hier (<http://support.automation.siemens.com/WW/view/de/19860219/133300>).

## 13.2.7 WinAC für Multi Panels

### 13.2.7.1 WinAC for Multi Panel

Informationen zu WinAC MP finden Sie hier (<http://support.automation.siemens.com/WW/view/de/10997567/130000>).

## 13.3 PLC

### 13.3.1 SIMATIC S7-1200

#### 13.3.1.1 CPU

#### CPU 1211C (6ES7 211-1xxx-0XB0)

Informationen zu den CPUs 1211C AC/DC/Rly, 1211C DC/DC/DC und 1211C DC/DC/Rly finden Sie hier (<http://support.automation.siemens.com/WW/lisapi.dll?>

[func=cslib.csinfo&lang=de&objid=6ES72111BD300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view](http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72111BD300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view)).

### CPU 1212C (6ES7 212-1xxx-0XB0)

Informationen zu den CPUs 1212C AC/DC/Rly, 1212C DC/DC/DC und 1212C DC/DC/Rly finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72121BD300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### CPU 1214C (6ES7 214-1xxx-0XB0)

Informationen zu den CPUs 1214C AC/DC/Rly, 1214C DC/DC/DC, 1214C DC/DC/Rly finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72141BE300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### CPU 1215C (6ES7 215-1xxx-0XB0)

Informationen zu den CPUs 1215C DC/DC/DC, 1215C AC/DC/Rly und 1215C DC/DC/Rly finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72151HG400XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### CPU 1217C (6ES7 217-1xxx-0XB0)

Informationen zur CPU 1217C DC/DC/DC finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72171AG400XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).



### 13.3.1.2 Signalboards (6ES7 2xx-xxx30-0XB0)

Informationen zu den Signalboards für S7-1200 finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72213BD300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### 13.3.1.3 CB 1241 (6ES7 241-1CH30-1XB0)

Informationen zum Kommunikationsboard CB 1241 finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72411CH301XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### 13.3.1.4 BB 1297 (6ES7 297-0AX30-0XA0)

Informationen zum Kommunikationsboard CB 1241 finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?query=BB+1297&func=cslib.cssearch&content=adsearch%2Fadsearch.aspx&lang=de&siteid=csius&objaction=cssearch&searchinprim=0&nodeid=41886045>).

### 13.3.1.5 Digitaleingabemodule (6ES7 221-1Bx30-0XB0)

Informationen zu den Digitaleingabemodulen DI8 x 24VDC und DI16 x 24VDC finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?>

[func=cslib.csinfo&lang=de&objid=6ES72221BF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view](http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72221BF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view)).

### 13.3.1.6 Digitalausgabemodule (6ES7 222-1xx30-0XB0)

Informationen zu den folgenden Digitalausgabemodulen finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72221BF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>):

- DQ8 x 24VDC
- DQ16 x 24VDC
- DQ16 x relay
- DQ8 x relay
- DQ8 x NO/NC relay

### 13.3.1.7 Digitalein- und Digitalausgabemodule (6ES7 223-1xx30-0XB0)

Informationen zu den folgenden Digitalein- und -ausgabemodulen finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72231BH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>):

- DI8/DQ8 x 24VDC
- DI16/DQ16 x 24VDC
- DI8 x 24VDC / DQ8 x relay
- DI16 x 24VDC / DQ16 x relay
- DI8 x 120VDC / DQ8 x relay

### 13.3.1.8 Analogeingabemodule (6ES7 231-xxx30-0XB0)

Informationen zu den folgenden Analogeingabemodulen finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll>)

[func=cslib.csinfo&lang=de&objid=6ES72314HD300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view](http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72314HD300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view)):

- AI4 x 13Bit
- AI4 x 16Bit
- AI8 x 13Bit
- AI4 x RTD
- AI8 x RTD
- AI4 x TC
- AI8 x TC

#### 13.3.1.9 Analogausgabemodule (6ES7 234-4Hx30-0XB0)

Informationen zu den Analogausgabemodulen AQ2 x 14Bits und AQ4 x 14Bits finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72324HB300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

#### 13.3.1.10 Analogein- und Analogausgabemodul (6ES7 234-4HE30-0XB0)

Informationen zu den Analogein- und -ausgabemodulen AI4 x 13Bits / AQ2 x 14Bits finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72324HB300XB0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

#### 13.3.1.11 Kommunikationsmodule

##### Industrial Remote Control

#### CP 1242-7 (6GK7 242-7KX30-0XE00)

Informationen zum Telecontrol Kommunikationsmodul CP 1242-7 finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6GK72427KX300XE0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### CP 1243-1 (6GK7 243-1JX30-0XE0)

Informationen zum Ethernet Kommunikationsmodul CP 1243-1 finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6GK72431JX300XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

## PROFIBUS

### CM 1242-5 (6GK7 242-5DX3x-0XE0)

Informationen zum PROFIBUS Kommunikationsmodul CM 1242-5 finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6GK72425DX300XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### CM 1243-5 (6GK7 243-5DX3x-0XE0)

Informationen zum PROFIBUS Kommunikationsmodul CM 1243-5 finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6GK72435DX300XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

## Punkt-zu-Punkt

### CM 1241 RS232 (6ES7 241-1AH30-0XB0)

Informationen zum Kommunikationsmodul für Punkt-zu-Punkt-Verbindungen CM 1241 (RS232) finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72411AH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### CM 1241 RS232 (6ES7 241-1AH32-0XB0)

Informationen zum Kommunikationsmodul für Punkt-zu-Punkt-Verbindungen CM 1241 (RS232) finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72411AH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### CM 1241 RS485 (6ES7 241-1CH30-0XB0)

Informationen zum Kommunikationsmodul für Punkt-zu-Punkt-Verbindungen CM 1241 (RS485) finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll>).

[func=cslib.csinfo&lang=de&objid=6ES72411CH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view](http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72411CH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view)).

#### CM 1241 RS422/485 (6ES7 241-1CH31-0XB0)

Informationen zum Kommunikationsmodul für Punkt-zu-Punkt-Verbindungen CM 1241 (RS422/485) finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72411CH310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### CM 1241 RS422/485 (6ES7 241-1CH32-0XB0)

Informationen zum Kommunikationsmodul für Punkt-zu-Punkt-Verbindungen CM 1241 (RS422/485) finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72411CH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### AS-Interface

#### AS-Interface CM 1243-2 (3RK7243-2AA30-0XB0)

Informationen zum AS-i Kommunikationsmodul CM 1243-2 finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=3RK72432AA300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### 13.3.1.12 Technologiemodule

#### SIWAREX WP231 (7MH4960-2AA01)

Informationen zum Wägemodul SIWAREX WP231 finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=7MH4960-2AA01&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### 4SI IO-Link (6ES7 278-4BD32-0XB0)

Informationen zum IO-Link-Master Technologiemodul 4SI IO-Link finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES72784BD320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

## 13.4 Dezentrale Peripherie

### 13.4.1 ET 200MP

#### 13.4.1.1 Interfacemodule

##### PROFINET

###### IM 155-5 PN ST (6ES7 155-5AA00-0AB0)

Informationen zu dem Dezentralen Peripheriemodul IM 155-5 PN ST finden Sie hier (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71555AA000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### 13.4.2 ET 200SP

#### 13.4.2.1 Interfacemodule

##### PROFINET

###### IM 155-6 PN ST (6ES7155-6AU00-0BN0)

Informationen zu dem Interface-Modul IM 155-6 PN ST hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71556AU000BN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

###### IM 155-6 PN ST V1.1 (6ES7 155-6AU00-0BN0)

Informationen zu dem Interface-Modul IM 155-6 PN ST V1.1 hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71556AU000BN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

**IM 155-6 PN HS (6ES7155-6AU00-0NN0)**

Informationen zu dem Interface-Modul IM 155-6 PN HS hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71556AU000NN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

**IM 155-6 PN HF (6ES7155-6AU00-0CN0)**

Informationen zu dem Interface-Modul IM 155-6 PN HF hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71556AU000CN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

**IM 155-6 PN HF V2.0 (6ES7 155-6AU00-0CN0)**

Informationen zu dem Interface-Modul IM 155-6 PN HF V2.0 finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71556AU000CN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

**Busadapter****Busadapter BA 2xRJ45 (6ES7 193-6AR00-0AA0)**

Informationen zu dem Busadapter BA 2xRJ45 finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71936AR000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

**Busadapter FC (6ES7 193-6AF00-0AA0)**

Informationen zu dem Busadapter FC finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71936AF000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

## PROFIBUS

### IM 155-6 DP ST (6ES7155-6BA00-0BN0)

Informationen zu dem Interface-Modul IM 155-6 DP ST hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71556BA000BN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

## Point-to-Point

### CM PtP (6ES7137-6AA00-0BA0)

Informationen zu dem Kommunikationsmodul CM PtP finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71376AA000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

## Weitere Feldbusse

### IM 155-6 Receive (6ES7155-6DU00-0BN0)

Informationen zu dem Interface-Modul IM 155-6 Receive hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71386AA000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

## Busadapter

### Busadapter Send (6ES7 193-6AS00-0AA0)

Informationen zu dem Busadapter Send finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71936AS000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### Busadapter Receive (6ES7 193-6AE00-0AA0)

Informationen zu dem Busadapter Receive finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll>).



[func=cslib.csinfo&lang=de&objid=6ES71936AE00AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view](https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71936AE00AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view)).

### 13.4.2.2 Digitaleingabemodule

#### DI 8x24VDC ST (6ES7 131-6BF00-0BA0)

Informationen zu dem Digitaleingabemodul DI 8x24VDC ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71316BF000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### DI 8x24VDC HF (6ES7 131-6BF00-0CA0)

Informationen zu dem Digitaleingabemodul DI 8x24VDC HF finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71316BF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### DI 8x24VDC HF V2.0 (6ES7 131-6BF00-0CA0)

Informationen zu dem Digitaleingabemodul DI 8x24VDC HF finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71316BF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### DI 8xNAMUR (6ES7 131-6TF00-0CA0)

Informationen zu dem Digitaleingabemodul DI 8x24VDC HF finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71316TF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### DI 16x24VDC ST (6ES7 131-6BH00-0BA0)

Informationen zu dem Digitaleingabemodul DI 16x24VDC ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71316BH000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### 13.4.2.3 Digitalausgabemodule

#### DQ 16x24VDC/0.5A ST (6ES7 132-6BH00-0BA0)

Informationen zu dem Digitalausgabemodul DQ 16x24VDC/0.5A ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71326BH000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### RQ 4x24VDC..230VAC/5A NO (6ES7 132-6HD00-0BB0)

Informationen zu dem Relaisausgabemodul RQ 4x24VUC...230VUC/5A finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71326HD000BB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### DQ 4x24VDC/2A ST (6ES7 132-6BD20-0BA0)

Informationen zu dem Digitalausgabemodul DQ 4x24VDC/2A ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71326BD200BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### DQ 8x24VDC/0.5A ST (6ES7 132-6BF00-0BA0)

Informationen zu dem Digitalausgabemodul DQ 8x24VDC/0.5A ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71326BF000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### DQ 8x24VDC/0.5A HF (6ES7 132-6BF00-0CA0)

Informationen zu dem Digitalausgabemodul DQ 8x24VDC/0.5A HF finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71326BF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### DQ 8x24VDC/0.5A HF V2.0 (6ES7 132-6BF00-0CA0)

Informationen zu dem Digitalausgabemodul DQ 8x24VDC/0.5A HF V2.0 finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll>).

[func=cslib.csinfo&lang=de&objid=6ES71326BF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view](https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71326BF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view)).

#### 13.4.2.4 Analogeingabemodule

##### AI 4xRTD/TC 2-/3-/4-wire HF (6ES7 134-6JD00-0CA1)

Informationen zu dem Analogeingabemodul AI 4xRTD/TC 2-/3-/4-wire HF finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71346JD000CA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

##### AI 4xRTD/TC 2-/3-/4-wire HF V2.0 (6ES7 134-6JD00-0CA1)

Informationen zu dem Analogeingabemodul AI 4xRTD/TC 2-/3-/4-wire HF V2.0 finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71346JD000CA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

##### AI 4xU/I 2-wire ST (6ES7 134-6HD00-0BA1)

Informationen zu dem Analogeingabemodul AI 4xU/I 2-wire ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71346HD000BA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

##### AI 4xI 2-/4-wire ST (6ES7 134-6GD00-0BA1)

Informationen zu dem Analogeingabemodul AI 4xI 2-/4-wire ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71346GD000BA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

##### AI 3x400VAC/1-5A ST (6ES7 134-6PA00-0BD0)

Informationen zu dem Analogeingabemodul AI 3x400VAC/1-5A ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71346PA000BD0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### AI 2xU/I 2-/4-wire HS (6ES7 134-6HB00-0DA0)

Informationen zu dem Analogeingabemodul AI 2xU/I 2-/4-wire HS finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES7134-6HB00-0DA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### AI 2xU/I HS (6ES7 134-6HB00-0DA1)

Informationen zu dem Analogeingabemodul AI 2xU/I HS finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71346HB000DA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### 13.4.2.5 Analogausgabemodule

#### AQ 2xU/I HS (6ES7 135-6HB00-0DA1)

Informationen zu dem Analogausgabemodul AQ 2xU/I HS finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71356HB000DA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### AQ 4xU/I ST (6ES7 135-6HD00-0BA1)

Informationen zu dem Analogausgabemodul AQ 4xU/I ST finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71356HD000BA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

#### AQ 2xU/I HS (6ES7 135-6HB00-0DA1)

Informationen zu dem Analogausgabemodul AQ 2xU/I HS finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71356HB000DA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view>).

### 13.4.2.6 Kommunikationsmodule

#### CM PtP (6ES7 137-6AA00-0BA0)

Informationen zum Kommunikationsmodul CM PtP finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71376AA000BA0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### 13.4.2.7 Spezialmodule

#### Servermodul (6ES7 193-6PA00-0AA0)

Informationen zu dem Servermodul finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71936PA000AA0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

#### Servermodul V2.0 (6ES7 193-6PA00-0AA0)

Informationen zu dem Servermodul finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71936PA000AA0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### 13.4.2.8 Technologiemodule

#### TM Count 1x24V (6ES7 138-6AA00-0BA0)

Informationen zu dem Relaisausgabemodul TMCOUNT 1x24V finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71386AA000BA0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

#### TM PosInput 1 (6ES7 138-6BA00-0BA0)

Informationen zu dem Positionierungsmodul TM PosInput 1 finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71386BA000BA0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### AI Energy Meter (6ES7 134-6PA00-0BD0)

Informationen zu dem Relaisausgabemodul AI Energy Meter finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71346PA000BD0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

### IO-Link Master Module (6ES7 137-6BD00-0BA0)

Informationen zu dem Relaisausgabemodul IO-Link Master Module finden Sie hier (<https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=de&objid=6ES71376BD000BA0&objaction=csviewmlfbbeitraege&sbtype=133300&caller=view>).

# Index

-

-, 1365

6ES7 155-6DU00-0BN0, 4410  
6ES7 155-6AU00-0CN0, 4409  
Absolutwert, 1737, 2003  
ANLAUF  
    Neustart (Warmstart), 836  
Ansicht vergrößern/verkleinern, (Siehe Zoom)  
Baugruppe  
    stecken, 402  
Externe Quelldatei  
    bearbeiten, 1450  
    öffnen, 1450  
Quality Code, 953  
SetzeFarbeHintergrundbeleuchtung, 3400  
SIMATIC PC Station  
    Laden, 68  
Übersetzen  
    Projekt, 3938  
Verbindungsparametrierung  
    Allgemein, 458

**&**

&, 1369, 1370

**\***

\*, 1365  
\*\*, 1365  
\*.bmp, 3062  
\*.cer, 588  
\*.dat, 588  
\*.emf, 3062  
\*.gif, 3062  
\*.ico, 3062  
\*.jpeg, 3062  
\*.jpg, 3062  
\*.p12, 522, 588  
\*.tif, 3062  
\*.wmf, 3062

.

.Net Service Packs, 29  
.Net Versionen, 29

**/**

/, 1365

**:**

:=, 1370, 1371  
:P, 847

**+**

+, 1365

**<**

<, 1367  
<=, 1367  
<>, 1367, 1370

**=**

=, 1367  
==, 1370

**>**

>, 1367  
>=, 1367

**3**

32-Bit Rückwärtszähler, 919  
3DES, 567  
3RK7243-2AA30-0XB0, 4407

**6**

6AV2124-0GC01-0AX0, 4400  
6AV2124-0JC01-0AX0, 4400

6AV2124-0MC01-0AX0, 4400  
6AV2124-0QC02-0AX0, 4400  
6AV2124-0UC02-0AX0, 4400  
6AV2124-0XC02-0AX0, 4400  
6AV2124-1DC01-0AX0, 4400  
6AV2124-1GC01-0AX0, 4400  
6AV2124-1JC01-0AX0, 4400  
6AV2124-1MC01-0AX0, 4400  
6AV3688-3AF37-0AX0, 4401  
6AV3688-3AY36-0AX0, 4401  
6AV3688-3EH47-0AX0, 4401  
6AV3688-4CX07-0AA0, 4401  
6AV3688-4EY06-0AA0, 4401  
6AV3688-4EY07-0AA0, 4401  
6AV6545-0CC10-0AX0, 4400  
6AV6591-1DC20-0AB0, 4400  
6AV6641-0AA11-0AX0, 4399  
6AV6641-0BA11-0AX0, 4399  
6AV6641-0CA01-0AX0, 4399  
6AV6642-0AA11-0AX0, 4400  
6AV6642-0AA11-0AX1, 4400  
6AV6642-0BA01-1AX0, 4400  
6AV6642-0BA01-1AX1, 4400  
6AV6642-0BC01-1AX0, 4400  
6AV6642-0BC01-1AX1, 4400  
6AV6642-0BD01-3AX0, 4400  
6AV6642-0DA01-1AX0, 4400  
6AV6642-0DA01-1AX1, 4400  
6AV6642-0DC01-1AX0, 4400  
6AV6642-0DC01-1AX1, 4400  
6AV6642-0EA01-3AX0, 4400  
6AV6642-8BA10-0AA0, 4400  
6AV6643-0AA01-1AX0, 4400  
6AV6643-0CB01-1AX0, 4400  
6AV6643-0CB01-1AX1, 4400  
6AV6643-0CD01-1AX0, 4400  
6AV6643-0CD01-1AX1, 4400  
6AV6643-0DB01-1AX0, 4400  
6AV6643-0DB01-1AX1, 4400  
6AV6643-0DD01-1AX0, 4400  
6AV6643-0DD01-1AX1, 4400  
6AV6643-0ED01-2AX0, 4400  
6AV6643-8AD10-0AA0, 4400  
6AV6644-0AA01-2AX0, 4400  
6AV6644-0AC01-2AX0, 4400  
6AV6644-0AC01-2AX1, 4400  
6AV6644-0BA01-2AX0, 4400  
6AV6644-0BA01-2AX1, 4400  
6AV6644-2AB01-2AX0, 4400  
6AV6645-0AA01-0AX0, 4401  
6AV6645-0AB01-0AX0, 4401  
6AV6645-0AC01-0AX0, 4401  
6AV6645-0BA01-0AX0, 4401  
6AV6645-0BB01-0AX0, 4401  
6AV6645-0BC01-0AX0, 4401  
6AV6645-0BE02-0AX0, 4401  
6AV6645-0CA01-0AX0, 4401  
6AV6645-0CB01-0AX0, 4401  
6AV6645-0CC01-0AX0, 4401  
6AV6645-0DD01-0AX1, 4401  
6AV6645-0DE01-0AX1, 4401  
6AV6645-0EB01-0AX1, 4401  
6AV6645-0EC01-0AX1, 4401  
6AV6645-0EF01-0AX1, 4401  
6AV6645-0FD01-0AX1, 4401  
6AV6645-0FE01-0AX1, 4401  
6AV6645-0GB01-0AX1, 4401  
6AV6645-0GC01-0AX1, 4401  
6AV6645-0GF01-0AX1, 4401  
6AV6647-0AA11-3AX0, 4399  
6AV6647-0AB11-3AX0, 4399  
6AV6647-0AC11-3AX0, 4399  
6AV6647-0AD11-3AX0, 4399  
6AV6647-0AE11-3AX0, 4399  
6AV6647-0AF11-3AX0, 4399  
6AV6647-0AG11-3AX0, 4399  
6AV6647-0AH11-3AX0, 4399  
6AV6647-0AJ11-3AX0, 4399  
6AV6647-0AK11-3AX0, 4399  
6AV6651-1AA01-0AA0, 4399  
6AV6651-1BA01-0AA0, 4399  
6AV6651-2AA01-0AA0, 4400  
6AV6691-1DA01-0AA1, 4399  
6AV6691-1DG01-0AA1, 4400  
6AV6691-1DJ01-0AA0, 4400  
6AV6691-1DJ01-0AB0, 4400  
6AV6691-1DJ01-0AC0, 4400  
6AV6691-1DJ01-0AD0, 4400  
6AV6691-1DJ01-0AE0, 4400  
6AV6691-1DR01-0AB0, 4400  
6ES7 131-6BF00-0BA0, 4411  
6ES7 131-6BF00-0CA0, 4411  
6ES7 131-6BH00-0BA0, 4411  
6ES7 132-6BD20-0BA0, 4412  
6ES7 132-6BF00-0BA0, 4412  
6ES7 132-6BF00-0CA0, 4412, 4413  
6ES7 132-6BH00-0BA0, 4412  
6ES7 132-6HD00-0BB0, 4412, 4416  
6ES7 134-6GD00-0BA1, 4413  
6ES7 134-6HB00-0DA1, 4414  
6ES7 134-6HD00-0BA1, 4413  
6ES7 134-6JD00-0CA1, 4413  
6ES7 134-6PA00-0BD0, 4413  
6ES7 135-6HB00-0DA1, 4414



---

|                           |                           |
|---------------------------|---------------------------|
| 6ES7 135-6HD00-0BA1, 4414 | 6ES7 221-1BH32-0XB0, 4404 |
| 6ES7 137-6AA00-0BA0, 4415 | 6ES7 221-3AD30-0XB0, 4403 |
| 6ES7 137-6BD00-0BA0, 4416 | 6ES7 221-3BD30-0XB0, 4403 |
| 6ES7 138-6AA00-0BA0, 4415 | 6ES7 222-1AD30-0XB0, 4403 |
| 6ES7 138-6BA00-0BA0, 4416 | 6ES7 222-1BD30-0XB0, 4403 |
| 6ES7 155-5AA00-0AB0, 4408 | 6ES7 222-1BF30-0XB0, 4404 |
| 6ES7 155-6AU00-0BN0, 4408 | 6ES7 222-1BF32-0XB0, 4404 |
| 6ES7 155-6AU00-0CN0, 4409 | 6ES7 222-1BH30-0XB0, 4404 |
| 6ES7 155-6AU00-0NN0, 4409 | 6ES7 222-1BH32-0XB0, 4404 |
| 6ES7 155-6BA00-0BN0, 4410 | 6ES7 222-1HF30-0XB0, 4404 |
| 6ES7 193-6AE00-0AA0, 4411 | 6ES7 222-1HF32-0XB0, 4404 |
| 6ES7 193-6AF00-0AA0, 4409 | 6ES7 222-1HH30-0XB0, 4404 |
| 6ES7 193-6AR00-0AA0, 4409 | 6ES7 222-1HH32-0XB0, 4404 |
| 6ES7 193-6AS00-0AA0, 4410 | 6ES7 222-1XF30-0XB0, 4404 |
| 6ES7 193-6PA00-0AA0, 4415 | 6ES7 222-1XF32-0XB0, 4404 |
| 6ES7 211-1AD30-0XB0, 4402 | 6ES7 223-0BD30-0XB0, 4403 |
| 6ES7 211-1AE31-0XB0, 4402 | 6ES7 223-1BH30-0XB0, 4404 |
| 6ES7 211-1AE40-0XB0, 4402 | 6ES7 223-1BH32-0XB0, 4404 |
| 6ES7 211-1BD30-0XB0, 4402 | 6ES7 223-1BL30-0XB0, 4404 |
| 6ES7 211-1BE31-0XB0, 4402 | 6ES7 223-1BL32-0XB0, 4404 |
| 6ES7 211-1BE40-0XB0, 4402 | 6ES7 223-1PH30-0XB0, 4404 |
| 6ES7 211-1HD30-0XB0, 4402 | 6ES7 223-1PH32-0XB0, 4404 |
| 6ES7 211-1HE31-0XB0, 4402 | 6ES7 223-1PL30-0XB0, 4404 |
| 6ES7 211-1HE40-0XB0, 4402 | 6ES7 223-1PL32-0XB0, 4404 |
| 6ES7 212-1AD30-0XB0, 4402 | 6ES7 223-1QH30-0XB0, 4404 |
| 6ES7 212-1AE31-0XB0, 4402 | 6ES7 223-1QH32-0XB0, 4404 |
| 6ES7 212-1AE40-0XB0, 4402 | 6ES7 223-3AD30-0XB0, 4403 |
| 6ES7 212-1BD30-0XB0, 4402 | 6ES7 223-3BD30-0XB0, 4403 |
| 6ES7 212-1BE31-0XB0, 4402 | 6ES7 231-4HA30-0XB0, 4403 |
| 6ES7 212-1BE40-0XB0, 4402 | 6ES7 231-4HD30-0XB0, 4405 |
| 6ES7 212-1HD30-0XB0, 4402 | 6ES7 231-4HD32-0XB0, 4405 |
| 6ES7 212-1HE31-0XB0, 4402 | 6ES7 231-4HF30-0XB0, 4405 |
| 6ES7 212-1HE40-0XB0, 4402 | 6ES7 231-4HF32-0XB0, 4405 |
| 6ES7 214-1AE30-0XB0, 4402 | 6ES7 231-5ND30-0XB0, 4405 |
| 6ES7 214-1AG31-0XB0, 4402 | 6ES7 231-5ND32-0XB0, 4405 |
| 6ES7 214-1AG40-0XB0, 4402 | 6ES7 231-5PA30-0XB0, 4403 |
| 6ES7 214-1BE30-0XB0, 4402 | 6ES7 231-5PD30-0XB0, 4405 |
| 6ES7 214-1BG31-0XB0, 4402 | 6ES7 231-5PD32-0XB0, 4405 |
| 6ES7 214-1BG40-0XB0, 4402 | 6ES7 231-5PF30-0XB0, 4405 |
| 6ES7 214-1HE30-0XB0, 4402 | 6ES7 231-5PF32-0XB0, 4405 |
| 6ES7 214-1HG31-0XB0, 4402 | 6ES7 231-5QA30-0XB0, 4403 |
| 6ES7 214-1HG40-0XB0, 4402 | 6ES7 231-5QD30-0XB0, 4405 |
| 6ES7 215-1AG31-0XB0, 4402 | 6ES7 231-5QD32-0XB0, 4405 |
| 6ES7 215-1AG40-0XB0, 4402 | 6ES7 231-5QF30-0XB0, 4405 |
| 6ES7 215-1BG31-0XB0, 4402 | 6ES7 231-5QF32-0XB0, 4405 |
| 6ES7 215-1BG40-0XB0, 4402 | 6ES7 232-4HA30-0XB0, 4403 |
| 6ES7 215-1HG31-0XB0, 4402 | 6ES7 232-4HB30-0XB0, 4405 |
| 6ES7 215-1HG40-0XB0, 4402 | 6ES7 232-4HB32-0XB0, 4405 |
| 6ES7 217-1AG40-0XB0, 4403 | 6ES7 232-4HD30-0XB0, 4405 |
| 6ES7 221-1BF30-0XB0, 4404 | 6ES7 232-4HD32-0XB0, 4405 |
| 6ES7 221-1BF32-0XB0, 4404 | 6ES7 234-4HE30-0XB0, 4405 |
| 6ES7 221-1BH30-0XB0, 4404 | 6ES7 234-4HE32-0XB0, 4405 |

6ES7 241-1AH30-0XB0, 4406  
 6ES7 241-1AH32-0XB0, 4406  
 6ES7 241-1CH30-0XB0, 4407  
 6ES7 241-1CH30-1XB0, 4403  
 6ES7 241-1CH31-0XB0, 4407  
 6ES7 241-1CH32-0XB0, 4407  
 6ES7 278-4BD32-0XB0, 4408  
 6ES7137-6AA00-0BA0, 4410  
 6ES7671-4EE00-0YA0, 4401  
 6ES7671-5EF01-0YA0, 4401  
 6GK7 242-5DX30-0XE0, 4406  
 6GK7 242-5DX31-0XE0, 4406  
 6GK7 242-7KX30-0XE0, 4406  
 6GK7 243-1JX30-0XE0, 4406  
 6GK7 243-5DX30-0XE0, 4406  
 6GK7 243-5DX31-0XE0, 4406

## 7

7MH4960-2AA01, 4407  
 7-Segment-Anzeige, 1880, 2151, 2345

## A

A-Adresse, 628  
 abgebaut, 3420  
 Abhängigkeitsstruktur, 1509  
     Ansichtsoptionen einstellen, 1512  
     anzeigen, 1512  
     Aufbau, 1510  
     Bedeutung der Symbole, 1511  
     Einführung, 1509  
 Ablage nicht gesteckter Baugruppen, 396  
 abmelden  
     Benutzer, 3996  
 Abmelden, 3352  
 Abmeldezeit, 3993  
     ändern, 3317  
     ändern in Runtime, 3324  
 ABS, 1737, 2003, 2201  
 Absolute Adressierung  
     einer Variable, 3164, 3484  
 Absolute Operanden ein- und ausblenden, 1375  
 Absolutwert, 2201  
 ACK, 4004  
 AcknowledgeAlarm, 3382  
 ACOS, 1754, 2022, 2217  
 ACT\_TINT, 2526  
 ActivatePreviousScreen, 3357  
 ActivateScreen, 3354  
 ActivateScreenByNumber, 3355

ADD, 1727, 1992  
 Addieren, 1727, 1992  
     Zeiten addieren mit T\_ADD, 2351  
 Admin, 3999  
 Administrator, 524  
 Adressband, 546  
 Adressbereich  
     verändern, 630  
 Adressbildung bei gepackten Adressen, 782  
 Adresse  
     Bereichszeiger, 3489  
     Mitsubishi, 3759  
     Omron Host Link, 3815  
 Adressen  
     Alarm bei gepackten, 789  
     einer Baugruppe ermitteln mit GEO\_LOG, 2628  
     entpacken, 782  
     packen, 782  
     Stationsadresse auslesen mit GetStationInfo, 2567  
     zuordnen, 631  
 adressieren  
     multiplexen, 3186  
     Variable indirekt adressieren, 3186  
 Adressieren von Operanden, 1065, 1066, 1067, 1069, 1070, 1072, 1074, 1076, 1077, 1078, 1079, 1081, 1082, 1083, 1085  
 Adressierung, 847  
     Allen-Bradley, 3718, 3736  
     Allen-Bradley Ethernet IP, 3714  
     allgemein, 628  
     EtherNet/IP, 3718, 3736  
     MPI, 3631  
     verändern, 630  
 Adress-Multiplexen  
     mit absoluten Adressen, 3189  
     mit symbolischen Adressen, 3190  
 Adressregister, 1082, 1083  
 Adressübersicht, 411  
 Adressumwandlung  
     Anweisungen zur, 2621  
 Advanced Encryption Standard (AES), 567  
 AES, 567  
 AES-128, 616, 626  
 Aggressive Mode, 567  
 Aging Time, 739  
 Aktionen  
     Grundlagen zum Rückgängig machen, 300  
     Grundlagen zum Wiederholen von Aktionen, 300  
     rückgängig machen, 302  
     wiederholen, 303

- Aktiv
  - Bereichszeiger, 3489
- Aktive Teilnehmer, 571
- Aktives Busmodul (ET 200M), 951
- AktiviereBild, 3354
- AktiviereBildMitNummer, 3355
- aktivieren, 3421
  - Projektsprache, 3886
- AktivierePutzbild, 3356
- AktiviereVorherigesBild, 3357
- aktualisieren
  - Betriebssystem, 3954
- AktualisiereVariable, 3358
- Aktualisierung
  - Variable, 3182, 3188
- Aktualisierung der Firmware , 3953
- Aktualisierung der Geräteversion, 3953
- Aktualisierung des Betriebssystems, 3953
- Aktualisierungszeit, 811
- Aktualisierungszyklus, 3200
- Alarmer
  - bei gepackten Adressen, 789
  - empfangen von Peripheriebaugruppe mit RALRM, 2399
  - Uhrzeitalarm abfragen mit QRY\_TINT, 2527
  - Uhrzeitalarm aktivieren mit ACT\_TINT, 2526
  - Uhrzeitalarm stellen mit SET\_TINT, 2522
  - Uhrzeitalarm stellen mit SET\_TINTL, 2524
  - Uhrzeitalarm stornieren mit CAN\_TINT, 2525
  - Verzögerungsalarm abfragen mit QRY\_DINT, 2532
  - Verzögerungsalarm starten mit SRT\_DINT, 2530
  - Verzögerungsalarm stornieren mit CAN\_DINT, 2531
- Alarmereignis
  - freigeben mit EN\_AIRT, 2546
  - freigeben mit EN\_IRT, 2544
  - sperrern mit DIS\_IRT, 2543
  - verzögern mit DIS\_AIRT, 2546
- Alle remanenten Daten zurücksetzen, 1834, 2105, 2306
- Allen-Bradley, 3704, 3726, 3727
  - Analogmeldung, 3745
  - Basic Panel, 3494
  - Datentyp, 3745
  - DF1, 3706
  - EtherNet/IP, 3706
  - Kommunikationstreiber, 3706
  - Kommunikationstreiber Allen-Bradley DF1, 3726, 3727
  - Mitsubishi, 3774
- Allen-Bradley
  - Bereichszeiger, 3743
- Allen-Bradley DF1
  - Anschluss, 3725
  - CPU-Typ, 3736
  - KF2 Modul, 3726, 3727
  - KF3 Modul, 3726, 3727
  - Migration der Datentypen, 139
  - Verbindung, 3722
  - Verbindung projektieren, 3722
  - Verbindungsparameter, 3723
  - zulässiger Datentyp, 3734
- Allen-Bradley Ethernet IP
  - Adressierung, 3714
  - Adressierungstyp, 3716
  - Adress-Multiplexen, 3717
  - Migration der Datentypen, 139
- Allen-Bradley EtherNet/IP
  - Anschluss, 3710
  - Datentyp, 3711
  - Verbindung, 3706
  - Verbindung projektieren, 3706
- Allen-Bradley EtherNET/IP
  - Verbindungsparameter, 3708
- Allgemeine Hinweise für die Fehlersuche, 4391
- Allgemeine Regeln, 1045
- Analogmeldeverfahren, 3205
- Analogmeldung, 3205
  - Allen-Bradley, 3745
  - Grenzwertüberschreitung, 3179
  - Mitsubishi, 3774
  - Omron, 3819
  - projektieren, 3219, 3226
- Analogmeldung importieren
  - Struktur für den Import, 3860
- Analogmeldungen
  - projektieren, 3214
- Analogmodul
  - auf Werkseinstellungen zurücksetzen, 1009
- AND, 1369, 1370, 1836, 1885, 2107
- andere Steuerungen
  - Adressierung, 3705
  - Besonderheiten, 3705
  - Datentypen, 3705
- Anfangsadresse, 628, 846
- Animation
  - Diagonale Bewegung, 3102
  - Direkte Bewegung, 3102
  - grüner Pfeil in Übersicht, 3098
  - Horizontale Bewegung, 3100
  - Mehrfachauswahl, 3105
  - Objektgruppe, 3105

- projektieren, 3098
- Übersicht, 3098
- Vertikale Bewegung, 3102
- Anlagenrelevante Informationen
  - Baugruppeninformation, 975
  - Herstellerinformation, 975
- Anlauf
  - Organisationsbausteine, 885
- ANLAUF
  - Anlauftätigkeiten, 836
  - Funktion, 835
- Anlaufart, 856
- Anlauf-OB
  - Beschreibung, 885
- Anlaufparameter, 838
- Anlaufprogramm, 885
- anlegen, 3286
  - Array, 3198
  - Benutzer, 3314, 3337, 3995, 3997
  - Benutzer in Runtime, 3321
  - Benutzergruppe, 3312, 3335
  - Bild, 3031
  - externe Variable, 3167
  - Globale Bibliothek, 3849
  - interne Variable, 3169
  - Rezeptur, 3286
  - Verbindung, 3473, 3477, 3499, 3548, 3603
  - Vorlage, 3037
- Anlegen
  - Meldegruppe, 3217
- Anmeldedialog
  - Zugriffsschutz projektieren, 3329
- anmelden
  - Anmeldung nicht erfolgreich, 3326
  - Benutzer, 3325
  - protokollieren, 3330
- Anmelden, 3358
- Anmelden für Webserver, 635
- anordnen
  - Objekt im Bild, 3043, 3054
- Anordnung der Bytereihenfolge, 1781, 2050, 2232
- anschließen
  - Steuerung, 3710, 3752, 3766, 3781, 3809
- Anschließen eines TS Adapter mit externem Modem, 4349
- Anschließen eines TS Adapter mit internem Modem, 4349
- Anschluss
  - Allen-Bradley DF1, 3725
  - Modicon, 3792
- Anschlussbelegung
  - 6XV1440-2P für Mitsubishi PG-Protokoll, 3766
- Allen-Bradley, 3730
- Allen-Bradley-Kabel 1747-CP3, 3732
- Allen-Bradley-Kabel 1761-CBL-PM02, 3733
- Allen-Bradley-Kabel 1784-CP10, 3731
- Mehr-Punkt-Kabel 1:MP/TP/PC, 3811
- Mehr-Punkt-Kabel 2:RS422, MP/TP/PC, 3812
- Punkt-zu-Punkt-Kabel 1,
  - Punkt-zu-Punkt-Kabel 2, 3796
- Punkt-zu-Punkt-Kabel-PP1 für Omron, 3812
- Punkt-zu-Punkt-Kabel-PP2 für Omron, 3813
- Ansicht verschieben
  - Tastaturbedienung, 386
  - Übersichtsnavigation, 370, 373, 376, 3452, 3458, 3461
- Ansichten der Querverweisliste, 1520, 3879
- Ansichtsoptionen
  - für den Belegungsplan, 1498
  - für die Abhängigkeitsstruktur einstellen, 1512
  - für die Aufrufstruktur einstellen, 1507
- Ansprechüberwachungszeit, 812
- Anstoß durch projektierte Funktion
  - Daten übertragen, 3533, 3586, 3645, 3685, 3836
- Antriebssignale, 4132
- Anweisung
  - ausschneiden, 1399
  - Datentyp festlegen, 1294, 1296, 1336, 1338
  - einfügen, 1400
  - kopieren, 1399
  - löschen, 1400
  - Regeln, 1380
  - Versionen, 1247
- Anweisung "WWW", 655
- Anweisungsprofil
  - aktivieren und deaktivieren, 1252
  - bearbeiten, 1251
  - erzeugen, 1249
  - Grundlagen, 1248
  - löschen, 1253
  - öffnen, 1251
- Anwenderdefinierte Textlisten
  - anlegen, 308
  - bearbeiten, 309
  - Wertebereich und Texte bearbeiten, 309
- Anwenderdefinierte Webseiten, 639, 642, 651, 652, 654, 655
- Anwenderdefinierte Webseiten synchronisieren, 3014
- Anwenderprogramm
  - Fehler suchen, 1401
  - Funktion, 1025
  - testen, 1524
- Anwendertexte, 241

- Anwendungsgebiet, 821  
 ANY, 1122  
 Anzeige einer Forcetabelle, 1565  
 anzeigen  
   Backup HMI, 3934  
   Grenzwertlinien beim Balken, 3128  
   Querverweis, 3881  
   Tooltip, 3978  
   von Querverweisen, 3878  
 Anzeigen  
   Aufrufstruktur, 1506  
   Abhängigkeitsstruktur, 1512  
   Belegungsplan, 1497  
   der Programminformationen, 1493  
   Ladespeicher, 1519  
   maximal verfügbaren Ladespeicher, 1519  
   Speicherauslastung der CPU, 1518  
   von Querverweisen, 1522  
 Anzeigeobjekt  
   Verfügbarkeit für Basic Panel, 3127  
 Anzeigename  
   Bereichszeiger, 3489  
 Applet, 526  
 Arbeitsbereich  
   Abgedockte Elemente einbetten, 189  
   Anordnung von Editoren und Tabellen sichern, 206  
   Elemente abdocken, 189  
   Elemente maximieren, 191  
   Elemente minimieren, 191  
   Funktion, 184  
   Gruppierte Elemente verwenden, 190  
   maximieren, 186  
   minimieren, 186  
   teilen, 188  
   zwischen den Elementen wechseln, 192  
 Arbeitsschritt  
   zum Erstellen von Bildern, 3031  
 Arbeitsspeicher, 842, 1515  
 Archiv  
   Runtime-Sprache, 3906  
   Sprachumschaltung, 3906  
 Archivieren von Projekten, 249, 250  
 Archivierungssprache, 3906  
 Arcuscosinus, 1754, 2022, 2217  
 Arcussinus, 1753, 2020, 2216  
 Arcustangens, 2218  
 Arcustangenswert, 1756, 2023  
 Array, 60, 164, 3196, 3198  
   anlegen, 3198  
   indirekt adressieren, 3186, 3196  
 ARRAY, 1079  
 Adressieren, 1070  
 Beispiel, 1117, 1118  
 Deklaration in der Bausteinschnittstelle, 1265  
 Deklaration in globalen Datenbausteinen, 1416  
 Deklaration in PLC-Datentypen, 1440  
 Format, 1114, 1115  
 Array-Datenbaustein, 1783, 1785, 1786, 1789, 2052, 2053, 2055, 2057, 2234, 2235, 2237, 2239, 2274  
 ARRAY-Datenbaustein, 1030, 1034, 1067, 1070, 1214, 1408, 1411, 1432  
 Arrayelement  
   Name, 60  
   Verwendungsstelle von HMI-Variable, 60  
 Arrayelemente, 3196  
 ARRAY-Grenzen, 1372  
 Arrayvariable, 3196  
   Char, 60  
 ASCII-Codetabelle, 482  
 ASCII-TSAP, 482  
 AS-i Netzkonfiguration parametrieren, 432  
 asiatische Sprachen  
   Projektierung, 3971  
   Schriftgröße, 3971  
   Textfeldlänge, 3971  
 asiatische Zeichen  
   Eingabe am Bediengerät, 3971  
   Interpretation, 3971  
   Speicherbedarf, 3971  
 asiatisches Betriebssystem, 3884  
 ASIN, 1753, 2020, 2216  
 AS-Interface, 793  
 Assistent  
   Geräte-Assistent, 3838  
 asynchron  
   Daten übertragen, 3530, 3583, 3640, 3686, 3831  
 Asynchronfehlerereignis  
   freigeben mit EN\_AIRT, 2546  
   freigeben mit EN\_IRT, 2544  
   sperrern mit DIS\_IRT, 2543  
   verzögern mit DIS\_AIRT, 2546  
 AT , 1074  
 ATAN, 1756, 2023, 2218  
 ATH, 2374  
 ATTACH, 852, 2517  
 ATTR\_DB, 2618  
 Audit-Log, 584, 586  
 Auf Anforderung, 3182  
 Aufbau, 932, 944  
   Aufrufstruktur, 1505  
   der Abhängigkeitsstruktur, 1510  
   der Querverweisliste, 1520, 3879  
   Register Speicherauslastung, 1517

Aufbau der Forcetabelle, 1562  
Aufbauen einer Fernverbindung, 4368  
    Vorgehen, 4369  
Aufgabe, 3434  
    deaktivieren, 3437  
Aufgabenplaner, 3432  
    Arbeitsbereich, 3435  
    deaktivierte Aufgabe, 3437  
    Ereignis-Trigger, 3437, 3440  
    Funktionsliste, 3436  
    Trigger, 3437  
aufheben  
    Gruppe, 3068  
auflösen  
    Gruppe, 3068  
Aufrufhierarchie, 1040  
Aufrufstruktur, 1502  
    Bedeutung der Symbole, 1504  
    Ansichtsoptionen einstellen, 1507  
    anzeigen, 1506  
    Aufbau, 1505  
    Einführung, 1502  
Aufzeichnen einer Protokolldatei für das Modem,  
4391  
Ausdruck  
    Grundlagen, 1364  
    Logischer Ausdruck, 1369  
    Vergleichsausdruck, 1367  
Ausdruck  
    arithmetischer Ausdruck, 1365  
Ausführen, 3422  
Ausgabe von Meldungen  
    projektieren:Ausgabe von Meldungen, 3228  
Ausgabefelder  
    Multiplexvariable, 61  
Ausgang  
    einfügen, 1309, 1351  
    entfernen, 1311, 1353  
Ausgang (A), 842  
Ausgangsbyte (AB), 842  
Ausgangsdoppelwort (AD), 842  
Ausgangswort (AW), 842  
auslesen  
    Rezepturdatensatz, 4017  
Auslesen  
    LED-Zustand mit LED, 2561  
    Lokalzeit mit RD\_LOC\_T, 2357  
ausrichten  
    Objekt bündig, 3052  
Ausschaltverzögerung, 1653, 1661, 1679, 1688,  
1914, 1941, 1951, 2164, 2184  
Auswahl geändert, 3422

auswählen  
    mehrere Objekte, 3043, 3050  
Authentifizierung, 695, 760  
Authentifizierungsmethoden, 565, 566  
Automatisierungssystem  
    aufbauen, 3442  
Automatisierungssystem, 3442, 3467  
    entfernt, 4385  
    lokal, 4385  
Autonegation aktivieren, 814, 3508, 3559, 3612  
Autonegotiation, 815  
    deaktiviert, 3509, 3560, 3613  
Autovervollständigung  
    Anweisung einfügen, 1255  
    Funktion, 1253  
    Variable einfügen, 1254, 1255  
AWL, 1606  
AWL-Quelle  
    Bausteine generieren, 1451  
AWP\_In\_Variable, 646  
AWP\_Out\_Variable, 645  
AWP-Kommando, 642, 644

## Ä

ÄndereKontrast, 3353  
ändern  
    Abmeldezeit in Runtime, 3324  
    Benutzer, 3998  
    Benutzergruppe in Runtime, 3325  
    Benutzergruppen Anzeigenname, 3318  
    Kennwort, 3316  
    Kennwort in Runtime, 3324  
    Name des Benutzers, 3324  
    Objektgröße, 3049  
Ändern  
    Bediengeräteversion, 3932  
Änderung, 3421

## B

Backup  
    löschen, 3935  
    umbenennen, 3935  
Backup von online Gerät laden, 4277  
Backup wiederherstellen, 4277  
Balken, 3128, 3980  
    Farbverlauf, 3128  
    Grenzwertlinien anzeigen, 3128  
Balkensegment  
    definieren, 3129

## Basic Panel

- Anzeige- und Bedienobjekt, 3127
- Bereichszeiger, 3496
- ETHERNET, 3495
- HTTP Protocol, 3494
- IF1B, 3495
- Kommunikationstreiber, 3494
- Mitsubishi, 3494
- Modicon Modbus, 3494
- Omron, 3494
- OPC, 3494
- Runtime-Start, 3942, 3964
- S7 1200, 3494
- S7 200, 3494
- S7 300, 3494
- S7 400, 3494
- Schnittstelle, 3495

## Basismodus, 1534

## Baugruppe

- adressieren, 628
- Anfangsadresse ermitteln mit GEO\_LOG, 2628
- Datensatz asynchron lesen mit RD\_DPARA, 2511
- Datensatz lesen mit RD\_DPAR, 2508
- Diagnosestatus ermitteln, 986
- einfügen, 403
- kopieren, 405
- logische Adresse des Steckplatzes ermitteln mit LOG\_GEO, 2630
- logische Adresse ermitteln mit GADR\_LGC, 2633
- logische Adressen ermitteln mit RD\_LGADR, 2631
- löschen, 404
- parametrieren und konfigurieren, 37
- selektieren, 394
- stecken, 38
- Steckplatz einer logischen Adresse ermitteln mit LGC\_GADR, 2634
- tauschen, 408
- Uhrzeit einer Baugruppe, 994
- verschieben, 407
- ziehen, 38

## Baugruppen simulieren, 365

## Baugruppenadressierung, 846

## Baugruppenanordnung, 390

## Baugruppenbeschriftung, 393

## Baugruppenrechte, 526

## Baugruppentausch, 150

## Baugruppenträger, 392

- Baugruppe stecken, 402

## Baugruppenträgerfehler-OB, 898

## Baugruppenwechsel im Betrieb, 951

## Baustein

- Arten, 1027
- Bibliothek verwenden, 1215
- Codebausteine vergleichen, 1454
- Datenbausteine vergleichen, 1455
- Eigenschaften, 1224
- Eigenschaften anzeigen, 1229
- Eigenschaften bearbeiten, 1229
- einfügen, 1217, 1221
- in externe Quelldatei exportieren, 1448
- know-how-geschützte Bausteine drucken, 1490
- know-how-geschützte Bausteine öffnen, 1489
- Know-how-Schutz, 1484
- Kommentar eingeben, 1222
- Konsistenzprüfung, 1468, 1508, 1513, 1514
- Konsistenzprüfung in der Aufrufstruktur, 1508
- kopieren, 1217, 1220
- Kopierschutz einrichten, 1487
- Kopierschutz entfernen, 1487
- Laden auf eine Memory Card, 1482
- Laden in Gerät, 1473, 1477
- Laden in Gerät im Betriebszustand RUN, 1476
- Laden von einer Memory Card, 1483
- Laden von Gerät, 1473
- offline löschen, 1233
- öffnen, 1230
- online löschen, 1234
- Optimierter Zugriff, 1034, 1036
- Passwort für know-how-geschützte Bausteine ändern, 1491
- schließen, 1232
- speichern, 1232
- suchen und öffnen, 1231
- Titel eingeben, 1222
- übersetzen, 1468
- Übersetzen im Programmiereditor, 1472
- Übersetzen in der Projektnavigation, 1471
- umbenennen, 1233
- Vergleich, 1452
- vergleichen, 1456
- verlassen, 2293
- von Gerät laden, 1481
- Zeitstempel, 1227

## Bausteinaufruf

- aktualisieren, 1300, 1342, 1396
- Aufruf als Einzel- oder Multiinstanz, 1041
- Aufruftyp ändern, 1301, 1343
- einfügen, 1299, 1341, 1388, 1389, 1391, 1392, 1393, 1394
- Einzelinstanz, 1041
- Grundlagen, 1039
- Multiinstanz, 1042

- Schachtelungstiefe, 1040
- Bausteineigenschaft
  - anzeigen, 1229
  - bearbeiten, 1229
  - Funktion, 1223
  - Übersicht, 1224
- Bausteinkommentar
  - ausblenden, 1245
  - einblenden, 1245
- Bausteinkonsistenz
  - in der Abhängigkeitsstruktur prüfen, 1514
  - prüfen, 1508
- Bausteinordner, 1211
- Bausteinparameter, 1044, 1046, 1048, 1050, 1052, 1053, 1054, 1055
  - Bausteinschnittstelle, 1258
  - Grundlagen, 900
- Bausteinschnittstelle
  - aktualisieren, 1269
  - ARRAY deklarieren, 1265
  - Aufbau, 1258
  - ausblenden, 1244
  - Eigenschaften von Variablen, 1272, 1274
  - einblenden, 1244
  - Gültige Datentypen, 1262
  - Multiinstanz, 1269
  - PLC-Datentyp deklarieren, 1267
  - Remanenz, 1273
  - STRUCT deklarieren, 1266
  - Variable deklarieren, 1262, 1263, 1268
  - Variablen importieren und exportieren, 1278
  - Zweck der Variablendeklaration, 1257
- Bausteinvergleich
  - Aktionen ausführen, 273
  - Baustein ändern, 1463
  - Bildlauf synchronisieren, 1463
  - Codebausteine vergleichen, 1454
  - Darstellung des Detailvergleichs, 1460
  - Datenbausteine vergleichen, 1455
  - Detailvergleich, 1458
  - Grundlagen, 1452
  - Navigation, 1462
  - Vergleichsergebnisse aktualisieren, 270, 1464
- Bausteinzugriff
  - Datenbaustein, 1422
- BCDCPL, 1882, 2153, 2347
- BearbeiteteMeldung, 3359
- bearbeiten
  - Gruppe, 3066
  - Meldung, 4007
  - Objekt innerhalb einer Gruppe, 3070
  - Ordnerverknüpfung, 3062
  - Rezepturdatensatz, 4015
- Bearbeiten
  - Funktionsliste, 3345
- Bedeutung der Spalten in der Forcetabelle, 1562
- Bedienelement
  - einfache Meldeanzeige, 3240, 4004
  - Kurvenanzeige, 3140, 3988
  - Meldeanzeige, 3143, 3145
- Bediengerät, 89
  - auf Werkseinstellungen zurücksetzen, 3954
  - Daten sichern, 3952
  - Daten wiederherstellen, 3950, 3952, 3966
  - Datensicherung, 3949, 3950, 3966
  - Erstinbetriebnahme, 3960
  - Gerätetyp ändern, 3957
  - Image, 3949
  - Laden, 3949
  - Leistungsmerkmale, 4020
  - MPI-Parameter, 3628
  - PROFIBUS-Parameter, 3516, 3566, 3619
  - PROFINET-Parameter, 3502, 3552, 3606
  - Software, 3949
  - Systemgrenzen, 4020
  - Wiederinbetriebnahme, 3960
- Bediengerät an einer Steuerung
  - in Betrieb nehmen, 3720, 3741, 3760, 3770, 3787, 3799, 3816
- Bediengeräte-Assistent, 3838
- Bediengerätetyp
  - ändern, 3957
- Bediengeräteversion, 3931, 3932, (Geräteversion)
  - prüfen, 3932
- Bediengerätewechsel, 49
- Bedienoberfläche
  - Ansichten, 176
  - Arbeitsbereich, 184
  - Arbeitsbereich maximieren, 186
  - Arbeitsbereich minimieren, 186
  - Detailansicht, 198
  - Inspektorfenster, 192
  - Palette "Referenzprojekte", 196
  - Portalansicht, 176
  - Projektansicht, 178
  - Projektnavigation, 181
  - Task Card, 194
  - Übersichtsfenster, 199
- Bedienoberfläche sichern, 203
- Bedienoberfläche speichern
  - Fensterlayout speichern, 203
- Bedienoberflächenlayout zurücksetzen, 206
- Bedienobjekt
  - Verfügbarkeit für Basic Panel, 3127



- Bedienung
  - Datum-Uhrzeit-Feld, 3981
  - EA-Feld, 3983
  - einfache Benutzeranzeige, 3995
  - einfache Meldeanzeige, 3240, 4003
  - einfache Rezepturanzeige, 3296, 4012
  - Grafikanzeige, 3985
  - grafisches EA-Feld, 3986
  - Kurvenanzeige, 3987
  - Meldeindikator, 3241
  - Schalter, 3990
  - Schaltfläche, 3989
  - symbolisches EA-Feld, 3991
- Bedienung in der Rezepturanzeige
  - Daten übertragen, 3583, 3642, 3682, 3833
- Bedienung in Runtime
  - Mehrtastenbedienung, 71
- beenden
  - Runtime, 3966
- Beenden
  - Anwenderprogramm, 1828, 2099, 2300
- Beenden des Forcens, 1583, 1585
- Befüllen
  - Bereich, 1777, 1796, 2045, 2064, 2228, 2257
  - Bereich ununterbrechbar, 1779, 2047, 2230
- Bei Überschreitung, 3422
- Bei Unterschreitung, 3423
- Beim Öffnen eines Dialogs, 3423
- Beim Schließen eines Dialogs, 3423
- Beispiel
  - Anwendung Meldeklassen, 3208
  - Berechnen einer Gleichung, 1609
  - Betriebsart wechseln und Betriebsart anzeigen, 3346
  - Erfassen der Richtung eines Förderbandes, 1606, 1616
  - Erfassen der Richtung eines Förderbands, 1591, 1598
  - Erfassen des Füllstands eines Lagerbereichs, 1592, 1599, 1607, 1617
  - für die Anzeige von Ein- und Ausgängen im Belegungsplan, 1495
  - für die Anzeige von Merkern im Belegungsplan, 1495
  - für die Eingabe von Forcewerten in die Forcetabelle, 1568
  - Heizen eines Ofens, 1610
  - Kontrollieren der Raumtemperatur, 1595, 1602
  - Steuern eines Förderbands, 1589, 1604, 1614
  - Steuerwerte in der Beobachtungstabelle, 1541
  - Systemmeldung, 3205
- Beispiel/
  - Bitmeldung, 3206
- Beispiele
  - Steuern eines Förderbands, 1597
- Belegen
  - eine Funktionstaste, 3108, 3111, 3112
- Belegungsplan
  - Bedeutung der Symbole, 1496
  - Ansichtsoptionen einstellen, 1498
  - Anzeige der remanenten Merker aktivieren, 1502
  - anzeigen, 1497
  - Aufbau, 1495
  - Beispiel für die Anzeige von Ein- und Ausgängen, 1495
  - Beispiel für die Anzeige von Merkern, 1495
  - Einführung, 1494
  - Filter definieren, 1499
  - Filter löschen, 1499
  - Filtermöglichkeiten, 1498
  - filtern, 1500
- Benutzer, 3993
  - abmelden, 3996
  - Admin, 3999
  - ändern, 3324
  - ändern, Touch-Bedienung, 3998
  - anlegen, 3314, 3337, 3995
  - anlegen in Runtime, 3321
  - anlegen, Touch-Bedienung, 3995, 3997
  - anmelden, 3325
  - Anmeldung protokollieren, 3330
  - bei Benutzerwechsel aktualisieren, 3439
  - Benutzergruppe zuordnen, 3315
  - einrichten, 523
  - entsperren, 3325
  - löschen, 3317, 3999
  - löschen in Runtime, 3324
  - Name ändern, 3316
  - Rollen anlegen, 524
  - Rollen zuweisen, 525
  - verwalten, 3316, 3323
- Benutzeranzeige, 3319
  - Anzahl Zeilen, 3130
  - Darstellung, 3130
  - Einfache Benutzeranzeige, 3320
  - projektieren, 3321
  - verwendbare Zeichen, 69
  - Verwendung, 3129
- Benutzerdaten
  - sichern, 3993
  - wiederherstellen, 3993
- Benutzerdefinierte Funktionen
  - in Funktionslisten, 3343

- Benutzerdefinierte Meldeklassen, 3209
- Benutzerdefinierte Rollen, 524
- Benutzergruppe, 3993
  - ändern in Runtime, 3325
  - anlegen, 3335
  - Anzeigename ändern, 3318
  - Benutzer zuordnen, 3315
  - Berechtigung verwalten, 3317
  - löschen, 3318
  - Name ändern, 3317
  - Unauthorized, 3326
  - verwalten, 3317
  - zuordnen, 3337
- Benutzername, 523
- Benutzeroberfläche sichern
  - Anordnung in Editoren sichern, 203
- Benutzerrechte erwerben
  - Am Betriebssystem mit Administratorrechten anmelden, 220
  - Mit der Windows Benutzerkontensteuerung, 219
- Benutzerverwaltung, 3304
  - aufbauen, 3331
  - Migration, 131
  - Objekt mit Zugriffsschutz, 3328
  - Runtime-Einstellungen, 3034, 3309, 3325
- Benutzerwechsel, 3424, 3439
- Beobachten
  - Befehl "Alle beobachten", 1550, 1574
  - Befehl "Sofort beobachten", 1551, 1575
- Beobachtungs- und Steuermodus, 1548
- Beobachtungsmodus, 1548
- Beobachtungstabelle
  - Bedeutung der Symbole, 1536
  - Aufbau, 1534
  - Basismodus, 1534
  - Bedeutung der Spalten, 1534
  - Beispiel für das Ausfüllen einer Beobachtungstabelle, 1539
  - Beobachtungs- und Steuermodus, 1548
  - Einsatzmöglichkeiten, 1533
  - erstellen, 1537
  - Erweiterter Modus, 1534
  - Funktion "Peripherieausgänge freischalten", 46
  - kopieren, 1538
  - Laden von Datenbausteinen bei laufendem Steuerauftrag, 45
  - Mehrfachzugriffe auf dieselbe CPU, 45
  - öffnen, 1538
  - speichern, 1539
  - Syntaxprüfung, 1539
  - Testen der Verdrahtung, 1533
  - Übersicht der Testmöglichkeiten, 1533
  - Übersicht über die Anzeigeformate, 1543
  - Umschalten zwischen Basismodus und Erweiterter Modus, 1535
  - zugelassene Operanden für Steuerwerte, 1541
  - zugelassenen Operanden, 1540
- Berechnen, 1302, 1344, 1723, 1988
- Berechtigung, 3993
  - erstellen, 3312, 3333
  - löschen, 3319
  - Name ändern, 3318
  - projektieren, 3327
  - verwalten, 3317
  - zuweisen, 3313, 3335
- Bereich
  - befüllen, 1777, 1796, 2045, 2064, 2228, 2257
  - Bereich, 2059
  - kopieren, 1769, 1791, 2037, 2221, 2252
  - ununterbrechbar befüllen, 1779, 2047, 2230
  - ununterbrechbar kopieren, 1775, 1793, 2043, 2062, 2226, 2254
  - Wert außerhalb Bereich, 1720, 1985
  - Wert innerhalb Bereich, 1719, 1984
- Bereich (... - ...)
  - Grafikliste, 3091
- Bereich kopieren, 1771, 2039, 2223
- Bereichszeiger, 3487, 3520, 3571, 3631, 3673, 3821
  - Adresse, 3489
  - Aktiv, 3489
  - Allen-Bradley, 3743
  - anlegen, 3491
  - Anzeigename, 3489
  - Basic Panel, 3496
  - Bildnummer, 3521, 3577, 3632, 3674, 3823
  - Datensatz, 3278, 3529, 3582, 3640, 3681, 3831
  - Datum/Uhrzeit, 63, 3522, 3572, 3573, 3633, 3675, 3824
  - Datum/Uhrzeit Steuerung, 63, 3523, 3634, 3676, 3825
  - Editor Verbindungen, 3488
  - Erfassungszyklus, 3489
  - Kommentar, 3489
  - Koordinierung, 3525, 3575, 3635, 3677, 3826
  - Kurven, 3536, 3699
  - Länge, 3489, 3491
  - Mitsubishi, 3772
  - Modicon Modbus, 3800
  - projektieren, 3491
  - Projektkennung, 3526, 3578, 3636, 3678, 3827
  - Registerkarte, 3489
  - Steuerauftrag, 3527, 3579, 3637, 3679, 3828
  - Steuerungsvariable, 3489

- Verbindung, 3488, 3492
- Bereichzeiger
  - Omron, 3817
- Beschränkungen durch Benutzerrechte, 218, 219
  - erkennen, 219
- Beschriftungsstreifen drucken, 297
  - Korrekturwert für Druckverschiebung ermitteln, 299
- Beschriftungsstreifen erstellen, 295
- Besonderheiten
  - Omron Host Link, 3818
- Bestandteile
  - Meldung, 3213
- Betrag, 1737, 2003
- Betriebsart wechseln und Betriebsart anzeigen (Beispiel), 3346
- Betriebsstundenzähler
  - hantieren mit RTM, 2363
- Betriebssystem
  - aktualisieren, 3954
  - asiatische Spracheinstellung, 3884
  - Einstellung auf westlichen, 3884
- Betriebssystem , 1025
- Betriebsverhalten
  - Systemmerker verwenden, 879
  - Taktmerker verwenden, 879
  - Uhrzeitfunktionen nutzen, 863
- Betriebszustand, 604
  - ANLAUF, 835
  - Einführung, 833
  - RUN, 838
  - STOP, 839
  - Übergänge, 834
- Betriebszustand umschalten, 992
- Bezeichnung
  - TS Adapter, 4363
  - TS Adapter, 4346, 4355
- Bibliothek, 3839
  - aufräumen, 363
  - Baustein hinzufügen und verwenden, 1215
  - Bibliothekselemente ausschneiden, 358
  - Bibliothekselemente vergleichen, 364
  - Bibliothekselemente verschieben, 359
  - Bibliothekselement kopieren, 3842, 3847
  - Bibliothekselement löschen, 3847
  - Buttons and Switches, 3843
  - Eigenschaften eines Typen anzeigen, 342
  - Elementansicht verwenden, 317
  - Filteransicht verwenden, 336
  - Freigegebene Typ-Version, 340
  - Freigegebene Typ-Version öffnen, 341
  - Grundlagen, 313
  - Instanzen löschen, 359
  - Kopiervorlage, 3841
  - Kopiervorlage hinzufügen, 335
  - Kopiervorlagen, 313
  - Kopiervorlagen aus der Zwischenablage einfügen, 359
  - Kopiervorlagen ausschneiden, 358
  - Kopiervorlagen einsetzen, 334
  - Kopiervorlagen verwenden, 337
  - Namen und Pfadstruktur harmonisieren, 362
  - Objekt hinterlegen, 3848
  - öffnen, 3851
  - Ordner erstellen, 333
  - speichern, 3850
  - systemerzeugte PLC-Datentypen, 43
  - Task Card, 315
  - Typ, 3842
  - Typen, 313
  - Typen aus der Zwischenablage einfügen, 359
  - Typen ausschneiden, 358
  - Typen einsetzen, 337
  - Typen in die Zwischenablage kopieren, 358
  - Typen und Versionen löschen, 359
  - Typ-Version im Test, 340
  - Typ-Version in Bearbeitung, 339
  - Versionierung von Typen, 338
- Bibliothek aktualisieren, 360
- Bibliotheksansicht
  - öffnen, 320
  - Übersicht, 318
  - verlassen, 321
- Bibliotheksobjekt, 3839, 3841
  - einfügen, 3849
- Bibliotheksobjekte vergleichen, 314
- Bibliotheksverwaltung
  - öffnen, 323
  - Übersicht, 321
- Bild
  - anlegen, 3031
  - Arbeitsschritt zum Erstellen, 3031
  - einfügen, 3032
  - Geräteabhängigkeit, 3027
  - kopieren, 3032, 3922
  - löschen, 3032
  - Schriftart, 3027
  - umbenennen, 3032
  - verschieben, 3032
  - Vorlage verwenden, 3039
  - zoomen, 3030
- Bildnummer, 3448
- Bildobjekt
  - Bewegung dynamisieren, 3100

- BildobjektCursorAbwärts, 3360
  - BildobjektCursorAufwärts, 3360
  - BildobjektSeiteAbwärts, 3361
  - BildobjektSeiteAufwärts, 3361
  - Bildschirmdarstellung, 33
  - Bildschirmtastatur
    - alphanumerische, 3975, 3977
    - KTP400 Basic, 3974
    - KTP600, KTP1000, TP1500 Basic, 3977
    - numerische, 3976, 3978
    - Tastenbelegung, 3975, 3977
  - Bildschirmtastatur verwenden, 218
  - Bildvorlage, 3111
  - Bildwechsel, 3424
  - Bilschirmtastatur
    - Sprachverhalten, 69
  - Bit (0, 1)
    - Grafikliste, 3092
    - Textliste, 3083
  - Bitfeld
    - rücksetzen, 1633, 1895
    - setzen, 1632, 1894
  - Bitfolge, 1091, 1092, 1093
    - 64 Bit, 1094
  - Bitgetriggerte Kurven, 3536, 3699
  - Bitmaske, 1872, 2143, 2337
  - Bitmeldeverfahren, 3205
  - Bitmeldung, 3205, 3206
    - Allen-Bradley, 3745
    - Mitsubishi, 3774
    - Omron, 3819
    - projektieren, 3217, 3225
  - Bitmeldung importieren
    - Struktur für den Import, 3863
  - Bitmeldungen
    - projektieren, 3214
  - Bitnummer (0 - 31)
    - Grafikliste , 3094
    - Textliste, 3085
  - Bits
    - zählen, 1883, 2154, 2348
  - BITSUM, 1883, 2154, 2348
  - Bitverknüpfung
    - Eingang einfügen, 1889
    - EXKLUSIV ODER, 1888
    - ODER, 1885, 1886
    - UND, 1884, 1885
  - blinken, 3099
  - Blinktest, 1014
  - BLKMOV, 1791, 2059, 2252
  - BLOCK, 1126
  - BOOL, 1091, 1136, 1152
  - BOOL\_TO\_, 1152
  - Boundaries, 816, 3510, 3561, 3614
  - BRCV, 2929
  - Broadcast, 547, 554, 741
  - BSEND, 2926
  - Buttons and Switches
    - Bibliothek, 3843
  - BYTE, 1091, 1136, 1153
  - BYTE\_TO\_, 1153
  - Bytes
    - Anordnung ändern, 2232
    - Anordnung ändern, 1781, 2050
- ## C
- CA-Gruppenzertifikat, 521
  - CA-Gruppenzertifikat erneuern, 571
  - CALC, 1723, 1988
  - CALCULATE, 1302, 1344
  - CalibrateTouchScreen, 3368
  - CAN\_DINT, 2531
  - CAN\_TINT, 2525
  - CASE, 2282
  - CA-Zertifikat, 518, 521, 4372
  - CA-Zertifikat installieren, 4373
  - CA-Zertifikat löschen, 4376
  - CEIL, 1801, 2070, 2263
  - Certificate Authority, 518
  - Change word order
    - Modicon MODBUS TCP/IP, 3780
  - ChangeConnection, 3411
  - Char
    - Arrayvariable, 60
  - CHAR, 1112, 1150, 1178
  - CHAR\_TO\_, 1178
  - Chars\_TO\_Strg, 2373
  - Checkliste bei Schwierigkeiten mit dem Modem, 4395
  - CJ1, 3814
  - CJ2, 3814
  - Class of Service, 708
  - ClearAlarmBuffer, 3378
  - ClearAlarmBufferProtoolLegacy, 3379
  - CMP <, 1717, 1982
  - CMP <=, 1713, 1978
  - CMP <>, 1709, 1974
  - CMP ==, 1706, 1972
  - CMP >, 1715, 1980
  - CMP >=, 1711, 1976
  - Codeabschnitte auf- und zuklappen, 1377
  - Comfort Panel
    - S7-1200, 68

- Compact, 3785, 3798
  - CompactLogix, 3713
  - CONCAT, 2379
  - CONTINUE, 2290
  - ControlLogix, 3713
  - CONVERT, 1798, 2067, 2260
  - Cookie, 648
  - CoS, 708
    - Warteschlange, 708
  - COS, 1750, 2017, 2214
  - Cosinus, 1750, 2017, 2214
  - COUNTER, 1126
  - CP 1242-7, 41
  - CP 1613, 31
  - CP 1623, 31
  - CP 343-2, 40
  - CP1, 3814
  - CPM, 3814
  - CPU
    - aktuellen LED-Zustand anzeigen, 983
    - Auswahl aus Hardware-Katalog, 397
    - Betriebszustand umschalten, 992
    - Datum und Uhrzeit lesen mit RD\_SYS\_T, 2356
    - Diagnosepuffer auslesen, 987
    - Eigenschaften, 855
    - Füllstand aller Speicherarten, 983, 984
    - Signalboard stecken, 633
    - Uhrzeit stellen mit WR\_SYS\_T, 2355
  - CPU-Bedienpanel
    - Anzeigebereich, 983
  - CPU-Belastung durch Trace, 4323
  - CPU-Datenbaustein
    - Definition, 1032
    - löschen, 1235
  - CPU-Eigenschaften, 856
  - CPU-Firmware, 908
  - CPU-Typ
    - Allen-Bradley DF1, 3736
    - Allen-Bradley EtherNet/IP, 3713
    - FX3 series, 3758
    - Mitsubishi FX, 3768
    - Mitsubishi MC TCP/IP, 3758
    - Modicon Modbus RTU, 3798
    - Modicon Modbus TCP, 3785
    - Omron Host Link, 3814
  - CREATE\_DB, 2611
  - CS1, 3814
  - CSV-Datei, 3854, 3856, 3865
  - CTD, 1692, 1956, 2189
  - CTRL\_HSC, 2670
  - CTRL\_PWM, 2583
  - CTU, 1689, 1953, 2186
  - CTUD, 1694, 1959, 2192
  - Cu10-Sensor, 941
  - Cyclic interrupt, 892
- ## D
- D\_ACT\_DP, 2425
  - Darstellung
    - Balken, 3128
    - Benutzeranzeige, 3130
    - Datum/Uhrzeit-Feld, 3130
    - Datum-Uhrzeit-Feld, 3981
    - E/A-Feld, 3132
    - EA-Feld, 3982
    - einfache Benutzeranzeige, 3994
    - einfache Rezepturanzeige, 3296, 4012
    - Ellipse, 3133
    - Grafikanzeige, 3134, 3985
    - Grafisches E/A-Feld, 3136
    - Kreis, 3138
    - Kurvenanzeige, 3139, 3987
    - Linie, 3141
    - Meldeanzeige, 3142, 3145, 3239, 4003
    - Meldeindikator, 3147, 3241
    - Rechteck, 3148
    - Schalter, 3150, 3990
    - Schaltfläche, 3151, 3989
    - Symbolisches E/A-Feld, 3153
    - Textfeld, 3156
  - Data Encryption Standard (DES), 568
  - Data Log, 638
    - erstellen mit DataLogCreate, 2593
    - erstellen mit DataLogNewFile, 2608
    - erstellen mit DataLogTypedNewFile, 2609
    - leeren mit DataLogClear, 2601
    - löschen mit DataLogDelete, 2606
    - öffnen mit DataLogOpen, 2597
    - öffnen mit DataLogTypedOpen, 2599
    - schließen mit DataLogClose, 2605
    - schreiben mit DataLogWrite, 2602
  - DataLogClear, 2601
  - DataLogClose, 2605
  - DataLogCreate, 2593
  - DataLogDelete, 2606
  - DataLogNewFile, 2608
  - DataLogOpen, 2597
  - DataLogTypedNewFile, 2609
  - DataLogTypedOpen, 2599
  - DataLogWrite, 2602
  - DATE, 1108, 1149, 1175
  - DATE\_AND\_LTIME, 1110
  - DATE\_AND\_TIME, 1109

- DATE\_TO\_, 1175
- Daten lesen
  - aus DP-Normslaves/PROFINET IO-Device mit DPRD\_DAT, 2436
  - aus remoter CPU mit GET, 2918
- Daten schreiben
  - in DP-Normslaves/PROFINET IO-Device mit DPWR\_DAT, 2438
  - in remote CPU mit PUT, 2920
- Daten übertragen
  - Anstoß durch projektierte Funktion, 3533, 3586, 3645, 3685, 3836
  - Bedienung in der Rezepturanzeige, 3583, 3642, 3682, 3833
  - Bereichszeiger, 3487, 3520, 3571, 3631, 3673, 3821
  - Bereichszeiger Bildnummer, 3521, 3577, 3632, 3674, 3823
  - Bereichszeiger Datensatz, 3529, 3582, 3640, 3681, 3831
  - Bereichszeiger Datum/Uhrzeit, 3522, 3572, 3573, 3633, 3675, 3824
  - Bereichszeiger Datum/Uhrzeit Steuerung, 3523, 3634, 3676, 3825
  - Bereichszeiger Koordinierung, 3525, 3575, 3635, 3677, 3826
  - Bereichszeiger Projektkennung, 3526, 3578, 3636, 3678, 3827
  - Bereichszeiger Steuerungsauftrag, 3527, 3579, 3637, 3679, 3828
  - mit Synchronisation, 3530, 3641, 3687, 3832
  - mögliche Fehlerursache, 3534, 3587, 3646, 3686, 3837
  - ohne Synchronisation, 3530, 3583, 3640, 3686, 3831
  - Steuerungsauftrag, 3531, 3584, 3643, 3683, 3834
- Daten wiederherstellen
  - Bediengerät, 3950, 3952, 3966
- Datenaustausch
  - DP-Slave, 769
  - I-Slave - DP-Master, 769
  - Kurven, 3536, 3699
  - über Bereichszeiger, 3448
  - Variablen, 3447
- Datenaustausch über AS-AS-Fernkopplung
  - Kommunikationsanweisung "AS\_DIAL", 4386
- Datenaustausch, 3441
- Datenbaustein
  - aktualisieren, 1413
  - anlegen, 1214, 1411
  - ARRAY deklarieren, 1416
  - ARRAY-Datenbaustein, 1030, 1034, 1067, 1070, 1214, 1408, 1411, 1432
  - Attribute lesen mit ATTR\_DB, 2618
  - aus Ladespeicher lesen mit READ\_DBL, 2614
  - basierend auf einem PLC-Datentyp, 1418
  - Bausteinänderungen ohne Reinitialisierung laden, 1414
  - CPU-Datenbaustein, 1032
  - Datenwerte online beobachten, 1431
  - Datenwerte während der Inbetriebnahme anpassen, 1421, 1432, 1433, 1434, 1435
  - Defaultwert, 1419
  - Deklarationstabelle, 1410
  - Eigenschaften von Variablen, 1424, 1426, 1427
  - Einstellwerte nutzen, 1421, 1432, 1433, 1434, 1435
  - erzeugen mit CREATE\_DB, 2611
  - Globaler Datenbaustein, 1030, 1408
  - Instanz-Datenbaustein, 1032, 1408
  - löschen mit DELETE\_DB, 2619
  - Optimierter Zugriff, 1034, 1036
  - programmieren, 1408, 1415
  - Remanenzverhalten, 1422, 1423
  - schreiben in Ladespeicher mit WRIT\_DBL, 2616
  - Startwert, 1419, 1420
  - STRUCT deklarieren, 1417
  - Variable importieren und exportieren, 1430
  - Werte anzeigen, 1431
- Datenbereiche
  - Bereichszeiger, 3490, 3823
  - lesen, 3490, 3823
  - schreiben, 3490, 3823
- Datenbit (DBX), 842
- Datenbyte (DBB), 842
- Datendoppelwort (DBD), 842
- Datenfach
  - für Rezepturen, 3278
- Datenfluss, 4011
- Datenflusskontrolle, 871
- Datenkonsistenz, 2913
- Datensatz, 3448
  - aus projektierten Systemdaten lesen mit RD\_DPARM, 2513
  - bereitstellen auf I-Device mit PRVREC, 2442
  - Datensätze schreiben und lesen, 2507
  - einer Baugruppe asynchron lesen mit RD\_DPARA, 2511
  - einer Baugruppe lesen mit RD\_DPAR, 2508
  - empfangen auf I-Device mit RCVREC, 2440
  - lesen, 3300
  - lesen mit RD\_REC, 2430
  - lesen mit RDREC, 2392

- schreiben mit WR\_REC, 2435
- übertragen, 3301
- übertragen mit WRREC, 2394
- vordefinierte Parameter schreiben mit WR\_DPARM, 2515
- Datensatzliste, 3282
- Datensicherung, 91
  - Bediengerät, 3950, 3966
- Datentyp
  - explizit konvertieren, 1798, 2067, 2260
  - Allen-Bradley, 3745
  - Allen-Bradley EtherNet/IP, 3711
  - ANY, 1122
  - ARRAY, 1114, 1115
  - Bitmeldung, 3745, 3774, 3819
  - BOOL, 1091, 1136, 1152
  - BYTE, 1091, 1136, 1153
  - CHAR, 1112, 1150, 1178
  - DATE, 1108, 1149, 1175
  - DINT, 1098, 1143, 1168
  - DT, 1109
  - DTL, 1111, 1148, 1177
  - DWORD, 1093, 1138, 1157
  - Explizite Konvertierung, 1152, 1153, 1155, 1157, 1161, 1163, 1164, 1166, 1168, 1170, 1171, 1173, 1174, 1175, 1176, 1177, 1178, 1179
  - Gültigkeit, 1087
  - Implizite Konvertierung, 1136, 1137, 1139, 1140, 1141, 1142, 1143, 1144, 1145, 1146, 1148, 1150, 1151
  - INT, 1096, 1141, 1164
  - interne Variable, 3166, 3486
  - Konvertierung, 1133, 1138, 1147, 1148, 1149
  - LDT, 1110
  - LINT, 1099
  - LREAL, 1103, 1146, 1173
  - LTIME, 1107
  - LTIME\_OF\_DAY, 1109
  - LWORD, 1094
  - Mitsubishi, 3774
  - Mitsubishi FX, 3767
  - Mitsubishi MC TCP/IP, 3757
  - Modicon Modbus RTU, 3797
  - Modicon Modbus TCP, 3784
  - Omron, 3819
  - Omron Host Llink, 3813
  - PLC-Datentyp, 1127, 1436
  - POINTER, 1120
  - REAL, 1102, 1145, 1171
  - S5TIME, 1105
  - SCL-Anweisung, 1383, 1384, 1385
  - SINT, 1095, 1139, 1161
  - STRING, 1112, 1151, 1179
  - STRUCT, 1119
  - TIME, 1107, 1147, 1174
  - TIME\_OF\_DAY, 1108
  - TOD, 1148, 1176
  - UDINT, 1099, 1144, 1170
  - UINT, 1097, 1142, 1166
  - ULINT, 1101
  - USINT, 1095, 1140, 1163
  - VARIANT, 1125
  - WORD, 1092, 1137, 1155
  - zulässig, 3711, 3757, 3767, 3784, 3797, 3813
- Datentyp DTL
  - Einschränkung, 63
- Datentypen, 156
  - Kurven, 3536, 3701
  - Migration, 138
  - S7 1200 V1, 3592
  - S7 1500, 3540
  - S7 200, 3692
  - S7 300/400, 3650
  - SIMATIC LOGO, 3704
  - zulässig, 3540, 3592, 3650, 3692, 3704
- Datenwort (DBW), 842
- Datum, 1108, 1109, 1110, 1111
- Datum/Uhrzeit, 3448
  - Bereichszeiger, 3572
- Datum/Uhrzeit PLC, 3448
- Datum/Uhrzeit-Feld, 3130
  - Format, 3131
  - Systemzeit anzeigen, 3131
  - Variable verwenden, 3131
- Datum-Uhrzeit-Feld
  - Bedienung, 3981
  - Darstellung, 3981
  - Verhalten, 3981
  - Verwendung, 3981
- DB-Variable
  - Definition, 1057
- DCAT, 1865, 2136, 2330
- DCP (Primary Setup Tool), 539
- DCP Server, 683
- Dead-Peer-Detection (DPD), 574
- deaktivieren, 3424
  - Projektsprache, 3886
- Deaktivieren von Projekten, 251
- DEC, 1736, 2001
- Deckblatt
  - Platzhalter für Dokument-Informationen, 290
- DECO, 1843, 2114, 2310
- Decodieren, 1843, 2114, 2310
- DecreaseFocusedValue, 3409

- DecreaseTag, 3410
- defekte Lizenz, 80, 92
- Definieren
  - Filter für den Belegungsplan, 1499
- Deinstallation starten, 102
- Deinstallation von License Keys, 79
- Dekrementieren, 1736, 2001
- DELETE, 2382
- DELETE\_DB, 2619
- Demultiplexen, 1850, 2121, 2316
- DEMUX, 1850, 2121, 2316
- DES, 568, 616, 626
- Deskalieren, 1812, 2082, 2278
- DETACH, 2518
- Detailansicht, 198
- Detailvergleich, 270
- Device Tool, 822, 824
- DeviceStates, 2570
- Dezentrale Peripherie, 767, 926
- Dezentrales Peripheriegerät ET 200iSP
  - Definition, 912
- Dezentrales Peripheriegerät ET 200M, 951
- Dezimalstellen, 32
- DHCP
  - Client, 691
  - Server-Konfiguration, 596
- DHCP-Server, 597
- Diagnose, 577, 928
  - DP-Slave, Diagnosedaten lesen mit DPNRM\_DG, 2450
  - Gerät, 3455
  - Geräte-Information, 3455
  - Verbindung, 3455
  - Verbindungsinformation, 3455
- Diagnosealarm-OB, 896
- Diagnose-Benutzer, 524
- Diagnosedaten
  - auslesen mit GET\_DIAG, 2578
- Diagnoseinformation generieren, 2576
- Diagnosepuffer
  - Grundlagen, 846, 1009
  - Organisation, 1009
- Diagnosepuffer auslesen, 987
- Diagnose-Repeater, 1015
- Diagnosestatus
  - online ermitteln und anzeigen, 966
- Diagnosestatus ermitteln, 986
- Diagnostic error interrupt, 896
- Diagonale Bewegung
  - Animation, 3102
- Dienstgruppen, 536, 539
- Diffie-Hellman-Schlüsselvereinbarung, 567
- DINT, 1098, 1143
- DINT , 1168
- DINT\_TO\_, 1168
- Direkter Zugriff auf die Peripherie, 847
- Direkttaste
  - projektieren, 3118
- DIS\_AIRT, 2546
- DIS\_IRT, 2543
- DIV, 1731, 1996
- Dividieren, 1731, 1996
- Divisionsrest, 1732, 1998
- DMSK\_FLT, 2541
- Dokumentation individuell erstellen, 229
- Dokumentationseditor
  - Aufbau, 285
- Dokumentationsfunktion
  - Fertige Rahmen und Deckblätter verwenden, 283
- Dokument-Information, 280
- doppelklicken, 3425
- Doppelwort, 1093
- Download, (Siehe Laden in Gerät)
- DP, 3569, 3622
- DP/DP-Koppler, 768
- DP\_TOPOL, 2454
- DP-Master, 628, 913
  - Anzeige im DP-Slave, 773
  - DP-Mastersystem hinzufügen, 775
  - DP-Schnittstelle, 772
  - Geräte und Baugruppen, 772
  - Trennung vom DP-Mastersystem, 774
- DP-Mastersystem
  - anlegen, 767, 771
  - DP-Slave einfügen, 776
  - DP-Slave trennen, 777
  - Eigenschaften bearbeiten, 775
  - erstellen, 770
  - hervorheben, 773
  - Topologie ermitteln mit DP\_TOPOL, 2454
  - Trennung der Teilnehmer, 774
  - Trennung vom Subnetz, 774
- DP-Normslave, 791
- DP-Normslaves
  - alle Ausgänge schreiben mit SETIO, 2396
  - alle Eingänge lesen mit GETIO, 2395
  - einen Teil der Ausgänge schreiben mit SETIO\_PART, 2398
  - einen Teil der Eingänge lesen mit GETIO\_PART, 2397
  - konsistente Daten lesen mit DPRD\_DAT, 2436
  - konsistente Daten schreiben mit DPWR\_DAT, 2438
- DPNRM\_DG, 2450



- DPRD\_DAT, 2436  
 DP-Schnittstelle, 772  
 DP-Slave, 628
  - am DP-Mastersystem einfügen, 776
  - Anzeige des DP-Masters, 773
  - Datenaustausch, 769
  - deaktivieren und aktivieren mit D\_ACT\_DP, 2425
  - Diagnosedaten lesen mit DPNRM\_DG, 2450
  - DP-Mastersystem zuweisen, 777
  - Gruppen synchronisieren mit DP\_SYC\_FR, 2445
  - Hardware-Katalog, 768
  - intelligenter, (Siehe I-Slave)
  - konfigurieren, 769, 777
  - mit Vorverarbeitung, (Siehe I-Slave)
  - Typen, 775
  - vernetzen, 770
  - vom DP-Mastersystem trennen, 777
 DPSYC\_FR, 2445  
 DPV1
  - ET 200S konfigurieren, 788
 DPWR\_DAT, 2438  
 Drahtbruch, 929, 945  
 drehen
  - Objekt, 3043, 3055
 Druck abbrechen, 295  
 Druckbare Inhalte, 277  
 Drucken
  - Aufbau des Ausdrucks, 276
  - Deckblatt anlegen, 283
  - Deckblatt bearbeiten, 284
  - Deckblatt nutzen, 279
  - Dokumentationsfunktion, 276
  - Druckfläche festlegen, 288
  - Druckinhalte, 280
  - Druckvorlagen verwenden, 279
  - Einstellungen ändern, 278
  - Elemente in der Bibliothek, 281
  - Elemente in der Projektnavigation, 281
  - Gerätesicht drucken, 377
  - Layout festlegen, 279
  - Netzansicht drucken, 377
  - Nicht druckbare Elemente, 277
  - Nicht druckbare Objekte, 278
  - Rahmen anlegen, 282
  - Rahmen bearbeiten, 284
  - Rahmen nutzen, 281
  - Rahmen und Deckblätter verwenden, 276
 Drücken, 3425  
 Druckumfang, 277  
 Druckvorschau erstellen, 290  
 DRUM, 1861, 2133, 2327  
 DSCP, 708  
 DT, 1109  
 DTL, 1111, 1148, 1177  
 DTL\_TO\_, 1177  
 Durchgängige Symbolik, 155  
 DWORD, 1093, 1138, 1157  
 DWORD\_TO\_, 1157  
 Dynamische Referenztemperatur, 948  
 dynamisieren
  - Bewegung eines Objekts, 3100
  - blinken, 3099
  - Direkte Bewegung, 3102
  - Farbe eines Objekts, 3099
  - Gestaltung eines Objekts, 3099
  - grüner Pfeil in Übersicht, 3098
  - Objekt, 3096**E**  
 E/A-Feld, 3131
  - Darstellungsformat, 57
  - Datenformat, 3132
  - Format Dezimal, 57
  - Modus, 3132
  - Verdeckte Eingabe, 3133
 E-Adresse, 628  
 EA-Feld, 3982
  - Bedienung, 3983
  - Darstellung, 3982
  - EA-Feld, 3984
  - Touch-Bedienung, 3983
  - Verhalten, 3983
  - Verwendung, 3982
  - Wechsel zwischen Eingabefeldern, 3983
 Ebene
  - Objekt einer Ebene zuordnen, 3122
 EC31-RTX, 40  
 EditAlarm, 3359  
 Editiersprache, 240, 3882
  - wählen, 3887
 Editor
  - Geräte und Netze, 3475
  - Grafiken, 3896
  - Verbindung, 3477
  - Verbindungen, 3650, 3652, 3654
 Eigenschaften (CPU), 856  
 Eigenschaften der VPN-Gruppe, 566  
 Ein- und Ausrücken
  - Zeilen, 1376
 Eindeutigkeit
  - von Objektnamen, 115
 Einerkomplement, 1841, 2112  
 einfache Benutzeranzeige, 3994

- Bedienung, 3995
- Darstellung, 3994
- Verwendung, 3994
- einfache Meldeanzeige, 4003
  - Bedienelement, 3240, 4004
  - Bedienung, 3240, 4003
  - Verwendung, 3239, 4003
- einfache Rezepturanzeige, 3282
  - Bedienung, 3296, 4012
  - Darstellung, 3296, 4012
  - Menübefehl, 3297, 4012, 4013
  - Verhalten, 3297, 4013
- einfaches Meldefenster
  - Verwendung, 3239, 4003
- einfügen
  - Bibliotheksobjekt, 3849
  - externe Grafik , 3898
  - Grafik, 3044
  - Grafik in Grafiksammlung, 3897
  - Objekt, 3043, 3044, 3058
  - Rechteck, 3076
- Einfügen
  - Bildgröße anpassen, 3922
  - Farbe, 3922
  - Funktionstaste, 3922
  - Prinzip, 3920
  - Schriftart, 3922
  - ungültiges Objekt, 3921
- Eingabe abgeschlossen, 3426
- Eingang
  - einfügen, 1309, 1889
  - entfernen, 1311, 1353
- Eingang (E), 842
- Eingang einfügen, 1351
- Eingangsbyte (EB), 842
- Eingangswort (EW), 842
- Einsatz, 3274
  - Systemfunktion, 3341
  - von Rezepturen, 3274
- Einschaltverzögerung, 1650, 1659, 1674, 1685, 1911, 1936, 1947, 2161, 2179
  - speichernd, 1677, 1686, 1939, 1949, 2181
- Einschränkung
  - Modicon Modbus RTU, 3801
  - Modicon Modbus TCP/IP, 3801
- Einsetzen eines TS Adapters für TeleService, 4345, 4370
- einstellen
  - Sprache, 3979
- Einstellen
  - Ansichtsoptionen für den Belegungsplan, 1498
  - Ansichtsoptionen für die Abhängigkeitsstruktur, 1512
  - Totzone, 3221
  - Verzögerungszeit, 3221
- Einstellung
  - Sprachen im Betriebssystem, 3883
- Einstellungen
  - allgemein, 1256
  - Allgemein, 171, 173, 174
  - ändern, 175, 1257, 1282, 1326, 1373
  - FUP, 1282, 1326
  - KOP, 1282, 1326
  - SCL, 1372
- Einzelinstanz
  - Aufruftyp ändern, 1301, 1343
  - Beispiel, 1041
  - Definition, 1041
- Elementliste, 3282
- Ellipse
  - Radius horizontal, 3134
  - Radius vertikal, 3134
- Ellipse , 3133
- ELSE, 2282
- E-Mail
  - E-Mail übertragen mit TM\_MAIL, 3016
  - E-Mail übertragen mit TMAIL\_C, 2956
- Empfangen von Daten
  - auf I-Device mit RCVREC, 2440
- Empty-Zeile einfügen, 4178
- Empty-Zeile hinzufügen, 4178
- EN-/ENO-Mechanismus
  - AWL-Beispiel, 1185
  - FUP-Beispiel, 1183
  - Grundlagen, 1180
  - KOP-Beispiel, 1183
  - SCL-Beispiel, 1184
- EN\_AIRT, 2546
- EN\_IRT, 2544
- ENCO, 1844, 2115, 2311
- Encodieren, 1844, 2115, 2311
- Ende der Erfassung erreichbarer Teilnehmer, 816, 3510, 3561, 3614
- Ende der Sync-Domain, 816, 3510, 3561, 3614
- Ende der Topologieerkennung, 816, 3510, 3561, 3614
- ENDIS\_PW, 1823, 2094, 2295
- Engineering System
  - Leistungsmerkmale, 4019
- ENO, 38, 161
- ENO aktivieren, 38
- ENO automatisch setzen, 1372

- entfernen
  - Formatierung im Meldetext, 3223
  - Objekt aus der Gruppe, 3069
  - Ordnerverknüpfung, 3063
- entsperren
  - Benutzer, 3325
- Enumerationen
  - Webserver, 651
- Enumerationstypen, 650
- Ereignis
  - abgebaut, 3420
  - aktivieren, 3421
  - Änderung, 3421
  - Ausführen, 3422
  - Auswahl geändert, 3422
  - Bei Überschreitung, 3422
  - Bei Unterschreitung, 3423
  - Beim Öffnen eines Dialogs, 3423
  - Beim Schließen eines Dialogs, 3423
  - Benutzerwechsel, 3424
  - Bildwechsel, 3424
  - deaktivieren, 3424
  - doppelklicken, 3425
  - Drücken, 3425
  - Eingabe abgeschlossen, 3426
  - ESC doppelt drücken, 3426
  - für Funktionslisten, 3187
  - Gegangen, 3426
  - Gekommen, 3426
  - Inspektorfenster, 55
  - Klicken, 3427
  - Klicken bei Blinken, 3427
  - Loop-In-Alarm, 3427
  - loslassen, 3428
  - Meldungspufferüberlauf, 3428
  - Organisationsbaustein (OB) zuordnen mit ATTACH, 2517
  - Quittieren, 3428
  - Rand erreicht, 3429
  - Runtime-Stopp, 3429
  - Taste drücken, 3430
  - Taste loslassen, 3430
  - Überlauf, 3430
  - umschalten AUS, 3431
  - Umschalten EIN, 3431
  - Variablen, 3187
  - wenig freier Speicherplatz, 3431
  - Wenig freier Speicherplatz, kritisch, 3431
  - Wertänderung, 3432
  - Zeit abgelaufen, 3432
  - Zuordnung zu Organisationsbaustein (OB) aufheben mit DETACH, 2518
- Ereignisgesteuerte Aufgaben
  - projektieren:Ereignisgesteuerte Aufgaben, 3219
- Ereignisname, 903
- Ereignisse
  - Konfiguration, 688
  - Severity-Filter, 690
- Ereignis-Trigger, 3437, 3440
- Erfassungsart
  - Variable, 3182
- Erfassungszyklus
  - Bereichszeiger, 3489
  - Variable, 3182, 3188, 3200
- ErhöheFokussiertenWert, 3362
- ErhöheVariable, 3362
- Erreichbare Teilnehmer, 816, 3510, 3561, 3614
- Ersatzwert
  - Reservemodul, 785
- Erscheinungsbild
  - eines Objekts dynamisieren, 3099
- erstellen
  - Gruppe, 3066
  - Rezepturdatensatz, 4014
  - Rezepturdatensatz am Bediengerät, 3299, 3300
- Erstellen
  - Hilfetext, 3219, 3221
- Erstellen einer Beobachtungstabelle, 1537
- Erstinbetriebnahme
  - Bediengerät, 3960
- Erstmaliges Öffnen, 29
- Erweiterte Statusinformation, 1372
- Erweiterter Modus, 1534
  - Globale Firewall-Regeln, 531
- Erweitertes Laden, 67
- ESC doppelt drücken, 3426
- ESP-Protokoll, 568
- ET 200eco, 911
- ET 200eco PN, 911, 926
- ET 200iSP, 910
- ET 200L, 910
- ET 200M, 910, 951
  - Definition, 951
- ET 200MP, 943
- ET 200pro, 911
- ET 200R, 911
- ET 200S, 910
  - DPV1-Modus, 788
  - Optionenhandling, 784, 787
  - Positioniermodul, 40
  - Steckplatzregeln, 779
  - Vergleichsstellen, 780
- ET 200S COMPACT, 910
- ET 200SP, 932

- Einsatzgebiet, 932
- Ethernet
  - Parameter, 3543, 3598, 3658, 3668, 3697
- ETHERNET
  - Basic Panel, 3495
- Ethernet-Baugruppe
  - Ziehen/Stecken, 38
- Ethernet-Non-IP-Telegramme, 530
- Ethernet-Schnittstelle
  - Parameter anzeigen, 4284
- Ethernet-Schnittstelle parametrieren, 4285
  - Änderbare Parameter, 4283
  - Hinzufügen einer temporären IP-Adresse, 4285
  - Mit Subnetz verbinden, 4284
  - Möglichkeiten zur Parametrierung, 4282
  - Temporäre IP-Adressen löschen, 4286
- EXIT, 2291
- EXKLUSIV ODER, 1887, 1888
  - Bitverknüpfung, 1887
- EXP, 1748, 2015, 2211
- Exponentialwert, 1748, 2015
- Exponentialwert bilden, 2211
- Export
  - Excel-Format, 3853
  - Meldung, 3858
  - Projekttexte, 3875, 3893
  - Rezeptur, 3854
  - Textliste, 3871
  - Variable, 3865
- Export von Beschriftungsstreifen, 296
- EXPT, 1758, 2026
- externe Grafik
  - Ordner bearbeiten, 3062
  - Ordner umbenennen, 3062
  - Ordner verknüpfen, 3062
  - Ordnerverknüpfung entfernen, 3062
- externe Grafikdatei
  - in Grafiksammlung einfügen, 3898
  - in Grafiksammlung hinterlegen, 3063
  - verwalten, 3044
- Externe Quelldatei
  - Baustein exportieren, 1448
  - Dateitypen mit Editor verknüpfen, 1450
  - einbinden, 1449
  - Grundlagen, 1446
  - Regeln zur Programmierung, 1447
- externe Variablen
  - Datenaustausch, 3447
- F**
- F\_TRIG, 1646, 1907, 2157
- Facility, 552
- FAQs, 384, 3466
- Farbe
  - Diagnose von Ethernet-Leitungen, 972
  - Diagnose von Ports, 972
- Farbverlauf
  - Balken, 3128
- Favoriten
  - ausblenden, 1245
  - einblenden, 1245
  - entfernen, 1298, 1340, 1387
  - hinzufügen, 1297, 1339, 1385
  - verwenden, 1297, 1339, 1386
- FB, 1029
- FC, 1028
- Fehlende Versorgungsspannung, 945
- Fehlende Versorgungsspannung, 938
- Fehlerauswertung mit RET\_VAL, 1622
- Fehlerbehandlung, 1407
  - GetError, 1828, 2099, 2301
  - GetErrorID, 1832, 2103, 2304
  - Grundlagen, 1403
  - Lokale Fehlerbehandlung, 1405
- Fehlerinformation, 1405, 1622, 1828, 1832, 2099, 2103, 2301, 2304
- Fehlermeldung
  - Laden, 3955
  - Laden von Daten auf die PLC, 67
- Fehlerüberwachung
  - Redundanz, 703
  - Spannungsversorgung, 701
  - Verbindungszustandsänderung, 702
- Feld, 1114, 1115
  - siehe ARRAY, 1265, 1416
- Feld lesen, 1765, 2033
- Feld schreiben, 1767, 2035
- Feldgerät, 791, 819
- Fensterlayout laden
  - Laden über Schnellzugriff, 204
  - Weitere Fensterlayouts laden, 204
- Fensterlayout speichern, 203
- Fensterlayout wiederherstellen, 205
- Fensterlayouts
  - Fensterlayouts löschen, 205
  - Reihenfolge ändern, 205
- Fernverbindung
  - als CPU-gesteuerte Fernverbindung, 4383
  - als VPN-Verbindung, 4370
  - als Wählverbindung, 4345
- Fernverbindung vom TS Adapter wird nicht aufgebaut, 4394

- Fernverbindung zum TS Adapter wird nicht aufgebaut, 4392
- Fernverbindungsaufbau nicht möglich, 4369
- Feste Referenztemperatur, 948
- festlegen
  - Referenzobjekt, 3051
- FETCH/WRITE, 884
- FieldRead, 1765, 2033
- FieldWrite, 1767, 2035
- File Browser, 638
- FILL, 1796, 2064, 2257
- FILL\_BLK, 1777, 2045, 2228
- Filter
  - auswählen, 1500
  - für den Belegungsplan definieren, 1499
  - Hardware-Katalog, 391
  - im Belegungsplan, 1498
  - löschen, 1499
- Filtern
  - Belegungsplan, 1500
- FIND, 2384
- Firewall
  - Dienstgruppen anlegen, 536, 539
  - Dienstgruppen verwalten, 537, 540
  - Firewall-Regeln, 530
  - ICMP-Dienste definieren, 535
- Firewall aktivieren
  - SCALANCE S, 591, 593
- Firmware, 908
  - aktualisieren, 3953
- Firmware-Update, 995
- Firmware-Update Memory Card, 840
- Firmwareversion V4, 29
- Flanke
  - negativ, 1638, 1641, 1644, 1900, 1902, 1905, 2157
  - positiv, 1637, 1640, 1642, 1899, 1901, 1904, 2155
- Flutterüberwachung, 955
- Flipflop
  - rücksetzen/setzen, 1636, 1897
  - setzen/rücksetzen, 1634, 1896
- FLOOR, 1802, 2072, 2264
- FOR, 2284
- Force-Auftrag
  - auf SD-Karte, 31
- Forcen
  - Alle forcen, 1581
  - Forcen beenden, 1583, 1585
- Forcen von Variablen, 46
  - Sicherheitsvorkehrungen, 1562, 1578
- Forcen von Variablen bei direktem Peripheriezugriff, 46
- Forcetabelle
  - Bedeutung der Symbole, 1564
  - Anzeige, 1565
  - Aufbau, 1562
  - Basismodus, 1563
  - Bedeutung der Spalten, 1562
  - Beobachtungs- und Steuermodus, 1548
  - Erweiterter Modus, 1563
  - Funktionalität, 1560
  - öffnen, 1565
  - speichern, 1566
  - Syntaxprüfung, 1566
  - Testmöglichkeiten, 1560
  - Übersicht der Testmöglichkeiten, 1560
  - Übersicht über die Anzeigeformate, 1569
  - Umschalten zwischen Basismodus und Erweiterter Modus, 1563
  - Variablen sofort beobachten, 1575
  - zugelassene Operanden, 1567
  - zugelassene Operanden für Forcewerte, 1568
- Forcewert
  - zugelassene Operanden, 1568
- Format, 3131
- Formatieren
  - Meldetext, 3223
- Forward Delay, 724
- FRAC, 1757, 2025, 2220
- Fragment, 652, 658
- Freier Kommentar
  - bearbeiten, 1304, 1346
  - einfügen, 1304, 1346
  - Einführung, 1303, 1345
  - löschen, 1305, 1347
- Freigabeausgang ENO, 161
- Fremdtreiber
  - Besonderheiten, 3705
  - Kommunikation, 3704
- Frequenzmesser, 923, 926
- Frequenzmesser Funktionsweise, 924
- FTP, 526, 527, 884, 2782
- FTP\_CMD, 2782
- FTPS, 2782
- FTPS-Zertifikate, 518
- Funktion
  - einer Funktionstaste zuweisen, 3113
- Funktion (FC)
  - anlegen, 1213
  - Definition, 1028
  - in externe Quelldatei exportieren, 1448
- Funktionalität der Forcetabelle, 1560

Funktionen

Variablenwert aktualisieren, 3163, 3483

Funktionsbaustein (FB)

anlegen, 1213

Definition, 1029

in externe Quelldatei exportieren, 1448

Instanz-Datenbaustein, 1029

Funktionsliste, 3187, 3341, 3436

Abarbeitung, 3343

Abarbeitung in Runtime, 3346

asynchrone Abarbeitung, 3346

Benutzerdefinierte Funktionen, 3343

synchrone Abarbeitung, 3346

Systemfunktion, 3343

Funktionslisten

bearbeiten, 3345

Funktionsplan, 1323

Funktionstaste, 3108

Funktion zuweisen, 3113

global belegen, 3111

Globale Zuordnung verwenden, 3036

Globales Bild, 3035

Grafik zuweisen, 3116

lokal belegen, 3036, 3112

mit Kennwort schützen, 3115

zur Bildnavigation verwenden, 3120

Funktionsumfang

ProSave, 3950

FUP, 1323, 1597, 1598, 1599

FUP-Element

aus Zwischenablage einfügen, 1350

ausschneiden, 1349

einfügen, 1335

ersetzen, 1351

kopieren, 1348

löschen, 1355

Operand einfügen, 1355

Regeln beim Einfügen, 1333

selektieren, 1348

FUP-Netzwerk

Bausteinaufruf einfügen, 1299, 1341

Programmstatusanzeige, 1531

Regeln zu Zweigen, 1359

Zweig, 1358

Zweig einfügen, 1359

Zweig löschen, 1360

FX1 series, 3768

FX2 series, 3768

FX3

Steuerung, 3754

FX3 series, 3758

G

GADR\_LGC, 2633

Ganzzahl, 1804, 2073, 2266

16 Bit, 1096, 1097

32 Bit, 1098, 1099

64 Bit, 1099, 1101

8 Bit, 1095

Gebertyp Wechsler, 954

Gegangen, 3426

Gehen, 3207

GeheZumAnfang, 3363

GeheZumEnde, 3364

Gekommen, 3426

GEN\_DIAG, 2576

GEO\_LOG, 2628

GEO2LOG, 2622

geografische Koordinaten, 686

Gerät

einer Hardware-Konfiguration hinzufügen, 399

einfügen, 3443

gestörte Geräte, 965

kopieren, 49, 405

löschen, 404

umbenennen, 507

verschieben, 407

Gerät tauschen

Beschränkung bei Verbindungen, 3910

Prinzip, 3910

Gerät wiederherstellen, 4277

Geräte

verbinden, 3447

vernetzen, 3446, 3471, 3472

Geräte & Netze, 3446, 3471

Geräte simulieren, 365

Geräte und Netze, 3446, 3471

HMI-Verbindungen, 3473

Verbindungen, 3473

Geräte vernetzen

Grundlagen zum Projektieren von Netzen, 413

Kommunikationsarten, 413

Netze innerhalb eines Projekts, 413

Netzparameter bearbeiten, 419

Schnittstellenparameter bearbeiten, 419

Vernetzen bei nicht vorhandenem Subnetz, 415

Vernetzen bei vorhandenem Subnetz, 416

Vernetzen mehrerer Schnittstellen gleichzeitig, 415

Voraussetzungen, 419

Geräteabhängigkeit

Bild, 3027

- S7 1200 V1, 3591
- S7 1200 V2, 3591
- S7 1200 V3, 3591
- S7 1200 V4, 3591
- SIMATIC S7 1500 V1.0, 3540
- Geräte-Assistent, 3838
- Geräte-Information, 965
  - Diagnose, 3455
- Gerätekonfiguration online ändern, 4268
- Gerätename, 796, 799, 802
- Gerätenummer, 798
- Gerätesicht
  - Ablage nicht gesteckter Baugruppen, 396
  - Baugruppe einfügen, 402
  - Baugruppenträger, 392
  - Eigenschaften bearbeiten, 409
  - Hardware- und Netzwerkeditor, 372, 3457
  - Parameter bearbeiten, 409
  - Signalboard einfügen, 632
- Gerätetausch ohne Wechselmedium, 817
- Geräteübersicht
  - Adressbereich, 628
- Geräteversion, 3940
  - aktualisieren, 3953
- Gestörte Geräte, 965
- GET, 2918
- Get\_AlarmState, 2550
- GET\_DIAG, 44, 2578
- GET\_NAME, 2565
- GetError, 1406, 1407, 1828, 2099, 2301
- GetError , 1405
- GetErrorID, 1405, 1406, 1407, 1832, 2103, 2304
- GetGroupNumber, 3375
- GETIO, 2395
- GETIO\_PART, 2397
- GetPassword, 3375
- GetStationInfo, 2567
- Getunnelte Kommunikation aktivieren
  - CP x43-1 Adv., 606
  - SCALANCE S, 591, 593
- GetUserName, 3374
- Gigabit-Adresse, 521
- Glättung, 929, 946
- Gleitpunktzahl, 1102, 1103
  - Gültigkeit prüfen, 1721, 1986
  - Ungültigkeit prüfen, 1722, 1987
- Gleitpunktzahlen, 1104
  - ungültige, 1104
- Globale Belegung
  - einer Funktionstaste, 3108, 3111
- Globale Bibliothek, 3839
  - anlegen, 3849
  - archivieren, 332
  - aufräumen, 363
  - Eigenschaften anzeigen, 327
  - Elementeansicht verwenden, 317
  - erstellen, 324
  - Filteransicht verwenden, 336
  - löschen, 330
  - migrieren, 130
  - öffnen, 325, 3851
  - Ordner erstellen, 333
  - Projekt aktualisieren, 357
  - Protokolle anzeigen, 328
  - schließen, 329
  - speichern, 328, 3850
  - Typen hinzufügen, 355
  - Typen verwenden, 356
- Globale Bibliothek hochrüsten, 326
- Globale Bibliotheken, 314
  - archivieren, 331
  - dearchivieren, 333
- Globale Firewall-Regeln, 531
  - zuweisen, 533
- Globale Paketfilter-Regeln, 533
- Globale Variable, (Siehe PLC-Variable)
- Globale Zuordnung verwenden
  - Funktionstaste, 3036
- globalen Datenbaustein
  - anlegen, 3491
  - projektieren, 3491
- Globaler Datenbaustein, 1030
  - Remanenzverhalten, 1423
- Globaler Softkey, 3111
- Globales Bild, 3035
  - Funktionstaste, 3035
  - Vorlage, 3036
- GMRP, 739
- GOTO, 2292
- GoToEnd, 3364
- GoToHome, 3363
- GPRS, 666
- Grafik
  - anpassen, 3135, 3137
  - aus der Grafiksammlung verwenden, 3061
  - Editor, 3896
  - einer Funktionstaste zuweisen, 3116
  - einfügen, 3044
  - einfügen in Grafiksammlung, 3897
  - Grafikanzeige, 3135
  - mit transparentem Hintergrund, 3061
  - Schaltfläche, 3152
  - verwalten, 3061
- Grafikanzeige, 3134, 3984

- Bedienung, 3985
- Darstellung, 3985
- transparente Farbe, 3135
- Verwendung, 3984
- Grafikdatei
  - in Grafiksammlung hinterlegen, 3063
  - in Grafiksammlung hinterlegen , 3898
- Grafikliste
  - Bereich (... - ...), 3091
  - Bereich (0 - 31), 3088
  - Bit (0, 1), 3092
  - Bitnummer (0 - 31), 3094
  - erstellen, 3087
  - Grafisches E/A-Feld, 3095
  - Projektierungsdaten ausgeben, 3095
  - Verwendung, 3087
- Grafiksammlung, 3061, 3063
- Grafiksammlung , 3898
- Grafisches E/A-Feld, 3136
  - Grafikliste ausgeben, 3095
- grafisches EA-Feld, 3985
  - Bedienung, 3986
  - Verhalten, 3986
  - Verwendung, 3985
- Grenzbereich
  - Variable, 3178
- Grenzfrequenzen Impulsausgänge, 4130
- Grenzwert, 938, 1742, 2008, 2206, 3214
  - Variable, 3179
- Grenzwertprüfung, 3976, 3978
- Grenzwertüberschreitung
  - Analogmeldung, 3179
- Größe
  - mehrerer Objekte ändern, 3060
- Grundlagen
  - Migration, 115
- Gruppe
  - aufheben, 3068
  - aflösen, 3068
  - Bearbeiten, 3066
  - erstellen, 3066
  - Objekt entfernen, 3069
  - Objekte hinzufügen, 3069
- Gruppeneigenschaften, 566
- Gruppenamen, 534, 538
- gruppieren
  - Objekt, 3066
- GSD-Dateien
  - Geräte konfigurieren (PROFIBUS), 791
  - GSD-Revisionen (PROFIBUS), 789
  - Installieren, 790
- GSD-Dateien (PROFINET), 818

- Ausgabestand ändern, 820
- Installieren, 819
- GSDML, (Siehe GSD-Dateien (PROFINET))
- GSM-Netz, 666

## H

- Handbücher, 384, 3466
- Handbücher zusammenstellen, 229
- Handshaking, 872
- Hardware
  - Eigenschaften bearbeiten, 409
  - Erkennung, 41
  - Konfigurieren und Parametrieren, 388
  - Parameter bearbeiten, 409
- Hardware interrupt, 894
- Hardware simulieren, 365
- Hardware Support Package
  - installieren, 97
- Hardware- und Netzwerkeeditor
  - Gerätesicht, 372, 3457
  - Netzsicht, 369, 3451
  - Topologiesicht, 375, 3460
- Hardware- und Software-Endschalter: Funktion, 4135
- Hardwareaufbau für Motion Control S7-1200, 4128
- Hardwaredatentypen, 1130
- Hardware-Diagnose, 964
- Hardware-Dokumentation, 4399
- Hardwareeditor
  - Funktion, 367, 3449
  - Hardware-Katalog, 382, 3464
  - Inspektorfenster, 380, 3462
  - Komponenten, 367, 3449
- Hardware-Erkennung, 401
- Hardwaregesteuerte Datenflusskontrolle, 871
- Hardware-Katalog
  - Auswahl der HW-Komponente, 397
  - DP/DP-Koppler, 768
  - DP-Slave, 768
  - durchsuchen, 391
  - Gerät hinzufügen, 400
  - I-Slave, 768
  - Task Card, 382, 3464
- Hardware-Kennung, 628
  - ermitteln mit LOG2MOD, 2625
- Hardware-Konfiguration
  - Baugruppe hinzufügen, 402
  - Gerät hinzufügen, 399
- Hardwarevoraussetzungen, 81
- Hardware-Voraussetzungen
  - Kommunikationsanweisung "AS\_MAIL", 4389



- Kommunikationsanweisung "PG\_DIAL", 4384
  - Kommunikationsanweisung "SMS\_SEND", 4388
  - Hardware-Voraussetzungen für AS-AS-  
Fernkopplung
    - Kommunikationsanweisung "AS\_DIAL", 4386
  - HART, 914
  - HART-Variablen
    - Aufbau, 953
    - projektieren, 952
    - Quality Code, 953
  - Herunterfahren von Windows XP, 31
  - Hervorheben
    - Verbindung, 3475
  - Hilfe
    - durchsuchen, 224
    - Hilfethemen drucken, 226
    - Index verwenden, 224
    - Kennzeichnung von Hilfethemen, 221
    - öffnen, 223
    - Stichwörter suchen, 224
    - Themen aus Favoriten aufrufen, 226
    - Themen aus Favoriten löschen, 226
    - Themen in Favoriten speichern, 225
    - Tooltip-Kaskaden automatisch aufklappen, 227
    - Volltextsuche, 224
  - Hilfeindikator, 3137
  - Hilfetext, 3214
    - erstellen, 3219, 3221
  - Hintergrund gestalten, 3058
  - Hintergrundfarbe
    - dynamisch ändern, 3099
  - hinterlegen
    - externe Grafik, 3063
  - Hinweise zum Parametrieren eines entfernten  
Modems, 4349
  - hinzufügen
    - Objekte zur Gruppe, 3069
  - HMI Backup
    - anzeigen, 3934
    - löschen, 3935
    - umbenennen, 3935
  - HMI-Variablen verknüpfen, 153
  - HMI-Verbindung, 151, 446, 884, 3447
    - anlegen, 3499, 3548, 3603
    - Kennwort, 3518, 3570
    - MPI, 3624
    - MPI-Parameter, 3626
    - Passwort, 3506, 3557
    - PROFIBUS, 3511, 3616
    - PROFIBUS-Parameter, 3514, 3564, 3617
    - PROFINET, 3498, 3499, 3547, 3548, 3602, 3603
    - PROFINET-Parameter, 3501, 3550, 3605
    - S7 1200, 3548
    - S7 1500, 3499
    - S7 300/400, 3603
  - HMI-Verbindungen
    - Geräte & Netze, 3473
  - Hochrüsten
    - Projekt, 3931
    - WinCC-Version eines Projekts, 3931
  - Hotkey festlegen, 3152
  - HSC, 863
  - HSP, (Siehe Support Package)
  - HTA, 2376
  - HTTP, 535
  - HTTP Protocol
    - Basic Panel, 3494
  - HTTPS, 592, 637
  - HW-ID
    - siehe HW-Kennung, 628
  - HW-Kennung, 2801, 2804, 2807, 2811, 2817, 2818,  
2819, 2820
- I**
- ICMP, 541
  - I-Device, 805
  - IE/AS-i Link PN IO, 793
  - IEC-Prüfung, 1133
    - einstellen, 1135
  - IEC-Zähler, 159
  - IEC-Zeit, 1107
  - IEC-Zeiten, 159
  - IEEE 802.3, 530
  - IEEE-Variable, 913
  - IF, 2280
  - IF1B
    - Basic Panel, 3495
  - IGMP, 739
  - IKE-Einstellungen, 566
  - Image, 3931, 3939, 3940, 3953
    - Bediengerät, 3949
  - IMC, 1872, 2143, 2337
  - Import
    - Meldung, 3859
    - Projekttexte, 3877, 3895
    - Rezeptur, 3856
    - Struktur der Analogmeldung, 3860
    - Struktur der Bitmeldung, 3863
    - Struktur der Rezepturdaten, 3857
    - Textliste, 3872
    - Variable, 3867
  - Impuls, 1648, 1658, 1668, 1682, 1909, 1930, 1944,  
2158, 2173

- verlängert, 1671, 1683, 1933, 1945, 2176
- Impuls- und Richtungsausgang, 4131
- Impulsgenerator
  - aktivieren / deaktivieren mit CTRL\_PWM, 2583
- Impulsschnittstelle: Prinzip, 4133
- Impulsverlängerung, 937, 955
- IN\_RANGE, 1719, 1984
- INC, 1735, 2000
- IncreaseFocusedValue, 3362
- IncreaseTag, 3362
- Indexvariablen, 3186
- IndexVariablen, 3186
- indirekt adressieren , 3186
- Indirekte Adressierung, 157, 1076, 1077, 1078, 1079, 1081, 1082
- Industrial Ethernet, 3467
- Informationssystem
  - Komponenten des Informationssystems, 220
  - Roll-out, 222
  - Tooltip, 222
- INIT\_RD, 1834, 2105, 2306
- Initialisierung (anwenderdefinierte Webseiten), 640
- Inkrementieren, 1735, 2000
- INSERT, 2383
- Inspektorfenster
  - Aufbau, 192
  - Automatisch reduzieren, 187
  - Ereignis, 55
  - Hardware- und Netzwerkeditor, 380, 3462
  - Querverweis, 55
  - Register Diagnose, 965
- Installation
  - Lizenzen, 75
  - Produkte hochrüsten, 98
  - Produkte modifizieren, 98
  - Produkte reparieren, 100
  - ProSave, 67
  - Protokoll, 93
  - Smartdrive, 51
  - Software anzeigen, 97
  - starten, 94
  - Support Package, 97
  - Systemvoraussetzungen, 76
  - Zielverzeichnis, 34
- Installation von License Keys, 78
- Installieren des lokalen Modems, 4348
- Instanz-Datenbaustein
  - Datentyp bei IEC-Zeiten oder -Zählern ändern, 1385
  - Definition, 1032
  - erzeugen, 1032
  - Remanenzverhalten, 1422
- INT, 1096, 1141, 1164
- INT\_TO\_, 1164
- integriert
  - Verbindung, 3473, 3477
- Integrierte Verbindung, 3164, 3484
- Integriertes Projekt
  - HMI-Variablen verknüpfen, 153
  - Integrierte HMI-Verbindung erstellen, 151
  - migrieren, 134, 145
  - nacharbeiten, 148
  - Unspezifizierte CPU wandeln, 150
  - Unspezifizierte Verbindung löschen, 154
- Intelligenter DP-Slave, (Siehe I-Slave)
- Internal Ethernet Port
  - Open settings, 3757
- Interne Netzknoten
  - Diagnose, 582
  - konfigurieren, 572
- Interne Subnetze manuell konfigurieren, 602
- interne Variablen
  - Datenaustausch, 3447
- Interne Vergleichsstelle, 948
- Interner
  - Ethernet Port, 3757
- Internet Key Exchange (IKE), 567
- Intrusion Detection System (IDS), 83, 86
- INV, 1841, 2112
- InverseLinearScaling, 3367
- InvertBit, 3365
- InvertBitInTag, 3366
- InvertiereBit, 3365
- InvertiereBitInVariable, 3366
- InvertiereLineareSkalierung, 3367
- Invertieren, 1628, 1841, 1890, 2112
- IO2MOD, 2626
- IO-Device
  - Aktualisierungszeit, 811
  - Ansprechüberwachungszeit, 812
  - vernetzen, 808
- IO-Link, 823
- IO-System, 808
  - erstellen, 808
- IP Access Control-Liste, 526
- IP-Adresse, 31, 796
- IP-Adresse zuweisen
  - aus dem Projektkontext, 1002
  - Grundsätzliches, 1000
  - über "Erreichbare Teilnehmer", 1001
- IP-Adressparameter, 798, 805
- IP-Dienste, 534

- IP-Konfiguration
  - Parameter aus dem Anwenderprogramm ändern, 3008
- IP-Netzknoten manuell konfigurieren, 601
- IP-Paketfilter-Regeln, 541
- IP-Parameter, 977
- IP-Protokoll, 3553
- IP-Regelsätze, 531
- IPsec-Einstellungen, 566
- IPv4-Adressen, 546
- IPv6
  - Notation, 679
- IPv6-Adressen, 547
- iQ series, 3758
- IQ-Sense, 956
- ISAKMP, 576
- I-Slave
  - Datenzugriff, 779
  - Hardware-Katalog, 768
- I-Slave, 769, 777
  - Datenaustausch, 769
  - konfigurieren, 778
- ISO-on-TCP
  - Merkmale, 471
  - TSAP, 480
- ISO-Protokoll, 600
  
- J**
- JMP, 1814, 2085
- JMP\_LIST, 1817, 2088
- JMPN, 1815, 2086
  
- K**
- kalibrieren, 1006
  - Überblick, 1006
- KalibriereTouchscreen, 3368
- Kalibriervorgang abbrechen, 1008
- Kaskadierzählfunktion, 919
- Kennwort, 3993
  - ändern, 3316
  - ändern in Runtime, 3324
  - für Funktionstaste, 3115
  - Hierarchieebene, 3310
  - HMI-Verbindung, 3518, 3570
  - Kennwortalterung, 3311
  - Kennwortkomplexität, 3311
  - sichern, 3993
  - wiederherstellen, 3993
- Kennwortliste, 3993
- Klicken, 3427
- Klicken bei Blinken, 3427
- Know-how-Schutz
  - Baustein drucken, 1490
  - Bausteine aus V10.5 konvertieren, 43
  - Bausteine öffnen, 1489
  - Einführung, 1484
  - einrichten, 1488
  - entfernen, 1492
  - Passwort ändern, 1491
- Kommen, 3207
- Kommentar
  - Bereichszeiger, 3489
- Kommentarabschnitt einfügen, 2294
- Kommentare
  - in SCL-Programm einfügen, 1397
- Kommunikation, 884, 3441
  - Definition, 3441
  - Fremdtreiber, 3704
  - Grundlagen, 3442
  - S7 1200, 3546
  - S7 1500, 3497
  - S7 200, 3663
  - S7 300, 3601
  - S7 400, 3601
  - SIMATIC LOGO!, 3692
  - über Bereichszeiger, 3448
  - Zyklusbelastung, 861
- Kommunikation MPI
  - S7 300/400, 3623
- Kommunikation PROFIBUS
  - S7 1200, 3562
  - S7 1500, 3513
  - S7 300/400, 3615
- Kommunikation über PUT/GET-Anweisung
  - Grundlagen zur PUT/GET-Anweisung, 484
  - Verbindung anlegen und parametrieren, 489
  - Verbindung löschen, 490
  - Verbindungsparametrierung starten, 488
  - Verbindungsparametrierung Übersicht, 485
  - Voraussetzungen, 485
- Kommunikationsanweisung
  - "AS\_DIAL", 4385, 4386
  - "AS\_MAIL", 4389
  - "PG\_DIAL", 4384, 4385
  - "SMS\_SEND", 4387
- Kommunikationsarten
  - Allen-Bradley, 3710
  - Einschränkung, 3782, 3793
  - EtherNet/IP, 3710
  - freigegeben, 3753, 3765, 3782, 3793, 3809
  - Koppelbare Steuerung, 3794

- Koppelbare Steuerungen, 3783
- Kommunikationslast, 856, 861
- Kommunikationslast einstellen, 429
- Kommunikationsmodul (CM), 2848
- Kommunikationsmodule, 869
  - Eigenschaften, 869
- Kommunikationsnetz, 3467
  - PROFINET, 3468
- Kommunikationspartner
  - Bediengerät, 3441
  - SIMATIC S7, 3546, 3601, 3660, 3670, 3673
  - Steuerung, 3441
  - vernetzen, 3472
- Kommunikationsport
  - konfigurieren, 870
- Kommunikationsprotokoll
  - festlegen, 873
  - Überblick, 872
- Kommunikationstreiber, 3448
  - Allen-Bradley, 3706
  - Basic Panel, 3494
  - Mitsubishi, 3747
  - Modicon Modbus, 3776
  - Omron, 3804
- Kompaktibilität
  - Rückwärtskompatibilität von Projekten, 235
- Kompatibilität, 39
  - Projekte aus älteren Programmversionen, 234
  - Projekte aus neueren Programmversionen, 235
  - Projekte mit Zusatzsoftware, 235
- Kompatibilität (Firmware-Versionen), 908
- Komponenten
  - in Betrieb nehmen, 3720, 3741, 3760, 3770, 3787, 3799, 3816
- Konfiguration
  - DP-Slave, 777
  - DP-Slave, einfach, 769
  - Hardware, 388
  - in Gerät laden, 825
  - in PG/PC laden, 826
  - Variable, 3173
- Konfiguration online ändern, 4268
- Konfigurationsmodus, 684
- Konfigurationssteuerung, 934
- konfigurieren
  - mehrere Variablen, 3174
  - Rechteck, 3076
- Konfigurieren
  - I-Slave, 778
- Konfigurieren des Netzes über Ethernet, 430, 3555, 3610
  - Einstellen der IP-Adresse, 430, 3555, 3610
  - Netze verbinden, 432, 3556, 3611
  - Private Subnetze bilden, 431, 3556, 3611
  - Zusammenhang IP-Adresse und Subnetzmaske, 431, 3555, 3610
- Konsistenz
  - Steckplatzregeln, 390
- Konsistenzprüfung, 598, 1508, 1513, 1514
  - Einführung, 1508, 1513
  - lokal, 517
  - projektweit, 517
- Konsistenzprüfung, 1508
- Konstante
  - deklarieren, 1202
  - Eigenschaften, 1205, 1206
  - Eingeben, 1085
  - Grundlagen, 1058
  - PLC-Variablen-tabelle, 1187
  - Regeln, 1202
- Kontaktplan, 1279
- Kontextfilter, 391
- Konvertieren, 1798, 2067, 2260
  - einer Zeichenkette in Hexadezimal mit ATH, 2374
  - Hexadezimal in Zeichenkette mit HTA, 2376
  - Zahl in Zahlenzeichenkette konvertieren mit VAL\_STRG, 2369
  - Zahlenzeichenkette in Zahl konvertieren mit STRG\_VAL, 2367
  - Zahlenzeichenkette in Zeichen kopieren mit Strg\_TO\_Chars, 2372
  - Zeichen in Zahlenzeichenkette kopieren mit Chars\_TO\_Strg, 2373
  - Zeichenkette mit S\_CONV, 2366
  - Zeiten konvertieren mit T\_CONV, 2350
- konvertierter Name (PROFINET), 802
- Konvertierung, 1133
  - explizit, 1152, 1153, 1161, 1163, 1164, 1166, 1168, 1170, 1171, 1173, 1174, 1175, 1176, 1177, 1178, 1179
  - explizit, 1155, 1157
  - implizit, 1136, 1137, 1138, 1139, 1140, 1141, 1142, 1143, 1144, 1145, 1146, 1147, 1148, 1149, 1150, 1151
- Koordinierte Übertragung, 3278
  - mit Steuerung, 3278
- Koordinierung, 3448
- KOP, 1279, 1591
- KOP-Element
  - aus Zwischenablage einfügen, 1308
  - ausschneiden, 1307
  - einfügen, 1293

- ersetzen, 1309
  - kopieren, 1307
  - löschen, 1313
  - Operand einfügen, 1313
  - Regeln beim Einfügen, 1290
  - selektieren, 1306
  - kopieren
    - Bediengerät, 49
    - Bild, 3032
    - Meldeanzeige, 54
    - Meldefenster, 54
    - Meldeindikator, 54
    - Variable, 3172
    - Vorlage, 3038
  - Kopieren
    - benutzerdefinierte Ordner, 3921
    - Bereich, 1769, 1791, 2037, 2059, 2221, 2252
    - Bereich ununterbrechbar, 1775, 1793, 2043, 2062, 2226, 2254
    - Bild, 3922
    - Bildgröße anpassen, 3922
    - Excel-Format, 3853
    - Farbe, 3922
    - Funktionstaste, 3922
    - Hardware-Komponente, 405
    - Prinzip, 3920
    - Schriftart, 3922
    - ungültiges Objekt, 3921
    - verknüpfte Objekte, 3924
    - Wert, 1760, 2028
  - Kopierschutz, 1487
  - Kopiervorlage
    - Bibliothek, 3841
  - KOP-Netzwerk
    - Baustein aufrufen einfügen, 1299, 1341
    - Kreuzung, 1319
    - Kreuzung einfügen, 1320
    - Kreuzung löschen, 1321
    - Kreuzung umhängen, 1320
    - Programmstatusanzeige, 1530
    - Regeln zur parallelen Verzweigungen, 1317
    - Unzulässige Verschaltungen, 1292
    - Zweig, 1316
    - Zweig einfügen, 1318
    - Zweig löschen, 1319
    - Zweig schließen, 1318
  - Kopplungen
    - Allen-Bradley DF1, 3726, 3727
  - Kreis, 3138
    - Radius, 3138
  - Kreuzung
    - Definition, 1319
    - einfügen, 1320
    - löschen, 1321
    - umhängen, 1320
  - Kurvenanforderung
    - Kurvenübertragung, 3535, 3700
  - Kurvenanforderungsbereich, 3535, 3700
  - Kurvenanzeige, 3139, 3987
    - Bedienelement, 3988
    - Bedienung, 3987
    - Darstellung, 3987
    - für Werte aus der Steuerung projektieren, 3201
    - Schaltfläche, 3140
    - Verwendung, 3987
  - KurvenanzeigeBlättereVorwärts, 3369
  - KurvenanzeigeBlättereZurück, 3370
  - KurvenanzeigeErweitern, 3370
  - KurvenanzeigeKomprimieren, 3371
  - KurvenanzeigeLinealLinks, 3372
  - KurvenanzeigeLinealRechts, 3371
  - KurvenanzeigeSetzeLinealmodus, 3372
  - KurvenanzeigeStartenStoppen, 3373
  - KurvenanzeigeZurückZumAnfang, 3373
  - Kurvenübertragungsbereich, 3535, 3700
  - Kurvenübertragungsbereich 1
    - Kurvenübertragungsbereich 2, 3535, 3700
  - Kurzschluss nach L+, 937
  - Kurzschluss nach M, 936
- L**
- LABEL, 1816, 2087
  - LACP, 731
  - laden
    - Projekt, 3941, 3960
  - Laden
    - Allgemeine Informationen, 256
    - auf eine Memory Card laden, 258
    - Bausteine auf eine Memory Card laden, 1482
    - Bausteine in Gerät, 1476
    - Bausteine in Gerät laden, 1478, 1479
    - Bausteine von einer Memory Card laden, 1483
    - Bediengerät, 3949
    - Erweitertes Laden, 67
    - Fehlermeldung, 3955
    - im Betriebszustand RUN, 1476
    - in Gerät, 38, 825
    - in PG/PC, 826
    - Projektdateien in Gerät laden, 257
    - SIMATIC PC Station, 68
    - taktsynchrone Applikationen, 825
    - von einem Gerät, 38, 43, 259

- Laden eines Projekts
  - mit angeschlossenem Bediengerät, 3939
- Laden über S7-Routing, 3940
- Laden von Daten auf die PLC
  - Fehlermeldung, 67
- Ladespeicher, 842, 1515
  - anzeigen, 1519
- Länge
  - Bereichszeiger, 3489, 3491
- Laufzeitmessung, 2308
- Layer 2, 530
- Layer 3, 530
- LDT, 1110
- Lead- und Lag Algorithmus, 1877, 2148, 2342
- LEAD\_LAG, 1877, 2148, 2342
- Lebensdauer von Zertifikaten, 565
- LED, 2561
  - Abbild, 3539
  - Funktion, 3539
  - Zuordnung, 3539
- LED ACK, 4004
- LED-Zustand
  - auslesen mit LED, 2561
- Leerbox
  - FUP-Element einfügen, 1336
  - KOP-Element einfügen, 1294
- LEFT, 2380
- Leistungsbilanz, 944
- Leistungsmerkmale
  - Bediengerät, 4020
  - Engineering System, 4019
- Leistungsbruch, 929
- LEN, 2378
- Lernfunktionalität, 572
- Lernmodus, 599
- LeseBenutzername, 3374
- LeseGruppennummer, 3375
- LeseKennwort, 3375
- Lesen
  - asynchron - Datensatz einer Baugruppe mit RD\_DPARA, 2511
  - aus einem Datenbaustein im Ladespeicher mit READ\_DBL, 2614
  - Datensatz einer Baugruppe mit RD\_DPAR, 2508
  - eines Datensatzes mit RD\_REC, 2430
- Lesezeichen
  - Funktion, 1377
  - löschen, 1379
  - navigieren, 1379
  - setzen, 1378
- LGC\_GADR, 2634
- License Key, 79, 91
  - Umgang mit License Keys, 79, 91
- License Manager Panel Plugin, 89
- LIMIT, 1742, 2008, 2206
- Limitieren, 1742, 2008, 2206
- Lineare Programmierung, 1026
- LineareSkalierung, 3377
- LinearScaling, 3377
- Linie, 3141
  - gestalten, 3058
  - Linienanfang, 3141
  - Linienende, 3058, 3141
- Linienanfang
  - Linie, 3141
- Linienende
  - Linie, 3141
- LINT, 1099
- Lizensierung Bediengerät
  - nicht PC-basiert, 89
- Lizenz
  - defekt, 80, 92
  - defekte Lizenz, 80, 92
  - Starten ohne gültige Lizenz, 77, 88
- Lizenzen, 75
- LLDP, 526, 527
- LLDP (Link Layer Discovery Protocol), 816, 3510, 3561, 3614
- LN, 1747, 2013, 2210
- LOG\_GEO, 2630
- LOG2GEO, 2624
- LOG2MOD, 2625
- Logarithmus, 1747, 2013, 2210
- Log-Datei öffnen
  - Paketfilter-Ereignisse, 584, 586
  - Sicherheits-Ereignisse, 584, 586
  - System-Ereignisse, 583, 585
- Logging, 577
  - CP x43-1 Adv., 606
  - SCALANCE S, 591, 593
- Logikanalysatorfunktion, 4296
- Login Webserver, 635
- Logische Adresse
  - einer Baugruppe ermitteln mit GADR\_LGC, 2633
  - einer Baugruppe ermitteln mit RD\_LGADR, 2631
  - zugehörigen Steckplatz ermitteln mit LGC\_GADR, 2634
  - zugehörigen Steckplatz ermitteln mit LOG\_GEO, 2630
- Logoff, 3352
- Logon, 3358
- Lokaldatenbit (L), 842
- Lokaldatenbyte (LB), 842
- Lokaldatendoppelwort (LD), 842

- Lokaldatenwort (LW), 842  
 Lokale Belegung  
   einer Funktionstaste, 3108, 3112  
 Lokale Fehlerbehandlung, 1405  
   Anweisungen, 1405  
   Fehlerinformation, 1405  
   Prioritäten, 1406  
 Lokale Variable, 1257  
 Lokales Logging, 577  
 Lokalzeit  
   auslesen mit RD\_LOC\_T, 2357  
   berechnen mit SET\_TIMEZONE, 2359  
   schreiben mit WR\_LOC\_T, 2358  
 Long word, 1094  
 Loop-In-Alarm, 3427  
   projektiert, 3223  
 LöscheMeldepuffer, 3378  
 LöscheMeldepufferProTool, 3379  
 löschen  
   Benutzer, 3317, 3999  
   Benutzergruppe, 3318  
   Berechtigung, 3319  
   Bild, 3032  
   Objekt, 3046  
   Rezepturdatensatz, 4016  
   Variable, 3173  
   Vorlage, 3038  
 Löschen  
   Filter im Belegungsplan, 1499  
   Hardware-Komponente, 404  
 loslassen, 3428  
 LREAL, 1103, 1146, 1173  
 LREAL\_TO\_, 1173  
 LTIME, 1107  
 LTOD, 1109  
 LWORD, 1094
- M**
- MAC-Dienste, 538  
 MAC-Netzknoten manuell konfigurieren, 602  
 MAC-Paketfilter-Regeln, 544  
 MAC-Regelsätze, 531  
 Main, 885  
 Main Mode, 567  
 Manuelles Fragment, 658  
 Mathematische Funktionen  
   CALCULATE, 1302, 1344  
 Matrixscanner, 1874, 2146, 2340  
 MAX, 1740, 2006, 2204  
 MAX\_LEN, 2373  
 Maximal verfügbaren Ladespeicher  
   anzeigen, 1519  
 Maximale Anzahl ansteuerbarer Antriebe, 4129  
 maximale Länge  
   Variable, 69  
 Maximale Zykluszeit, 860  
 Maximum, 1740, 2006, 2204  
 MB\_CLIENT, 2877, 2894  
 MB\_COMM\_LOAD, 2859  
 MB\_MASTER, 2863  
 MB\_SERVER, 2885, 2905  
 MB\_SLAVE, 2871  
 MC\_ChangeDynamic: Anweisung, 2667  
 MC\_ChangeDynamic: Parameter, 2668  
 MC\_CommandTable: Anweisung, 2665  
 MC\_CommandTable: Parameter, 2666  
 MC\_Halt: Anweisung, 2646  
 MC\_Halt: Funktionsdiagramm, 2649  
 MC\_Halt: Parameter, 2647  
 MC\_Home: Anweisung, 2643  
 MC\_Home: Parameter, 2644  
 MC\_MoveAbsolute: Anweisung, 2650  
 MC\_MoveAbsolute: Funktionsdiagramm, 2652  
 MC\_MoveAbsolute: Parameter, 2651  
 MC\_MoveJog: Anweisung, 2661  
 MC\_MoveJog: Funktionsdiagramm, 2664  
 MC\_MoveJog: Parameter, 2662  
 MC\_MoveRelative: Anweisung, 2653  
 MC\_MoveRelative: Funktionsdiagramm, 2656  
 MC\_MoveRelative: Parameter, 2654  
 MC\_MoveVelocity: Anweisung, 2657  
 MC\_MoveVelocity: Funktionsdiagramm, 2660  
 MC\_Power: Anweisung, 2636  
 MC\_Power: Funktionsdiagramm, 2640  
 MC\_Power: Parameter, 2636  
 MC\_Reset: Anweisung, 2641  
 MC\_Reset: Parameter, 2641  
 MCAT, 1868, 2139, 2333  
 MD5, 568, 616, 626  
 MDM, 229  
 Mehrfachauswahl, 3050  
 Mehrfachselektion, 394  
 Mehrpunkt-Kopplung  
   Allen-Bradley DF1, 3729, 3730  
 Mehrstastenbedienung, 71  
 Meldeanzeige, 54, 3142, 3144, 3228, 3240, 4004  
   Bedienelement, 3143  
   Bedienung, 3240, 4004  
   Darstellung, 3142, 3239, 4003  
   Darstellung projektieren, 3231  
   Meldetextfenster, 3240, 4004  
   projektieren, 3230

- Spalte, 3143
- MeldeanzeigeBearbeiteMeldung, 3380
- MeldeanzeigeQuittiereMeldung, 3381
- MeldeanzeigeZeigeHilfetext, 3381
- Meldearchiv
  - in Runtime, 4002
- Meldeereignis
  - Gehen, 3207
  - Kommen, 3207
  - Quittieren, 3207
- Meldefenster, 54, 3228
  - in Runtime, 3238, 4002
  - projektieren, 3232
- Meldegruppe, 3211, 3214
  - anlegen, 3217
  - Migration, 125
  - projektieren, 3217
- Meldeindikator, 54, 3147, 3229, 3241, 4005
  - Bedienung, 3241
  - Darstellung, 3147, 3241
  - Ereignisse, 3147
  - in Runtime, 3238, 4002
  - Meldeklassen, 3147
  - projektieren, 3232
- Meldekategorie, 3213
  - Darstellung, 3239, 4003
  - in Runtime, 3238, 4000
- Meldekategorie, 3208
  - benutzerdefinierte, 3209
  - Namensänderung durch die Migration, 126
  - verwenden, 3209
  - vordefinierte, 3209
- Meldenummer, 3212, 3213
- Meldepuffer
  - in Runtime, 3238, 4001
- Meldesystem, 3204
- Meldetext, 3213
  - Ausgabefelder, 61
  - formatieren, 3223
  - Formatierung entfernen, 3223
  - Sonderzeichen, 61
- Meldeverfahren, 3205
- Meldezustand, 3213
  - Gegangen, 3207
  - Gekommen, 3207
  - Quittiert, 3207
- Meldung
  - anzeigen, 3142
  - bearbeiten, 4007
  - Bestandteile, 3213
  - exportieren, 3858
  - importieren, 3859
  - in Runtime, 3237, 4000
  - Inspektorfenster, 55
  - projektieren, 3219, 3225
  - quittieren, 4006
  - Systemfunktion, 3242
- Meldung mit Einfachquittierung, 3211
- Meldung ohne Quittierung, 3211
- Meldungen
  - Datentypen, 3588, 3647, 3689, 3701
  - Einschränkung, 3774
  - Modicon Modbus, 3802
  - nicht integrierte Verbindung, 3802
  - projektieren, 3588, 3647, 3689, 3701
  - Wert einer Variable ausgeben, 3222
- Meldungen anzeigen, 828
- Meldungen des Modems
  - Protokolldatei, 4396
- Meldungen projektieren
  - Allen-Bradley, 3744
  - Besonderheiten, 3773
  - Datentypen, 3774
  - nicht integrierte Verbindung, 3773
- Meldungsanzeige, 828
  - Ansicht "Aktive Meldungen", 830
  - Archiv exportieren, 829
  - Archiv leeren, 830
  - Archivansicht, 828
  - Aufbau der Meldungen in der Ansicht "Aktive Meldungen", 830
  - Aufbau der Meldungen in der Archivansicht, 828
  - Meldung quittieren, 831
  - Meldungen empfangen, 829
  - Meldungen ignorieren, 832
  - Tastaturbedienung, 832
- Meldungspufferüberlauf, 3428
- Memory Card, 840
  - Eigenschaften anzeigen, 312
  - Einführung, 310
  - formatieren, 1000
  - Kartenleser hinzufügen, 311
  - Ziehen/Stecken, 38
  - zugreifen, 312
- Memory Cards, 32
- Menübefehl
  - einfache Rezepturanzeige, 3297, 4013
- Merker
  - Anzeige der remanenten Merker aktivieren, 1502
- Merkerbit (M), 842
- Merkerbyte (MB), 842
- Merkerdoppelwort (MD), 842
- Merkerwort (MW), 842
- Messfenster (TOR), 926



- MIB, 526
- Micro, 3785
- MicroLogix, 3713, 3736
- MID, 2381
- Migration, 32
  - Ablauf der Migration, 108
  - Anpassungen vorher, 121
  - Datentypen Allen-Bradley DF1, 139
  - Datentypen Allen-Bradley Ethernet IP, 139
  - Datentypen Mitsubishi FX, 140
  - Datentypen Modicon Modbus, 140
  - Datentypen Modicon Modbus TCP/IP, 141
  - Datentypen Omron Hostlink/Multilink, 141
  - Datentypen SIMATIC S7 200, 142
  - Datentypen SIMATIC S7 300/400, 142
  - der Benutzerverwaltung, 131
  - der Rezepturdaten, 131
  - Einführung, 114
  - Einführung zur Migration, 107
  - Einschluss der Hardware-Konfiguration, 108
  - Globale Bibliothek, 130
  - Grundlagen, 115
  - Historie anzeigen, 113
  - Integriertes Projekt, 145
  - Integriertes Projekt migrieren, 134, 145
  - Projektbibliothek, 130
  - Projekte migrieren, 118, 136
  - Protokolldatei anzeigen, 114
  - Schrifteinstellungen, 123
  - unterstützte Bediengeräte, 120
  - Unterstützte Produkte, 107
  - von externen Variablen, 138
  - von Meldegruppen, 125
  - von Projekttexten, 128
  - von Runtime-Daten, 131
  - von Sprachen, 128
  - von Texten, 128
  - von Variablen, 124
  - WinCC V7.0 SP3, 54
- Migrationstool, 116
- Migrations-Tool
  - Deinstallation, 105
  - Hardware-Konfiguration einschließen, 110
  - Migrations-Datei erstellen, 111
  - Migrations-Tool verwenden, 109
  - Systemvoraussetzungen, 104
  - Vertrieb und Bezugsquellen, 104
- Migrationstool starten, 110
- Migriertes Projekt
  - übersetzen, 119
- MIN, 1738, 2004, 2202
- Mindestzykluszeit, 838, 860
- Minimum, 1738, 2004, 2202
- Mirroring, 716
- Mischbetrieb, 569
- Mitsubish
  - Verbindungsunterbrechung, 63
- Mitsubishi, 3704
  - Adresse, 3759
  - Analogmeldung, 3774
  - Basic Panel, 3494
  - FX, 3747
  - Kommunikationstreiber, 3747
  - TCP/IP, 3747
- Mitsubishi FX
  - Anschluss, 3766
  - CPU-Typ, 3768
  - Datentyp, 3767
  - Migration der Datentypen, 140
  - Mitsubishi MC TCP/IP, 3753, 3765
  - Verbindung, 3762
  - Verbindung projektieren, 3762
  - Verbindungsparameter, 3763
- Mitsubishi MC TCP/IP
  - Anschluss, 3752
  - CPU-Typ, 3758
  - Datentyp, 3757
  - Verbindung, 3748
  - Verbindung projektieren, 3748
  - Verbindungsparameter, 3750
- Mnemonic einstellen, 1246
- MOD, 1365, 1732, 1998
- Modbus-Kommunikation
  - als Modbus-Master mit MB\_MASTER, 2863
  - als Slave mit MB\_SLAVE, 2871
  - Port konfigurieren mit MB\_COMM\_LOAD, 2859
- Modem
  - entferntes, 4349
  - lokales, 4348
  - Probleme, 4391
- Modemverbindung kommt nicht zustande, 4395
- Modemverbindung wird getrennt, 4396
- Modemverbindung wird unterbrochen, 4395
  - Ursachen und Behebung, 4395
- Modicon
  - Anschluss, 3792
  - Einschränkungen mit Modbus RTU, 3793
  - freigegebene Kommunikation mit Modbus RTU, 3793
  - Verbindungskabel, 3792
- Modicon M340, 3785
- Modicon Modbus, 3704
  - Basic Panel, 3494
  - Kommunikationstreiber, 3776

- Migration der Datentypen, 140
  - RTU, 3776
- MODicon Modbus
  - TCP, 3776
- Modicon Modbus RTU, 3798
  - Datentyp, 3797
  - Verbindung, 3788
  - Verbindung projektieren, 3788
  - Verbindungsparameter, 3790
- Modicon Modbus TCP, 3785
  - Anschluss, 3781
  - Datentyp, 3784
  - Verbindung, 3777
  - Verbindung projektieren, 3777
  - Verbindungsparameter, 3778
- Modicon Modbus TCP/IP
  - Migration der Datentypen, 141
- Modicon MODBUS TCP/IP
  - Change word order, 3780
- Modicon Modbus Verbindung
  - Datentypen, 3802
- Module
  - Kommunikationszustand, 581
- Module der S7-1200, 39
- ModuleStates, 2574
- Modulversion
  - aktualisieren, 412
- Modus, 3152
  - E/A-Feld, 3132
  - Grafisches E/A-Feld, 3136
- mögliche Fehlerursache
  - Daten übertragen, 3534, 3587, 3646, 3686, 3837
- Momentum, 3785, 3798
- Mono-Master-System, 769
- Motion Control CPU S7-1200
  - Leitfaden, 4138
- MOVE, 162, 1760, 2028
- MOVE\_BLK, 1769, 2037, 2221
- MOVE\_BLK\_VARIANT, 1771, 2039, 2223
- MPI, 3467
  - Adressierung, 3631
  - HMI-Verbindung, 3624
  - Netzwerk, 3470
  - Netzwerkarchitektur, 3470
  - Parameter, 3626, 3628, 3629
  - S7 200, 3661, 3671
  - S7 300/400, 3623, 3624
  - Verbindung, 3654
- MPI/PROFIBUS Einstellungen zurücksetzen, 4294
- MPI-Adresse
  - S7 300, 3631
  - S7 400, 3631

- MPI-Parameter
  - S7 300/400, 3626, 3628, 3629
- MRP-Domain, 980
- MRP-Domain-Eigenschaften, 980
- MSK\_FLT, 2540
- MSTP, 727
  - Port, 724
- MUL, 1729, 1995
- Multicast, 554, 737
- Multiinstanz, 1069
  - Aufruftyp ändern, 1301, 1343
  - Definition, 1042
  - deklarieren, 1269
- Multiple Spanning Tree, 724, 727
- Multiplex-/Synchronbetrieb, 960
- multiplexen, 3186
  - Adress-Multiplexen Allen-Bradley Ethernet IP, 3717
- Multiplexen, 1847, 2118, 2314
  - Adress-Multiplexen, 3189
  - mit absoluten Adressen, 3189
  - mit symbolischen Adressen, 3190
- Multiplizieren, 1729, 1995
- MUX, 1847, 2118, 2314
- My Documentation Manager, 229

## N

- N, 1638, 1641, 1900, 1902
- N\_TRIG, 1644, 1905
- Nachkommastellen, 1757, 2025, 2220, 3976, 3978
- Nachricht
  - Beginn festlegen, 876
  - Ende festlegen, 877
  - senden, 875
- Name
  - Arrayelement, 60
- Name ändern
  - Benutzer, 3316
  - Benutzergruppe, 3317
  - Berechtigung, 3318
- Namen von Meldeklassen
  - Änderung durch die Migration, 126
- NAT/NAPT
  - Routing, 553
- NEG, 1733, 1999
- Negieren, 1733, 1999
- Negotiation, 699
- Netz-ID, 596
- Netzlaufwerk, 32
- Netzsicht, 3471
  - Gerät hinzufügen, 400

- Hardware- und Netzwerkeitor, 369, 3451
- Netzübersicht
  - Grundfunktionen, 417
  - Grundfunktionen zur Bearbeitung der Netzübersichtstabelle, 417
- Netzwerk, 3441
  - aufklappen, 1286, 1330
  - einfügen, 1283, 1285, 1327, 1329
  - Ethernet, 3468
  - in Bausteinen navigieren, 1289, 1332
  - Kommentar eingeben, 1288, 1331
  - kopieren, 1285, 1329
  - löschen, 1286, 1329
  - MPI, 3470
  - PPI, 3471
  - PROFIBUS, 3469
  - PROFINET, 3468
  - selektieren, 1284, 1328
  - Titel eingeben, 1287, 1330
  - verwenden, 1283, 1327
  - zuklappen, 1286, 1330
- Netzwerkadapter, 604
- Netzwerkarchitektur
  - PPI, 3471
- Netzwerkeitor
  - Funktion, 367, 3449
  - Hardware-Katalog, 382, 3464
  - Inspektorfenster, 380, 3462
  - Komponenten, 367, 3449
- Netzwerkcommentar
  - ausblenden, 1245
  - einblenden, 1245
- Netzwerk-Syslog, 577
- Netzzugang
  - Eigenschaften öffnen, 4281
- Neu verbinden
  - Variable, 3177
- Neue Datentypen, 156
- Neustart (Warmstart), 836
- Nicht gesteckte Baugruppe, 396
- nicht integriert
  - Verbindung, 3477
- Nicht lizenzierter Modus
  - Bediengeräte, 90
  - Engineering-System, 77, 88
- Nicht Plug&Play-fähige Modems, 4348
- Nicht spezifizierte CPU, 401
- NORM\_X, 1807, 2077, 2269
- Normdevice, 818
- Normieren, 1807, 2077, 2269
- Normslave, 791
- NOT, 1369, 1370, 1628
- NOT\_OK, 1722, 1987
- NTP, 737
- NTP (secure), 560
- NTP-Server, 560
- NTP-Server exportieren, 559
- NTP-Server importieren, 559
- Nur HTTPS-Server, 683
- Nutzdaten
  - Bereich, 847
- O**
- O, 4270
- OB
  - Ereignisse und OBs, 847
  - Übersicht, 847
- OB 1, 885
- OB 80, 895
- OB 82, 896
- OB 83, 897
- OB für herstellerepezifischen Alarm, 891
- OB für profilspezifischen Alarm, 891
- Oberflächensprache, 3882
  - auswählen, 3885
- Objekt
  - anordnen, 3043, 3054
  - außerhalb des Bereichs, 3055
  - bearbeiten innerhalb einer Gruppe, 3070
  - bündig ausrichten, 3052
  - des gleichen Typs einfügen, 3043
  - Direkte Bewegung dynamisieren, 3102
  - drehen, 3043, 3055
  - dynamisieren, 3096
  - einer Ebene zuordnen, 3122
  - einfügen, 3043, 3044
  - Erscheinungsbild dynamisieren, 3099
  - gestalten, 3058
  - gleichmäßig ausrichten, 3053
  - Größe ändern, 3049
  - gruppieren, 3066
  - in Bibliothek hinterlegen, 3848
  - löschen, 3046
  - mehrere auswählen, 3043, 3050
  - mehrfach einfügen, 3058
  - OLE-Objekt aus Datei erstellen, 3044
  - OLE-Objekt neu erstellen, 3044
  - Position ändern, 3047
  - Position und Größe mehrerer Objekte ändern, 3044, 3060
  - spiegeln, 3043
  - Tab-Reihenfolge, 3043
  - Verfügbarkeit für Basic Panel, 3127

- vervielfachen, 3058
- Objektgröße
  - vereinheitlichen, 3049
- Objektgruppe
  - Animation, 3105
  - Objekt entfernen, 3069
  - Objekt innerhalb einer Gruppe bearbeiten, 3070
- Objektliste
  - Variable, 3169
- Objektnamen
  - Eindeutigkeit, 115
- ODER, 1885, 1886
- OF, 2282
- Offline, 512
- Offline-/Online-Vergleich
  - automatische Gerätezuordnung, 507
- OK, 1721, 1986
- OLE-Objekt
  - aus Datei erstellen, 3044
  - in Grafiksammlung hinterlegen, 3063
  - in Grafiksammlung hinterlegen , 3898
  - neu erstellen, 3044
- Omron, 3704
  - Analogmeldung, 3819
  - Basic Panel, 3494
  - Hostlink, 3804
  - Kommunikationstreiber, 3804
- Omron Host Link, 3814
  - Adresse, 3815
  - Anschluss, 3809
  - Datentyp, 3813
  - Verbindungskabel, 3809
  - Verbindungsparameter, 3807
- Omron Hostlink
  - Verbindung, 3805
  - Verbindung projektieren, 3805
- Omron Hostlink/Multilink
  - Migration der Datentypen, 141
- Online, 512
  - Hardware-Erkennung, 401
- Online- und Diagnosesicht, 964
- Online- und Diagnosesicht starten, 973
- Online verbinden, 1012
  - mehrere Geräte verbinden, 4274, 4275
  - mehrere TIA-Portal-Instanzen, 37
- Online-Anzeige
  - Farbe orange, 973
- Online-Betrieb, 29, 37
- Online-Diagnose, 578
- Online-Modus, 4269
  - Alle erreichbaren Teilnehmer als Liste anzeigen, 4267
  - Anzeige des Online-Modus, 4271
  - Erreichbare Teilnehmer an einer Schnittstelle anzeigen, 4267
  - Online verbinden, 4272
  - Online-Verbindung trennen, 4273
  - Standby oder Ruhezustand des PG/PC, 4270
- Online-Tools, 964, 974
- Online-Verbindung, 4269
  - mehrere TIA-Portal-Instanzen, 37
- Online-Verbindung trennen, 1014
- Online-Zugang, 4278
- Online-Zustand anzeigen, 4273
- OP 73
  - Baudrate am PROFIBUS, 3544, 3599, 3659, 3661, 3669, 3671, 3672
- OP 77A
  - Baudrate am PROFIBUS, 3544, 3599, 3659, 3661, 3669, 3671, 3672
- OP73
  - Projekt laden, 68
- OP77A
  - Projekt laden, 68
- OPC
  - Basic Panel, 3494
- Open User Communication
  - Parameterwerte ändern, 479
  - Allgemein, 457
  - Anweisungen, 457
  - Daten empfangen mit TRCV, 2984, 2988
  - Daten senden mit TSEND, 2978, 2981
  - Daten über UDP empfangen mit TURCV, 2997
  - Daten über UDP senden mit TUSEND, 2994
  - IP-Konfigurationsparameter ändern mit T\_CONFIG, 3008
  - Kommunikationsverbindung abbauen mit TDISCON, 2976
  - Kommunikationsverbindung aufbauen mit TCON, 2968, 2971
  - Port-Nummern, 478
  - Rücklesbarkeit, 479
  - TCON\_IP\_RFC, 477
  - TCON\_IP\_v4, 475
  - TCON\_Param, 473
  - TSAP, 480
  - Verbindung anlegen, 466
  - Verbindung aufbauen und Daten lesen mit TRCV\_C, 2942, 2948
  - Verbindung aufbauen und Daten senden mit TSEND\_C, 2931, 2935
  - Verbindung löschen, 469
  - Verbindungsaufbau, 457
  - Verbindungsbeschreibung, 472, 475, 476

Verbindungsparameter, 462  
 Verbindungsparametrierung, 458, 459  
 Verbindungsparametrierung starten, 465  
 Verwendete Protokolle, 470  
 Operand, 1056, 1058, 1059, 1065, 1066, 1067, 1069, 1070, 1072, 1074, 1085  
     einfügen, 1313, 1355  
 Operand auf negative Signalflanke abfragen, 1638, 1900  
 Operand auf positive Signalflanke abfragen, 1637, 1899  
 Operand bei negativer Signalflanke setzen, 1641, 1902  
 Operand bei positiver Signalflanke setzen, 1640, 1901  
 Operandenbereich, 842  
 Optimierter Bausteinzugriff, 155  
 Optimierter Zugriff, 1034, 1036  
 Optionenhandling, 784, 787, (Siehe Konfigurationssteuerung (ET 200SP))  
 OR, 1370, 1838, 1886, 2109  
 Ordner verknüpfen  
     externe Grafik, 3062  
 Ordnerverknüpfung  
     bearbeiten, 3062  
     entfernen, 3063  
     umbenennen, 3063  
 Organisationsbaustein (OB)  
     anlegen, 1212  
     Ereignis zuordnen mit ATTACH, 2517  
     Ereigniszuordnung aufheben mit DETACH, 2518  
     Funktion, 1028  
     in externe Quelldatei exportieren, 1448  
     Startinformation, 1028  
     Startinformationen auslesen mit RD\_SINFO, 2552  
     Weckalarm-OB, Parameter abfragen mit QRY\_CINT, 2521  
     Weckalarm-OB, Parameter setzen mit SET\_CINT, 2519  
 Ortszeit, 856  
 OSPF  
     Area Range, 753  
     Areas, 752  
     Configuration, 751  
     Schnittstellen, 754  
     Virtual Links, 756  
 OUT\_RANGE, 1720, 1985  
 Overlay-Icon, 972

## Ö

öffnen  
     Globale Bibliothek, 3851  
 Öffnen einer Beobachtungstabelle, 1538  
 Öffnen einer Forcetabelle, 1565  
 Öffnen von Telefonbüchern, 4340  
 Öffnerkontakt, 1627  
 östliche Zeichen  
     Eingabe am Bediengerät, 3971

## P

P, 1637, 1640, 1899, 1901  
 P\_TRIG, 1642, 1904  
 Packen von Adressen, 782  
 PageDown, 3395  
 PageUp, 3394  
 Paketfilter-Log, 584, 586  
 Palettenmodus, 196  
 Panning, (Siehe Ansicht verschieben)  
 Parameter  
     für CPU, 856  
     MPI, 3626, 3628, 3629  
     PROFIBUS, 3514, 3516, 3517, 3564, 3566, 3567, 3617, 3619, 3621  
     S7 1200, 3596  
     S7 1500, 3541  
     S7 200, 3656, 3661, 3666, 3671, 3672  
     SIMATIC LOGO!, 3695  
     Verbindung, 3708, 3723, 3750, 3763, 3778, 3790, 3807  
     Versteckte Parameter, 1315, 1357  
     vordefinierte - schreiben mit WR\_DPARM, 2515  
 Parametertypen, 1126  
 Parameterübergabe, 1044, 1046, 1048, 1050, 1052, 1053, 1054, 1055  
 Parameterversorgung, 1045  
 Parametrieren des entfernten Modems, 4349  
 Parametrierung  
     Hardware, 388  
 Passwort  
     für CPU-Zugriffsschutz, 856  
     HMI-Verbindung, 3506, 3557  
 Passwort-Legitimierung, 1823, 2094, 2295  
 Passwort-Legitimierung einschränken und freigeben, 1823, 2094, 2295  
 Passwortschutz, 635, 856, 880, 881  
     Schutzkonzept, 274  
     Zugriffsrechte zurücknehmen, 275

- PC-CP, 511
- PDF-Dateien anzeigen, 76
- PE\_CMD, 2471
- PE\_DS3\_Write\_ET200S, 2476
- PE\_End\_RSP, 2496
- PE\_Error\_RSP, 2493
- PE\_Get\_Mode\_RSP, 2499
- PE\_I\_DEV, 2489
- PE\_List\_Modes\_RSP, 2497
- PE\_Measurement\_List\_RSP, 2504
- PE\_Measurement\_Value\_RSP, 2505
- PE\_PEM\_Status\_RSP, 2501
- PE\_Start\_End, 2467
- PE\_Start\_RSP, 2495
- PEEK, 2241
  - Speicheradresse lesen, 2241
- PEEK\_BOOL, 2244
  - Speicherbit lesen, 2244
- Perfect Forward Secrecy, 568
- Performance, 155
- Peripherie, 1066
  - direkter Zugriff auf, 847
- Peripherieadresse, 628, 847
- Peripherieausgang, 842
- Peripherieeingang, 842
- Peripheriezugriffsfehler, 845
- Permanente Station, 667
- Permanente Verbindung, 667
- PG/PC-Schnittstelle
  - Einstellbare Parameter an der MPI-Schnittstelle, 4287
  - Einstellbare Parameter an der PROFIBUS-Schnittstelle, 4288
  - online verbinden, 4273
- PG/PC-Schnittstelle parametrieren, 4280
  - Automatische Erkennung der Busparameter, 4288
  - MPI-Schnittstelle parametrieren, 4289
  - PROFIBUS-Schnittstelle parametrieren, 4290
- PID\_3Step
  - Anweisung, 2718, 2750
  - Ausgangsparameter, 2729, 2759
  - Durchgangsparameter, 2731
  - Eingangsparameter, 2727, 2758
  - Statische Variablen, 2761
- PID\_Compact
  - Eingangsparameter, 2680, 2703
  - Anweisung, 2699
  - Ausgangsparameter, 2681, 2703
  - Durchgangsparameter, 2682
  - Statische Variablen, 2683, 2704
- PLC5, 3736
- PLC-Datentyp
  - Adressieren, 1071
  - anlegen, 1438
  - ARRAY deklarieren, 1440
  - Definition, 1127, 1436
  - Deklaration in Datenbausteinen, 1418
  - Deklarationstabelle für PLC-Datentypen, 1437
  - deklarieren, 1441
  - deklarieren in der Bausteinschnittstelle, 1267
  - Eigenschaften von Variablen, 1442, 1443
  - löschen, 1439
  - Offline/Offline-Vergleich, 1455
  - STRUCT deklarieren, 1440
  - Struktur programmieren, 1439
  - systemerzeugte PLC-Datentypen in Bibliotheken, 43
  - vergleichen, 1467
- PLC-Datentypen, 156
- PLC-Variable
  - importieren- und exportieren, 1586, 1587, 1588
  - beobachten, 1206
  - Definition, 1057
  - deklarieren, 1194, 1195, 1198, 1200
  - Eigenschaften, 1204
  - kopieren, 1208
  - neu verbinden, 3177
  - Offline/Offline-Vergleich, 1455
  - PLC-Variablen-tabelle, 1186, 1187, 1192, 1193, 1194
  - PLC-Variablen-tabelle vergleichen, 1455
  - Regeln, 1192
  - Remanenzverhalten, 1196, 1197
  - Vergleich, 1452
  - vergleichen, 1465
  - Vergleichsergebnisse aktualisieren, 270
  - Zeilen sortieren, 1209
  - Zellen automatisch ausfüllen, 1209
  - Zulässige Adressen und Datentypen, 1192
- PLC-Variable neu verbinden, 3177
- PLC-Variablen, 644, 645, 646
- PN/PN Coupler
  - Ethernet-Subnetze koppeln, 822
  - Gruppieren, 822
- POINTER, 1120
- POKE, 2245
  - Speicheradresse schreiben, 2245
- POKE\_BLK, 2249
  - Speicherbereich schreiben, 2249
- POKE\_BOOL, 2247
  - Speicherbit schreiben, 2247
- Port
  - 102 (S7-Protokoll - TCP), 535

- 123 (NTP), 554, 558
- 161 (SNMP), 554
- 20/21 (FTP), 535
- 443 (HTTPS), 554
- 4500 (IPsec), 554
- 500 (IPsec), 554
- 500 (ISAKMP - UDP), 576
- 514 (Syslog), 554
- 80 (HTTP), 535
- 8448 (Security-Diagnose), 619
- Konfiguration, 700
- Portkonfiguration, 699, 701
- umbenennen, 507
- verschalten, 810
- PORT\_CFG, 2828
- Portalansicht, 176
- Portkonfiguration, 701
- Port-Nummern, 478
- Portoptionen
  - Autonegation aktivieren, 3508, 3559, 3612
  - Übertragungsrate/Duplex, 3508, 3559, 3612
  - Überwachen, 3508, 3559, 3612
- Port-Optionen, 815, 3509, 3560, 3613
  - Autonegation aktivieren, 814
  - Übertragungsmedium/Duplex, 814
  - Überwachen, 814
- Ports verschalten
  - grafische Ansicht, 503, 505
  - tabellarische Ansicht, 504, 506
- Portverschaltung ändern
  - grafische Ansicht, 503
- Position
  - mehrerer Objekte verändern, 3060
  - von Objekten, 3047
- Potenzieren, 1758, 2026
- PPI, 3467
  - Netzwerk, 3471
  - Netzwerkarchitektur, 3471
  - S7 200, 3672
- Premium, 3785
- Preshared Keys, 566
- Prioritäten
  - Lokale Fehlerbehandlung, 1406
- Produktsupport, 384, 3466
  - aktivieren, 384
- PROFIBUS, 426, 3467, 3569, 3622
  - ET 200S im DPV1-Modus, 788
  - HMI-Verbindung, 3511, 3563, 3616
  - Kopplung mit PROFINET, 793
  - Netzwerk, 3469
  - OP 73, 3544, 3599, 3659, 3661, 3669, 3671, 3672
  - OP 77A, 3544, 3599, 3659, 3661, 3669, 3671, 3672
  - Parameter, 3514, 3516, 3517, 3544, 3564, 3566, 3567, 3599, 3617, 3619, 3621, 3659, 3669
  - S7 1200, 3562, 3563
  - S7 1500, 3513
  - S7 300/400, 3511, 3615, 3616
  - Standard, 3569, 3622
  - Universell, 3569, 3622
- PROFIBUS DP, 770, 3469
  - Verbindung, 3652
- PROFIBUS Leitungskonfiguration
  - Optischer Ring, 428
- PROFIBUS Profile, 426, 3569, 3622
  - Auswirkungen auf die Übertragungsgeschwindigkeit, 426, 3569, 3622
  - Bedeutung der Profile, 427, 3569, 3623
  - Unterschiedliche Profile am selben Subnetz, 426, 3569, 3622
- PROFIBUS Subnetz konfigurieren
  - Bedeutung der Busparameter, 425
  - Busparameter einstellen, 424
  - Parameter aufeinander abstimmen, 423
- PROFIBUS-Parameter
  - S7 1200, 3564, 3566, 3567
  - S7 1500, 3514, 3516, 3517
  - S7 300/400, 3617, 3619, 3621
- PROFIenergy
  - Abgefragte Energiedaten als Antwort generieren, 2499
  - Abgefragte Engiesparmodi als Antwort generieren, 2497
  - Angeforderte Messwerte als Antwort generieren, 2505
  - Antwort auf Kommando zum Pausenende generieren, 2496
  - Antwort auf Kommando zum Pausenstart generieren, 2495
  - Beschreibung, 2465
  - Energiesparmodus starten und beenden, 2467, 2471
  - Liste der unterstützten Messwerte als Antwort generieren, 2504
  - Negative Antwort auf Kommando generieren, 2493
  - PE\_CMD, 2471
  - PE\_DS3\_Write\_ET200S, 2476
  - PE\_End\_RSP, 2496
  - PE\_Error\_RSP, 2493
  - PE\_Get\_Mode\_RSP, 2499
  - PE\_I\_DEV, 2489
  - PE\_Identify\_RSP, 2502

- PE\_List\_Modes\_RSP, 2497
- PE\_Measurement\_List\_RSP, 2504
- PE\_Measurement\_Value\_RSP, 2505
- PE\_PEM\_Status\_RSP, 2501
- PE\_Start\_End, 2467
- PE\_Start\_RSP, 2495
- PEM-Status als Antwort generieren, 2501
- PROFINET-Kommandos im I-Device steuern, 2489
- Schaltverhalten der Powermodule einstellen, 2476
- Unterstützte PROFINET-Kommandos als Antwort generieren, 2502
- Zustandsinformationen auslesen, 2471
- Profil, (Siehe Busprofil)
- Profile, 891
- PROFINET, 600, 3467
  - Gerätetausch ohne Wechselmedium, 817
  - HMI-Verbindung, 3498, 3499, 3547, 3548, 3602, 3603
  - Kopplung mit PROFIBUS, 793
  - S7 1200, 3547
  - S7 1500, 3498
  - S7 300/400, 3602
  - Verbindung, 3650
- PROFINET IO, 791, 796, 798, 807, 808
  - Port-Optionen, 814, 3508, 3559, 3612
- PROFINET IO-Device
  - Namensvergabe aus dem Projektkontext, 1004
  - Namensvergabe über "Erreichbare Teilnehmer", 1004
- PROFINET IO-Devices
  - alle Ausgänge schreiben mit SETIO, 2396
  - einen Teil der Ausgänge schreiben mit SETIO\_PART, 2398
  - einen Teil der Eingänge lesen mit GETIO\_PART, 2397
  - konsistente Daten lesen mit DPRD\_DAT, 2436
  - konsistente Daten schreiben mit DPWR\_DAT, 2438
- PROFINET IO-System, 808
- PROFINET RT-Klasse, 792
- PROFINET-Adresse, 521
- PROFINET-Gerätename, 802
- PROFINET-IO-Device
  - Namen vergeben, 1002
- PROFINET-Parameter
  - Bediengerät, 3502, 3552, 3606
  - HMI-Verbindung, 3501, 3550, 3605
  - S7 1200, 3550, 3553
  - S7 1500, 3501, 3502, 3504
  - S7 300/400, 3605, 3606, 3608
- Steuerung, 3504, 3553, 3608
- PROFINET-Schnittstelle, 805, 856, 977
- Program\_Alarm, 2547
- Programmbearbeitung
  - zyklisch, 885
- Programmiereditor
  - Allgemeine Einstellungen, 171, 173, 174
  - Allgemeine Einstellungen, 1256
  - Aufbau, 1236
  - Bausteinschnittstelle, 1237
  - Favoriten, 1237
  - Fläche des Programmierfensters vergrößern, 1243
  - Funktion, 1235
  - Funktionsleiste, 1237
  - Programmierfenster, 1237
  - Task Card Anweisungen, 1237
  - Task Card Testen, 1237
  - Tastaturbedienung, 1239
- Programmierfehler, 44
- Programmierfenster
  - anpassen, 1375
  - SCL, 1374
- Programmiersprache
  - FUP, 1323
  - KOP, 1279
  - Regeln zum Umschalten, 1402
  - umschalten, 1402
- Programmierung
  - Linear, 1026
  - Strukturiert, 1026
- Programminformationen
  - Ansichten, 1493
  - anzeigen, 1493
  - im Belegungsplan, 1494
  - im Register , 1515
  - in der Abhängigkeitsstruktur, 1509
  - in der Aufrufstruktur, 1502
- Programmmeldung
  - generieren mit Program\_Alarm, 2547
- Programmressourcen, 1211
- Programmstatus
  - ausschalten, 1527
  - Bausteine bearbeiten, 1528
  - einschalten, 1527
  - Funktion, 1525
  - FUP, 1531
  - KOP, 1530
  - SCL, 1532
  - Variable steuern, 1529
- Programmsteuerung, 1828, 2099, 2300



- Projekt
  - Daten vergleichen, 38, 261
  - Detailvergleich, 270
  - Eigenschaften anzeigen, 237
  - erstellen, 232
  - Grundlagen, 231
  - harmonisieren, 362
  - hochrüsten, 3931
  - laden, 3941, 3960
  - laden (Übersicht), 3936
  - löschen, 239
  - mehrere Verbindungen, 3526, 3579, 3637, 3678
  - mehrsprachig, 3888
  - migrieren, 118, 136
  - Offline/Offline-Vergleich durchführen, 262
  - Online/Offline-Vergleich durchführen, 262
  - Projektdaten übersetzen, 255
  - schließen, 238
  - speichern, 238
  - über Variablen-Simulator simulieren, 3945
  - übersetzen , 3938
  - übersetzen (Übersicht), 3936
- Projekt anpassen
  - für anderes Bediengerät, 3957
- Projekt drucken, 294
- Projekt öffnen, 233
- Projekt übertragen
  - Bediengerät, 72
  - Rezepturdatensatz, 72
- Projektansicht, 178
- Projektarchive, 249
- Projektbibliothek, 50
- Projektbibliothek, 314, 3839
  - aufräumen, 363
  - Elementansicht verwenden, 317
  - Filteransicht verwenden, 336
  - Konsistenzprüfung für eine Typ-Version durchführen, 347
  - migrieren, 130
  - Ordner erstellen, 333
  - Projekt aktualisieren, 353
  - Typen gemeinsame Version zuweisen, 351
  - Typen hinzufügen, 342
  - Typen verwenden, 343
  - Typ-Version verwerfen, 348
- Projektdaten
  - auf eine Memory Card laden, 258
  - Informationen zum Laden, 256
  - Informationen zum Übersetzen, 254
  - laden, 257
  - Meldung exportieren, 3858
  - Meldung importieren, 3859
- Rezeptur exportieren, 3854
- Textliste exportieren, 3871
- übersetzen, 255
- Variable exportieren, 3865
- von einem Gerät laden, 259
- Projekte migrieren
  - Voraussetzungen, 112
  - Vorgehen, 112
- Projektbibliothek
  - Typ-Version freigeben, 349
- projektieren
  - Berechtigung, 3327
  - Kurvenanzeige für Werte aus der Steuerung, 3201
  - Meldungen, 3588, 3689, 3701
  - Rezepturanzeige, 3293
  - Verbindungen, 3650, 3652, 3654, 3663, 3693
  - Zugriffsschutz, 3333
- Projektieren, 3219, 3228
  - Analogmeldung, 3219, 3226
  - Analogmeldungen, 3214
  - Bitmeldung, 3217, 3225
  - Bitmeldungen, 3214
  - Darstellung der Meldeanzeige, 3231
  - Ereignisgesteuerte Aufgaben, 3221
  - HART-Variable, 952
  - Meldeanzeige, 3230
  - Meldegruppe, 3217
  - Quittierung Meldung, 3234, 3235, 3236
  - Steuerungsmeldungen, 3214
  - Systemmeldungen, 3214
- Projektiert
  - Loop-In-Alarm, 3223
- Projektierung optimieren
  - Allen-Bradley DF1, 3720, 3742, 3760, 3771, 3787, 3799, 3816
- Projektierungsphase, 3959
- Projektierungsrechte, 526
- Projektkennung, 3448
- Projektnavigation
  - ausblenden, 1244
  - Automatisch reduzieren, 187
  - einblenden, 1244
  - Funktion, 181
  - Gerät hinzufügen, 399
- Projektsprache, 240, 3882
  - aktivieren, 3886
  - deaktivieren, 3886
- Projektsprachen
  - Anwendertexte, 3889
  - Editiersprache ändern, 243
  - Projektsprachen festlegen, 242

- Systemtexte, 3889
  - Verwendung, 240
- Projekttext
  - exportieren, 3875, 3893
- Projekttexte
  - Alle Texte exportieren, 246
  - Anwendungsbeispiel, 247
  - Einzelne Texte exportieren, 246
  - Einzelne Texte übersetzen, 3890
  - importieren, 247, 3877, 3895
  - migrieren, 128
  - Projekttexte übersetzen, 243
  - Referenztext anzeigen, 244, 3892
  - Texte von selektierten Objekten ändern, 244
  - Übersetzung in Projektsprachen, 240, 3889
- ProSave, 3950
  - Installation, 67
- Protokoll, 535
- Protokolldatei
  - Meldungen des Modems , 4396
- Prozessabbild, 847
  - Aktualisieren, 845
  - Ausgänge schreiben mit SETIO, 2396
  - der Ausgänge aktualisieren mit UPDAT\_PO, 2387
  - der Ausgänge synchronisieren mit SYNC\_PO, 2390
  - der Eingänge synchronisieren mit SYNC\_PI, 2388
  - Eingänge aktualisieren mit UPDAT\_PI, 2385
  - Eingänge lesen mit GETIO, 2395
  - Grundlagen, 844
  - Peripheriezugriffsfehler, 845
  - Prozessabbildbereich lesen mit GETIO\_PART, 2397
  - Prozessabbildbereich übertragen mit SETIO\_PART, 2398
- Prozessalarm
  - Funktionsweise, 852
- Prozessalarm-OB
  - Beschreibung, 894
  - parametrieren, 903
- Prozessdatenbereich
  - Baugruppenadressierung, 846
  - Baugruppen-Anfangsadresse, 846
- Prozessführungsphase, 3959
- Prüfen
  - Bediengeräteversion, 3932
- PRVREC, 2442
- PT, 1928
- PTC-Widerstand, 941
- PTO, 43

- Punkt-zu-Punkt-Kommunikation (PtP), 869
  - Freeport-Protokoll, 869
- Punkt-zu-Punkt-Kopplung
  - Allen-Bradley DF1, 3728
  - Ausgangssignale setzen mit SGN\_SET, 2845
  - Datenübertragung anstoßen mit SEND\_PTP, 2839
  - Empfang freigeben mit RCV\_PTP, 2842
  - Empfangsparameter projektieren mit RCV\_CFG, 2833
  - Empfangspuffer löschen mit RCV\_RST, 2843
  - Kommunikationsparameter projektieren mit PORT\_CFG, 2828
  - serielle Übertragungsparameter projektieren mit SEND\_CFG, 2831
  - Signalzustand abfragen mit SGN\_GET, 2844
- PUT, 2920
- PUT/GET, 884
- PWM, 43

## Q

- Q
  - Steuerung, 3755
- Q series, 3758
- Q0xUDEH CPU
  - Steuerung, 3757
- QRY\_CINT, 2521
- QRY\_DINT, 2532
- QRY\_TINT, 2527
- Quadrat, 1744, 2010, 2208
- Quadratwurzel, 1745, 2012, 2209
- Quantum, 3785
- Querverweis
  - Inspektorfenster, 55
- Querverweise
  - anzeigen, 1522, 3878, 3881
  - Einführung, 1519, 3878
  - Nutzen, 1519, 3878
- Querverweisliste, (Siehe Querverweise), (Siehe Querverweise)
  - Ansichten, 1520, 3879
  - anzeigen, 1522, 3878
  - Aufbau, 1520, 3879
  - Einstellungen, 1520, 3879
  - Spalten sortieren, 1520, 3879
  - Überblick, 1522, 3878
- Quitterkonzept
  - Meldung mit Einfachquittierung, 3211
  - Meldung ohne Quittierung, 3211
- QuittiereMeldung, 3382

quittieren  
 Meldung, 4006  
 Störmeldung, 4006  
 Quittieren, 3207, 3428  
 Quittierkonzept, 3211  
 Quittierung, 3210  
 projektieren, 3234, 3235, 3236  
 Quittierung projektieren  
 Störmeldung, 3537, 3589, 3648, 3690, 3702,  
 3746, 3774, 3803, 3819

## R

R, 426, 1630, 1892  
 R\_TRIG, 1645, 1906, 2155  
 Rack or station failure, 898  
 Radius, 3138  
 RADIUS, 759  
 Radius horizontal, 3134  
 Radius vertikal, 3134  
 Rahmen  
 Druckfläche festlegen, 288  
 Platzhalter für Dokument-Informationen, 290  
 Rahmen gestalten, 3058  
 Rahmen und Deckblätter aus der Bibliothek nutzen,  
 283  
 Rand erreicht, 3429  
 RCV\_CFG, 2833  
 RCV\_PTP, 2842  
 RCV\_RST, 2843  
 RCVREC, 2440  
 RD\_ADDR , 2627  
 RD\_DPAR, 2508  
 RD\_DPARA, 2511  
 RD\_DPARM, 2513  
 RD\_LOC\_T, 2357  
 RD\_REC, 2430  
 RD\_SINFO, 2552  
 RDREC, 2392  
 RE\_TRIGR, 1827, 2098, 2300  
 READ\_DBL, 45, 2614  
 READ\_ERR, 2541  
 ReadFromArrayDB, 1783, 2052, 2234, 2274  
 ReadFromArrayDBL, 1786, 2055, 2237  
 REAL, 1102, 1145, 1171  
 REAL\_TO\_, 1171  
 Rechteabhängigkeiten, 527  
 Rechteck  
 einfügen, 3076  
 einfügen und konfigurieren, 3075  
 konfigurieren, 3076  
 Radius Ecken X, 3148

Radius Ecken Y, 3148  
 Recipe, 638  
 RecipeExport, 2585  
 RecipelImport, 2586  
 Referenzen, 157  
 Referenzieren: Referenziermodi, 4137  
 Referenzkanal, 938  
 Referenzkanal des Moduls, 948  
 Referenzkanalfehler, 946  
 Referenzobjekt  
 festlegen, 3051  
 Referenzprojekt  
 Grundlagen, 252  
 öffnen, 252  
 schließen, 252  
 vergleichen, 253  
 Referenzsprache, 240, 3882  
 wählen, 3887  
 Referenztemperatur, 938, 947, 948  
 Regeln für die Netzkonfiguration von MPI-Netzen  
 Regeln für die MPI-Adressvergabe, 422  
 Regeln für die Netzkonfiguration von PROFIBUS-  
 Netzen  
 Teilnehmeradressen vergeben, 422  
 Register Speicherauslastung  
 Aufbau, 1517  
 Relaispule, 1628  
 negiert, 1629  
 Remanente Merker  
 Anzeige aktivieren, 1502  
 Remanenz, 844, 1034, 1036  
 Bausteinschnittstelle, 1273  
 Datenbaustein, 1422  
 Datenbaustein , 1423  
 Merker, Zeiten, Zähler, 1196  
 PLC-Variable, 1196, 1197  
 Remanenz von IP-Adressparametern, 805  
 Remanenzspeicher, 1515  
 Remanenzverhalten  
 Merker, Zeiten, Zähler, 1197  
 Remote Desktop, 30  
 Remote-Access-Benutzer, 524  
 REPEAT, 2288  
 REPLACE, 2383  
 Reservemodul  
 Ersatzwert, 785  
 RESET\_BF, 1633, 1895  
 ResetBit, 3388  
 ResetBitInTag, 3389  
 Ressource  
 SIMATIC S7 1200, 65

- Restore-Vorgang
    - Autorisierung, 90
    - License Key, 90, 91
  - RET, 1822, 2092
  - RETURN, 2293
  - Rezept
    - in CSV-Datei exportieren, 2585
    - in Datenbaustein importieren, 2586
  - Rezeptur, 3272, 3274, 3275, 3857, 4008
    - Datenfluss, 4011
    - Datensatz, 3275, 4008
    - Einsatz, 3274
    - Einsatzbeispiel: Maschinenparametrierung, 3274
    - Einsatzbeispiel:Chargenorientierte Produktion, 3274
    - Element, 3275, 4009
    - exportieren, 3854
    - Grundlagen, 3272, 3274
    - importieren, 3856
    - neu anlegen, 3286
    - Speicherbedarf, 4025
    - Verwendung von Textlisten, 3281
  - Rezeptur bedienen
    - Datensatz lesen, 3300
    - Datensatz übertragen, 3301
    - Rezepturdatensatz erstellen, 3299, 3300
  - Rezeptur importieren
    - Struktur für den Import, 3857
  - Rezepturanzeige, 3282
    - Aktualisierung, 3283
    - Darstellung, 3149
    - einfach, 3282
    - mit Funktionstaste bedienen, 3283
    - projektieren, 3293
    - Rezepturdatensatz, 3276, 3284
    - Symbolleiste, 3149
    - Verhalten bei Bildwechsel, 3283
    - Verwendung, 3149
  - RezepturanzeigeDatensatzNeu, 3383
  - RezepturanzeigeLeseDatensatzAusSteuerung, 3383
  - RezepturanzeigeLöscheDatensatz, 3384
  - RezepturanzeigeMenü, 3384
  - RezepturanzeigeÖffnen, 3385
  - RezepturanzeigeSchreibeDatensatzInSteuerung, 3385
  - RezepturanzeigeSpeichereDatensatz, 3386
  - RezepturanzeigeSpeichernUnterDatensatz, 3386
  - RezepturanzeigeUmbenennenDatensatz, 3387
  - RezepturanzeigeZeigeHilfetext, 3387
  - RezepturanzeigeZurück, 3388
- Rezepturdaten
    - Migration, 131
  - Rezepturdatensatz
    - am Bediengerät erstellen, 3299, 3300
    - aus Steuerung lesen, 4017
    - bearbeiten, 4015
    - erstellen, 4014
    - löschen, 4016
    - mit Steuerung abgleichen, 4015
    - neu anlegen, 3286
    - Projekt übertragen, 72
    - Verwendung von Textlisten, 3281
    - zur Steuerung übertragen, 4018
  - Rezepturelement
    - neu anlegen, 3286
  - Rezepturliste, 3282
  - Richtungsausgang und Fahrtrichtung:  
Zusammenhang, 4134
  - RIGHT, 2380
  - Ringredundanz, 721
  - ROL, 1859, 2131, 2325
  - Rollen, 524
    - benutzerdefiniert, 524
    - systemdefiniert, 524
  - Rollenname, 525
  - ROR, 1857, 2129, 2323
  - Rotieren
    - links, 1859, 2131, 2325
    - rechts, 1857, 2129, 2323
  - ROUND, 1800, 2069, 2262
  - Route anlegen, 595
  - Router-IP-Adresse, 596
  - Routing-Modus, 552
  - RS, 1636, 1897
  - RS232/PPI-Multi-Master-Kabel, 873
  - RT, 1665, 1927, 2170
  - RT-Klasse, 792
  - Ruckbegrenzung: Funktion, 4136
  - Rückgängig machen von Aktionen
    - Grundlagen zum Rückgängig machen, 300
    - letzte Aktion rückgängig machen, 302
    - mehrere Aktionen rückgängig machen, 302
  - Rücklesbarkeit von Verbindungsparametern, 479
  - Rückruf zu einer beim Verbindungsaufbau angegebenen Rufnummer, 4353
  - Rückrufmöglichkeiten in TeleService, 4353
  - RücksetzeBit, 3388
  - RücksetzeBitInVariable, 3389
  - Rücksetzen
    - Bitfeld, 1633, 1895
    - IEC-Zeit, 1665, 2170
    - Operand, 1630, 1636, 1892, 1897
  - Rücksetzen auf Werkseinstellungen, 3953
  - Ruhezustand, 31

RUN, 838  
 Runden, 1800, 1801, 1802, 1804, 2069, 2070, 2072, 2073, 2262, 2263, 2264  
 Runtime  
   Abarbeitung der Funktionsliste, 3346  
   Mausrad, 70  
   simulieren, 3943  
   Startbild, 3033  
 RUNTIME, 2308  
   Laufzeitmessung, 2308  
 Runtime-Daten  
   Migration, 131  
 Runtime-Einstellungen  
   Benutzerverwaltung, 3034, 3309, 3325  
 Runtime-Sprache, 3882, 3900, 3967  
   Archiv, 3906  
   auswählen, 3901, 3968  
   Reihenfolge bei Sprachumschaltung, 3903  
   Schriftart, 3905, 3969  
 Runtime-Start  
   Basic Panel, 3942, 3964  
 Runtime-Stopp, 3429  
 Runtime-Version, 3937

## S

S, 1631, 1893  
 S\_AVERZ, 1679, 1941  
 S\_COMP, 2364  
 S\_CONV, 2364, 2366  
 S\_CU, 2195  
 S\_EVERZ, 1674, 1936  
 S\_IMPULS, 1668, 1930  
 S\_ODT, 1674, 1936, 2179  
 S\_ODTS, 1677, 1939, 2181  
 S\_OFFDT, 1679, 1941, 2184  
 S\_PEXT, 1671, 1933, 2176  
 S\_PULSE, 2173  
 S\_SEVERZ, 1677, 1939  
 S\_VIMP, 1671, 1933  
 S5TIME, 1105  
 S7 1200  
   Basic Panel, 3494  
   HMI-Verbindung, 3548  
   Kommunikation, 3546  
   Parameter, 3596  
   PROFIBUS, 3562, 3563  
   PROFIBUS-Parameter, 3564, 3566, 3567  
   PROFINET, 3547  
   Verbindungsparameter, 3596  
 S7 1200 V2  
   Datentypen, 3592

S7 1500  
   Datentypen, 3540  
   HMI-Verbindung, 3499  
   Kommunikation, 3497  
   Parameter, 3541  
   PROFIBUS, 3511, 3513  
   PROFIBUS-Parameter, 3514, 3516, 3517  
   PROFINET, 3498  
   PROFINET-Parameter, 3501, 3502, 3504  
   Verbindungsparameter, 3541  
 S7 200  
   Datentypen, 3692  
   Kommunikation, 3663  
   MPI, 3661, 3671  
   Parameter, 3661, 3666, 3671, 3672  
   PPI, 3672  
   Verbindung, 3663, 3693  
   Verbindungsparameter, 3666  
 S7 300  
   Basic Panel, 3494  
   Kommunikation, 3601  
   MPI-Adresse, 3631  
 S7 300/400  
   Datentypen, 3650  
   HMI-Verbindung, 3603  
   MPI, 3623, 3624  
   MPI-Parameter, 3626, 3628, 3629  
   Parameter, 3656  
   PROFIBUS, 3615, 3616  
   PROFIBUS-Parameter, 3617, 3619, 3621  
   PROFINET, 3602  
   PROFINET-Parameter, 3605, 3606, 3608  
   Verbindungsparameter, 3656  
 S7 400  
   Basic Panel, 3494  
   Kommunikation, 3601  
   MPI-Adresse, 3631  
 S7-1200, 39  
 S7-CP, 511  
 S7-CPU  
   Arbeitsspeicher, 842  
   Ladespeicher, 842  
   Operandenbereich, 842  
 S7-Kommunikation  
   Daten an eine remote Partner-Anweisung mit USEND, 2923  
   Daten an eine remote Partner-Anweisung senden mit BSEND, 2926  
   Daten von einer remoten Partner-Anweisung empfangen mit BRCV, 2929  
   Daten von einer remoten Partner-Anweisung empfangen mit URCV, 2925

- Nutzdatengröße, 2916
- Parameter, 2914
- Sende- und Empfangsparameter, 2916
- S7-PCT, 824
- S7-Routing
  - Laden, 3940
- S7-Verbindung, 433
  - TSAP, 444
- SA, 1688, 1951
- SA-Lebensdauer, 568
- SCALANCE XR500, "Übersetzen"/"Laden", 42
- SCALANCE S, 511
- SCALE, 1810, 2080, 2275
- SCALE\_X, 1805, 2075, 2267
- Schachtelungstief, 1040
- Schalter, 3150, 3990
  - Bedienung, 3990
  - Darstellung, 3990
  - Typ, 3150
  - Verhalten, 3990
  - Verwendung, 3990
- Schaltfläche, 3151, 3989
  - Bedienung, 3989
  - Darstellung, 3989
  - Grafik, 3152
  - Hotkey festlegen, 3152
  - Modus, 3152
  - projektieren, 3333
  - Text, 3152
  - Verhalten, 3989
  - Verwendung, 3989
  - Zugriffsschutz projektieren, 3329
- SchaltflächeDrücken, 3391
- SchaltflächeLoslassen, 3391
- Schieben
  - links, 1855, 2127, 2321
  - rechts, 1853, 2124, 2319
- Schleife, 2280, 2284, 2287, 2288, 2290, 2291
- Schließerkontakt, 1626
- Schlüsselwörter, 1059
- Schlüsselw orthervorhebung, 1372
- Schneller Zähler
  - Allgemeines, 863
  - Funktionsweise, 864
  - Konfigurieren, 866
- Schnittstelle, (Siehe Bausteinschnittstelle)
  - anzeigen, 977
  - Basic Panel, 3495
  - hinzufügen, 4282
  - umbenennen, 507
- Schnittstelleneigenschaften, 977
- Schreiben
  - eines Datensatzes mit WR\_REC, 2435
- Schreibschutz, 880
- Schriftart, -größe und farbe einstellen , 1375
- Schrifteinstellungen
  - migrieren, 123
- Schriftgröße
  - asiatische Sprachen , 3971
- Schrittmotor, 4128
- Schrittschaltwerk, 1861, 2133, 2327
- schützen
  - Funktionstaste mit Kennwort, 3115
- Schutzkonzept, 882, 884
  - Einführung, 274
  - Zugriffsrechte zurücknehmen, 275
- Schutzstufe, 880, 881
  - Zugriffsrechte zurücknehmen, 275
- SCL, 1363, 1614, 1616
  - Kommentar einfügen, 2294
  - Programmierfenster, 1374
- SCL-Anweisung
  - Datentyp ändern, 1384, 1385
  - einfügen, 1381, 1382
  - Grundlagen zu den Datentypen, 1383
  - Regeln, 1380
  - selektieren, 1399
- SCL-Programm
  - Anweisung einfügen, 1381, 1382
  - Anweisung selektieren, 1399
  - Baustein aufruf einfügen, 1388, 1389, 1391, 1392, 1393, 1394
  - Kommentare einfügen, 1397
  - Parameterliste anzeigen und ausblenden, 1397
  - Programmstatusanzeige, 1532
- SD-Karte
  - offener Force-Auftrag, 31
- SD-Karten, 30
- SE, 1683, 1685, 1947
- Security Configuration Tool
  - Projektierungssichten, 512
- Security-Modul, 511
- SEG, 1880, 2151, 2345
- Segmentdiagnose
  - grafische Anzeige, 1016
  - Symbole, 1016
  - textuelle Anzeige, 1016
- SeiteAbwärts, 3395
- SeiteAufwärts, 3394
- SEL, 1845, 2116, 2312
- Selektieren, 1845, 2116, 2312
- SEND\_CFG, 2831
- SEND\_PTP, 2839

- Sendetakt, 811
- Separator-Zeile einfügen, 4178
- Separator-Zeile hinzufügen, 4178
- Service & Support, 384, 3466
  - aktivieren, 384
- Servicedaten
  - speichern, 991
- Servomotor, 4128
- SET\_BF, 1632, 1894
- SET\_CINT, 2519
- SET\_TIMEZONE, 2359
- SET\_TINT, 2522
- SET\_TINTL, 2524
- SetBit, 3396
- SetBitInTag, 3397
- SetConnectionMode, 3404
- SetDeviceMode, 3395
- SETIO, 2396
- SETIO\_PART, 2398
- SetLanguage, 3402
- SetPLCMode, 3376, 3400, 3401
- SetzeBetriebsartGerät, 3395
- SetzeBit, 3396
- SetzeBitInVariable, 3397
- SetzeBitWährendTasteGedrückt, 3398
- SetzeFarbeHintergrundbeleuchtung, 3400
- Setzen
  - Bitfeld, 1632, 1894
  - Operand, 1631, 1634, 1893, 1896
  - Zähleranfangswert, 1703, 1968
- SetzePLCModus, 3376, 3400, 3401
- SetzeSprache, 3402
- SetzeVariable, 3403
- SetzeVerbindungsmodus, 3404
- Severity, 552
- SGN\_GET, 2844
- SGN\_SET, 2845
- SHA1, 568, 616, 626
- SHA-Algorithmus, 694
- Shared Device
  - zugehörige IO-Controller, 979
- SHL, 1855, 2127, 2321
- ShowAlarmWindow, 3414
- SHR, 1853, 2124, 2319
- SI, 1682, 1944
- Sicherheitseinstellungen, 693
- Sicherheits-Ereignisse, 584, 586
- Sicherheitshinweis
  - Rezepturdatensatz im Hintergrund ändern, 3283
  - Tooltip wechseln, 3979
- Sicherheitshinweise, 227
- Sicherheitsprogramm, 83, 86
- Sicherheitssystem, 3992
- Sicherheitsvorkehrungen beim Forcen von Variablen, 1562, 1578
- sichern, 3993
  - Daten des Bediengeräts, 3950, 3952, 3966
- Sicherung von online Gerät laden, 4277
- Sicherung wiederherstellen, 4277
- SiClock, 539
- Sieben-Segment-Anzeige, 1880, 2151, 2345
- Signalboard, 632
  - stecken, 633
- SIMATIC ISB Treiber, 35
- SIMATIC LOGO
  - Datentypen, 3704
- SIMATIC LOGO!
  - Kommunikation, 3692
  - Parameter, 3695
  - Verbindungsparameter, 3695
- SIMATIC Memory Card
  - formatieren, 1000
- SIMATIC S7
  - Kommunikationspartner, 3546, 3601, 3660, 3670, 3673
- SIMATIC S7 1200
  - Ressource, 65
  - Verbindungsressource, 65
- SIMATIC S7 200
  - Migration der Datentypen, 142
- SIMATIC S7 300/400
  - Migration der Datentypen, 142
- SIMATIC S7 Embedded Controller, 40
- Simulation, 71
  - Runtime, 3943
  - Steuerungsanbindung, 72
- Simulation starten, 366
- simulieren
  - Projekt über Variablen-Simulator, 3945
- SimuliereSystemtaste, 3406
- SimuliereVariable, 3407
- SIN, 1749, 2016, 2213
- SINT, 1095, 1139, 1161
- SINT\_TO\_, 1161
- Sinus, 1749, 2016, 2213
- skalieren
  - Variable linear skalieren, 3183, 3184
- Skalieren, 1805, 1810, 2075, 2080, 2267, 2275
- Skript
  - Variablenwert aktualisieren, 3163, 3483
- SLC, 3736
- Slice-Zugriff, 1072
- Slicing, 157

- Smartdrive
  - Installation, 51
- SMC, 1874, 2146, 2340
- SMS empfangen, 2810
- SMS senden, 2806
- SMS-Text, 2808, 2812, 2819
- SMTP-Client, 691
- SNC\_RTCB, 2361
- SNMP, 526, 684, 693
  - Benutzer, 694
  - Gruppen, 693
  - SNMP-Trap, 693
- SNMPv1, 616, 626
- SNMPv3, 616, 626
- Softkey, 3108, 3112
  - global, 3111
- SOFTNET Security Client
  - Datenbasis, 588
  - Einsatz des, 587
  - im Projekt konfigurieren, 587
  - Konfigurationsdatei erstellen, 587
- SOFTNET Security Client, 511
- Software
  - Bediengerät, 3949
- Software simulieren, 365
- Softwaregesteuerte Datenflusskontrolle, 872
- Software-Regler
  - Projektieren, 4052
- Software-Voraussetzungen auf Anlagen-Seite
  - Kommunikationsanweisung "AS\_MAIL", 4389
  - Kommunikationsanweisung "PG\_DIAL", 4385
  - Kommunikationsanweisung "SMS\_SEND", 4388
- Software-Voraussetzungen auf PG/PC-Seite
  - Kommunikationsanweisung "PG\_DIAL", 4385
- Software-Voraussetzungen für AS-AS-Fernkopplung
  - Kommunikationsanweisung "AS\_DIAL", 4387
- Sonderzeichen
  - Meldetext, 61
- Spalte
  - Meldeanzeige, 3143
- Spalten ein- und ausblenden, 1210, 1278, 1430, 1445
- Speedy Splitter, 369, 372, 375, 385
- Speicheradresse lesen, 2241
- Speicheradresse schreiben, 2245
- Speicherarten einer CPU anzeigen, 983, 984
- Speicherauslastung, 1515
  - Arbeitsspeicher, 1515
  - Code-Arbeitsspeicher, 1515
  - Daten-Arbeitsspeicher, 1515
  - Einführung, 1515
  - Ladespeicher, 1515
  - Remanenzdaten, 1515
  - Remanenzspeicher, 1515
- Speicherauslastung der CPU
  - anzeigen, 1518
- Speicherbedarf
  - Rezeptur, 4025
- Speicherbereich
  - Arbeitsspeicher, 842
  - Ladespeicher, 842
  - Remanente Speicherbereiche, 844
- Speicherbereich schreiben, 2249
- Speicherbit lesen, 2244
- Speicherbit schreiben, 2247
- Speicherkarte, (Siehe Memory Card)
- speichern
  - globale Bibliothek, 3850
- Speichern einer Beobachtungstabelle, 1539
- Speichern einer Forcetabelle, 1566
- Speichern von Telefonbüchern, 4341
- Speicherplatzwarnung, 31
- Speicherreserve, 43
- Spezialvariable (Webserver), 648
- spiegeln
  - Objekt, 3043
- Sprache
  - Archiv, 3906
  - asiatische Sprachen, 3884
  - asiatisches Betriebssystem, 3884
  - Editiersprache, 3887
  - Grafik sprachabhängig, 3896
  - mehrsprachiges Projekt, 3888
  - Projektsprache aktivieren, 3886
  - Projektsprache deaktivieren, 3886
  - Referenzsprache, 3887
  - sprachabhängige Darstellung, 3883
  - sprachabhängige Darstellung bei Datum, Uhrzeit, Geldbeträgen und Zahlen, 3883
  - Sprachunterstützung, 3884
- Sprache einstellen, 3979
- Sprachen
  - migrieren, 128
- Sprachkürzel, 662
- Sprachumschaltung, 662, 3900, 3970
  - Archiv, 3906
  - Runtime-Sprache, 3903
- Sprachverhalten
  - Bildschirmtastatur, 69
- Springen, 1814, 1815, 1816, 1817, 1819, 1822, 2085, 2086, 2087, 2088, 2089, 2092
- Sprungliste, 1817, 2088
- Sprungmarke, 1816, 2087, 2292
- Sprungverteiler, 1819, 2089



- SQR, 2208  
 SQR , 1744, 2010  
 SQRT, 1745, 2012, 2209  
 SR, 1634, 1896  
 SRT\_DINT, 2530  
 SS, 1686, 1949  
 SSH-Server, 683  
 SSL-Zertifikat, 521  
 Stammzertifizierungsstellen, 520  
 Standard  
     PROFIBUS, 3569, 3622  
 Standard-Benutzer, 524  
 Standard-Router, 595  
 Standardschriftart, 3905, 3969  
 Standby Modus, 31  
 ständig lesen  
     Variable, 3182  
 Startbild, 3033  
 Startinformation, 1028  
 Startup, 885  
 Startwert  
     Variable, 3180, 3181  
 Stateful Packet Inspection, 530  
 Station  
     Informationen auslesen mit GetStationInfo, 2567  
     kopieren, 405  
     löschen, 404  
     umbenennen, 507  
     verschieben, 407  
 Stationsname, (Siehe Geräte name)  
 Status, 890  
     Baugruppe gestört, 986  
 Status der Online-Verbindung, 1012  
 Statusalarm, 890  
 Statusalarm-OB, 890  
 Steckplatz  
     Baugruppenträger, 393  
     logische Adresse ermitteln mit LGC\_GADR, 2634  
     logische Adresse ermitteln mit LOG\_GEO, 2630  
     selektieren, 394  
 Steckplatzregeln, 390  
     ET 200S, 779  
 stempeln, 3058  
 Steuermodus, 1548  
 Steuern  
     Schnelle Zähler, 2670  
 Steuerung  
     anschließen, 3710, 3752, 3766, 3781, 3809  
     MPI-Parameter, 3629  
     PROFIBUS-Parameter, 3517, 3567, 3621  
     PROFINET-Parameter, 3504, 3553, 3608  
     Rezepturdatensatz auslesen, 4017  
     Rezepturdatensatz übertragen, 4018  
     Variable, 3185  
 Steuerungsauftrag, 3448  
     Daten übertragen, 3531, 3584, 3643, 3683, 3834  
 Steuerungsmeldungen  
     Projektieren, 3214  
 Steuerungszeitalarm, 1865, 1868, 2136, 2139, 2330, 2333  
 Steuerwert  
     zugelassene Operanden, 1541  
 STOP, 839  
 StoppeRuntime, 3408  
 StopRuntime, 3408  
 Störfrequenzunterdrückung, 946  
 Störmeldung  
     quittieren, 4006  
     Quittierung am Bediengerät, 3538, 3590, 3649, 3691, 3703, 3746, 3775, 3804, 3820  
     Quittierung projektieren, 3537, 3589, 3648, 3690, 3702, 3746, 3774, 3803, 3819  
     Quittierung von der Steuerung, 3538, 3590, 3648, 3690, 3702, 3746, 3775, 3803, 3820  
 STP, 1828, 2099, 2300  
 Strg\_TO\_Chars, 2372  
 STRG\_VAL, 2367  
 String  
     Variablen vergleichen mit S\_COMP, 2364  
 STRING, 1081, 1112, 1151, 1179  
     Adressieren, 1071  
 STRING\_TO\_, 1179  
 Strompfad  
     einfügen, 1323  
     löschen, 1323  
     Verwendung, 1321  
 STRUCT  
     Adressieren, 1071  
     Aufbau, 1119  
     Deklaration in globalen Datenbausteinen, 1417  
     Deklarieren in der Bausteinschnittstelle, 1266  
     Deklarieren in einem PLC-Datentyp, 1440  
 Struktur  
     STRUCT, 1119  
 Strukturierte Programmierung, 1026  
 SUB, 1728, 1993  
 Subnetzmaske, 798  
 Subtrahieren, 1728, 1993  
     Zeiten subtrahieren mit T\_SUB, 2353  
 Suche  
     Hardware-Katalog, 391  
 Suchen und Ersetzen, 304  
     Ersetzen von Suchbegriffen, 306  
     Suche starten, 305

- Suchfunktion verwenden, 304
- Zusatzoptionen für die Suche, 304
- Support Package installieren, 97
- SV, 1683, 1945
- SWAP, 1781, 2050, 2232
- SWITCH, 1819, 2089
- Switchport, 815, 3509, 3560, 3613
- Symbol
  - für Betriebszustand, 972
  - für Hardware-Diagnose, 969
  - für Software-Diagnose, 971
  - für Verbindungsstatus, 1019
  - für Vergleich, 971
  - für Vergleichsstatus, 1019
  - für Wertevergleich, 4059
  - Meldeklassen, 3229
  - Overlay-Icon, 972
- Symbol zuordnen, 631
- Symbole
  - im Belegungsplan, 1496
  - in der Abhängigkeitsstruktur, 1511
  - in der Aufrufstruktur, 1504
  - in TeleService, 4339
- Symbole in der Beobachtungstabelle, 1536
- Symbole in der Forcetabelle, 1564
- Symbolik, 155
- Symbolische Adressierung, 157
  - einer Variable, 3165, 3485
- Symbolische Konstante, 1058
- Symbolische Programmierung
  - Einblenden absoluter Adressen, 1246
- Symbolisches E/A-Feld, 3153
  - Modus, 3153
  - Textliste, 3085, 3154
- symbolisches EA-Feld, 3991
  - Bedienung, 3991
  - Verhalten, 3992
  - Verwendung, 3991
- Symbolleiste
  - Reihenfolge, 3043, 3054
- SYNC\_PI, 2388
- SYNC\_PO, 2390
- Sync-Domain, 816, 979, 3510, 3561, 3614
- Sync-Domain-Eigenschaften, 979
- synchron
  - Daten übertragen, 3530, 3641, 3687, 3832
- Synchronfehlerereignisse
  - abfragen aus Ereignisstatusregister mit READ\_ERR, 2541
  - demaskieren mit DMSK\_FLT, 2541
  - maskieren mit MSK\_FLT, 2540
- Synchronisation
  - Uhrzeitslaves mit SNC\_RTCB, 2361
  - von DP-Slaves mit DP\_SYC\_FR, 2445
- Synchronisierung (anwenderdefinierte Webseiten), 640
- Syntax
  - Adressierung, 3715
- Syntax für AWP-Kommandos, 644
- Syntaxfehler
  - Fehler finden, 1401
  - Grundlagen, 1400
- Syslog-Client, 699
- Syslog-Server, 577
- System
  - Allgemeine Informationen, 685
  - Systemkonfiguration, 683
- Systembaustein
  - Ordner Systembausteine, 1211
- Systemdefinierte Rolle
  - administrator, 524
  - diagnostics, 524
  - remote access, 524
  - standard, 524
- Systemdefinierte Textlisten
  - bearbeiten, 309
  - Texte ändern, 310
- Systemdiagnose, 3268
  - Detailansicht, 3268
  - Diagnosepuffer-Ansicht, 3269
  - Geräteansicht, 3268
  - System-Diagnoseanzeige, 3268
- System-Diagnoseanzeige
  - Detailansicht, 3155
  - Diagnosepuffer-Ansicht, 3155
  - Geräteansicht, 3155
  - projektieren, 3271
- System-Ereignisse, 583, 585
- Systemfunktion
  - Einsatz, 3341
  - Einsatzmöglichkeit, 3341
  - Funktionsliste, 3343
  - in der Funktionsliste, 3341
  - Verwendung, 3340
- Systemfunktionen, 3339
  - Abmelden, 3352
  - AktiviereBild, 3354
  - AktiviereBildMitNummer, 3355
  - AktivierePutzbild, 3356
  - AktiviereVorherigesBild, 3357
  - AktualisiereVariable, 3358
  - ÄndereKontrast, 3353
  - Anmelden, 3358

- BearbeiteMeldung, 3359
  - BildobjektCursorAbwärts, 3360
  - BildobjektCursorAufwärts, 3360
  - BildobjektSeiteAbwärts, 3361
  - BildobjektSeiteAufwärts, 3361
  - ErhöheFokussiertenWert, 3362
  - ErhöheVariable, 3362
  - GeheZumAnfang, 3363
  - GeheZumEnde, 3364
  - InvertiereBit, 3365
  - InvertiereBitInVariable, 3366
  - InvertiereLineareSkalierung, 3367
  - KalibriereTouchscreen, 3368
  - KurvenanzeigeBlättereVorwärts, 3369
  - KurvenanzeigeBlättereZurück, 3370
  - KurvenanzeigeErweitern, 3370
  - KurvenanzeigeKomprimieren, 3371
  - KurvenanzeigeLinealLinks, 3372
  - KurvenanzeigeLinealRechts, 3371
  - KurvenanzeigeSetzeLinealmodus, 3372
  - KurvenanzeigeStartenStoppen, 3373
  - KurvenanzeigeZurückZumAnfang, 3373
  - LeseBenutzername, 3374
  - LeseGruppennummer, 3375
  - LeseKennwort, 3375
  - LineareSkalierung, 3377
  - LöscheMeldepuffer , 3378
  - LöscheMeldepufferProTool , 3379
  - MeldeanzeigeBearbeiteMeldung, 3380
  - MeldeanzeigeQuittiereMeldung, 3381
  - MeldeanzeigeZeigeHilfetext, 3381
  - QuittiereMeldung, 3382
  - RezepturanzeigeDatensatzNeu, 3383
  - RezepturanzeigeLeseDatensatzAusSteuerung, 3383
  - RezepturanzeigeLöscheDatensatz, 3384
  - RezepturanzeigeMenü, 3384
  - RezepturanzeigeÖffnen, 3385
  - RezepturanzeigeSchreibeDatensatzInSteuerung, 3385
  - RezepturanzeigeSpeichereDatensatz, 3386
  - RezepturanzeigeSpeichernUnterDatensatz, 3386
  - RezepturanzeigeUmbenennenDatensatz, 3387
  - RezepturanzeigeZeigeHilfetext, 3387
  - RezepturanzeigeZurück, 3388
  - RücksetzeBit, 3388
  - RücksetzeBitInVariable, 3389
  - SchaltflächeDrücken, 3391
  - SchaltflächeLoslassen, 3391
  - SeiteAbwärts, 3395
  - SeiteAufwärts, 3394
  - SetzeBetriebsartGerät, 3395
  - SetzeBit, 3396
  - SetzeBitInVariable, 3397
  - SetzeBitWährendTasteGedrückt, 3398
  - SetzeSprache, 3402
  - SetzeVariable, 3403
  - SetzeVerbindungsmodus, 3404
  - SimuliereSystemtaste, 3406
  - SimuliereVariable, 3407
  - StoppeRuntime, 3408
  - VerfolgeBenutzeränderung, 3409
  - verfügbare, 3349
  - VerringereFokussiertenWert, 3409
  - VerringereVariable, 3410
  - WechseleVerbindung, 3411
  - ZeigeAnmeldedialog, 3412
  - ZeigeHilfetext, 3413
  - ZeigeMeldefenster, 3414
  - Systemgrenzen
    - Bediengerät, 4020
  - System-Log, 583, 585
  - Systemmeldung, 3205
    - Bedeutung, 3244, 3245, 3248, 3249, 3252, 3254, 3255, 3256, 3257, 3260, 3262, 3263
    - Parameter, 3243
  - Systemmeldungen
    - projektieren, 3214
  - Systemmerker, 856, 879
  - Systemspeicher
    - Diagnosepuffer, 846, 1009
    - Operandenbereiche, 842
    - Prozessabbild der Ein- und Ausgänge, 844
  - Systemstromversorgung, 944
  - Systemtexte, 241
  - Systemvoraussetzungen STEP 7 Basic, 80
  - Systemvoraussetzungen WinCC Basic, 84
  - Systemzeit
    - NTP-Client, 696
    - PTP-Client, 697
    - SNTP-Client, 696
  - Systemzeit anzeigen, 3131
  - Systemzeit lesen mit TIME\_TCK, 2362
  - SZ, 1703, 1968
- T**
- T\_ADD, 2351
  - T\_COMBINE, 2355
  - T\_COMP, 2349
  - T\_CONFIG, 44, 3008
  - T\_CONV, 2350
  - T\_DIAG, 3002
  - T\_DIFF, 2354

- T\_RESET, 3000
- T\_SUB, 2353
- Tabelle
  - Verbindung, 3471
- Tabellenanordnung sichern, 203
- Tabulatorweite einstellen , 1375
- Taktmerker, 856, 879
- Taktsynchronität, 825
- Tan, 2215
- TAN, 1752, 2019
- Tangens, 1752, 2019, 2215
- Task Card, 3028
  - Automatisch reduzieren, 187
  - Funktion, 194
  - Hardware-Katalog, 382, 3464
  - Online-Tools, 974
  - Palettenmodus ändern, 196
  - Werkzeuge, 3040
- Tastaturbedienung, 385
  - Anpassen von Editoren, 212
  - AWL, 1239
  - Editieren von Texten, 215, 1239
  - Fenster, 209
  - FUP, 1239
  - GRAPH, 1239
  - Grundfunktionen des TIA-Portals, 208
  - KOP, 1239
  - Navigation im TIA-Portal, 211
  - Objekte bearbeiten, 214
  - Objekte selektieren, 213
  - Online-Funktionen, 217
  - Programmieditor, 1239
  - Projektbearbeitung, 209
  - SCL, 1239
  - Tabellen, 216
- Taste drücken, 3430
- Taste loslassen, 3430
- Tasten-Bedienung
  - Datum-Uhrzeit-Feld, 3982
  - EA-Feld, 3984
- Tauschen
  - Baugruppe, 408
- TCI (Tool Calling Interface), 822
- TCON, 2968, 2971
- TCON\_IP\_RFC, 477
- TCON\_IP\_v4, 475
- TCON\_Param, 473
- TCP, 535
  - Merkmale, 471
  - Port-Nummern, 478
- TCP/IP Einstellungen zurücksetzen, 4287
- TDISCON, 2976
- Technologieobjekt Achse und Auftragstabelle: Liste der ErrorIDs und ErrorInfos, 4233
- Technologieobjekt Achse: Ablauf aktives Referenzieren, 4168
- Technologieobjekt Achse: Ablauf passives Referenzieren, 4166
- Technologieobjekt Achse: Aktualisierung der Variablen, 4256
- Technologieobjekt Achse: Erweiterte Parameter, 4146
- Technologieobjekt Achse: Grundparameter, 4146
- Technologieobjekt Achse: Hardware- und Softwarekomponenten, 4141
- Technologieobjekt Achse: Konfiguration Achsname, 4147
- Technologieobjekt Achse: Konfiguration Anfahr- / Referenzierrichtung, 4165
- Technologieobjekt Achse: Konfiguration Antrieb Bereit, 4150
- Technologieobjekt Achse: Konfiguration Antriebsfreigabe, 4150
- Technologieobjekt Achse: Konfiguration Antriebssignale, 4150
- Technologieobjekt Achse: Konfiguration Anwendereinheit, 4149
- Technologieobjekt Achse: Konfiguration Beschleunigung, 4156
- Technologieobjekt Achse: Konfiguration Dynamik Allgemein, 4156
- Technologieobjekt Achse: Konfiguration Eingang Referenzpunktschalter, 4164
- Technologieobjekt Achse: Konfiguration Einheit der Geschwindigkeitsbegrenzung, 4156
- Technologieobjekt Achse: Konfiguration Erlaube Richtungsumkehr am HW-Endschalter, 4165
- Technologieobjekt Achse: Konfiguration Hardware - Schnittstelle, 4147
- Technologieobjekt Achse: Konfiguration Hochlaufzeit, 4156
- Technologieobjekt Achse: Konfiguration Impulse pro Motorumdrehung, 4150
- Technologieobjekt Achse: Konfiguration Maximale Geschwindigkeit / Start/Stop-Geschwindigkeit, 4156
- Technologieobjekt Achse: Konfiguration Mechanik, 4150
- Technologieobjekt Achse: Konfiguration Notstopp-Verzögerung, 4159
- Technologieobjekt Achse: Konfiguration PTO und HSC, 4148
- Technologieobjekt Achse: Konfiguration Referenzieren - Passiv, 4164

- Technologieobjekt Achse: Konfiguration Referenzieren Aktiv, 4165
- Technologieobjekt Achse: Konfiguration Referenziergeschwindigkeit, 4166
- Technologieobjekt Achse: Konfiguration Referenzpunktposition, 4164, 4166
- Technologieobjekt Achse: Konfiguration Referenzpunktverschiebung, 4166
- Technologieobjekt Achse: Konfiguration Richtungssinn invertieren, 4151
- Technologieobjekt Achse: Konfiguration Ruckbegrenzung, 4157
- Technologieobjekt Achse: Konfiguration Rücklaufzeit, 4156
- Technologieobjekt Achse: Konfiguration Seite des Referenzpunktschalters, 4164, 4165
- Technologieobjekt Achse: Konfiguration Verrundungszeit, 4158
- Technologieobjekt Achse: Konfiguration Verzögerung, 4156
- Technologieobjekt Achse: Konfiguration Weg pro Motorumdrehung, 4150
- Technologieobjekt Achse: Konfigurationsparameter Dynamik im Anwenderprogramm ändern, 4162
- Technologieobjekt Achse: Konfigurationsparameter Referenzieren im Anwenderprogramm ändern, 4169
- Technologieobjekt Achse: Neues Objekt hinzufügen, 4146
- Technologieobjekt Achse: Symbole des Konfigurationsfensters, 4147
- Technologieobjekt Achse: Übersicht Diagnose, 4145
- Technologieobjekt Achse: Übersicht Inbetriebnahme, 4145
- Technologieobjekt Achse: Übersicht Konfiguration, 4144
- Technologieobjekt Achse: Variable Config.DriveInterface, 4243
- Technologieobjekt Achse: Variable Config.DynamicDefaults, 4245
- Technologieobjekt Achse: Variable Config.DynamicLimits, 4245
- Technologieobjekt Achse: Variable Config.General, 4242
- Technologieobjekt Achse: Variable Config.Homing, 4249
- Technologieobjekt Achse: Variable Config.Mechanics, 4244
- Technologieobjekt Achse: Variable Config.PositionLimits\_HW, 4248
- Technologieobjekt Achse: Variable Config.PositionLimits\_SW, 4247
- Technologieobjekt Achse: Variable ErrorBits, 4255
- Technologieobjekt Achse: Variable MotionStatus, 4251
- Technologieobjekt Achse: Variable StatusBits, 4252
- Technologieobjekt Achse: Verhalten bei aktivierter Ruckbegrenzung, 4161
- Technologieobjekt Achse: Werkzeuge, 4143
- Technologieobjekt Auftragstabelle: Erweiterte Parameter, 4173
- Technologieobjekt Auftragstabelle: Grundparameter, 4173
- Technologieobjekt Auftragstabelle: Konfiguration Aktiviere Warnungen, 4174
- Technologieobjekt Auftragstabelle: Konfiguration Allgemein, 4173
- Technologieobjekt Auftragstabelle: Konfiguration Auftragstabelle, 4174
- Technologieobjekt Auftragstabelle: Konfiguration Auftragstyp, 4174
- Technologieobjekt Auftragstabelle: Konfiguration Dauer, 4176
- Technologieobjekt Auftragstabelle: Konfiguration Geschwindigkeit, 4176
- Technologieobjekt Auftragstabelle: Konfiguration Nächster Schritt, 4177
- Technologieobjekt Auftragstabelle: Konfiguration Position / Fahrweg, 4175
- Technologieobjekt Auftragstabelle: Konfiguration Schrittcode, 4177
- Technologieobjekt Auftragstabelle: Konfiguration Verwende Achsparameter von, 4174
- Technologieobjekt Auftragstabelle: Kontextmenübefehle, 4177
- Technologieobjekt Auftragstabelle: Neues Objekt hinzufügen, 4172
- Technologieobjekt Auftragstabelle: Symbole des Konfigurationsfensters, 4173
- Technologieobjekt Auftragstabelle: Verwendung, 4170
- Technologieobjekt Auftragstabelle: Werkzeuge, 4171
- Technologieobjekte  
     PID\_3Step, 4094  
     PID\_Compact, 4061
- Teilnehmer mit unbekannter IP-Adresse, 570
- Teilnehmeradresse, 628
- TELECONTROL SERVER BASIC, 666
- Telecontrol-Server, 666
- Telefonbuch bei TeleService, 4337
- TeleService  
     Bedeutung der Symbole, 4339  
     Eigenschaften des Telefonbuchs, 4337  
     Arbeiten mit dem Telefonbuch, 4337

- AS\_DIAL, 4385
- Aufbau des Telefonbuches, 4338
- Definieren von Wählregeln, 4344
- Einsatz des TS Adapter IE Advanced, 32
- E-Mail übertragen mit TM\_MAIL, 3016
- Funktionalität, 4336
- Kommunikationsanweisung:"AS\_DIAL", 4336
- Kommunikationsanweisung:"AS\_MAIL", 4336
- Kommunikationsanweisung:"PG\_DIAL", 4336
- Kommunikationsanweisung:"SMS\_SEND", 4336
- Kommunikationsanweisung:"TM\_MAIL", 4336
- Kommunikationsanweisung:"TMAIL\_C", 4336
- Leistungsfähigkeit in Telefonnetzen, 4348
- Modemtypen/Medien, 4347
- Modem-Unterstützung, 4347
- Möglichkeiten zum
  - Verbindungsaufbau:S7-1200er CPUs, 4383
  - Möglichkeiten zum
    - Verbindungsaufbau:S7-1500er CPUs, 4383
    - Möglichkeiten zum
      - Verbindungsaufbau:S7-300/400er CPUs, 4383
  - Netzübergänge, 4347
  - Rückrufvarianten, 4351
  - Spalten im Telefonbuch ein- oder ausblenden, 4340
  - Telefonbuch, 4337
  - Telefonbuch drucken, 4343
  - Telefonbuch exportieren, 4343
  - Telefonbuch importieren, 4341
  - Telefonbuch öffnen, 4340
  - Telefonbuch speichern, 4341
  - Verbindung zur AS aufbauen, 4385
  - Wählregeln definieren, 4344
  - Zeilen ins Telefonbuch einfügen, 4340
  - Zugriff auf Telefonbücher, 4338
- TeleService über GPRS, 672
  - Verbindung aufbauen, 673
- Temperaturkoeffizient, 928, 946
- Temperaturkompensation, 915
- Template, 3108
- temporär, 667
- Temporäre Station, 667
- Temporäre Verbindung, 667
- Terminal- und Elektronikmodule, 913
- Testmöglichkeiten in der Forcetabelle, 1560
- Text
  - Schaltfläche, 3152
  - Textfeld, 3156
- Texte
  - migrieren, 128
- Textfeld, 3156
  - Größe, 3157
- Textfeldlänge
  - asiatische Sprachen , 3971
- Textliste
  - Bit (0, 1), 3083
  - Bitnummer (0 - 31), 3085
  - erstellen, 3079
  - exportieren, 3871
  - importieren, 3872
  - Symbolisches E/A-Feld, 3085, 3154
  - Verwendung, 3078
  - Wert/Bereich, 3082
- Textlisten
  - anwenderdefiniert, 307
  - Einführung, 306
  - systemdefiniert, 307
  - Verwendung in Rezepturdatensätzen, 3281
- TFTP
  - Laden/Speichern, 686
- TIA-Portal
  - beenden, 171
  - starten, 171
- TIME, 1107, 1147, 1174
- Time delay interrupt, 891
- Time error interrupt, 895
- TIME\_TCK, 2362
- TIME\_TO\_, 1174
- TIMER, 1126
- TM\_MAIL, 3016
- TMAIL\_C, 2956
- TO\_Axis\_PTO, 4146
- TOD, 1108, 1148, 1176
- TOD\_TO\_, 1176
- TOF, 1653, 1661, 1914, 1923, 2164
- TON, 1650, 1659, 1911, 1920, 2161
- TONR, 1655, 1663, 1916, 1925, 2167
- Tool Calling Interface (TCI), 822
- Tooltip
  - anzeigen, 3978
- Topologieerkennung, 816, 3510, 3561, 3614
- Topologiesicht
  - Diagnosestatus in der grafischen Ansicht, 498
  - Diagnosestatus in der tabellarischen Ansicht, 499
  - Funktionen, 494
  - Gerät hinzufügen, 400
  - Hardware- und Netzwerkeditor, 375, 3460
  - Offline-/Online-Vergleich, 500
  - Online ermittelte Geräte übernehmen, 510
  - Online ermittelte Port-Verschaltungen übernehmen, 509
  - Ports verschalten, 502
  - projektierte Topologie, 496, 497
  - Unterschiede zur Netzsicht, 494

- Topologiesicht starten, 496
- Totzone
  - einstellen, 3221
- Touch-Bedienung
  - EA-Feld, 3983
- TP, 1648, 1658, 1909, 1919, 2158
- TP177A
  - Projekt laden, 68
- Trace, 4296
  - Abtastung, 4311
  - Aufzeichnung, 4298, 4317
  - Aufzeichnungsbedingungen, 4326, 4327, 4329
  - Aufzeichnungsdauer, 4333
  - Aufzeichnungsebenen, 4323
  - Aufzeichnungstakt, 4311, 4333
  - Bitspur, 4301
  - CPU-Belastung, 4323
  - Datenablage, 4298
  - drucken, 4321
  - erfassbare Variablen, 4322
  - Kurvendiagramm, 4301, 4320
  - Lebensdauer der Werte, 4323
  - Mengengerüst, 4323
  - Mess-Cursor, 4320
  - Messung, 4298, 4307, 4318
  - Oberfläche, 4299, 4324
  - Pretrigger, 4329
  - Projektnavigator, 4301
  - Schnelleinstieg, 4309
  - Signale, 4325, 4332
  - Signaltafel, 4303, 4320
  - Trace-Editor, 4315
  - Tracekonfiguration, 4298, 4305, 4315, 4316, 4319, 4332
  - Tracekonfiguration speichern, 4316
  - Traces im Gerät, 4306
  - Traceverwaltung, 4304
  - Trigger, 4334
  - Triggermodus, 4328
  - Triggervariable, 4326, 4328
  - Übersetzung, 4327
  - unterstützte Geräte, 4296
  - Zeitstempel, 4308
- Trace S7-1200/1500, 4322
- Tracefunktion, 4296
- transparente Farbe, 3135
  - Anzeige auf Panel-Bediengerät, 3985
  - Anzeige auf Panels, 3135
- Transparenz
  - in Grafik, 3061
- TRCV, 2984, 2988
- TRCV\_C, 44, 2942, 2948
- Trennen der Fernverbindung, 4369
- Trigger, 3437
  - Ereignis-Trigger, 3437, 3440
- Triggervariable, 3214
- TRUNC, 1804, 2073, 2266
- TS Adapter, 71
- TS Adapter IE
  - Funktionsweise, 4363
  - Kurzbeschreibung, 4362
  - parametrieren, 4367, 4381
  - Standard-Parametrierung, 4364
- TS Adapter IE Advanced
  - Anschlussarten, 4379
  - Kurzbeschreibung, 4379
  - parametrieren, 4381
  - Parametriermöglichkeiten, 4380
- TS Adapter MPI
  - Aufbau des Direktanschlusses, 4357
  - Aufbau des Modemanschlusses, 4357
  - Direktanschluss, 4356
  - Funktionsweise, 4355
  - Kurzbeschreibung, 4355
  - Modemanschluss, 4357
  - parametrieren, 4359
  - Parametriermöglichkeiten, 4358
  - Parametrierung durchführen, 4359
  - Standard-Parametrierung wiederherstellen, 4360
- TS Adapter MPI
  - Standard-Parametrierung, 4356
- TS Adapter IE Basic
  - Anschluss an das GSM-Netz, 4365
  - Anschluss an das Telefonnetz, 4365
  - Anschluss an das Telefonnetz über ein externes Modem, 4366
  - Anschlussarten, 4364
- TS Adapter MPI
  - Adapterparameter exportieren, 4361
  - Adapterparameter importieren, 4362
  - Zugriffsschutz einrichten, 4352
- TSAP
  - ASCII-Codetafel, 482
  - Aufbau, 444, 480
- TSAP-Vergabe
  - Beispiele, 482
- TSEND, 2978, 2981
- TSEND\_C, 2931, 2935
- TURCV, 2997
- Türkisch, 34
- Türkische Regions- und Sprachoptionen, 34
- TUSEND, 2994
- Typ
  - Bibliothek, 3842

Typen auf neueste Version aktualisieren, 360  
Typen von DP-Slaves, 775

## U

UART-Datenübertragung, 868  
UBLKMOV, 1793, 2062, 2254  
UDINT, 1099, 1144, 1170  
UDINT\_TO\_, 1170  
UDP, 535, 541, 558  
    Merkmale, 472  
    Port-Nummern, 478  
UDT, 156, 166  
UFILL\_BLK, 1779, 2047, 2230  
Uhrzeit, 856, 1108, 1109, 1110, 1111  
    der CPU stellen mit WR\_SYS\_T, 2355  
    Lokalzeit berechnen mit SET\_TIMEZONE, 2359  
    Precision Time Protocol, 697  
    SNTP (Simple Network Time Protocol), 695  
    Systemzeit der CPU lesen mit TIME\_TCK, 2362  
    Uhrzeit einstellen in Online- und Diagnosesicht, 994  
    Uhrzeit und Datum der CPU lesen mit RD\_SYS\_T, 2356  
    Uhrzeitsynchronisation, 695  
Uhrzeit ermitteln, 994  
Uhrzeitalarm  
    abfragen mit QRY\_TINT, 2527  
    aktivieren mit ACT\_TINT, 2526  
    Funktion, 887  
    Regeln, 887  
    Status abfragen, 887  
    stellen mit SET\_TINT, 2522  
    stellen mit SET\_TINTL, 2524  
    stellen und aktivieren, 887  
    stornieren, 887  
    stornieren mit CAN\_TINT, 2525  
Uhrzeitalarm-OB  
    parametrieren, 900  
Uhrzeitfunktion  
    Grundlagen, 863  
    Lesen der Uhrzeit, 863  
    Parameter der Uhr, 863  
    Stellen der Uhrzeit, 863  
    Uhrzeitformat, 863  
Uhrzeitsynchronisation, 856  
UINT, 1097, 1142, 1166  
UINT\_TO\_, 1166  
ULINT, 1101  
umbenennen  
    Bild, 3032  
    Variable, 3172  
    Vorlage, 3038  
UMOVE\_BLK, 1775, 2043, 2226  
umschalten  
    zwischen Runtime-Sprachen, 3900, 3970  
umschalten AUS, 3431  
Umschalten EIN, 3431  
Umschalten zwischen Basismodus und Erweiterterem Modus in der Beobachtungstabelle, 1535  
Umschalten zwischen Basismodus und Erweiterterem Modus in der Forcetabelle, 1563  
Umschaltzeit, 875  
Umwandler  
    Zeiten konvertieren mit T\_CONV, 2350  
UND, 1884, 1885  
ungültige Lizenz  
    bei Zeitonenwechsel, 80, 92  
Universell  
    PROFIBUS, 3569, 3622  
Universelles-serielles-Schnittstellen-Protokoll, (Siehe USS-Protokoll)  
Unknown Peers, 570  
UNSCALE, 1812, 2082, 2278  
Unspezifizierte CPU umwandeln, 150  
Unterlauf, 930, 938  
UPDAT\_PI, 2385  
UPDAT\_PO, 2387  
Update, 890  
Update-Alarm, 890  
Update-Alarm-OB, 890  
URCV, 2925  
Urlöschen, 839  
    durchführen, 994  
USB-Kartenleser, 32  
USEND, 2923  
USINT, 1095, 1140, 1163  
USINT\_TO\_, 1163  
USS\_DRIVE, 2852  
USS\_PORT, 2851  
USS\_RPM, 2855  
USS\_WPM, 2856  
USS-Kommunikation  
    Datenaustsch mit Antrieben über USS\_DRIVE, 2852  
    Parameter aus Antrieb lesen mit USS\_RPM, 2855  
    Parameter im Antrieb ändern mit USS\_WPM, 2856  
    Übertragung zu Antrieb steuern mit USS\_PORT, 2851  
USS-Protokoll  
    Anweisungen, 2846  
USS-Statuscodes, 2858



## Ü

Überblick über die Querverweisliste, 1522  
 Überlagern  
   Variable, 1074  
 Überlagern von Variablen, 1268  
 Überlauf, 930, 938, 3430  
 übersetzen  
   Editor, 3888  
 Übersetzen  
   Adressparameter, 67  
   Baustein, 1471  
   Bausteine, 1468  
   Konsistenzprüfung, 1468  
   migriertes Projekt, 119  
   Übersetzungsfehler beheben, 1472  
 Übersetzen von Texten, 240  
 Übersicht  
   über die Anzeigeformate in der  
   Beobachtungstabelle, 1543  
   über die Anzeigeformate in der Forcetabelle,  
   1569  
 Übersicht der einsetzbaren TS Adapter, 4346  
 Übersichtsfenster, 50, 199  
   Alle Bausteine anzeigen, 202  
   Detailansicht sortieren, 202  
   Objekte vergleichen, 201  
   Spalten einblenden und ausblenden, 202  
 Übersichtsnavigation, 370, 373, 376, 3452, 3458,  
 3461  
 übertragen  
   Rezepturdatensatz, 4018  
 Übertragungskarte, (Siehe Memory Card)  
 Übertragungsmedium/Duplex, 814  
 Übertragungsrate/Duplex, 3508, 3559, 3612  
 Überwachen, 814, 3508, 3559, 3612  
 Überwachungsfenster, 955

## V

VAL\_STRG, 2369

## Variable

absolut adressieren, 3164, 3484  
 Adressierung, 3167  
 Aktualisierung, 3182, 3188  
 Datentyp, 3169  
 Ereignis, 3187  
 Erfassungsart, 3182  
 Erfassungszyklus, 3182, 3188, 3200  
 exportieren, 3865

externe Variable, 3163, 3483  
 externe Variable anlegen, 3167  
 Grenzbereich, 3178  
 Grenzwert, 3179  
 importieren, 3867  
 in Meldung ausgeben, 3222  
 in Runtime, 3182  
 Indexvariablen, 3186  
 IndexVariablen, 3186  
 indirekt adressieren, 3186  
 interne Variable, 3166, 3486  
 interne Variable anlegen, 3169  
 Kommentar, 3169  
 Konfiguration, 3173  
 kopieren, 3172  
 Länge, 3169  
 linear skalieren, 3183, 3184  
 löschen, 3173  
 maximale Länge, 69  
 mehrere Variable konfigurieren, 3174  
 multiplexen, 3186  
 Name, 3169  
 negativ, 1646, 1907  
 neu verbinden, 3177  
 Objektliste, 3169  
 PLC-Variable, 1186  
 PLC-Variablen und DB-Variablen, 1057  
 positiv, 1645, 1906  
 ständig lesen, 3182  
 Startwert, 3180, 3181  
 Steuerung wechseln, 3185  
 symbolisch adressieren, 3165, 3485  
 überlagern, 1074  
 Übersicht, 1056  
 umbenennen, 3172  
 Variableninformationen ein- oder ausblenden,  
 1315, 1357  
 Verbindung zur Steuerung, 3167  
 Werte darstellen, 3201  
 Variable bei negativer Signalfanke setzen, 1646,  
 1907, 2157  
 Variable bei positiver Signalfanke setzen, 1645,  
 1906, 2155  
 Variable lesen, 645  
 Variable schreiben, 646  
 Variable verwenden, 3131  
 Variable zuordnen, 631  
 Variablen  
   Alle beobachten, 1550, 1574  
   Datenaustausch, 3447  
   Grundlagen, 3160, 3480  
   Migration, 138

- migrieren, 124
  - Sofort beobachten, 1551, 1575
- Variablendaten
  - Struktur für den Import, 3868, 3873
- Variablendeklaration
  - ARRAY deklarieren, 1265
  - basierend auf einem PLC-Datentyp, 1418
  - Bausteinschnittstelle, 1258
  - Bausteinschnittstelle aktualisieren, 1269
  - Eigenschaften von Variablen, 1272, 1274, 1424, 1426, 1427
  - Gültige Datentypen, 1262
  - Multiinstanz, 1269
  - PLC-Datentyp deklarieren, 1267
  - Remanenz, 1273
  - Reservierte Schlüsselwörter, 1059
  - Spalten ein- und ausblenden , 1210, 1278, 1430, 1445
  - STRUCT deklarieren, 1266
  - Tabellenzeile am Ende einfügen, 1276, 1428, 1444
  - Tabellenzeile einfügen, 1207, 1276, 1428, 1443
  - Variable deklarieren, 1262, 1263, 1268
  - Variable importieren und exportieren, 1430
  - Variable löschen, 1208, 1277, 1429, 1444
  - Variablen importieren und exportieren, 1278
  - Variablen überlagern, 1268
  - Zeilen sortieren, 1209
  - Zellen automatisch ausfüllen, 1209, 1277, 1430, 1445
  - Zweck der Variablendeklaration, 1257
- Variablenimport
  - Struktur der Variablendaten, 3868, 3873
- Variableninformationen ein- oder ausblenden, 1315, 1357
- Variablenliste
  - indirekt adressieren, 3186
- Variablen-Simulator, 3946
- Variablentabelle
  - anwenderdefinierte, 3161, 3481
  - für HMI-Geräte, 3161, 3481
  - Standard, 3161, 3481
- Variablenwert
  - ausgeben, 3201
- variabler Index, 164
- VARIANT, 1125
- VARIANT in DB\_ANY konvertieren, 2272
- VARIANT\_TO\_DB\_ANY, 2272
- Verbindung, 433, 446, 3447
  - Adressdetails, 1022
  - Allen-Bradley DF1, 3722
  - Allen-Bradley EtherNet/IP, 3706
  - anlegen, 466, 3473, 3477
  - bei fehlender oder nicht eindeutiger Netzuzuordnung projektieren, 438
  - Bereichszeiger, 3488, 3492
  - Hervorheben, 3475
  - integriert, 3473
  - integrierte Verbindung, 3164, 3484
  - Kennwort, 3518, 3570
  - löschen, 469
  - Mitsubishi MC TCP/IP, 3748
  - Modicon Modbus RTU, 3788
  - Modicon Modbus TCP, 3777
  - offline, 3526, 3579, 3637, 3678
  - Omron Hostlink, 3805
  - Omron Kommunikationstreiber, 3817
  - Parameter, 3708, 3723, 3750, 3763, 3778, 3790, 3807
  - Passwort, 3506, 3518, 3557, 3570
  - projektieren, 3473, 3477, 3706, 3722, 3748, 3762, 3777, 3788, 3805
  - S7 200, 3663, 3693
  - Tabelle, 3471
- Verbindung löschen, 154
- Verbindung projektieren
  - Allen-Bradley DF1, 3722
  - Allen-Bradley EtherNet/IP, 3706
  - Mitsubishi FX, 3762
  - Mitsubishi MC TCP/IP, 3748
  - Modicon Modbus RTU, 3788
  - Modicon Modbus TCP, 3777
  - Omron Hostlink, 3805
- Verbindung überprüfen, 3002
- Verbindung zurücksetzen, 3000
- Verbindungen
  - Editor, 3650, 3652, 3654
  - projektieren, 3650, 3652, 3654, 3663, 3693
- Verbindungsaufbau von einer entfernten Anlage ("PG\_DIAL"), 4384
- Verbindungsbeschreibung
  - Aufbau, 473, 475, 477
  - Datenbaustein , 472, 475, 476
  - Parameterwerte ändern, 479
- Verbindungsdetails, 1022
- Verbindungsdiagnose
  - detailliert, 1020
  - Überblick, 1018
- Verbindungsinformation, 1020
  - Diagnose, 3455
- Verbindungskabel
  - 6XV1440-2P für Mitsubishi PG-Protokoll, 3766
  - Allen-Bradley-Kabel 1747-CP3, 3732
  - Allen-Bradley-Kabel 1761-CBL-PM02, 3733

- Allen-Bradley-Kabel 1784-CP10, 3731
- Mehr-Punkt-Kabel 1:MP/TP/PC, 3811
- Mehr-Punkt-Kabel 2:RS422, MP/TP/PC, 3812
- Mitsubishi FX, 3766
- Modicon, 3792
- Omron Host Link, 3809
- Punkt-zu-Punkt-Kabel 1,
- Punkt-zu-Punkt-Kabel 2, 3796
- Punkt-zu-Punkt-Kabel-PP2 für Omron, 3812, 3813
- Verbindungskabel 9-pol.
  - Sub D, RS 422, 3730
- Verbindungsmechanismen, 884
- Verbindungsmodi, 667
- Verbindungsparameter
  - Allen-Bradley DF1, 3723
  - Allen-Bradley EtherNet/IP, 3708
  - Mitsubishi FX, 3763
  - Mitsubishi MC TCP/IP, 3750
  - Modicon Modbus RTU, 3790
  - Modicon Modbus TCP, 3778
  - Omron Host Link, 3807
- Verbindungsparameter PUT/GET-Anweisung, 487
- Verbindungsparametrierung
  - starten, 465
  - Übersicht, 459
  - Verbindungsparameter, 462
- Verbindungsparametrierung PUT/GET-Anweisung, 488
- Verbindungsregeln, 613, 623
- Verbindungsressource, 433, 447
  - SIMATIC S7 1200, 65
- Verbindungsressourcen
  - online, 1020, 1021
- Verbindungsstatus
  - mittels Symbolen anzeigen, 1019
- verdeckte Eingabe, 3133
- Verdrahtung testen, 1533
- vereinheitlichen
  - Objektgröße, 3049
- VerfolgeBenutzeränderung, 3409
- Verfügbarkeit
  - Objekt für Basic Panel, 3127
- Vergleichen
  - Bitmaske, 1872, 2143, 2337
  - Gleich, 1706, 1972
  - Größer, 1715, 1980
  - Größer gleich, 1711, 1976
  - Kleiner, 1717, 1982
  - Kleiner gleich, 1713, 1978
  - Ungleich, 1709, 1974
  - Zeichenketten mit S\_COMP, 2364
- Vergleichen
  - Zeitvariablen, 2349
- Vergleichseditor
  - Aktionen festlegen, 272
  - Ansicht ändern, 274
  - Ansicht filtern, 268
  - Übersicht, 264
- Vergleichsstelle, 780, 915, 930, 938, 947, 948
- Verhalten
  - Datum-Uhrzeit-Feld, 3981
  - EA-Feld, 3983
  - einfache Rezepturanzeige, 3297, 4013
  - grafisches EA-Feld, 3986
  - Schalter, 3990
  - Schaltfläche, 3989
  - symbolisches EA-Feld, 3992
- Verkabelungsregeln (PROFINET), 815, 3509, 3560, 3613
- Verknüpfen
  - EXKLUSIV ODER, 1840, 2111
  - ODER, 1838, 2109
  - UND, 1836, 2107
- Verknüpfte Objekte
  - kopieren, 3924
- Verknüpfungspfad
  - einfügen, 1362
  - löschen, 1362
  - Verwendung, 1361
- vernetzen
  - Geräte, 3471
- Vernetzen
  - von Kommunikationspartnern, 3472
- Vernetzen in der Gerätesicht
  - Möglichkeiten zur Vernetzung, 417
  - Vorgehen, 418
- Vernetzung bearbeiten
  - Subnetz kopieren, 421
  - Subnetze und Geräte kopieren, 421
- Vernetzungen bearbeiten
  - Netzanschluss aufheben, 420
- VerringereFokussiertenWert, 3409
- VerringereVariable, 3410
- verschieben
  - Bild, 3032
  - Vorlage, 3038
- Verschieben
  - Hardware-Komponenten, 407
- Verschlüsselung, 512
- Verschlüsselungssoftware, 83, 86
- Version eines Typen im Test bearbeiten, 346
- Versionierung von Typen
  - Freigegebene Version, 340

- Version im Test, 340
- Version in Bearbeitung, 339
- Versorgungsspannung, 938
- Versteckte Parameter, 1315, 1357
- verteilen
  - Objekte gleichmäßig, 3053
- Vertikale Bewegung
  - Animation, 3102
- verwalten
  - Benutzer, 3316
  - Benutzer in Runtime, 3323
  - Benutzergruppe, 3317
  - Grafik, 3061
- verwendbare Zeichen
  - Benutzeranzeige, 69
- verwenden
  - Funktionstaste zur Bildnavigation, 3120
- Verwendung
  - Benutzeranzeige, 3129
  - Datum-Uhrzeit-Feld, 3981
  - EA-Feld, 3982
  - einfache Benutzeranzeige, 3994
  - einfache Meldeanzeige, 4003
  - einfaches Meldefenster, 4003
  - Grafikanzeige, 3984
  - grafisches EA-Feld, 3985
  - Kurvenanzeige, 3987
  - Rezepturanzeige, 3149
  - Schalter, 3990
  - Schaltfläche, 3989
  - symbolisches EA-Feld, 3991
- Verwendung von UDTs, 166
- Verzögerungsalarm
  - abfragen mit QRY\_DINT, 2532
  - Anweisungen, 2529
  - starten mit SRT\_DINT, 2530
  - stornieren mit CAN\_DINT, 2531
- Verzögerungsalarm-OB, 891
- Verzögerungszeit
  - einstellen, 3221
- Verzweigung, 2280, 2282
- Virens scanner, 34, 83, 86
- Virtuelle Maschine (VM)
  - unterstützte Virtualisierungsplattformen, 82, 85
- VKE
  - invertieren, 1628, 1890
- VKE auf negative Signalfanke abfragen, 1644, 1905
- VKE auf positive Signalfanke abfragen, 1642, 1904
- VLAN
  - Port VID, 713
  - Priorität, 713
  - Tag, 713

- VLAN-Betrieb, 566
- VLAN-Tagging, 566
- VOID, 1126
- Voraussetzungen für das Senden einer E-Mail, 4389
- Voraussetzungen für das Senden einer SMS, 4387
- Voraussetzungen zum Aufbau einer Fernverbindung, 4368
- Voraussetzungen zum Aufbau einer VPN-Verbindung, 4377
- Vordefinierte Firewall-Regeln
  - CP 1543-1, 619
  - CP 1628, 609
  - CP x43-1 Adv., 607
  - SCALANCE S, 591, 593
- Vordefinierte Meldeklassen, 3209
- Vordergrundfarbe
  - dynamisch ändern, 3099
- Vorlage
  - anlegen, 3037
  - einfügen, 3038
  - Globales Bild, 3036
  - kopieren, 3038
  - löschen, 3038
  - umbenennen, 3038
  - verschieben, 3038
  - verwenden im Bild, 3039
- Vorteile durch den Einsatz von TeleService, 4335
- Vorzeichen, 1737, 2003
- VPN-Gruppe, 564
- VPN-Verbindung
  - aufbauen, 4377
  - CA-Zertifikat installieren, 4373
  - CA-Zertifikat löschen, 4376
  - trennen, 4378
- VPN-Verbindung wird nicht aufgebaut
  - Überprüfung und Behebung, 4397
- VRRP
  - Adressübersicht, 750
  - Konfiguration, 749
  - Router, 748

**W**

- Wählregeln in Telefonbüchern, 4344
- Wählverbindung
  - aufbauen, 4368
  - trennen, 4370
- WAIT, 1836, 2107, 2307
- WAN-IP-Adresse
  - festlegen, 575
- Webapplikation, 639
- Web-Control-DB, 656

- Web-Datenbausteine, 654
  - Webseiten im Browser, 661
  - Webserver, 627, 633, 635, 639, 642, 654
    - aktivieren, 636
    - Enumerationen, 651
    - Fragment, 652
    - HTTPS, 637
  - WechseleVerbindung, 3411
  - Wechselmedium, 817
  - Wechsepuffer, 3535, 3700
  - Wechsler, 954
  - Weckalarm-OB
    - Beschreibung, 892
    - Parameter abfragen mit QRY\_CINT, 2521
    - Parameter setzen mit SET\_CINT, 2519
    - parametrieren, 902
  - Weckruf, 668
  - Weck-SMS, 668
  - Weitere Feldgeräte (PROFIBUS und PROFINET), 791, 819
  - Welcome Tour anzeigen, 76
  - wenig freier Speicherplatz, 3431
  - Wenig freier Speicherplatz, kritisch, 3431
  - Werkseinstellungen
    - rücksetzen, 3953
    - zurücksetzen auf, 997, 998
  - Werkzeuge, 3040
  - Wert/Bereich
    - Textliste, 3082
  - Wertänderung, 3432
  - Werte
    - vergleichen, 4059
  - Werte darstellen
    - als Kurve, 3201
  - Wertzuzuweisung, 1371
  - WHILE, 2287
  - wiederherstellen, 3993
    - Daten des Bediengeräts, 3950, 3952, 3966
  - Wiederholen von Aktionen
    - Aktionen wiederholen, 303
    - Grundlagen zum Wiederholen von Aktionen, 300
  - Wiederinbetriebnahme
    - Bediengerät, 3960
  - WinCC flexible Projekt
    - migrieren, 136
  - WinCC flexible Runtime
    - beenden, 3966
  - WinCC flexible-Projekt
    - migrieren, 118
  - WinCC V7.0 SP3, 54
  - WinCC-Version eines Projekts
    - hochrüsten, 3931
  - Windows Server 2003, 35
  - Wireless-LAN/PB-Link, 793
  - Wissenswertes zum Parametrieren des TS Adapter MPI, 4358
  - WORD, 1092, 1137, 1155
  - WORD\_TO\_, 1155
  - WR\_DPARM, 2515
  - WR\_LOC\_T, 2358
  - WR\_REC, 2435
  - WR\_SYS\_T, 2355
  - WRIT\_DBL, 45, 2616
  - WriteToArrayDB, 1785, 2053, 2235
  - WriteToArrayDBL, 1789, 2057, 2239
  - WRREC, 2394
  - Wurzel Quadrat, 2209
  - WWW (Anweisung), 655
  - WWW Anwenderdefinierte Webseiten synchronisieren, 3014
- X**
- xlsx-Datei, 3859, 3866, 3871, 3872
  - XOR, 1369, 1370, 1840, 1887, 1888, 2111
- Z**
- Z\_RUECK, 1701, 1966
  - Z\_VORW, 1700, 1964
  - ZAEHLER, 1697, 1962, 2199
  - Zählen
    - rückwärts, 1701, 1705, 1966, 1971, 2189, 2197
    - Rückwärts, 1692, 1956
    - vorwärts, 1700, 1704, 1964, 1969, 2186, 2195
    - Vorwärts, 1689, 1953
    - vorwärts und rückwärts, 1697, 1962, 2192, 2199
    - Vorwärts und rückwärts, 1694, 1959
  - Zählen (8 DI NAMUR), 917
  - Zähler
    - Schnelle Zähler, 2670
    - Schnelle Zähler steuern, 2670
  - Zähler, schneller, 863
  - Zählerart, 865
  - Zählereingang, 865
  - Zehnerkomplement, 1882, 2153, 2347
  - Zeichen, 1112
  - Zeichenkette, 1112
    - aus Hexadezimal konvertieren mit HTA, 2376
    - in Hexadezimal konvertieren mit ATH, 2374
    - konvertieren mit S\_CONV, 2366
    - Länge ermitteln mit LEN, 2378
    - linke Zeichen auslesen mit LEFT, 2380

- Maximale Länge ermitteln mit MAX\_LEN, 2373
- mittlere Zeichen auslesen mit MID, 2381
- rechte Zeichen auslesen mit RIGHT, 2380
- String-Variablen vergleichen mit S\_COMP, 2364
- verschieben / kopieren mit S\_CONV, 2364
- Zahl in Zeichenkette konvertieren mit VAL\_STRG, 2369
- Zahlenzeichenkette konvertieren mit STRG\_VAL, 2367
- Zeichen einfügen mit INSERT, 2383
- Zeichen ersetzen mit REPLACE, 2383
- Zeichen finden mit FIND, 2384
- Zeichen in Zeichenkette kopieren mit Chars\_TO\_Strg, 2373
- Zeichen löschen mit DELETE, 2382
- Zeichenkette in Zeichen kopieren mit Strg\_TO\_Chars, 2372
- Zeichenketten zusammenfassen mit CONCAT, 2379
- ZeigeAnmeldedialog, 3412
- ZeigeHilfetext, 3413
- ZeigeMeldefenster, 3414
- Zeiger
  - ANY, 1122
  - POINTER, 1120
  - VARIANT, 1125
- Zeilennummern anzeigen, 1375
- Zeit, 1105, 1107, 1668, 1930
  - erfassen, 1663
  - UTC-Zeit in Lokalzeit umrechnen mit SET\_TIMEZONE, 2359
- Zeit abgelaufen, 3432
- Zeit akkumulieren, 1925
- Zeit als Ausschaltverzögerung starten, 1923
- Zeit als Einschaltverzögerung starten, 1920
- Zeit als Impuls starten, 1919
- Zeit rücksetzen, 1927
- Zeitakkumulator, 1655, 1916, 2167
- Zeitdauer, 1105, 1667, 1928, 2171
- Zeiteinstellung, 684
- Zeiten
  - addieren mit der Anweisung T\_ADD, 2351
  - Datum und Uhrzeit zusammenfassen mit T\_COMBINE, 2355
  - Differenz ermitteln mit T\_DIFF, 2354
  - konvertieren mit T\_CONV, 2350
  - subtrahieren mit T\_SUB, 2353
  - Zeitvariablen vergleichen mit T\_COMP, 2349
- Zeitfehler-OB, 895
- Zeitgetriggerte Kurven, 3536, 3699
- Zeitstempel, 3213
- Zeitstempelung, 916
- Zeitsynchronisierung, 559
- Zeitverzögerung, 1836, 2107, 2307
- Zerstörung von License Keys, 78
- Zertifikat, 518, 566
  - austauschen, 521
  - durch Zertifizierungsstelle signiert, 520
  - erneuern, 520
  - ersetzen, 521
  - exportieren, 518
  - importieren, 518
  - selbst-signiert, 520
- Zertifikatsmanager, 518
- Zertifizierungsstelle, 519
- Ziehen/Stecken-Alarm (ET 200M), 951
- Ziehen/Stecken-OB, 897
- Zoom
  - Einstellen der Zoom-Stufe, 370, 373, 376, 3452, 3458, 3461
  - Tastaturbedienung, 386
- zoomen
  - Bild, 3030
- ZR, 1705, 1971
- Zugelassene Operanden für die Forcetabelle, 1567
- Zugelassenen Operanden für die Beobachtungstabelle, 1540
- Zugriff auf Operanden, 1065, 1066, 1067, 1069, 1070, 1072, 1074, 1085
- Zugriff über PUT/GET-Kommunikation durch entfernte Partner erlauben, 884
- Zugriff von entfernten Anlagen auf ein PG/PC, 4384
- Zugriffsschutz, 3329
  - Benutzerverwaltung, 3328
  - Gültigkeit, 4350
  - projektieren, 3333
  - Vorteile, 4350
  - werkseitig, 4350
  - Zugriffsschutzstufe 1, 4351
  - Zugriffsschutzstufe 2, 4351
- zulässig
  - Datentyp, 3711, 3757, 3767, 3784, 3797, 3813
  - Datentypen, 3540, 3592, 3650, 3692, 3704
- zulässige Datentypen
  - Kurven, 3536, 3701
- zulässiger Datentyp
  - Allen-Bradley DF1, 3734
- Zurücksetzen
  - auf Werkseinstellungen, 997, 998
- zuweisen
  - Objekt einer Ebene, 3122
- Zuweisen
  - einer Funktionstaste eine Funktion, 3113
  - einer Funktionstaste eine Grafik, 3116

Zuweisung, 1371, 1628, 1890  
  negiert, 1629, 1891  
Zuweisung globaler Datenbausteine, 43  
ZV, 1704, 1969  
Zweierkomplement, 1733, 1999  
Zweig  
  Definition, 1316, 1358  
  einfügen, 1318, 1359  
  löschen, 1319, 1360  
  Regeln, 1317  
  Regeln , 1359  
  schließen, 1318  
Zyklisch  
  fortlaufend, 3182  
  im Betrieb, 3182  
Zyklische Programmbearbeitung  
  Programmieren, 885  
  Unterbrechungsmöglichkeiten, 885  
zyklischer Betrieb, 3662, 3665, 3698  
Zyklusbelastung, 861  
Zyklus-OB  
  Beschreibung, 885  
Zyklusüberwachungszeit, 838, 861, 1827, 2098,  
2300  
Zykluszeit, 856, 860  
  gemessene anzeigen, 981  
  parametrierte anzeigen, 976

